Context-Aware Hierarchical Taxonomy Generation for Scientific Papers via LLM-Guided Multi-Aspect Clustering

Kun Zhu[1*], **Lizi Liao**[2], **Yuxuan Gu**[1], **Lei Huang**[1], **Xiaocheng Feng**[1,3†], **Bing Qin**[1,3]
[1]Harbin Institute of Technology [2] Singapore Management University
[3] Peng Cheng Laboratory
{kzhu,yxgu,lhuang,xcfeng,qinb†}@ir.hit.edu.cn lzliao@smu.edu.sg

Abstract

The rapid growth of scientific literature demands efficient methods to organize and synthesize research findings. Existing taxonomy construction methods, leveraging unsupervised clustering or direct prompting of large language models (LLMs), often lack coherence and granularity. We propose a novel **context-aware hierarchical taxonomy generation framework** that integrates LLM-guided multi-aspect encoding with dynamic clustering. Our method leverages LLMs to identify key aspects of each paper (*e.g.*, methodology, dataset, evaluation) and generates aspect-specific paper summaries, which are then encoded and clustered along each aspect to form a coherent hierarchy. In addition, we introduce a new benchmark of 156 expert-crafted taxonomies encompassing 11.6 k papers, providing the first **naturally annotated** dataset for this task. Experimental results demonstrate that our method significantly outperforms prior approaches, achieving state-of-the-art performance in taxonomy coherence, granularity, and interpretability.[1]

1 Introduction

The rapid expansion of academic publications has created an overwhelming amount of information, making it increasingly challenging for researchers to stay up-to-date and systematically organize domain knowledge (Reisz et al., 2022; Hanson et al., 2024; Vineis, 2024). As a result, there is a growing demand for structured and concise taxonomies that can support the exploration and synthesis of more efficient literature (Shen et al., 2018; Zhu et al., 2023). Traditional approaches to building taxonomies of scientific papers typically rely on manual or narrowly defined schemes. Common solutions include supervised classification into a pre-defined hierarchy (*e.g.*, ACM CCS) (Zhang et al., 2021; Sadat and Caragea, 2022; Rao et al., 2023) and unsupervised clustering of papers followed by post-hoc keyword-based label extraction (Zhang et al., 2018; Shang et al., 2020). These methods often require substantial human curation or yield coarse topic structures, limiting their usefulness for in-depth literature understanding.

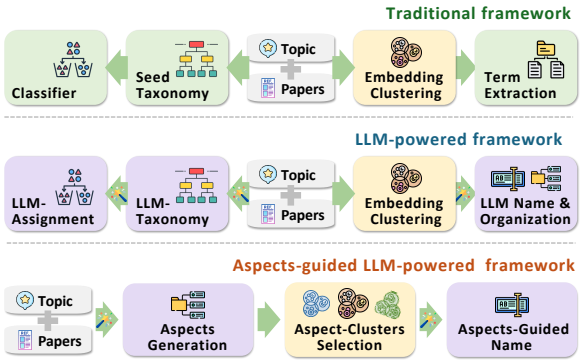


Figure 1: Comparison of taxonomy construction paradigms. Traditional methods typically use supervised classification or clustering with term extraction. Recent approaches incorporate LLMs to replace or enhance key components within these pipelines (purple). Our approach uniquely integrates LLMs with clustering in a context-aware multi-aspect framework, resulting in coherent and precise hierarchical taxonomies.

Recent advances utilize LLMs to automate the taxonomy construction. LLMs demonstrate strong capabilities in long-text understanding and abstraction (Achiam et al., 2023; Grattafiori et al., 2024), leading to approaches that generate taxonomy trees or assign papers to categories in an end-to-end fashion (Hsu et al., 2024; Wan et al., 2024). Hybrid strategies first cluster papers and then prompt LLMs to produce summaries or category labels for each cluster (Katz et al., 2024; Hu et al., 2025).

While these LLM-based methods have shown promise, studies have found that they struggle to capture highly specialized knowledge and fine-grained concepts specific to scientific domains.

[*] Work was done during an internship at SMU.
[†] Corresponding Author
[1]Code and dataset are available in https://github.com/zhukun1020/TaxoBench-CS.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 15616–15634
November 4-9, 2025

Moreover, taxonomies produced solely by LLMs are not guaranteed to align with the content of a given corpus, often resulting in missing or hallucinated categories. Effective taxonomy construction inherently demands context-aware representations, wherein the characterization of each paper dynamically adapts based on its relationships and similarities to surrounding papers. Without this context awareness, papers focusing on distinct aspects (*e.g.*, methodologies *v.s.* datasets) might be incorrectly categorized, leading to incoherent taxonomy structures. This gap calls for new techniques that consider multiple content dimensions and their corpus-level context during taxonomy generation.

In this paper, we propose a novel framework for paper taxonomy generation that leverages LLM-guided, multi-aspect representations in conjunction with adaptive clustering. Specifically, our approach uses a dynamic aspect generator to automatically determine salient semantic aspects (such as research objective, methodology, or data source) for a given collection of papers. Guided by these, the LLM produces aspect-specific summaries for each paper, ensuring that each document is represented in a manner that is both facet-specific and context-aware. We then employ a dynamic clustering algorithm to search for an optimal grouping of papers for each aspect dimension. By iteratively applying multi-aspect encoding and clustering in a top-down fashion, our framework constructs a hierarchical taxonomy tree that is tailored to the corpus at each level. This design allows the taxonomy to capture different facets of the literature at different branches, yielding more coherent and interpretable category structures.

In addition to methodological innovations, a significant bottleneck in this area has been the lack of high-quality, naturally annotated datasets for evaluating taxonomy construction. Most existing benchmarks are synthetic (Hsu et al., 2024) or rely on coarse (Katz et al., 2024), predefined categories that fail to reflect the nuanced hierarchies. To bridge this gap, we construct a new dataset of academic taxonomies **TaxoBench-CS**, by collecting 156 human-authored taxonomy trees (covering 11.6 k research papers) from survey and review articles on `arXiv`. These taxonomies, created by domain experts, provide realistic hierarchical structures that mirror a deep understanding of topic decomposition. This dataset offers a valuable resource for training and evaluating taxonomy generation methods under more natural conditions, and we will release it to foster further research.

In summary, our contributions are threefold:

- We curate a high-quality benchmark consisting of 156 expert-annotated taxonomies of 11.6 k papers, facilitating future research.
- We propose to combine multi-aspect paper encoding with a dynamic clustering algorithm, enabling context-aware, hierarchical organization of research papers.
- Our approach outperforms existing state-of-the-art methods, yielding interpretable and human-readable taxonomy trees with significantly improved coherence and granularity.

2 Preliminary

Here, we first formalize the task of taxonomy construction for scientific literature. We then describe the creation of a new benchmark dataset derived from human-authored taxonomies in survey papers.

2.1 Task Definition

Given a specific topic x and a collection of corresponding scientific papers $\mathcal{D} = \{d_1, d_2, \ldots, d_N\}$, the objective is to generate a hierarchical taxonomy $\mathcal{T}(V, E)$ that organizes these papers into a tree structure of semantically coherent categories. In detail, the taxonomy of depth L starts from a root node $r \in V^{(0)}$ and each node $v \in V^{(l)}$ corresponds to a depth l, where $V = \bigcup_{l=0}^{L} V^{(l)}$. In addition, each node v is associated with a subset of papers $D_v \subseteq \mathcal{D}$ and a topic facet x_v (*e.g.*, high-level methodological approaches, underlying mechanisms or learning paradigms, or specific research tasks and evaluation scenarios). The root node r represents the overarching topic x and encompasses all papers $D_r = \mathcal{D}$. For every non-leaf node $v \in V^{(l<L)}$, its k_v child nodes $\text{Child}(v)$ form a complete, non-overlapping partition of the papers subset D_v, satisfying the constraints:

$$\text{Child}(v) = \{v_1, v_2, \ldots, v_{k_v}\} \subseteq V^{(l+1)}, \quad \text{with} \begin{cases} \bigcup_{t=1}^{k_v} D_{v_t} = D_v \\ D_{v_t} \bigcap D_{v_{t'}} = \emptyset, \ \forall t \neq t' \end{cases}. \tag{1}$$

Edges typically represent hierarchical semantic relations (*e.g.*, *isA*, *instanceOf*) and are restricted to link nodes across adjacent layers, where

$$E = \bigcup_{l=0}^{L-1} E^{(l)}, \ E^{(l)} \subseteq V^{(l)} \times V^{(l+1)}. \tag{2}$$

Datasets	Clustering	Hierachy	Ground Truth	Source
CLUSTREC-COVID (Katz et al., 2024)	✓	✗	✓	synthetic
SCITOC (Katz et al., 2024)	✗	✓	✓	natural
SciPile (Gao et al., 2025)	✓	✓	✗	synthetic
CHIME (Hsu et al., 2024)	✓	✓	✗	synthetic
TaxoBench-CS (Ours)	✓	✓	✓	natural

Table 1: Comparison of existing taxonomy datasets: Datasets are evaluated based on three key criteria: clustering annotations, hierarchical structures, and ground-truth labels. We also distinguish whether datasets are synthetic or naturally derived. Our dataset uniquely meets all three criteria while being naturally sourced.

In our framework, the taxonomy is built iteratively by partitioning each subset D_v from the depth l into disjoint subsets assigned to its children.

2.2 Dataset Construction

Existing datasets for evaluating taxonomy generation methods generally rely on either topic-based retrieval followed by manual annotation (Katz et al., 2024) or LLM-assisted taxonomy creation and filtering (Hsu et al., 2024; Gao et al., 2025). However, these approaches often introduce noise into the structure and lack high-quality, reliably annotated ground-truth hierarchies.

