SLIDES2MINDMAP: RECONSTRUCTING COGNITIVELY GROUNDED KNOWLEDGE HIERARCHIES FROM LECTURE SLIDES

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating mind maps from lecture slides can help learners efficiently assimilate fragmented knowledge, promising substantial benefits for intelligent tutoring systems. However, dedicated automatic evaluation methods remain underexplored, while the generation is also challenging, requiring a global-local knowledge focus balance and handling large-scale, heterogeneous content. We formulate the **Slides2MindMap** task, which aims to reconstruct cognitively efficient knowledge hierarchies from a course's slide deck collection. For systematic evaluation, we introduce **S2M-Bench**, a benchmark comprising 12,774 slide pages with expert-annotated mind maps spanning 24 university courses. S2M-Bench includes a cognitive-science-grounded evaluation framework that integrates comparison with expert-annotated references, structure conformity analysis, and VLM-as-a-Judge. To address this task, we propose **AutoMindMap**, an agentic framework inspired by the Structure Building Framework Theory. AutoMindMap comprises *Skeleton Laying* for global scaffold anchoring, *Iterative Knowledge Integration* augmented by context-aware and branch-isolated summarization, and *Dual-Stage Refinement* with a local-global decoupling mechanism. The framework reconciles local knowledge faithfulness with global coherence, and adapts to slide-specific features. Experiments on S2M-Bench demonstrate that AutoMindMap outperforms baselines and achieves superior robustness across different models and scenarios, underscoring its educational application value.

1 INTRODUCTION

Mind maps play a significant role in modern education scenarios, revealing the progressive and hierarchical knowledge logic, and offering a visual scaffold that mirrors the associative nature of human cognition in learning (Buzan & Buzan, 2006). Grounded in Advance Organizer Theory (Ausubel, 1960) and Cognitive Load Theory (Sweller, 2011), mind maps effectively facilitate the assimilation of fragmented knowledge (Raiyn, 2016), and enable more pedagogical scenarios such as lecture planning (Zhang-Li et al., 2025) and exercise generation (Dong et al., 2024). Despite the educational value, constructing high-quality mind maps from course materials remains labor-intensive and expertise-demanding, motivating the pursuit of automated generation (Zhang et al., 2024), particularly for **slide-oriented mind mapping** (Low et al., 2025): Unlike well-structured textbooks, slides are often interspersed with visually noisy and pedagogy-driven content, lacking explicit hierarchical organization and narrative continuity (Tanaka et al., 2023). The nature risks learner misinterpretation and cognitive overload, and complicates automated knowledge extraction (Kasinathan & Mustapha, 2024), thereby underscoring both the application necessity and the technical difficulty of the task.

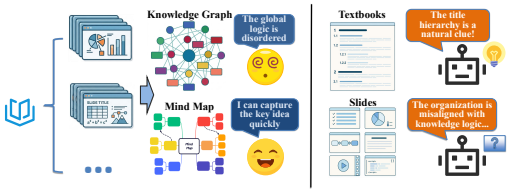


Figure 1: Illustration of Slides2MindMap. **Left**: mind maps provide more intuitive structure for visual learning, but need to strike a global-local balance. **Right**: the flexible organization of slides makes explicit hierarchy extraction difficult.

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

Although previous studies have conducted preliminary explorations on the automatic mind map generation, these efforts have primarily focused on supervised sentence dominance relation identification (Hu et al., 2021; Zhang et al., 2024) and hierarchical text summarization (Jain et al., 2024) in short text scenarios, and demos in the field of human-computer interaction (HCI) for user experiments (Han & Choi, 2025; Low et al., 2025). These methods suffer from limited generalization and scalability in large-scale and heterogeneous pedagogical materials, and lack comprehensive evaluation framework. Recent large language models (LLMs) bring new opportunities for the task due to the powerful zero-shot extraction and summarization capacities (Zhang et al., 2025b), thus LLM-based mind map generation has been favored by many intelligent tutoring systems (ITS) (Jia et al., 2025; Yu et al., 2026). However, as shown in Figure 1, the task still lacks systematic exploration, and the following challenges still exist:

First, mind map generation presents a paradigm gap with previous knowledge extraction tasks. Previous ones, like knowledge graph construction (KGC), focus on dense triple extraction and high-frequency entity anchoring. Such a structure targets retrieval and inference tasks, but lacks conciseness and overall hierarchical logic for *efficient human learner visual understanding* (Zhu et al., 2024). Concept Map is closer to educational scenarios, but still faces schema constraint (Davies, 2011). Comparatively, a cognitively efficient mind map should organize concepts into progressive and meaningful hierarchies around a central theme, distilling dense relational networks into an intuitive, branched structure that emphasizes core knowledge logic (Hua & Wind, 2019). This principle demands a ***balance*** *between local knowledge faithfulness and global logical coherence*, which existing LLM-based extraction and summarization methods struggle to strike.

The above challenge is compounded by slide-specific features (Tanaka et al., 2023). (1) Slides show flexible organizations, lacking explicit hierarchical logic in linear and fragmented structures. As shown in Figure 2, the narratives serve pedagogical purposes (e.g., guidance, interpretation, association, and consolidation), and exhibit interleaving, repetition, and jumps (Feng, 2020), presenting a **systematic heterogeneity with intrinsic knowledge logic**. (2) The slide collection of each course tends to be **large-scale** (see Table 1), comprising numerous decks and hundreds of slides with implicit and interlaced knowledge association (Low et al., 2025), which further complicates comprehensive knowledge understanding, especially for end-to-end methods (Liu et al., 2024). Although recent document hierarchy indexing (DHI) methods like PageIndex (Zhang et al., 2025c) and BookRAG (Wang et al., 2025) can generate analogous hierarchical structures, these methods rely heavily on well-structured documents and explicit formatting cues, thus are brittle in flexible and long-context slide scenarios. Accordingly, building coherent knowledge hierarchies from slides requires *further reasoning and reconstruction rather than direct extraction and merging*.

These observations further introduce an **evaluation problem**: a scalable benchmark should construct a proper evaluation slides dataset to cover a wider range of educational scenarios, while the evaluation framework should consider the knowledge reconstruction faithfulness, the task openness and the application value, i.e. improving the learner's understanding efficiency.

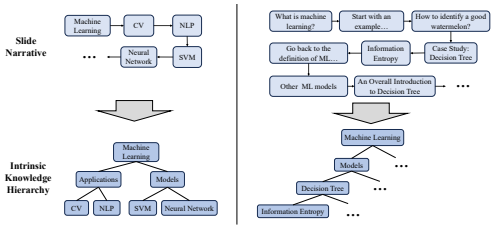


Figure 2: Two typical cases of the heterogeneity. **Left**: intermediate taxonomies of descendant concepts might be implicit and need to be inferred. **Right**: a lot of noise and knowledge interleaving are for pedagogical purposes, and hierarchy reconstruction needs further reasoning.

To address the challenges, we first formulate the **Slides2MindMap** task, aiming to reconstruct explicit, coherent, and faithful knowledge hierarchies from a course's large-scale slide deck collection. For comprehensive evaluation, we introduce **S2M-Bench**, the first dedicated benchmark focusing on hierarchical knowledge reconstruction from authentic course slide collections, and targeting efficient learner understanding. S2M-Bench comprises 12,774 slide pages across 24 university-level courses, with expert-annotated mind maps averaging 121.38 concepts per course. The courses are categorized by knowledge type and content modality to emphasize scenario diversity. Given the semi-open nature of the task, the evaluation framework is grounded in cognitive science and integrates ground-truth-based comparison, quantitative structure conformity, and VLM-as-a-Judge (Chen et al., 2024a).

To address the task, we propose **AutoMindMap**, an LLM-based agentic framework inspired by the *Structure Building Framework* (SBF) Theory (Gernsbacher, 2013) in cognitive psychology. We discard the traditional knowledge extraction paradigm and treat mind map construction as a *dynamic discourse comprehension process* that mirrors human cognitive models. AutoMindMap emulates foundation laying, mapping, and shifting processes in SBF through a three-stage design: **(1) Skeleton Laying** emulates *foundation laying*, capturing salient elements to scaffold an overall knowledge skeleton, and generating skills for further branch expansion. This strategy anchors global structure logic to prevent collapse. **(2) Iterative Knowledge Integration** emulates *mapping-shifting*, maintaining branch-isolated memory states as cross-discourse semantic bridges through context-aware summaries, then incrementally expanding the map through atomic tools and structural repair. The design alleviates knowledge fragmentation, logic breaks, and invalid structure. **(3) Dual-Stage Refinement** decouples macro-level (e.g., global balance and coherence) and micro-level (e.g., factual accuracy) issues for separate refinement, to avoid global-local mutual structural destruction. Macro-level refinement is performed with rendered map images for intuitive structural insight. Overall, the SBF-inspired design preserves global and local consistency, achieving secure harnessing through structural anchoring, tool-based expansion, and refinement.

Our contribution can be summarized as follows: (1) We establish S2M-Bench, the first dedicated benchmark for evaluating the lecture-slide-oriented automated knowledge hierarchy reconstruction task. (2) We propose AutoMindMap, an agentic framework that effectively addresses the task by emulating the human discourse cognitive model. (3) Extensive experiments on S2M-Bench show that AutoMindMap significantly outperforms baselines and exhibits strong generalization and robustness across different models and scenarios.

2 S2M-BENCH

2.1 PROBLEM DEFINITION

Definition 1 (Mind Map). A mind map $\mathcal{M}$ (Buzan & Buzan, 2006) is a connected directed acyclic graph (DAG) defined as $\mathcal{M} = (V, E, r)$, where V is the set of concept nodes, $r \in V$ is an only **central** root node representing the core topic. $E \subseteq V \times V$ is a set of hierarchical edges, where $(v_i, v_j) \in E$ indicates that v_j is a child of v_i, multiple parents of one node (termed **cross-links**) are permitted. Hierarchical edges indicate dominance relationships (e.g., hypernym–hyponym, whole–part, and entity–attribute). This paper excludes the map visual design (Hua & Wind, 2019) and only focuses on the knowledge structure.

Definition 2 (Slides2MindMap). Given a course C comprising m slide decks $\mathcal{D} = \{D_1, \ldots, D_m\}$, where $\mathcal{D}$ is not necessarily well-formed (e.g., without prerequisite or standard file name), and each deck D_i consists of several pages containing heterogeneous elements (e.g., titles, images, paragraphs, equations, tables). The **Slides2MindMap** task aims to generate a mind map $\mathcal{M}$ with the course name as r, such that $\mathcal{M}$ can faithfully generalize the intrinsic knowledge hierarchy of C, and present a coherent and cognitively-efficient global structure for visual understanding.

2.2 DATASET CONSTRUCTION AND QUALITY CONTROL

The collected courses span university-level computer science and mathematics. To ensure **data quality**, we converted all decks to PDF, removed slides with formatting anomalies, and verified page completeness. To keep **task particularity**, we manually selected course slides that exhibit enough scale, flexibly organized structures, and rich hierarchical knowledge, while excluding those with excessively structured narratives or overly abstract concepts. To maximize **scenario diversity**, we categorized all courses along two dimensions (i.e., knowledge type and content modality) inspired by Bloom's taxonomy of educational objectives (Armstrong, 2010) and multi-modal education theory (Hassett & Curwood, 2009). Following the categorization criteria (detailed in Appendix C), we further selected 24 high-quality courses totaling 267 decks and 12,774 slides. Table 1 summarizes the dataset categorization and statistics. For each course, we invited expert annotators, consisting of the course lecturer and PhD-level teaching assistants, to construct mind maps with unified instructions and rubrics. To ensure **annotation reliability**, we conducted cross-validation and multiple refinement rounds after initial annotation to eliminate redundant, ambiguous, and over-dense concepts, retaining only high-confidence elements that faithfully reflect the source material.

Table 1: Statistics for different course categories. All columns are averages except *Courses*. *Co.* abbreviates *Course*. The tokenization is based on *o200k_base*.