To address these limitations, we introduce a new benchmark dataset, **TaxoBench-CS**, constructed from naturally annotated taxonomy trees found in computer science review papers on `arXiv`[2]. We start by systematically selecting survey papers that contain explicit hierarchical taxonomy diagrams. By parsing the corresponding LaTeX source files, we extract citation identifiers directly linked to taxonomy structures, which are then mapped to their full titles using the citation metadata provided in each paper's associated `.bib` or `.bbl` files. Next, we retrieve detailed paper metadata from Semantic Scholar[3]. To ensure the dataset's accuracy and reliability, we manually verify all citation mappings, eliminating any incorrect or ambiguous entries.

The final TaxoBench-CS dataset consists of 156 author-curated taxonomy trees, serving as robust hierarchical annotations. Each taxonomy contains, on average, 74.4 referenced papers and spans 3.1 levels in depth. Excluding the paper citation indicators connected to the leaf-level nodes, each tree includes around 24.8 nodes that represent structured semantic categories, providing a rich and structurally sound resource. As shown in Table 1, our proposed TaxoBench-CS uniquely combines explicit clustering structures, hierarchical organization, and authoritative annotations derived directly from naturally occurring expert-curated taxonomies. This combination makes it an ideal benchmark for evaluating and developing taxonomy generation methods under realistic conditions.

[2]https://arxiv.org/

[3]https://www.semanticscholar.org/me/research

3 Method

The core of our method lies in appropriately decomposing the given node v of depth l according to the structure and semantics of its associated paper set D_v. We first represent papers in the associated paper set $d_i \in D_v$ using multi-aspect encoding (§3.1). Given the clustering results over the multi-aspect vectors of D_v, we apply a dynamic search algorithm to determine the most appropriate partitioning strategy (§3.2). Therefore, we can iteratively partition the paper set D_v and get the child nodes $\text{Child}(v)$ of node v from a top-down manner to construct the taxonomy tree (§3.3).

3.1 Multi-Aspect Paper Encoding

In this part, our goal is to obtain a global representation of the paper set D_v that captures its overall semantic structure. To this end, we propose to automatically generate a set of candidate aspects $\mathcal{A}_v$ using an LLM based on all papers in D_v. These aspects are then used in a parallel manner to guide the encoding of individual papers. The aspect generator is defined as follows:

$$\mathcal{A}_v \sim p_{\text{LLM}}(\mathcal{A}|v, D_v), \tag{3}$$

where we prompt the LLM such as GPT-4o to analyze the paper distribution in D_v according to the global trace of current node v (topic facets of v and all its ancestor nodes) before generating the detailed content of aspects $\mathcal{A}_v$. In addition, the LLM is required to infer the number of aspects $|\mathcal{A}_v|$ automatically. We demand the LLM to identify a set of salient semantic dimensions that can effectively characterize and classify the papers, such as research problem and application domain.

Given the discovered aspects $a \in \mathcal{A}_v$, we parallelly generate aspect-guided summaries s_a^d for each

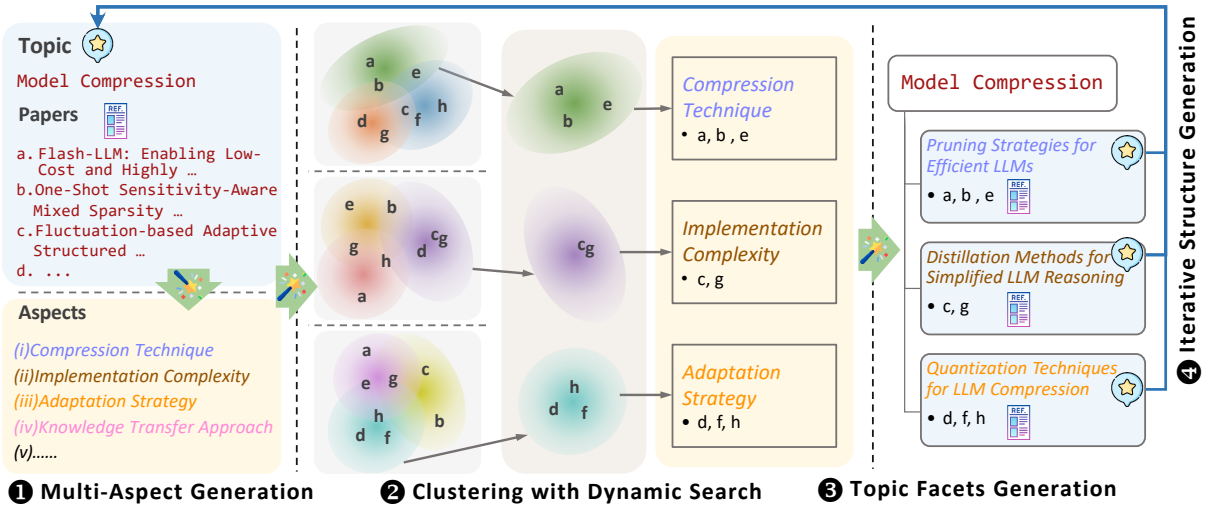


Figure 2: Our proposed Aspects-guided LLM-based Top-Down Clustering framework. Specifically, we dynamically generate multiple semantic aspects to represent each paper, and perform aspect-specific clustering via dynamic search. The abstract aspects are instantiated into concrete topic facets, which serves as the heading of nodes. This process is iteratively applied to construct a coherent and semantically meaningful taxonomy.

paper $d \in D_v$ by prompting the LLM. Each summary is then encoded into a n-dimensional vector $e_a^d \in \mathbb{R}^n$, where we have:

$$\begin{aligned} &\textit{For all } \ a, d \ \in \mathcal{A}_v \times D_v \textit{ in parallel}: \\ &\qquad e_a^d = \mathrm{Enc}(s_a^d), \quad s_a^d \sim p_{\mathrm{LLM}}(s|a,d)\ . \end{aligned} \tag{4}$$

We collect the encoding of paper set D_v for each aspect and obtain $\mathbf{e}_a = \{e_a^d \mid \forall d \in \mathcal{D}_v\}$, which can also be regarded as a matrix $\mathbf{e}_a \in \mathbb{R}^{|D_v| \times n}$.

3.2 Clustering with Dynamic Search

Given that encoding across different aspects may reside in heterogeneous semantic spaces with varying structures and scales, directly aggregating all representation vectors $\mathbf{e} = \{e_a^d \mid \forall d \in D_v, \forall a \in \mathcal{A}_v\}$ into a unified space for clustering would be inappropriate. Therefore, we perform clustering independently within each aspect space $\mathbf{e}_a$:

$$\begin{aligned} &\textit{For all } \ a \ \in \mathcal{A}_v \textit{ in parallel}: \\ &\quad f_a : \mathbf{e}_a \times \{1, 2, \ldots, k\} \to [0, 1] \\ &\quad \text{Expectation}: \ \forall i \in \{1, 2, \ldots, k\}, \\ &\qquad \mathcal{C}_a^i = \left\{ e_a^d \ \middle| \ \arg\max_j f_a(e_a^d, j) = i, \forall d \in D_v \right\} \\ &\quad \text{Maximization}: \\ &\qquad \mathcal{L}_a^{\text{cluster}} = -\sum\nolimits_{i=1}^{k} \sum\nolimits_{e \in \mathcal{C}_a^i} f_a(e, i), \end{aligned} \tag{5}$$

where $\mathcal{C}_a^i$ is the temporary allocation of the cluster index i and f_a is the clustering model that maps the encoding vector e to the cluster i with a probability of $f_a(e, i)$, $\sum_{i=1}^{k} f_a(e, i) = 1$. In addition, k is a hyperparameter that determines the number of clusters, where $k_v \leq |\mathcal{A}_v| \times k$.

Given the cluster assignment probabilities for each aspect, we need to select for each paper $d \in D_v$ a unique pair (a, i), where a is an aspect and i is a cluster index within that aspect, such that: (1) Each paper d will be assigned to only one cluster i. (2) The total number of unique pairs (a, i) used in the paper set D_v is k_v. (3) The total assignment probability is maximized. Therefore, we define a binary indicator $\delta_{a,i}^d \in \{0, 1\}$ and the objective:

$$\max_{\delta} \sum_{d \in D_v} \sum_{a \in \mathcal{A}_v} \sum_{i=1}^{k} \delta_{a,i}^d \cdot f_a(e_a^d, i), \tag{6}$$

which is subject to:

$$\begin{aligned} &\sum_{a \in \mathcal{A}_v} \sum_{i=1}^{k} \delta_{a,i}^d = 1, \forall d \in D_v \\ &\left| \left\{ (a, i) \ \middle| \ \exists d \in D_v \text{ s.t. } \delta_{a,i}^d = 1 \right\} \right| = k_v. \end{aligned} \tag{7}$$

As a result, we have the search process as illustrated in the algorithm 1, where we directly define a search space $\mathbb{S}$ containing all possible combinations $S \subseteq \mathcal{A}_v \times \{1, 2, \ldots, k\}$ that satisfy $|S| = k_v$. Each S encodes a specific clustering scheme with k_v unique aspect-cluster assignments (a, i). We adopt a real-time strategy that the `score` of every combination S is updated as each paper $d \in D_v$ arrives, where we trace the optimal assignment

Algorithm 1 Search with Pruning

Init. $\mathbb{S} \leftarrow \{S \subseteq \mathcal{A}_v \times \{1, \ldots, k\} \mid |S| = k_v\}$
Init. $\texttt{score}[S] \leftarrow 0,\ \forall S \in \mathbb{S}$
Init. $\texttt{state}[S][(a,i)] \leftarrow \{\},\ \forall S \in \mathbb{S},\ (a,i) \in S$
for all $d \in D_v$ in random order **do**
  for all $S \in \mathbb{S}$ **do**
    $\texttt{score}[S] \leftarrow \texttt{score}[S] + \max_{(a,i)\in S} f_a(e_a^d, i)$
    $\texttt{state}[S][\arg\max_{(a,i)\in S} f_a(e_a^d, i)].add(d)$
  end for
  if $\text{score}[S] \ll \text{avg score},\ \exists S \in \mathbb{S}$ **then**
    $\mathbb{S} \leftarrow \mathbb{S} \setminus S$
  end if
end for
$\texttt{max\_score} \leftarrow \max_{S\in\mathbb{S}} \texttt{score}[S]$
$S^* \leftarrow \arg\max_{S\in\mathbb{S}} \texttt{score}[S]$
$\texttt{optimal\_state} \leftarrow \texttt{state}[S^*]$
return S^*, max_score, optimal_state

trajectory via the state variable. Optionally, we can randomize the iterative order of the papers and prune the low-scoring combinations during the process to reduce search space and improve efficiency. After processing all documents, the algorithm returns the highest score combination S^* along with its trajectory optimal_state.