Category	Courses	Decks/Co.	Pages/Co.	Pages/Deck	Tokens/Co.	Tokens/Page	Nodes/Co.	Edges/Co.
Math & Programming	7	11.29	516.14	45.73	146.24K	283.32	121.71	136.57
Concept Understanding	6	10.83	658.50	60.78	170.39K	258.76	154.67	167.17
Synthetic Application	11	11.18	470.55	42.08	138.68K	294.72	103.00	111.82
Text-dominated	6	9.33	388.00	41.57	106.05K	273.33	138.50	147.33
Image-dominated	6	11.67	620.33	53.17	165.06K	266.08	111.00	120.50
Hybrid	12	11.75	557.50	47.45	162.07K	290.71	118.00	131.83
Overall	**24**	**11.12**	**532.25**	**47.72**	**148.81K**	**280.34**	**121.38**	**132.88**

This rigorous rule minimizes subjectivity. The annotation consumes over 200 hours. These human-annotated maps serve as **ground-truth** in subsequent experiments. Further analysis and taxonomy illustrations of the dataset are deferred to Appendix C.

2.3 Evaluation Metrics

Although high-quality human annotation can be regarded as a reference, the knowledge organization faithfulness, structure compliance, open-ended nature, and cognitive efficiency need to be further considered. Thus, we design a three-faceted framework for comprehensive evaluation. Specifically, the "ground truth" doesn't indicate absolute golden answer, and for the cognitive efficiency evaluation, we leverage cognitive science theories and existing mind map scoring rubrics to decompose it into multiple quantifiable structural criteria, and construct rubrics for VLM-as-a-Judge. To verify the evaluation reliability, we conducted detailed analysis in the Appendix G.

2.3.1 Ground-Truth (GT)-based Comparison

Let $\mathcal{M}^{\text{gt}} = (V^{\text{gt}}, E^{\text{gt}}, r^{\text{gt}})$ and $\mathcal{M}^{\text{pred}} = (V^{\text{pred}}, E^{\text{pred}}, r^{\text{pred}})$ denote the ground-truth and generated maps, respectively. We first obtain a matched node-pair set $\mathcal{C} \subseteq V^{\text{pred}} \times V^{\text{gt}}$ via a matching algorithm (in Appendix E.1). To emphasize the trade-off capacity, we design the F1-score from three aspects: **(1) Hierarchy-aware Node F1-Score (HN-F1)** assigns each matched pair a weight that penalizes depth misalignment based on standard F1-score calculation, thus better reflecting the ability to identify the knowledge hierarchy: $w(v_i^{\text{gt}}, v_i^{\text{pred}}) = 1 - \left| \frac{d(v_i^{\text{gt}})}{d^{\text{gt}}} - \frac{d(v_i^{\text{pred}})}{d^{\text{pred}}} \right|$, where the depth $d(v)$ denotes the shortest-path distance from the root r, and $d^{\text{gt}} = \max_{v \in V^{\text{gt}}} d(v)$, $d^{\text{pred}} = \max_{v \in V^{\text{pred}}} d(v)$ are the respective maximum depths. **(2) Edge Connectivity F1-Score (EC-F1)** evaluates edge-level similarity and reflects the ability to identify hierarchical relations between concepts. Considering the open-ended nature, we use directed reachability instead of strict edge matching to tolerate reasonable intermediate nodes and taxonomies. **(3) Matched-Edge Connectivity F1-Score (MEC-F1)** further restricts the denominator of EC-F1 to edges whose both endpoints are in the matched set $\mathcal{C}$, thus isolating the evaluation to the structurally shared concept backbone. The metric is more stringent and better reflects the quality of the aligned hierarchical skeleton. The detailed calculations of the above three metrics are in Appendix E.2.

2.3.2 Quantitative Structure Conformity

This aspect measures whether the generated mind map is *structurally compliant* through quantifiable indicators. **(1) Pass@1** serves as a *hard constraint*: a generated map *passes* if containing exactly one root, no isolated nodes, and no cycles. This metric verifies only the most *fundamental structural requirements* of a valid mind map (Buzan & Buzan, 2006). We test the strictest condition that only runs the algorithm once. **(2) Structural Load Factor (SLF)** serves as a *soft constraint*, which is grounded in the principles of working memory capacity (Miller, 1956) and progressive differentiation (Ausubel, 1960), quantifying cognitive load imposed by structural anomalies. The metric is derived from existing structural constraint rubrics (Kedaj et al., 2014) and is defined as the average proportion of *abnormal* leaf nodes and non-leaf nodes: $\text{SLF} = \frac{1}{2}\left(\frac{|V^{\text{ab}}_{\text{leaf}}|}{|V_{\text{leaf}}|} + \frac{|V^{\text{ab}}_{\text{non-leaf}}|}{|V_{\text{non-leaf}}|}\right)$, where a leaf node v is abnormal if $d(v) = 1$ or $d(v)) \geq 7$, and abnormal non-leaf nodes refer to the nodes with more than 7 direct child nodes.

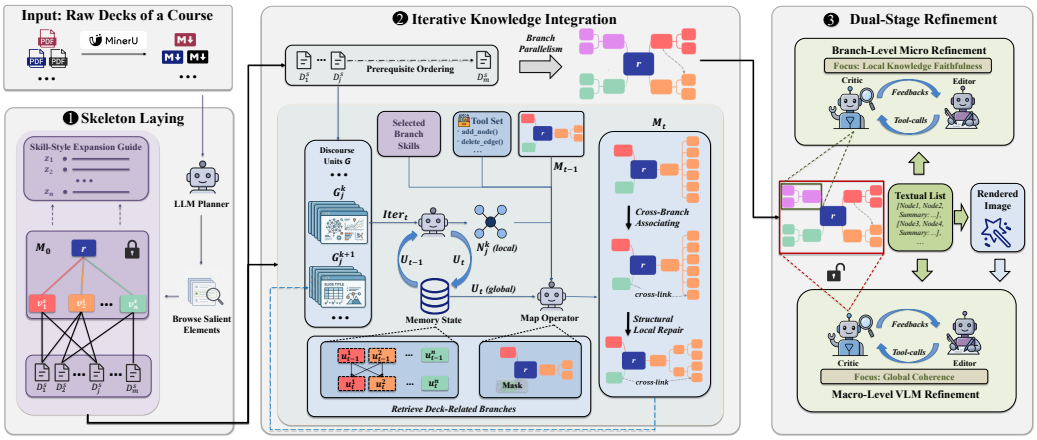


Figure 3: Overall framework of **AutoMindMap**. The main pipeline receives the parsed Markdown files of a course, then lays the initial skeleton with key associative information. The mind map is incrementally expanded with memory and tool-augmented integration. The map is further refined by an editor-critic framework that decouples macro-level and micro-level repair.