We can extract the partitioned paper sets D_{v_t} from the trajectory optimal_state and generate the topic facet x_{v_t} with LLM as follows:

$$\begin{aligned} &\textit{For all } (a,i) \in S^*,\ t \in \{1,\ldots,k_v\} \textit{ in parallel}: \\ &\quad D_{v_t} = \{d \mid \forall d \in \texttt{optimal\_state}[(a,i)]\} \\ &\quad x_{v_t} \sim p_{\text{LLM}}(x|v, D_{v_t}, S^*) \\ &\quad v_t \triangleq \langle x_{v_t}, D_{v_t} \rangle,\ E^{(l)} \leftarrow E^{(l)} \cup \{(v, v_t)\}, \end{aligned} \tag{8}$$

where the node v_t is connected to its parent v.

3.3 Iterative Structure Generation

As illustrated in Figure 2, our method constructs the taxonomy in a top-down manner, starting from the root node r and iteratively expanding the child nodes Child(v) for node v from each depth l, this is decomposing the associated paper set D_v and generating a corresponding topic facet x_v that characterizes the semantic focus of its substructure.

During each expansion step, we dynamically generate new aspects based on the current distribution of the papers in D_v. This process is tailored to capture the updated salient semantic dimensions and key distinctions among papers within the new partitioned subset. It is worth noting that we incorporate the topic facets of all ancestor nodes into the prompt context. This ensures that the newly generated aspects reflect not only local document features, but also the global structural direction of the taxonomy, thereby better understanding the direction in which the current node needs to be expanded. The expansion process continues until a stopping condition is met, such as reaching a maximum depth L or encountering the number of papers in the node below a predefined threshold. Once the expansion is complete, the resulting tree constitutes the taxonomy of given topic and papers.

4 Experiments

4.1 Baselines

We compare our approach with two categories of methods: pure LLM-based and clustering-incorporated taxonomy generation.

4.1.1 Pure LLM-based Methods

CHIME (Hsu et al., 2024) extracts claims and frequent entities from related papers, then prompts an LLM to generate root categories and assign claims into a hierarchical structure.

TNT-LLM (Wan et al., 2024) first prompts an LLM to summarize each input, then iteratively constructs and refines a taxonomy from the summaries.

GoalEx (Wang et al., 2023b) generates explanation-based candidate clusters given a goal, and assigns each document via entailment prompting. A integer linear programming step selects clusters that best cover the dataset with minimal redundancy.

4.1.2 Clustering-incorporated Methods

Knowledge Navigator (Katz et al., 2024) encodes paper abstracts into dense embeddings and applies traditional clustering algorithms to group them. The resulting clusters are named and organized into a hierarchical structure by LLM.

SCYCHIC (Gao et al., 2025) uses an LLM to extract structured contributions from each paper, which are then embedded and clustered hierarchically. A bidirectional clustering algorithm specifies the number of levels and clusters per level.

4.2 Experimental Settings

We employ GPT-4o (2024-08-06) for aspect generation (eq. 3) and topic facet generation (eq. 8), due to its superior reasoning and abstraction capabilities. Besides, we use LLaMA-3.1-8B to generate aspect-guided summaries (eq. 4), as it requires less complex reasoning to locate and extract relevant information from the paper. This division enables a balance between generation quality and computational cost across the pipeline.

	Categorization			Structure		Nodes	Human Assessment				
	NMI	ARI	Purity	CEDS	HSR		Cov.	Rel.	Str.	Val.	Ade.
Pure LLM-based											
CHIME	35.4	0.9	41.8	23.3	**74.7**	1.1	43.2	50.3	54.5	47.6	47.6
TnT-LLM	51.6	2.3	57.6	19.1	69.9	1.5	41.1	47.3	48.1	46.0	46.6
GoalEx	46.7	8.8	47.6	23.2	70.5	1.0	45.9	53.3	57.0	48.6	46.8
Clustering-incorporated											
KN	44.7	16.2	42.4	18.8	49.5	0.5	47.5	57.0	55.0	52.0	47.0
SCYCHIC	49.8	9.0	50.6	23.0	66.4	1.5	47.3	50.7	55.2	48.4	46.8
Ours	**60.1**	**19.1**	**62.2**	**23.8**	74.5	1.2	**50.6**	**57.1**	**59.6**	**52.9**	**54.4**

Table 2: Automatic and human evaluation results on taxonomy generation. We report categorization quality (**NMI**, **ARI**, **Purity**), structural consistency (**CEDS**, **HSR**), and normalized node count (**Nodes**), where 1.0 of **Nodes** indicates an exact match with the gold taxonomy in terms of node count. Human evaluation is conducted on five dimensions, **Cov**erage, **Rel**evance, **Str**ucture, **Val**idity, and **Ade**quacy, each rated on a scale of 1 to 100.

Following Katz et al. (2024), we adopt text-embedding-3-large for paper encoding (eq. 4) and use Gaussian Mixture Models (GMMs) as the aspect-specific clustering model $f_a(e, i)$ (eq. 5). In the main experiments, the number of clusters per aspect k and the number of child nodes per parent node k_v are both empirically set as 4. The maximum taxonomy depth is limited to $L = 3$. See the prompts that we use in the Appendix C. Due to computational and manual costs, we randomly sample 25 of the 156 taxonomy instances for human evaluation and ablation studies. Each configuration was executed once with a fixed random seed, and results are averaged over the sampled instances.

4.3 Evaluation Metrics

We evaluate taxonomy generation from two complementary perspectives: *papers categorization* and *topic structure*, using both automatic and human evaluation. Full metric definitions and annotation guidelines are provided in Appendix A.

Automatic Evaluation. To assess papers categorization, we report three widely used clustering metrics: Normalized Mutual Information (**NMI**), Adjusted Rand Index (**ARI**), and **Purity**. For topic quality and structural alignment, we adopt Heading Soft Recall (**HSR**) (Fränti and Mariescu-Istodor, 2023) and Catalogue Edit Distance Similarity (**CEDS**) (Zhu et al., 2023). In addition, we use a normalized **Nodes Ratio**, defined as the number of generated nodes divided by the number of nodes in the oracle taxonomy, as an auxiliary metric to monitor coarse-grained structural discrepancies.

Human Evaluation. Following Hu et al. (2024), we conduct human evaluation on five dimensions: **Coverage**, **Relevance**, **Structure**, **Validity**, and **Adequacy**. Each dimension is rated on a scale of 1 to 100 to allow fine-grained comparisons. The evaluation is performed by six reviewers: three PhD students in computer science and three advanced LLMs: GPT-4o (2024-11-20), Claude 3.7 Sonnet (2025-02-19), and LLaMA-3.3-70B Instruct.

4.4 Main Results

Best categorization performance. We obtain the best categorization performance, with NMI (60.1), ARI (19.1), and Purity (62.2), surpassing both pure LLM-based baselines (*e.g.*, TnT-LLM with NMI of 51.6) and clustering-incorporated baselines (*e.g.*, KN with ARI of 16.2). This proves the superiority of our multi-aspect framework in producing more coherent and well-separated clusters, offering a more reliable foundation for semantic organization.

Superior structure alignment. We achieve the highest CEDS score of 23.8, indicating strong structural consistency with oracle taxonomies. The HSR score of 74.5 confirms that our method possesses the ability to recover coherent hierarchical relations. In addition, the node ratio of 1.2 suggests a balanced taxonomy size, avoiding the situation of both over-fragmentation and under-segmentation.

Preferred by human evaluators. As shown in Table 2, our method receives the highest human evaluation scores in all five dimensions, with notable improvements in **Coverage** (50.6), **Structure** (59.6), and **Adequacy** (54.4). This indicates that our generated taxonomies cover more comprehensive contents and exhibit a more coherent organization of the structure, thereby enhancing the usability. The agreement between the annotators measured by Fleiss's Kappa on discretized scores (converted from a scale of 1 to 100 to a scale of

	Categorization			Structure	
	NMI	ARI	Purity	CEDS	HSR
Dynamic Aspects					
Search	57.8	20.1	66.4	23.7	69.9
Prune	58.6	20.4	66.0	23.9	69.4
Fixed Aspects					
Search	55.2	19.5	62.4	25.8	68.6
Prune	55.0	19.7	60.7	25.4	66.5
Abstract	57.1	22.3	64.3	24.2	66.3

Table 3: Ablation results on aspect generation and dynamic search. "Dynamic Aspects" means our dynamic aspect generation process, while "Fixed Aspects" is using fixed manual aspects. "Search" denotes dynamic clusters search and "Prune" is the pruning strategy in the search process. "Abstract" means only using the paper abstracts without aspect guidance.

5 points) is 0.24, indicating moderate consistency among the evaluators.

4.5 Ablation Study on Aspect Generation and Dynamic Search

We conduct an ablation study to examine the impact of aspect generation methods and clustering strategies on taxonomy quality in Table 3.

Dynamic *v.s.* Fixed Aspects. We first compare our proposed dynamic aspect generation (*Dynamic Aspects*) with a manually defined aspect template shared across all paper sets (*Fixed Aspects*). The results show that the dynamic aspects achieve consistently better performance in both categorization (*e.g.*, NMI 57.8 *v.s.* 55.2) and structural alignment (*e.g.*, HSR 69.9 *v.s.* 68.6). This highlights the benefit of tailoring semantic dimensions to each paper set, which better captures latent topical variations and improves clustering quality.

Full *v.s.* Pruning Search. Within each setting, we compare two clustering strategies: Full Search and Pruning Search. For the fixed-aspect setting, pruning significantly reduces categorization and structure performance, indicating that simple greedy filtering may break high-quality groupings formed under strong human priors. In contrast, under the dynamic aspect setting, pruning yields comparable performance to full dynamic search. This suggests that while LLM-generated aspects offer higher representational flexibility, they also introduce variability and redundancy, where pruning can help remove outliers with little degradation.

Effect of Using Abstracts Only. Finally, we include a baseline that uses only abstracts of papers without aspects. Although it performs reasonably well in ARI (22.3), its overall categorization and structure scores remain lower than our full model. This underscores the importance of aspect-guided representation beyond manual summarization.

k_v	Categorization			Structure		Nodes
	NMI	ARI	Purity	CEDS	HSR	
3	55.1	21.7	60.2	24.6	63.7	1.1
4	57.6	19.5	65.2	24.3	69.3	1.4
5	59.0	18.9	69.5	20.4	68.9	1.6
6	61.2	18.2	73.6	19.9	69.9	1.9
S	56.2	21.5	62.2	23.9	66.0	1.1

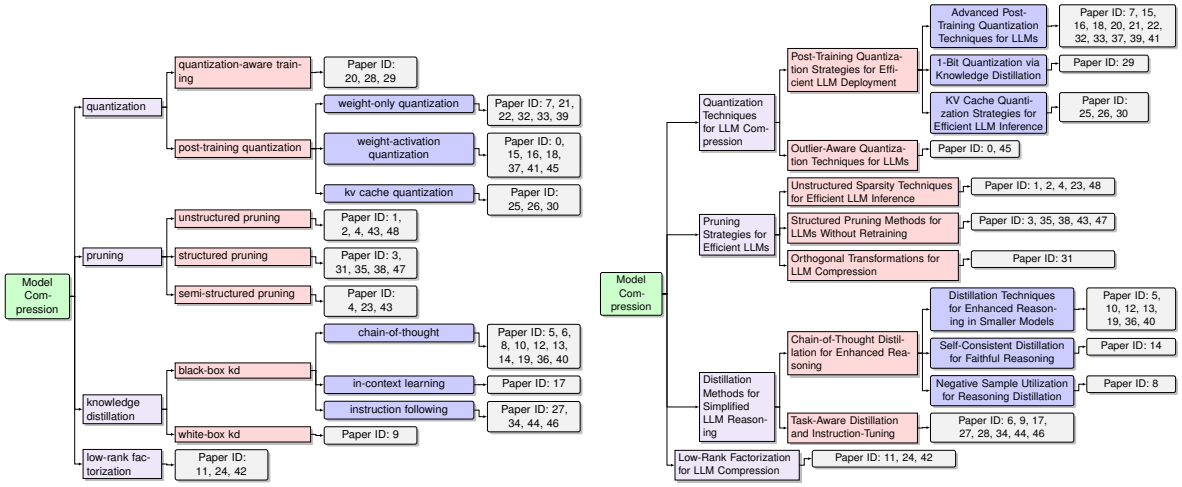
Table 4: Performance under different values of hyperparameter k_v, which controls the number of clusters per node. "S" denotes an adaptive selection strategy from our baseline. Fixed larger k_v improves purity but harms structural consistency (CEDS and Nodes), while adaptive k_v achieves a balanced yet unremarkable performance across all metrics.

4.6 Effect of Hyperparameter k_v

We analyze the influence of the hyperparameter k_v, which controls the number of clusters generated at each node during hierarchical taxonomy construction. Table 4 reports the results under fixed values of $k_v \in \{3, 4, 5, 6\}$, as well as an adaptive strategy ("S") (Katz et al., 2024) where the model dynamically selects from $3, 4, 5, 6$ based on the clustering result with the highest silhouette score.

Fixed *v.s.* Adaptive k_v. As k_v increases, we observe a steady improvement in categorization performance, with NMI rising from 55.1 (at $k_v = 3$) to 61.2 (at $k_v = 6$). Purity also increases substantially, reflecting finer-grained clustering. However, this comes at the cost of structural quality: CEDS decline and the normalized node count (Nodes) increase, indicating over-fragmented taxonomies with reduced alignment to the gold standard.

The adaptive strategy achieves relatively balanced performance across all metrics rather than a significant improvement in any individual metric (NMI 56.2, ARI 21.5, CEDS 23.9). Moreover, the adaptive strategy requires repeated clustering operations for all k_v, resulting in substantially higher computational overhead. Coupled with only marginal improvements, the high cost suggests that silhouette-based selection may offer limited practical benefit in taxonomy generation.



(a) Built by Zhu et al. (2024)

(b) Generated by our method

Figure 3: Taxonomy of "Model Compression methods for Large Language Models".

4.7 Robustness to Noisy Inputs

To evaluate the robustness of our method under more realistic settings, where the initial set of relevant papers is not perfectly curated, we conducted additional experiments simulating noisy input conditions, as suggested by Reviewer RHVv. Specifically, we injected 5%–30% unrelated papers into the curated dataset to mimic potential noise introduced by retrieval-based pipelines.

	Noise Ratio	Categorization NMI	ARI	Purity	Structure CEDS	HSR	Nodes
TnT-LLM	0%	43.76	3.61	56.16	20.06	72.15	1.66
	5%	44.68	3.91	56.26	20.87	75.48	1.92
	10%	50.73	3.88	63.45	15.88	81.57	2.36
	20%	41.25	1.90	52.42	17.71	72.58	1.79
	30%	43.55	3.43	55.12	19.10	76.54	2.20
SCYCHIC	0%	39.37	6.25	45.29	17.36	60.52	1.73
	5%	38.33	5.89	44.95	16.92	61.47	1.73
	10%	37.22	5.18	46.20	15.42	63.10	1.73
	20%	37.62	5.59	43.70	17.78	62.26	1.73
	30%	36.47	6.49	46.89	17.19	63.58	1.78
Ours	0%	53.80	13.94	61.93	23.99	68.28	1.48
	5%	54.46	15.74	62.93	23.15	69.30	1.49
	10%	52.93	14.02	61.05	23.22	70.22	1.65
	20%	54.17	16.54	61.33	23.72	70.69	1.73
	30%	54.42	17.53	61.64	21.66	72.14	1.80

Table 5: Performance comparison under different noise levels.

Experimental results (see Table 5) show that our method consistently outperforms baseline approaches and maintains superior performance and structural stability across all noise levels. In contrast, TnT-LLM suffers from significant performance fluctuations, and SCYCHIC experiences moderate degradation.

We attribute this robustness to two key design choices in our framework: Aspect-aware clustering with dynamic search, which selectively identifies the most relevant combination of aspect dimensions for each paper, effectively filtering out noise; Expanded representation space of aspect-cluster combinations, which allows noisy or outlier papers to be isolated into peripheral nodes without disrupting the core taxonomy structure.

These findings highlight the error-tolerant nature of our approach and demonstrate its effectiveness even when applied to noisy, less curated document sets. We believe this provides strong evidence of the method's practical applicability beyond oracle-like experimental conditions.

4.8 Case-Study

Comparison with Human-Annotated Taxonomy. Figure 3(a) shows the human-annotated taxonomy from Zhu et al. (2024) on "Model Compression Methods for Large Language Models," and Figure 3(b) presents our generated result. For comparison, additional case studies produced by baseline methods are included in the Appendix B.1 At the top level, both taxonomies adopt a method-based categorization (*e.g.*, quantization, pruning, distillation), which is largely consistent. Only one paper (28) is misclassified. In deeper layers, our taxonomy introduces more fine-grained and diverse subtopics. While these differ from the human taxonomy, they reflect alternative yet valid grouping strategies based on implementation details or use cases. This highlights the subjectivity of deeper-level structuring and the model's ability to surface meaningful semantic distinctions.

5 Related Work

Organizing the ever-growing scientific literature into coherent, hierarchical categories remains a core challenge in scholar knowledge management. Traditional approaches typically rely on manually curated taxonomies, where each paper is mapped to one or more predefined categories within a multi-level hierarchy (Zhang et al., 2021; Sadat and Caragea, 2022; Rao et al., 2023).

Recent advances in LLMs have significantly reshaped the landscape of topic modeling and document clustering by semantically rich and context-aware representations, allowing for more interpretable and scalable taxonomy construction (Zhang et al., 2023; Pham et al., 2024; Wang et al., 2023a; Qiu et al., 2024; Viswanathan et al., 2024). In general, there are two technical paradigms for taxonomy construction: **classification-based** and **clustering-based**, where each of them offers distinct advantages and trade-offs.

In the classification paradigm, an LLM first induces a taxonomy, and papers are subsequently assigned (Pham et al., 2024). CHIME (Hsu et al., 2024) produces the taxonomy and assigns papers in one pass. GoalEx (Wang et al., 2023b) aligns LLM-generated explanations with papers and applies integer linear programming to finalize a non-overlapping set of assignments. To better handle long-document settings, TnT-LLM (Wan et al., 2024) iteratively generates and updates the label taxonomy. More recently, TaxoAdapt (Kargupta et al., 2025) incrementally expands the taxonomy by analyzing papers one by one, formulating multi-dimensional taxonomy construction as iterative multi-label classification. Automatic literature review generation pipelines such as AutoSurvey (Wang et al., 2024), Storm (Shao et al., 2024), and SurveyForge (Yan et al., 2025) replace taxonomies with hierarchically structured outlines, i.e., they first draft an outline of target topics and then retrieve and attach relevant papers to each entry.

Despite their flexibility, label-first pipelines often produce redundant labels, hallucinated or missing categories, and imbalanced hierarchies. **Clustering-based methods** organize papers in the representation space, then leverage inter-paper relations to enforce global coherence and balance, and finally add labels respectively. A related line integrates clustering with LLM generation, where papers are first grouped by unsupervised methods and then semantic labels are produced for each cluster (Diaz-Rodriguez, 2025; Hu et al., 2024). Knowledge Navigator (Katz et al., 2024) performs single-stage flat clustering, while Gao et al. (2025) explore hierarchical strategies (bottom-up, top-down, and bi-direction). However, these approaches often rely on local, per-cluster descriptions in isolation, yielding redundant or inconsistent labels due to missing global context and weak structural constraints. In contrast, our method proposes dynamic and structure-aware hierarchical clustering with global aspects, maintaining the semantic distinctiveness and structural fidelity of the taxonomies.