2.3.3 VLM-AS-A-JUDGE

Considering the open-ended nature and the visual pedagogical effectiveness, we employ a vision-language model (VLM), GPT-5.4, as an evaluator to evaluate the open-ended metrics, scoring mind maps using a 5-level Likert scale (Nemoto & Beglar, 2014) based on both *textual relation* lists and *rendered images*. Grounded in established cognitive science theories (detailed in Appendix K) and adapted from existing rubrics (Hua & Wind, 2019), we propose three dimensions: **(1) Content Accuracy (CA)** measures whether the method can extract correct, concise, and unambiguous concept representations with reasonable taxonomies from complex documents. **(2) Structure Quality (SQ)** examines structural soundness. Guided by principles of cognitive balance (Sweller et al., 2019) and working memory capacity, this metric focuses on coherent hierarchy progressiveness, branch balance, and appropriate scale. **(3) Pedagogical Efficiency (PE)** measures cognitive-level utility for human learners, including alignment with human cognitive habits to minimize confusion, featuring clear visual rendering, appropriate scale and granularity, relevant cross-links, and distinct structural emphasis on core knowledge for efficient comprehension. Detailed criteria of VLM-as-a-Judge evaluation are in Appendix E.3.

3 AUTOMINDMAP

The challenges encountered in Slides2MindMap bear similarities to human discourse comprehension. The SBF theory reveals that humans organize large-scale discourse into a coherent knowledge system through a cognitive model, which involves laying a foundational understanding first, incrementally mapping coherent information into the understanding system, and shifting to new substructures when coherence breaks. Drawing inspiration from the model, AutoMindMap mirrors the process through three concrete stages: **Skeleton Laying**, **Iterative Knowledge Integration**, and **Dual-Stage Refinement**; each is further adapted to slide-specific obstacles through tailored mechanisms described below. These modules form a top-down global structuring and bottom-up gradual refinement to pursue a global-local balance. The overall framework is shown in Figure 3.

Before the pipeline, we first employ MinerU (Wang et al., 2024a) to parse raw PDF decks into Markdown format. Images are transformed into descriptive summaries extracted by the VLM, and each page is segmented and wrapped into *<page_i>* ... *</page_i>* style to facilitate understanding.

3.1 SKELETON LAYING

The stage mirrors the foundation laying in SBF, generating the course's overall understanding and planning the subsequent stages before the construction. For a course C consisting of decks

$\mathcal{D} = \{D_1, \ldots, D_m\}$, we extract salient elements K_i (e.g., file meta data, page titles) from each deck D_i, and ask LLM planner to produce: (1) Backbone second-level nodes $\mathcal{V}^s = \{v_1^s, \ldots, v_n^s\}$ directly connecting to r, to formulate the **initial skeleton**, which represents the top-level and most abstract concept set with proper size that outlines the course. The skeleton is locked before the final macro refinement. (2) A bipartite graph $B_{(V,D)} \subseteq \mathcal{V}^s \times \mathcal{D}$, mapping each $v_i^s \in \mathcal{V}^s$ to relevant decks to build content logical association between decks and second-level nodes. (3) A set $\mathcal{Z} = \{z_1, \ldots, z_n\}$ where each z_i is an expansion guideline for v_i^s and serves as a skill-style (Xu & Yan, 2026) principle for downstream branching. $\mathcal{Z}$ includes anticipated descendant structure, risks to be avoided, and actionable suggestions. (4) A prerequisite sequence $\mathcal{S} = [D_1^s, \ldots, D_m^s]$ guides sequential and parallel processing.

$\mathcal{V}^s$ and $\mathcal{Z}$ scaffold a global anchor for subsequent generation, and prevent the subsequent map expansion from structure shift and loss of overall knowledge architecture sight; $B_{(V,D)}$ and $\mathcal{S}$ bridge cross-deck logic, avoiding the structural disorder caused by document-knowledge heterogeneity.

3.2 Iterative Knowledge Integration

The integration mirrors the mapping (intra-branch) and shifting (cross-branch) processes. Concepts are incrementally assimilated via summarization-enhanced atomic tool calls and local structural adjustment. Following $\mathcal{S}$, we ask the LLM to segment pages based on discourse semantics (Feng, 2020) instead of format clues. Each deck D_j is partitioned into contiguous page groups $\mathcal{G}_j = \{G_j^1, \ldots, G_j^q\}$ based on K_j, where each group constitutes a coherent discourse unit (e.g., with the same pedagogical topic or rhetorical move). This strategy respects narrative boundaries and mitigates logic breaks. Then the following steps are executed iteratively:

Firstly, for group G_j^k processed in t^{th} iteration, LLM-based extraction and summarization are performed sequentially:

$$(\mathcal{N}_j^k,\, \mathcal{U}_t) = \Gamma(G_j^k,\, \mathcal{U}_{t-1}), \tag{1}$$

where Γ denotes the LLM operation, $\mathcal{N}_j^k = (V_j^k, E_j^k)$ is the extracted local concept relation subgraph, $\mathcal{U}_t = \{u_t^i \mid (v_i^s, D_j) \in B_{(V,D)}\}$ is a dynamic memory state that evolves across iterations. Each u_t^i is anchored to D_j-relevant v_i^s branches and retains the essence of prior content, reconstructing knowledge logic continuity from the pedagogy sequence. Critically, $\mathcal{U}_t$ provides the contextual scaffold for interpreting $\mathcal{N}_j^k$, and the synergies bridge the branch-level fragmented local discourses (O'Brien et al., 1998). This branch-isolated design avoids context explosion and collapse, while preserving intra-branch coherence.

Second, the integration of $\mathcal{N}_j^k$ into the global mind map $\mathcal{M}_t$ is formalized as

$$\mathcal{M}_t = \Theta(\mathcal{N}_j^k,\, Z_t,\, \mathcal{M}'_{t-1},\, \mathcal{U}_t), \tag{2}$$

where Θ is the integration operation, $Z_t = \{z_i \mid (v_i^s, D_j) \in B_{(V,D)}\}$ controls the update with prior guidance, and only retrieves deck-related z_i. $\mathcal{M}_0$ is the initial skeleton. $\mathcal{M}'_{t-1} \subseteq \mathcal{M}_{t-1}$ is a *partial view* of the whole map that exposes only the branches $\mathcal{M}_{t-1}(v_i^s)$ under v_i^s with $(v_i^s, D_j) \in B_{(V,D)}$. $\mathcal{M}'_{t-1}$ masks unrelated branches and constrains the LLM's context to the most relevant structural and semantic neighborhood, avoiding distraction by unrelated noise knowledge. Then $\mathcal{M}_t$ is further improved by the following two steps:

Cross-Branch Associating aims to mitigate the risk of cross-link loss caused by $\mathcal{M}'_{t-1}$. For each new updated node v_w, we retrieve candidate node set V_c from the nodes outside $\mathcal{M}'_{t-1}$ that each node $v_c^i \in V_c$ satisfies $v_c^i \in TopK(\text{Sim}(v_w, v))_{v \notin \mathcal{M}'_{t-1}}$ and $\text{Sim}(v_w, v_c^i) \geq \gamma_{sim}$, where $\text{Sim}(\cdot, \cdot)$ is performed on the text embeddings *cosine similarity* and token-level *Jaccard similarity* separately to capture lexical and semantic association. The neighbor subgraphs of v_w and v_c^i are then provided to the LLM to judge inter-branch conceptual dependencies and perform potential merging and cross-linking operations.

Structural Local Repair checks structural legitimacy (e.g., loops and broken structure) and health (e.g., abnormal nodes and confusing relations) issues through rules and the LLM, to identify risky structures around each v_w for further repair and prevent the accumulation of structural errors.

Then the pipeline will return to the first step to process the next group (or next deck). For acceleration, decks that have disjoint second-level node assignments are processed in parallel, and decks

under the same second-level node are processed based on S. All operations concerning the mind map update are implemented via **reasoning-enhanced atomic tool calls** (listed in Appendix J), with compliance checks on parameter validity and structural constraints, and stage-aware tool isolation. The mechanism constrains the mind map generation to verifiable and controllable graph editing, mitigating collapsing structures caused by hallucination.

3.3 Dual-Stage Refinement

Even with careful incremental integration and checking, the final mind map may exhibit residual structural imperfections such as overgrown branches, guideline violations, and suboptimal global balance. However, direct refinement may risk local-global mutual structural destruction, which can further cause repeated modifications and structure collapse. We address the issues through an extra **Editor-Critic** refinement framework, which decouples micro-level local anomaly repair and macro-level global coherence improvement.

First, each second-level branch $\mathcal{M}(v_i^s)$ is refined independently. A critic agent examines the branch and produces feedback; then an editor agent executes structural edits via tool calls:

$$F_i^{(t)} = \pi_{\text{critic-mic}}\big(\mathcal{M}(v_i^s)^{(t)},\ z_i\big), \tag{3}$$

$$\mathcal{M}(v_i^s)^{(t+1)} = \pi_{\text{editor-mic}}\big(\mathcal{M}(v_i^s)^{(t)},\ F_i^{(t)}\big), \tag{4}$$

where $F_i^{(t)}$ is a list of actionable feedback, $\mathcal{M}(v_i^s)^{(t+1)}$ is the refined result. The tools used by the editor are the same as the integration step. This stage focuses on **micro-level** repair (e.g., local conflict resolution, guideline adherence, and branch-level pruning). The second stage addresses **macro-level** issues such as overall branch imbalance and structural incoherence, which are difficult to assess from textual relations alone. We further render the entire mind map as an image $I^{(t)} = \text{Vis}(\mathcal{M}^{(t)})$, and feed **both the image and the textual list** to a VLM-based critic; the editor then revises the whole map:

$$F^{(t)} = \pi_{\text{critic-mac}}(\mathcal{M}^{(t)},\ I^{(t)}), \tag{5}$$

$$\mathcal{M}^{(t+1)} = \pi_{\text{editor-mac}}(\mathcal{M}^{(t)},\ F^{(t)},\ I^{(t)}). \tag{6}$$

In macro-level refinement, the rendered image provides intuitive spatial awareness that text alone cannot convey. Structure-irrelevant pure visual design (e.g., color) is excluded in the refinement. The whole refinements are repeated until reaching the maximum rounds or no issues are reported. Here the superscript t indicates the refinement round, distinguished from the previous subscript. The two-stage refinement reflects a bottom-up philosophy that prevents mutual destruction caused by mixing local editing and global editing.

4 Experiments

We compare AutoMindMap against existing baselines and conduct ablation studies to validate the contribution of each module. Additional experiments are discussed in the **Appendix**, including the performance across different LLMs and course categories, the running cost and stability (Appendix F), the evaluation reliability analysis (Appendix G), the statistic of structural indicator distribution (Appendix H) and the case study and failure analysis (Appendix I).

4.1 Experiment Setup

4.1.1 Baselines.

Since no prior work directly addresses the Slides2MindMap task, we design several heuristic methods and select representative DHI methods as baselines: **(1) Human**. Human-annotated ground-truth data serves as a competitive reference in structure conformity and VLM-Judge metrics. **(2) Direct**. The LLM directly outputs all *(concept, sub-concept)* tuples given the full course content and constraints as prompts. **(3) Chunk-Merge**. Each deck is treated as a chunk (with heuristic segmentation for special long decks); local mind maps are generated per chunk, then merged and refined by the LLM and statistical rules, following the mainstream LLM-based KG construction (KGC) and *Map-Reduce* paradigm (Edge et al., 2024; Dean & Ghemawat, 2008). **(4) Highlights-only**.

Table 2: Experimental results of different methods. DeepSeek-V4-Flash serves as the backbone model. **Bold** and underlined results indicate the global/baseline optimum (except for Human), respectively. Avg. is the average result of VLM-judge metrics. Human annotations serve as the ground truth; thus, GT-based metrics are inapplicable. Except for VLM-judge, all results are percentages.