6 Conclusion

In this work, we propose a novel framework for taxonomy generation that leverages multi-dimensional representations and dynamic clustering. By dynamically generating semantic aspects tailored to each document set and searching for optimal clustering configurations via dynamic search, our method constructs taxonomies that are both semantically coherent and structurally faithful. We further introduce a high-quality benchmark of 156 annotated taxonomies derived from CS survey papers to facilitate reliable evaluation. Extensive experiments demonstrate that our approach outperforms existing pure LLM-based and clustering-incorporated methods in both automatic and human evaluations. Ablation studies confirm the effectiveness of dynamic aspect modeling and adaptive clustering strategies.

Limitations

Although our method demonstrates strong performance, several limitations remain:

1. In practical applications, the system must first retrieve candidate papers from a broad and potentially noisy corpus, which introduces additional challenges such as incomplete coverage, irrelevant documents, and retrieval errors. Our framework focuses on a controllable experimental environment with oracle papers. Developing retrieval-integrated taxonomy construction methods that are robust to these issues constitutes an important direction for future work.

2. The quality of aspect extraction and summarization depends on the capabilities of the underlying LLM, which affects the generalization of our framework.

3. The combination of multi-aspect encoding and iterative clustering introduces computational overhead, which may limit scalability to very large corpora. We plan to explore more efficient clustering strategies and scalable approximations to support deployment on a greater scale.

4. Our evaluation benchmark focuses on survey papers in computer science, where its applicability to other domains or less-structured corpora remains to be explored. In future work, we will extend our framework to cross-domain settings.

5. We find that silhouette-based k-selection is not well suited for clustering in complex and semantic-driven tasks such as taxonomy generation, which leaves the development of more effective task-specific clustering selection strategies for future work.

6. Our current framework employs hierarchical clustering, which enforces a strict, non-overlapping partitioning of papers at each level. In contrast, expert-authored taxonomies (e.g., the oracle trees in our benchmark) sometimes allow a paper to be assigned to multiple branches. Enabling multi-label taxonomy construction is thus an important and challenging extension that we leave for future research.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) (grant 62276078, U22B2059), the Key R&D Program of Heilongjiang via grant 2022ZX01A32, and the Fundamental Research Funds for the Central Universities (XNJKKGYDJ2024013). It was also supported by the Ministry of Education, Singapore, under its AcRF Tier 2 Funding (Proposal ID: T2EP20123-0052). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Ministry of Education, Singapore. We thank the iFLYTEK Spark AI Assistant Team for providing application requirements and high-value feedback.

Ethics Statement

This work focuses on constructing paper taxonomies using large language models (LLMs), with the goal of assisting researchers and beginners in understanding domain knowledge, tracking research trends, and improving reading efficiency. While this technology has the potential to support scientific discovery and education, it also carries risks that warrant ethical consideration.

Use of LLMs and Potential Risks Our framework relies on LLMs to generate semantic aspects and organize papers into a hierarchical taxonomy. We acknowledge that LLMs are susceptible to hallucinations, which may lead to factually incorrect or misleading taxonomy structures. Nevertheless, any downstream use of the generated taxonomy for scientific analysis or educational purposes should be critically verified, especially in high-stakes or sensitive applications.

Dataset Collection and Licensing We construct our dataset using publicly available metadata and content from arXiv and Semantic Scholar, both of which provide research access under open licenses. The dataset used in this study includes paper titles, metadata (e.g., authors, publication years), and taxonomy structures extracted from the LaTeX source files of review papers collected from arXiv. Specifically, we target survey papers that explicitly include taxonomy structures in their source files. From these files, we extract the taxonomy tree as well as the titles of cited papers mentioned within the taxonomy.

For each cited paper in the taxonomy, we obtain its metadata using the Semantic Scholar API. In cases where the cited papers are also publicly available on arXiv, we further retrieve their LaTeX source files and extract their `Introduction` sections. This allows us to enrich the representation of each paper beyond the abstract and metadata, enabling more informed and semantically grounded taxonomy construction.

All data were obtained through open APIs and publicly accessible sources, and their use is restricted to academic research. We confirm that our use of these artifacts complies with their intended use and access conditions. No redistribution of full-text content outside permitted use cases has been conducted. The resulting dataset, including derived taxonomy annotations, is shared under a research-only license and should not be repurposed for commercial or non-academic use.

Privacy and Anonymization We conducted a manual check to ensure that the dataset does not contain personally identifiable information (PII) beyond standard academic author metadata, which

are already publicly accessible through the original platforms. No sensitive personal content, user-generated data, or non-consensual information is included. Our system does not process or generate user data, and all derived outputs (e.g., cluster labels, taxonomy facets) are generated from published research papers.

Human Annotation and Consent We recruited voluntary annotators to evaluate the quality of the generated taxonomies. All annotators were fully informed about the purpose of the study, the nature of the data, and how their assessments would be used. No personal information was collected from annotators, and consent was obtained prior to their participation.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jairo Diaz-Rodriguez. 2025. k-llmmeans: Summaries as centroids for interpretable and scalable llm-based text clustering. *arXiv preprint arXiv:2502.09667*.

Pasi Fränti and Radu Mariescu-Istodor. 2023. Soft precision and recall. *Pattern Recognition Letters*, 167:115–121.

Muhan Gao, Jash Shah, Weiqi Wang, and Daniel Khashabi. 2025. Science Hierarchography: Hierarchical Organization of Science Literature. *Preprint*, arXiv:2504.13834.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Mark A. Hanson, Pablo Gómez Barreiro, Paolo Crosetto, and Dan Brockington. 2024. The strain on scientific publishing. *Quantitative Science Studies*, 5(4):823–843.

Chao-Chun Hsu, Erin Bransom, Jenna Sparks, Bailey Kuehl, Chenhao Tan, David Wadden, Lucy Lu Wang, and Aakanksha Naik. 2024. Chime: Llm-assisted hierarchical organization of scientific studies for literature review support. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 118–132.

Yuntong Hu, Zhuofeng Li, Zheng Zhang, Chen Ling, Raasikh Kanjiani, Boxin Zhao, and Liang Zhao. 2024. Taxonomy tree generation from citation graph. *arXiv preprint arXiv:2410.03761*.

Yuntong Hu, Zhuofeng Li, Zheng Zhang, Chen Ling, Raasikh Kanjiani, Boxin Zhao, and Liang Zhao. 2025. Taxonomy Tree Generation from Citation Graph. *Preprint*, arXiv:2410.03761.

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)*, pages 29834–29850, Vienna, Austria. Association for Computational Linguistics.

Uri Katz, Mosh Levy, and Yoav Goldberg. 2024. Knowledge navigator: LLM-guided browsing framework for exploratory search in scientific literature. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8838–8855, Miami, Florida, USA. Association for Computational Linguistics.

Chau Pham, Alexander Hoyle, Simeng Sun, Philip Resnik, and Mohit Iyyer. 2024. Topicgpt: A prompt-based topic modeling framework. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2956–2984.

Mingjie Qiu, Wenzhong Yang, Fuyuan Wei, and Mingliang Chen. 2024. A topic modeling based on prompt learning. *Electronics*, 13(16):3212.

Susie Xi Rao, Peter H Egger, and Ce Zhang. 2023. Hierarchical classification of research fields in the" web of science" using deep learning. *arXiv preprint arXiv:2302.00390*.

Niklas Reisz, Vito D P Servedio, Vittorio Loreto, William Schueller, Márcia R Ferreira, and Stefan Thurner. 2022. Loss of sustainability in scientific work. *New Journal of Physics*, 24(5):053041.

Mobashir Sadat and Cornelia Caragea. 2022. Hierarchical multi-label classification of scientific documents. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8923–8937, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jingbo Shang, Xinyang Zhang, Liyuan Liu, Sha Li, and Jiawei Han. 2020. NetTaxo: Automated Topic Taxonomy Construction from Text-Rich Network. In *Proceedings of The Web Conference 2020*, pages 1908–1919, Taipei Taiwan. ACM.

Yijia Shao, Yucheng Jiang, Theodore Kanell, Peter Xu, Omar Khattab, and Monica Lam. 2024. Assisting in writing Wikipedia-like articles from scratch with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6252–6278, Mexico City, Mexico. Association for Computational Linguistics.

Jiaming Shen, Zeqiu Wu, Dongming Lei, Chao Zhang, Xiang Ren, Michelle T. Vanni, Brian M. Sadler, and Jiawei Han. 2018. HiExpan: Task-Guided Taxonomy Construction by Hierarchical Tree Expansion. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2180–2189, London United Kingdom. ACM.

Paolo Vineis. 2024. Scientific publishing: crisis, challenges, and new opportunities. *Frontiers in Public Health*, 12:1417019.

Vijay Viswanathan, Kiril Gashteovski, Kiril Gashteovski, Carolin Lawrence, Tongshuang Wu, and Graham Neubig. 2024. Large Language Models Enable Few-Shot Clustering. *Transactions of the Association for Computational Linguistics*, 12:321–333.

Mengting Wan, Tara Safavi, Sujay Kumar Jauhar, Yujin Kim, Scott Counts, Jennifer Neville, Siddharth Suri, Chirag Shah, Ryen W. White, Longqi Yang, Reid Andersen, Georg Buscher, Dhruv Joshi, and Nagu Rangan. 2024. TnT-LLM: Text Mining at Scale with Large Language Models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5836–5847, Barcelona Spain. ACM.

Han Wang, Nirmalendu Prakash, Nguyen Khoi Hoang, Ming Shan Hee, Usman Naseem, and Roy Ka-Wei Lee. 2023a. Prompting large language models for topic modeling. In *2023 IEEE International Conference on Big Data (BigData)*, pages 1236–1241. IEEE.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, Wei Ye, Shikun Zhang, and Yue Zhang. 2024. Autosurvey: Large language models can automatically write surveys. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zihan Wang, Jingbo Shang, and Ruiqi Zhong. 2023b. Goal-driven explainable clustering via language descriptions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10626–10649.