Methods		GT-based			Structure Conformity		VLM-Judge			
		HN-F1	EC-F1	MEC-F1	Pass@1	SLF ↓	CA	SQ	PE	Avg.
Human		-	-	-	100.00	1.05	3.96	3.92	3.79	3.89
Baselines	**Direct**	33.43	20.67	64.50	79.16	20.22	3.17	3.00	2.88	3.01
	Chunk-Merge	25.03	16.21	68.10	83.33	5.26	2.96	2.92	2.50	2.79
	Highlights-only	23.63	14.50	66.25	95.83	19.12	3.21	3.13	3.00	3.11
	BookRAG	25.12	10.12	47.85	**100.00**	12.56	2.29	2.21	1.92	2.14
	PageIndex	24.15	11.66	57.11	**100.00**	14.19	2.75	2.50	2.08	2.44
AutoMindMap	**+ GPT-4.1-mini**	37.80	22.96	74.01	**100.00**	**1.82**	3.58	3.63	**3.71**	3.64
	+ GPT-5.4-mini	**39.88**	**25.12**	**74.34**	**100.00**	1.93	**3.83**	**3.71**	3.63	**3.72**

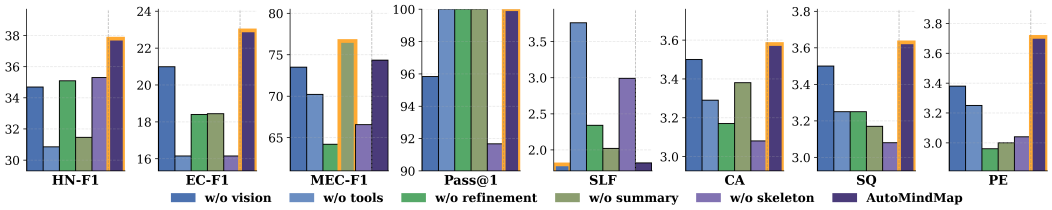


Figure 4: Ablation study result. The corresponding detailed data is in Table 12.

The LLM receives only salient elements (e.g., titles) from the parsed Markdown to infer hierarchy. **(5) BookRAG** (Wang et al., 2025). A tree-hierarchy indexing framework based on MinerU's parsing output. We only use its tree indexing algorithm, excluding the graph construction and retrieval parts. **(6) PageIndex** (Zhang et al., 2025c). A TOC-style vector-free indexing method that organizes hierarchy via formatting cues and content understanding. Notably, mainstream KGC methods (Zhu et al., 2024) are excluded because the generated KG structure is far from a mind map, and even beyond the basic constraints (i.e., pass@1 $= 0\%$). Training-based mind mapping methods (Hu et al., 2021; Zhang et al., 2024) are also excluded, as our data scale and scenarios are beyond their scope (i.e., sentence-level short text summarization).

4.1.2 Implementation.

Since AutoMindMap is training-free and to ensure evaluation effectiveness, all 24 courses in S2M-Bench are treated as the test set. All raw files are parsed into Markdown via MinerU for all non-Human methods. DeepSeek-V4-Flash serves as the backbone LLM for baselines and the vision-free modules of AutoMindMap; GPT-4.1-mini and GPT-5.4-mini are evaluated separately as the VLM for visual refinement in AutoMindMap. More implementation details are in Appendix D.

4.2 Experiment Results

4.2.1 Main Experiments.

As shown in Table 2, **AutoMindMap** (with either VLM) avoids illegal generation and consistently outperforms all baselines (except for Human) across every metric, while the stronger VLM (GPT-5.4-mini) yields further improvements. **Human** annotations achieve the highest scores, confirming the annotation quality, and AutoMindMap still has a gap with human experts. Among baselines, **Direct** and **Highlights-only** perform relatively better, as they directly generate the entire map, preserving global structural consistency at the cost of missing inconspicuous concepts and generating invalid structure. **Chunk-Merge** maintains competitive local knowledge faithfulness but suffers from poor global coherence due to the post-merging mechanism, reflected in lower VLM-Judge scores. **BookRAG** and **PageIndex** reliably produce valid tree structures (Pass@1 = 100%) but perform worst on both GT-based and VLM-Judge metrics, confirming that retrieval-oriented DHI

methods are ill-suited for the Slides2MindMap task. All the automated baselines except for Chunk-Merge have obviously high SLF, which indicates the risk of increased cognitive load of the map. We conduct further analysis on fine-grained GT-based metrics and case study in Appendix F.6 and Appendix I.

4.2.2 Ablation Study.

Figure 4 presents the ablation results, where we compare the full AutoMindMap against five variants: **(1) w/o vision**, which supplies only textual descriptions of the map during refinement; **(2) w/o tools**, which directly generates JSON tuples for map updates; **(3) w/o refinement**, which ablates the decoupled refinement mechanism; **(4) w/o summary**, which directly transforms the long corpus into a mind map without context-aware summarization for discourse-based grouping; **(5) w/o skeleton**, which builds the map directly from the root without any prior scaffolding and skill. We only use GPT-4.1-mini for visual refinement. Notably, AutoMindMap is not strictly optimal on every metric (e.g., slightly underperforms on MEC-F1 and SLF), indicating that some components trade off performance on specific dimensions for overall quality improvement. The trade-off is discussed in Appendix F.6 and Appendix G.3. We combine the visual generation effect and provide a more in-depth study on the function of each ablated module in Appendix I.

4.2.3 Further Analysis.

AutoMindMap exhibits robust performance across different LLMs and diverse course categories (Figure 5), low variance under stochastic multi-runs, and eligible runtime cost. These detailed experiments are in Appendix F. For the evaluation of S2M-Bench, in Appendix G, we confirm that VLM-Judge exhibits high agreement with human and cross-model judges, the robustness, interpretability and cross-metric correlation of the evaluation system. Besides, through re-annotation and cross-metric correlation analysis, we demonstrate that the GT-based metrics are affected by the task openness, resulting in low values and limited evaluation utility, but still retain discrimination in specific regime. Appendix H further shows that the structural statistical feature distribution of the maps generated by AutoMindMap is closer to that of humans, and part of the quantified structural indicators are consistent with the Structure Quality score. Appendix I provides more in-depth analysis of the main experiment and the ablation study. We also discuss some limitations in Appendix B.

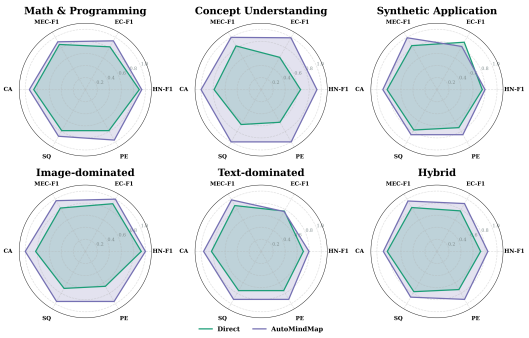


Figure 5: Comparison between *AutoMindMap* and *Direct* on different courses categories. Each metric is normalized by its maximum value across all results, resulting in a [0,1] scale. AutoMindMap performs better on each category, especially in *Concept Understanding*, showing the advantage in courses with more complex knowledge hierarchy. More analysis is in Appendix F.