Xiangchao Yan, Shiyang Feng, Jiakang Yuan, Renqiu Xia, Bin Wang, Lei Bai, and Bo Zhang. 2025. SURVEYFORGE : On the outline heuristics, memory-driven generation, and multi-dimensional evaluation for automated survey writing. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12444–12465, Vienna, Austria. Association for Computational Linguistics.

Chao Zhang, Fangbo Tao, Xiusi 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*, pages 2701–2709, London United Kingdom. ACM.

Yu Zhang, Xiusi Chen, Yu Meng, and Jiawei Han. 2021. Hierarchical metadata-aware document categorization under weak supervision. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 770–778.

Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. Clusterllm: Large language models as a guide for text clustering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13903–13920.

Kun Zhu, Xiaocheng Feng, Xiachong Feng, Yingsheng Wu, and Bing Qin. 2023. Hierarchical catalogue generation for literature review: A benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6790–6804, Singapore. Association for Computational Linguistics.

Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. 2024. A survey on model compression for large language models. *Transactions of the Association for Computational Linguistics*, 12:1556–1577.

A Evaluation Metrics

We evaluate taxonomy generation from two complementary perspectives: *clustering structure* and *heading quality*. In addition to automatic evaluation, we also conduct human evaluation to assess the practical quality of the generated taxonomies.

A.1 Clustering Evaluation

Hierarchical Mutual Information (HMI) extends mutual information to hierarchical structures by evaluating consistency across multiple levels of the taxonomy. It provides a structure-aware measure that rewards alignment not only at the leaf level but also across internal nodes.

Adjusted Rand Index (ARI) measures the agreement between the predicted and gold cluster assignments, correcting for random chance. It is widely used in clustering evaluation and is robust to varying cluster sizes.

Purity quantifies the extent to which each predicted cluster contains documents from a single ground-truth category. While intuitive, this metric may favor solutions with a large number of small clusters.

A.2 Heading Evaluation

Heading Soft Recall. We follow the calculation of Shao et al. (2024). This metric measures the proportion of ground-truth headings that are approximately matched by generated node names using soft string similarity. It allows for minor lexical variations and captures semantic overlap. It is worth noting that, in theory, longer generated outputs tend to achieve higher scores under soft matching metrics such as Soft Heading Recall. This is because longer outputs are more likely to semantically overlap with the reference headings, thereby increasing the chance of a successful match under relaxed similarity thresholds. However, this improvement may not necessarily reflect better quality, as it can be attributed to over-generation rather than more accurate content selection.

Catalogue Edit Distance Similarity (CEDS) (Zhu et al., 2023) evaluates the overall similarity between the generated taxonomy and the gold taxonomy by computing a **normalized tree edit distance**. It accounts for both structural alignment (e.g., insertion, deletion, reordering of nodes) and heading-level similarity, offering a holistic assessment of taxonomy quality.

A.3 Human Evaluation

To complement automatic metrics, we conduct a human evaluation based on five criteria followed Hu et al. (2025):

- **Coverage**: Does the taxonomy comprehensively cover the major themes and subtopics within the document collection?
- **Relevance**: Are the identified categories appropriate and meaningful for the given set of documents?
- **Structure**: Is the overall organization coherent and logically structured as a hierarchy?
- **Usefulness**: How helpful is the taxonomy for readers trying to understand or navigate the domain?
- **Validity**: Does the taxonomy align with expert expectations or established domain knowledge?

Each aspect is rated on a scale of 1 to 100 by multiple annotators with relevant domain expertise, and the final scores are averaged among the raters. To link the evaluation protocol with concrete outcomes, we further analyze inter-rater reliability by discretizing the scores into five bins of equal width and computing consistency both within and across rater groups. Inter-annotator agreement, measured by Fleiss' κ on the discretized ratings, shows the following: Human–Human $= 0.31$, LLM–LLM $= 0.38$, and Human–LLM $= 0.24$. Taken together, these results indicate that both human annotators and LLMs exhibit comparable levels of consistency within the group, while their agreement between groups remains relatively low, suggesting systematic differences in rating behavior between the two.

B Case Study

To qualitatively evaluate the effectiveness of our method, we conduct a case study on the topic of "Model Compression".

B.1 Taxonomy Trees

Figures 5(a) and 4 show the human-authored taxonomy tree and the corresponding set of papers from the survey paper "A Survey on Model Compression for Large Language Models" (Zhu et al., 2024). Our generated taxonomy is presented in Figure 5(b), while the taxonomies produced by other

A Survey on Model Compression for Large Language Models

[0] OliVe: Accelerating Large Language Models via Hardware-friendly Outlier-Victim Pair Quantization
[1] Flash-LLM: Enabling Low-Cost and Highly-Efficient Large Generative Model Inference With Unstructured Sparsity
[2] One-Shot Sensitivity-Aware Mixed Sparsity Pruning for Large Language Models
[3] Fluctuation-based Adaptive Structured Pruning for Large Language Models
[4] SparseGPT: Massive Language Models Can Be Accurately Pruned in One-Shot
[5] Large Language Models Are Reasoning Teachers
[6] Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes
[7] OWQ: Outlier-Aware Weight Quantization for Efficient Fine-Tuning and Inference of Large Language Models
[8] Turning Dust into Gold: Distilling Complex Reasoning Capabilities from LLMs by Leveraging Negative Data
[9] Less is More: Task-aware Layer-wise Distillation for Language Model Compression
[10] Teaching Small Language Models to Reason
[11] Matrix Compression via Randomized Low Rank and Low Precision Factorization
[12] Distilling Reasoning Capabilities into Smaller Language Models
[13] Democratizing Reasoning Ability: Tailored Learning from Large Language Model
[14] SCOTT: Self-Consistent Chain-of-Thought Distillation
[15] SmoothQuant: Accurate and Efficient Post-Training Quantization for Large Language Models
[16] ZeroQuant: Efficient and Affordable Post-Training Quantization for Large-Scale Transformers
[17] In-context Learning Distillation: Transferring Few-shot Learning Ability of Pre-trained Language Models
[18] RPTQ: Reorder-based Post-training Quantization for Large Language Models
[19] PaD: Program-aided Distillation Can Teach Small Models Reasoning Better than Chain-of-thought Fine-tuning
[20] LLM-QAT: Data-Free Quantization Aware Training for Large Language Models
[21] AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration
[22] SqueezeLLM: Dense-and-Sparse Quantization
[23] E-Sparse: Boosting the Large Language Model Inference through Entropy-based N: M Sparsity
[24] ASVD: Activation-aware Singular Value Decomposition for Compressing Large Language Models
[25] KVQuant: Towards 10 Million Context Length LLM Inference with KV Cache Quantization
[26] KIVI: A Tuning-Free Asymmetric 2bit Quantization for KV Cache
[27] Selective Reflection-Tuning: Student-Selected Data Recycling for LLM Instruction-Tuning
[28] BitDistiller: Unleashing the Potential of Sub-4-Bit LLMs via Self-Distillation
[29] OneBit: Towards Extremely Low-bit Large Language Models
[30] WKVQuant: Quantizing Weight and Key/Value Cache for Large Language Models Gains More
[31] SliceGPT: Compress Large Language Models by Deleting Rows and Columns
[32] QuIP: 2-Bit Quantization of Large Language Models With Guarantees
[33] SpQR: A Sparse-Quantized Representation for Near-Lossless LLM Weight Compression
[34] Lion: Adversarial Distillation of Proprietary Large Language Models
[35] Shortened LLaMA: A Simple Depth Pruning for Large Language Models
[36] Explanations from Large Language Models Make Small Reasoners Better
[37] LLM-FP4: 4-Bit Floating-Point Quantized Transformers
[38] LLM-Pruner: On the Structural Pruning of Large Language Models
[39] LUT-GEMM: Quantized Matrix Multiplication based on LUTs for Efficient Inference in Large-Scale Generative Language Models
[40] Specializing Smaller Language Models towards Multi-Step Reasoning
[41] OmniQuant: Omnidirectionally Calibrated Quantization for Large Language Models
[42] The Truth is in There: Improving Reasoning in Language Models with Layer-Selective Rank Reduction
[43] A Simple and Effective Pruning Approach for Large Language Models
[44] Self-Instruct: Aligning Language Models with Self-Generated Instructions
[45] Outlier Suppression+: Accurate quantization of large language models by equivalent and effective shifting and scaling
[46] LaMini-LM: A Diverse Herd of Distilled Models from Large-Scale Instructions
[47] Sheared LLaMA: Accelerating Language Model Pre-training via Structured Pruning
[48] Dynamic Sparse No Training: Training-Free Fine-tuning for Sparse LLMs

Figure 4: Papers in the taxonomy built by Zhu et al. (2024)

baseline methods are shown in Figures 6–10. As illustrated, our method produces a more coherent and semantically meaningful taxonomy structure, with clearer topic hierarchies and better alignment to the source papers, compared to other approaches.