5 Conclusion

In this work, we formulate a new task of Slides2MindMap, which aims to generate cognitively efficient concept mind maps from large-scale course lecture slides. We further propose a benchmark, S2M-Bench, to comprehensively evaluate the Slides2MindMap task, and an LLM-based agentic framework, AutoMindMap, which outperforms the existing baselines in S2M-Bench. Based on the Structure Building Framework Theory, AutoMindMap involves a pipeline consisting of skeleton laying, iterative knowledge integration, and dual-stage refinement, balancing local knowledge faithfulness and global logic coherence, and adapting to the large-scale fragmented slide content. However, our study has some limitations (see Appendix B). These issues can be further explored. AutoMindMap has been deployed on relevant intelligent education platforms and will be fully open-source. We hope that this work can advance the field of Educational AI, especially knowledge structure visualization.

AI USE STATEMENT

All research ideas, the framework of AutoMindMap, experimental design, and core paper-writing were conceived by humans. Codex (with GPT-5.3-Codex) was used to assist programming and prompt polishing. DeepSeek-V4-Pro and GPT-5.5 were used to polish the paper, without modifying the original idea. Some icons in the figures were generated by GPT-Image-2.

ETHICS STATEMENT

All slides used are approved by the lecturers of the courses as well as the relevant academic organization of the university. We remove the slides that may potentially leak personal privacy. All data annotators are informed about the purpose of the data they annotated. The datasets and code will be released upon publication.

REFERENCES

Hasan Abu-Rasheed, Constance Jumbo, Rashed Al Amin, Christian Weber, Veit Wiese, Roman Obermaisser, and Madjid Fathi. Llm-assisted knowledge graph completion for curriculum and domain modelling in personalized higher education recommendations, 2025.

Patricia Armstrong. Bloom's taxonomy. *Vanderbilt University Center for Teaching*, 12(05):2023, 2010.

David P Ausubel. The use of advance organizers in the learning and retention of meaningful verbal material. *Journal of educational psychology*, 51(5):267, 1960.

Tony Buzan and Barry Buzan. *The mind map book*. Pearson Education, 2006.

Dongping Chen, Ruoxi Chen, Shilin Zhang, Yinuo Liu, Yaochen Wang, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. Mllm-as-a-judge: Assessing multimodal llm-as-a-judge with vision-language benchmark, 2024a. URL https://arxiv.org/abs/2402.04788.

Zhe Chen, Heyang Liu, Wenyi Yu, Guangzhi Sun, Hongcheng Liu, Ji Wu, Chao Zhang, Yu Wang, and Yanfeng Wang. M^3AV: A multimodal, multigenre, and multipurpose audio-visual academic lecture dataset. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 9041–9060. Association for Computational Linguistics, 2024b. doi: 10.18653/V1/2024.ACL-LONG.489.

Martin Davies. Concept mapping, mind mapping and argument mapping: what are the differences and do they matter? *Higher education*, 62(3):279–301, 2011.

Jeffrey Dean and Sanjay Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

Harkirat S Dhindsa and O Roger Anderson. Constructivist-visual mind map teaching approach and the quality of students' cognitive structures. *Journal of Science Education and Technology*, 20 (2):186–200, 2011.

Linyi Ding, Sizhe Zhou, Jinfeng Xiao, and Jiawei Han. Automated construction of theme-specific knowledge graphs. *arXiv preprint arXiv:2404.19146*, 2024.

Junnan Dong, Zijin Hong, Yuanchen Bei, Feiran Huang, Xinrun Wang, and Xiao Huang. Clr-bench: Evaluating large language models in college-level reasoning. *arXiv preprint arXiv:2410.17558*, 2024.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

Alexander Richard Fabbri, Irene Li, Prawat Trairatvorakul, Yijiao He, Weitai Ting, Robert Tung, Caitlin Westerfield, and Dragomir Radev. Tutorialbank: A manually-collected corpus for prerequisite chains, survey extraction and resource recommendation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 611–620, 2018.

Dezheng William Feng. Genre, pedagogy, and powerpoint design: A multimodal move analysis of linguistics lecture slides. In *Approaches to specialized genres*, pp. 177–197. Routledge, 2020.

Rafael Ferreira-Mello, Máverick André, Anderson Pinheiro, Evandro Costa, and Cristobal Romero. Text mining in education. *WIREs Data Mining and Knowledge Discovery*, 9(6):e1332, 2019. doi: https://doi.org/10.1002/widm.1332.

Morton Ann Gernsbacher. *Language comprehension as structure building*. Psychology Press, 2013.

Jana Gonnermann-Müller, Jennifer Haase, Nicolas Leins, Thomas Kosch, and Sebastian Pokutta. Llm-based educational simulation: Evaluating temporal student persona stability across adhd profiles. *arXiv preprint arXiv:2605.06307*, 2026.

Junzhi Han and Jinho D Choi. Beyond linear digital reading: An llm-powered concept mapping approach for reducing cognitive load. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pp. 805–817, 2025.

Dawnene D Hassett and Jen Scott Curwood. Theories and practices of multimodal education: The instructional dynamics of picture books and primary classrooms. *The Reading Teacher*, 63(4): 270–282, 2009.

Mengting Hu, Honglei Guo, Shiwan Zhao, Hang Gao, and Zhong Su. Efficient mind-map generation via sequence-to-graph and reinforced graph refinement. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 8130–8141, 2021.

Cheng Hua and Stefanie A Wind. Exploring the psychometric properties of the mind-map scoring rubric. *Behaviormetrika*, 46(1):73–99, 2019.

Parag Jain, Andreea Marzoca, and Francesco Piccinno. Structsum generation for faster text comprehension. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7876–7896, 2024.

Ye Jia, Chen Li, Zackary PT Sin, Xiangzhi Eric Wang, Jiongning Lian, Peter HF Ng, Xiao Huang, George Baciu, Jiannong Cao, and Qing Li. Towards ai-assisted immersive learning: factor analysis of learning effect in k-cube edu-metaverse. In *2025 IEEE Smart World Congress (SWC)*, pp. 1795–1802. IEEE, 2025.