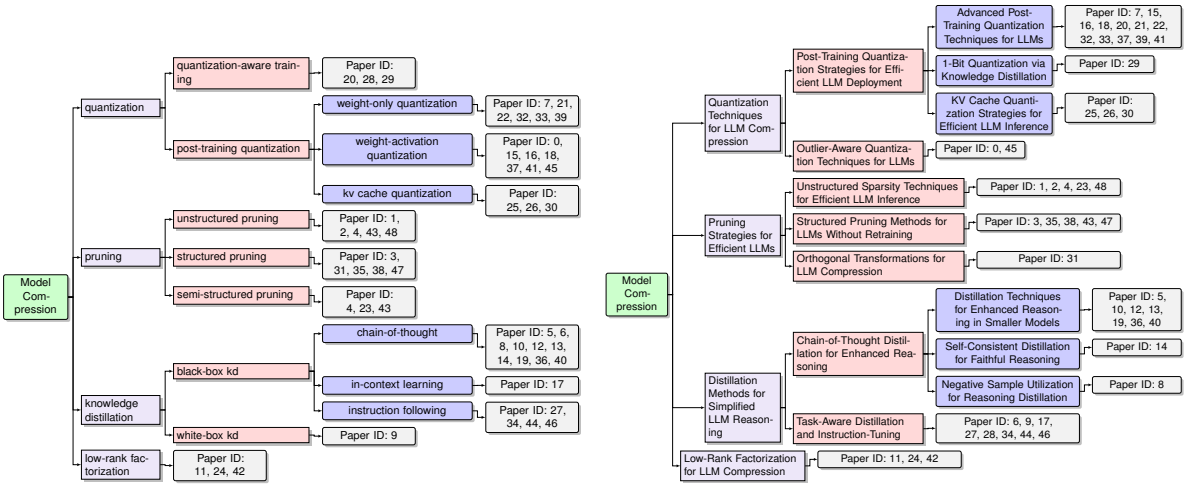
B.2 Generation Process

To complement the quantitative results in Table 3, we provide several representative case studies that qualitatively illustrate the role of aspect generation, aspect-guided summarization, and facet identification. These examples highlight how different components contribute to clustering outcomes and taxonomy construction.

Figure 11 shows the aspects generated under the topic *"Model Compression → Quantization Techniques for LLM Compression"*. The resulting aspects capture salient semantic dimensions that effectively characterize and differentiate relevant papers (e.g., Quantization Framework Type, Hardware Efficiency Techniques).

These aspects are then used in parallel to guide the encoding of individual papers. Figure 12 presents the aspect-guided summary for the paper *"Flash-LLM: Enabling Low-Cost and Highly-Efficient Large Generative Model Inference With Unstructured Sparsity"*. Compared with the original abstract, the aspect-based summary selectively foregrounds details aligned with the identified aspects, facilitating clearer alignment with clustering.

Figure 13 illustrates how facets are identified within the topic *"Model Compression"* and the selected aspects after clustering with dynamic search. The system generates corresponding topic facets that summarize the semantic focus of each substructure and render the resulting taxonomy more interpretable and navigable.



(a) Built by Zhu et al. (2024)

(b) Generated by our method

Figure 5: Taxonomy of "Model Compression methods for Large Language Models".

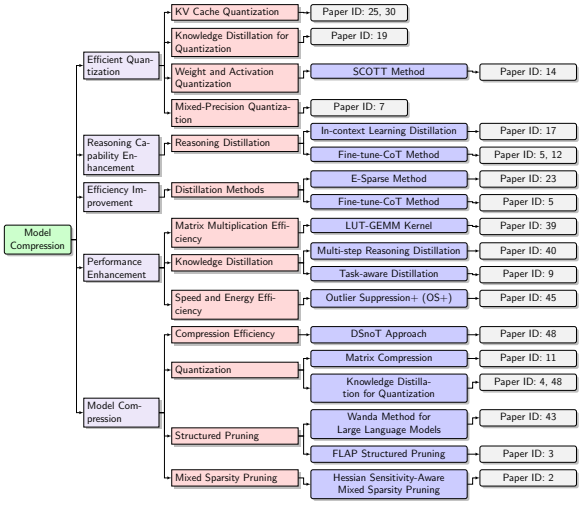


Figure 6: Taxonomy of "Model Compression methods for Large Language Models" generated by Chime (Hsu et al., 2024).

C Prompts

The prompts we used are shown in Figures 14–18.

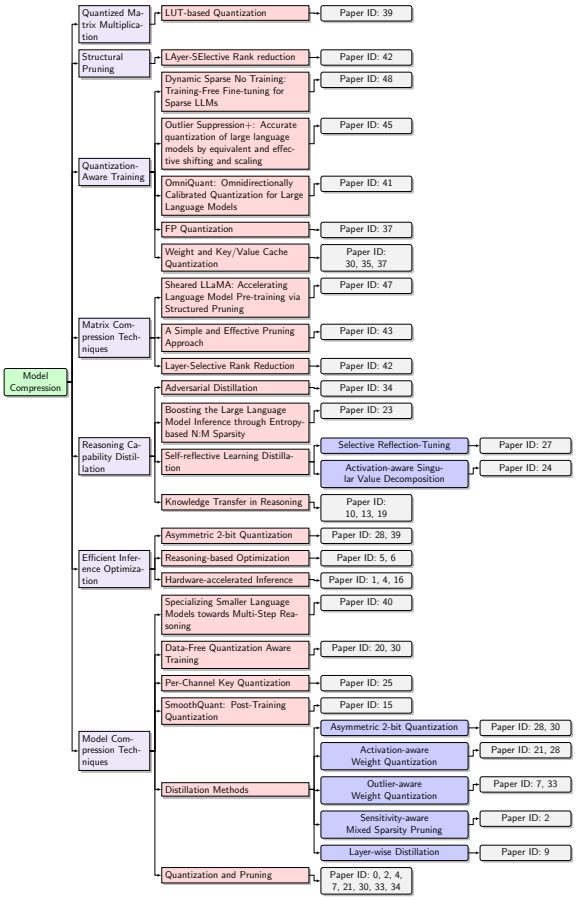


Figure 7: Taxonomy of "Model Compression methods for Large Language Models" generated by TnTLLM (Wan et al., 2024).

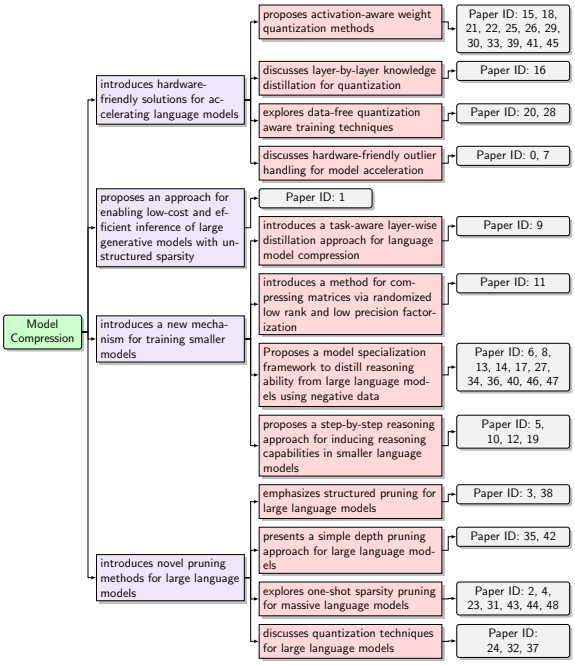


Figure 8: Taxonomy of "Model Compression methods for Large Language Models" generated by GoalEx (Wang et al., 2023b).

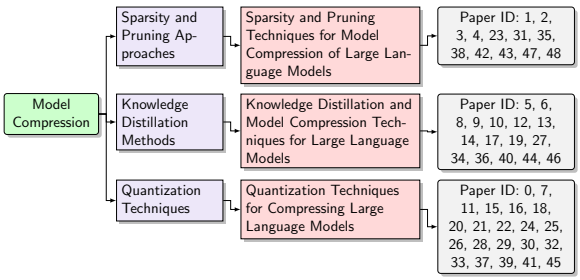


Figure 9: Taxonomy of "Model Compression methods for Large Language Models" generated by KnowledgeNavigator (Katz et al., 2024).

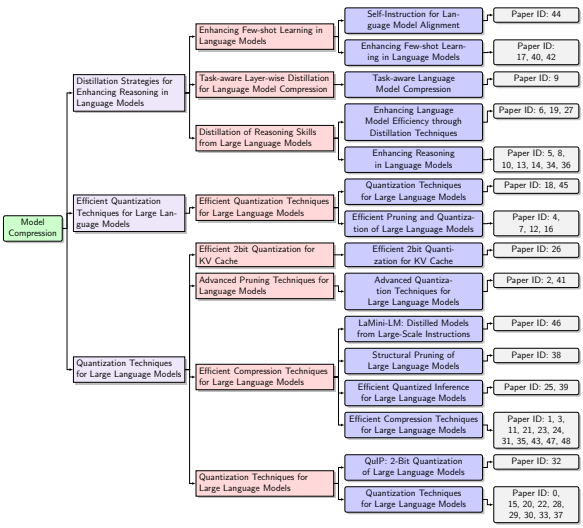


Figure 10: Taxonomy of "Model Compression methods for Large Language Models" generated by SCYCHIC (Gao et al., 2025).

Topic

Model Compression -> Quantization Techniques for LLM Compression

ASPECT

1. Outlier Management Techniques: "This sub-dimension captures methods that specifically address the challenges posed by outlier values in model weights and activations, including strategies for their identification, suppression, and quantization."

2. Precision Level Strategies: "This sub-dimension focuses on the various approaches to quantization precision, including mixed-precision techniques, low-bit quantization, and the use of floating-point representations to optimize model performance while reducing memory usage."

3. Quantization-Aware Training Methods: "This sub-dimension includes techniques that incorporate training processes specifically designed to enhance the quantization of large language models, ensuring that the models maintain performance despite reduced precision."

4. Quantization Framework Type: "This sub-dimension categorizes the different frameworks or methodologies employed for quantization, such as post-training quantization, quantization-aware training, or hybrid approaches, emphasizing their unique characteristics and applications."

5. Hardware Efficiency Techniques: "This sub-dimension highlights strategies that improve the hardware efficiency of quantization methods, including algorithm-architecture co-design and optimizations that reduce computational overhead during inference."

6. Knowledge Distillation in Quantization: "This sub-dimension captures the use of knowledge distillation techniques to enhance the performance of quantized models, particularly in scenarios where original training data is not available."

Figure 11: Aspect generation under a specific topic.

TITLE

"Flash-LLM: Enabling Low-Cost and Highly-Efficient Large Generative Model Inference With Unstructured Sparsity"

ASPECT-BASED SUMMARY

- **"Compression Technique"**: "The paper employs unstructured model pruning as a compression technique to reduce GPU memory usage and computational demands while maintaining model accuracy. This approach selectively removes less significant connections in neural networks, allowing for efficient deployment of large generative models."
- **"Reasoning Capability"**: "Not applicable"
- **"Quantization Strategy"**: "Not applicable"
- **"Sparsity Utilization"**: "Flash-LLM leverages unstructured sparsity to enhance the efficiency of large generative model inference. By implementing a Load-as-Sparse and Compute-as-Dense methodology, it addresses memory bandwidth bottlenecks and optimizes the performance of skinny matrix multiplications on tensor cores."
- **"Distillation Process"**: "Not applicable"
- **"Efficiency Improvement"**: "The paper presents Flash-LLM as a highly efficient software framework that significantly improves inference speed and reduces resource costs. It achieves up to 3.8X and 3.6X enhancements in performance over existing frameworks like DeepSpeed and FasterTransformer, respectively."
- **"Model Adaptation"**: "Not applicable"
- **"Training Paradigm"**: "Not applicable"

Figure 12: Aspect-based summary generated for the paper *"Flash-LLM: Enabling Low-Cost and Highly-Efficient Large Generative Model Inference With Unstructured Sparsity"*.

TOPIC

Model Compression

Aspect->Facet

1. ASPECT: Adaptation Strategy -> FACET: Quantization Techniques for LLM Compression

1.1. ASPECT: Outlier Management Techniques -> FACET: Outlier-Aware Quantization Techniques for LLMs

1.2 ASPECT: Quantization Framework Type -> FACET: Post-Training Quantization Strategies for Efficient LLM Deployment

1.2.1 ASPECT: Activation and Weight Quantization Schemes -> FACET: Post-Training Quantization Strategies for Efficient LLM Deployment

1.2.2 ASPECT: Contextual Cache Optimization -> FACET: KV Cache Quantization Strategies for Efficient LLM Inference

1.2.3 ASPECT: Precision Level -> FACET: 1-Bit Quantization via Knowledge Distillation

2. ASPECT: Compression Technique -> FACET: Pruning Strategies for Efficient LLMs

2.1. ASPECT: Sparsity Granularity -> FACET: Unstructured Sparsity Techniques for Efficient LLM Inference

2.2 ASPECT: Sparsity Granularity -> FACET: Structured Pruning Methods for LLMs Without Retraining

2.3 ASPECT: Sparsity Granularity -> FACET: Orthogonal Transformations for LLM Compression

3. ASPECT: Implementation Complexity -> FACET: Distillation Methods for Simplified LLM Reasoning

3.1 ASPECT: Training Paradigm -> FACET: Task-Aware Distillation and Instruction-Tuning

3.2 ASPECT: Reasoning Capability -> FACET: Chain-of-Thought Distillation for Enhanced Reasoning

3.2.1 ASPECT: Data Efficiency Techniques -> FACET: Distillation Techniques for Enhanced Reasoning in Smaller Models

3.2.2 ASPECT: Data Efficiency Techniques -> FACET: Negative Sample Utilization for Reasoning Distillation

3.2.3 ASPECT: Data Efficiency Techniques -> FACET: Self-Consistent Distillation for Faithful Reasoning

4. ASPECT: Compression Technique -> FACET: Low-Rank Factorization for LLM Compression

Figure 13: Facet identification within a topic.

Fixed Aspects

```
{
    "Research Problem": "A brief statement of
the problem addressed in this study and its
significance.",
    "Key Contributions": "A summary of the main
innovations and improvements introduced by this
study.",
    "Method": "A concise summary of the
methodological approach employed in the study",
    "Datasets": "The datasets used in the
study, their sources, and their characteristics
(size, type, domain).",
    "Experimental Setup": "Key details of the
experiment, including training strategies,
hyperparameter tuning, hardware setup, and
baseline implementations.",
    "Evaluation Metrics": "The metrics used to
assess performance (e.g., accuracy, BLEU,
ROUGE, F1-score, MSE).",
    "Results & Findings": "Summary of the main
experimental outcomes and how they compare with
state-of-the-art methods."
}
```

Figure 14: Fixed aspects we used.

First Level Aspects Generation

System

```
You are an expert in research survey writing and
taxonomy design.

Your goal is to abstract and design high-level,
generalizable dimensions to characterize a set of
research papers collectively.
Focus on identifying abstract dimensions, not on listing
concrete topics, methods, or datasets.

Each dimension should have:
- A clear and concise name
- A short explanation of what the dimension captures (no
more than 20 words)

Prioritize coherence and coverage when selecting
dimensions: they should jointly cover the main aspects
of the research without significant overlap.

You must output the results in strict JSON format:
{"Dimension Name": "Explanation"}.

Be concise, formal, and highly structured. Avoid free
text explanations.
Avoid mentioning any specific methods, dataset names,
model architectures, task examples, or experimental
details.
```

User

```
Here is a list of paper titles related to [TITLE]:

[PAPERS]

Analyze these papers based on their titles only.
Design and output a set of general, abstract dimensions
(no more than 10 and no less than 4) suitable for
characterizing the research collectively according to
the given instructions.
- Do not list topics, methods, or datasets individually.
- Keep each explanation within 20 words.
- Output only the dimension names and their explanations
in JSON format.
```

Figure 15: Prompt used for the first-level aspects generation.

Other Level Aspects Generation

System

```
You are an expert in research survey writing and
taxonomy design.

Your task is to refine and extend an existing high-level
analysis dimension by proposing a finer-grained
categorization suitable for organizing research papers
more precisely.

Given:
- A selected high-level analysis dimension (e.g.,
Research Focus, Methodology, or Evaluation Setting)
- A set of research papers, each with a brief
description relevant to the selected dimension

Your task is to:
- Analyze the papers and their descriptions
- Propose several finer-grained sub-dimensions under the
given high-level dimension
- Each sub-dimension must have:
  - A clear and concise name
  - A short explanation of what it captures

Guidelines:
- Sub-dimensions should be specific enough to
differentiate papers within the topic
- They must be generalizable and reusable, not overly
tied to individual papers
- Maintain formal academic tone
- Avoid listing specific paper names or copying text
from descriptions
- Output must be structured strictly in JSON format:
{"Sub-Dimension Name": "Short explanation"}
```

User

```
Here is the list of papers related to [TITLE] and their
corresponding descriptions about high-level dimension
[TOPIC]:

[PAPERS]

Task:
- Based on the descriptions, generate 2–6 sub-dimensions
that fall under the given high-level dimension.
- Each sub-dimension should have a concise name and a
short explanation.
- Output only the structured JSON as specified.
```

Figure 16: Prompt used for the other-level aspects generation.

Aspect-based Summary Generation

System

```
You are a research analysis assistant tasked with
generating concise, structured summaries of academic
papers under specific analytical dimensions.

Given:
- A paper's title, abstract, and optionally its
introduction
- One or more predefined analytical dimensions (e.g.,
Research Focus, Methodology, Evaluation Setting)
- For each dimension, you may optionally be given a more
specific sub-dimension (e.g., Research Focus →
Hallucination Detection)

Your goal is to:
- Generate for each paper a short, informative, and
targeted description under each given (sub-)dimension
- The description should be:
  - Specific to the dimension
  - Expressive of what the paper contributes,
investigates, or demonstrates under that angle
  - No longer than 100 words per dimension
  - Not copied or directly paraphrased from the abstract

If no meaningful content relates to a dimension, return
`"Not applicable"` as the value for that field.

Output must be structured JSON: {"Dimension Name or Sub-
Dimension Name": "Short description"}
```

User

Input Details

I am going to provide the target paper as follows, extract and summarize the details:

- Target aspects: **[ASPECTS]**
- Target paper title: **[TITLE]**
- Target paper abstract: **[ABSTRACT]**
- (Optional) Target paper introduction: **[INTRODUCTION]**

Figure 17: Prompt used for aspect-based summary generation.

Topic Facets Generation

System

```
You are an expert in scientific research analysis.

Your task is to generate meaningful and consistent names for
multiple paper clusters under the same semantic topic path.

**Input Information**

- Title: [TITLE] — the broader research theme (e.g., LLMs
for Causal Reasoning)

- Topic Path: [TOPIC] — the current semantic layer (e.g.,
Methodology or Methodology → LLMs as Reasoning Engines)

- Input: A dictionary of clusters, where each key is a
cluster topic, and the value is a list of paper summaries

{
  "cluster_1": [ {'Title': '...', 'Abstract': '...'}, ...],
  "cluster_2": [{'Title': '...', 'Abstract': '...'}, ...],
  ...
}

**Your Tasks**

For each cluster, you must:

1. Carefully examine the topic path and understand the
expected granularity:

 - If the topic path is broad (e.g., Methodology), your
output should be cluster names that describe the role, use,
or behavior of LLMs, such as:

  + LLMs as Reasoning Engines
  + LLMs as Planning Assistants
  + LLMs as Helpers to Traditional Methods

 - If the topic path is already specific (e.g., Methodology
→ LLMs as Reasoning Engines), your cluster names should
reflect specific modeling or training strategies, such as:

  + Prompt Engineering
  + Chain-of-Thought Tuning
  + Knowledge-Augmented Fine-Tuning



2. Generate one precise and specific name for each cluster
that captures its unifying theme.

**Output format (JSON)**:

{
  "cluster_1": "LLMs as Symbolic Reasoning Agents",
  "cluster_2": "Prompt Engineering for Causal Inference
Tasks",
  "cluster_3": "Fine-tuned LLMs for Structured Reasoning"
}


**Constraints**

 - Cluster Name should be specific, functional, and grounded
in the shared patterns of the papers

 - Do not include generic names like "LLM Applications" or
"Recent Advances"

 - Maintain strict JSON format
```

User

Here is the list of papers related to **[TITLE]** and their corresponding descriptions about high-level dimension **[TOPIC]**:

[PAPERS]

```
Task:
- Based on the descriptions, generate 2–6 sub-dimensions
that fall under the given high-level dimension.
- Each sub-dimension should have a concise name and a short
explanation.
- Output only the structured JSON as specified.
```

Figure 18: Prompt used for topic facets generation.