B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Proceeding Visualization '91*, pp. 284–291, 1991. doi: 10.1109/VISUAL.1991.175815.

Vinothini Kasinathan and Aida Mustapha. A presentation mining framework: From text mining to to mind mapping. In *International Conference on Soft Computing and Data Mining*, pp. 233–243. Springer, 2024.

Petr Kedaj, Josef Pavlíček, Petr Hanzlík, et al. Effective mind maps in e-learning. *Acta Informatica Pragensia*, 3(3):239–250, 2014.

Dong Won Lee, Chaitanya Ahuja, Paul Pu Liang, Sanika Natu, and Louis-Philippe Morency. Lecture presentations multimodal dataset: Towards understanding multimodality in educational videos. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20030–20041, 2023. doi: 10.1109/ICCV51070.2023.01838.

Irene Li, Alexander R Fabbri, Robert R Tung, and Dragomir R Radev. What should i learn first: Introducing lecturebank for nlp education and prerequisite chain learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 6674–6681, 2019.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024.

Coey Low, Vinothini Kasinathan, and Aida Mustapha. Boommapper: An innovative web-based system for transforming powerpoint presentations into mind maps. In *International Workshop on Learning Technology for Education Challenges*, pp. 128–139. Springer, 2025.

George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.

Tomoko Nemoto and David Beglar. Likert-scale questionnaires. In *JALT 2013 conference proceedings*, volume 108, pp. 1–6, 2014.

Songjie Niu, Kaisen Yang, Rui Zhao, Yichao Liu, Zonglin Li, Hongning Wang, and Wenguang Chen. Tree-kg: An expandable knowledge graph construction framework for knowledge-intensive domains. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18516–18529, 2025.

Edward J O'Brien, Michelle L Rizzella, Jason E Albrecht, and Jennifer G Halleran. Updating a situation model: a memory-based text processing view. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 24(5):1200, 1998.

Jamal Raiyn. The role of visual learning in improving students' high-order thinking skills. *Journal of education and practice*, 7(24):115–121, 2016.

Romil Rawat. Logical concept mapping and social media analytics relating to cyber criminal activities for ontology creation. *International Journal of Information Technology*, 15(2):893–903, 2023.

Pranab Kumar Sen. Estimates of the regression coefficient based on kendall's tau. *Journal of the American Statistical Association*, 63:1379–1389, 1968. URL https://api.semanticscholar.org/CorpusID:122323572.

Rand J. Spiro. Cognitive flexibility theory : advanced knowledge acquisition in ill-structured domains. 1988. URL https://api.semanticscholar.org/CorpusID:54151398.

John Sweller. Cognitive load theory. In *Psychology of learning and motivation*, volume 55, pp. 37–76. Elsevier, 2011.

John Sweller, Jeroen J. G. van Merriënboer, and Fred Paas. Cognitive architecture and instructional design: 20 years later. *Educational Psychology Review*, 31:261 – 292, 2019.

Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 13636–13645, 2023.

Anthony J. Viera and Joanne Mills Garrett. Understanding interobserver agreement: the kappa statistic. *Family medicine*, 37 5:360–3, 2005.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, et al. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*, 2024a.

Fei Wang, Weibo Gao, Qi Liu, Jiatong Li, Guanhao Zhao, Zheng Zhang, Zhenya Huang, Mengxiao Zhu, Shijin Wang, Wei Tong, et al. A survey of models for cognitive diagnosis: New developments and future directions. *arXiv preprint arXiv:2407.05458*, 2024b.

Shu Wang, Yingli Zhou, and Yixiang Fang. Bookrag: A hierarchical structure-aware index-based approach for retrieval-augmented generation on complex documents. *arXiv preprint arXiv:2512.03413*, 2025.

Jiongen Xiao and Zifeng Zhang. Eduvqa: A multimodal visual question answering framework for smart education. *Alexandria Engineering Journal*, 122:615–624, 2025.

Renjun Xu and Yang Yan. Agent skills for large language models: Architecture, acquisition, security, and the path forward. *arXiv preprint arXiv:2602.12430*, 2026.

Ji-Fan Yu, Daniel Zhang-Li, Zhe-Yuan Zhang, Yu-Cheng Wang, Hao-Xuan Li, Joy Jia Yin Lim, Zhan-Xin Hao, Shang-Qing Tu, Lu Zhang, Xu-Sheng Dai, et al. From mooc to maic: reimagine online teaching and learning through llm-driven agents. *Journal of Computer Science and Technology*, 41(1):394–414, 2026.

Jifan Yu, Chenyu Wang, Gan Luo, Lei Hou, Juanzi Li, Zhiyuan Liu, and Jie Tang. Course concept expansion in moocs with external knowledge and interactive game. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4292–4302. Association for Computational Linguistics, 2019. doi: 10.18653/v1/p19-1421.

Jifan Yu, Gan Luo, Tong Xiao, Qingyang Zhong, Yuquan Wang, Wenzheng Feng, Junyi Luo, Chenyu Wang, Lei Hou, Juan-Zi Li, Zhiyuan Liu, and Jie Tang. Mooccube: A large-scale data repository for nlp applications in moocs. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

Enming Zhang, Yuzhe Li, Yuliang Liu, Yingying Zhu, and Xiang Bai. Towards comprehensive lecture slides understanding: Large-scale dataset and effective method. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4455–4464, 2025a. doi: 10.1109/ICCV51701.2025.00424.

Haopeng Zhang, Philip S Yu, and Jiawei Zhang. A systematic survey of text summarization: From statistical methods to large language models. *ACM Computing Surveys*, 57(11):1–41, 2025b.

Mingtian Zhang, Yu Tang, and PageIndex Team. Pageindex: Next-generation vectorless, reasoning-based rag. *PageIndex Blog*, September 2025c. https://pageindex.ai/blog/pageindex-intro.

Zhuowei Zhang, Mengting Hu, Yinhao Bai, and Zhen Zhang. Coreference graph guidance for mind-map generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19623–19631, 2024.

Daniel Zhang-Li, Zheyuan Zhang, Jifan Yu, Joy Jia Yin Lim, Shangqing Tu, Linlu Gong, Haohua Wang, Zhiyuan Liu, Huiqin Liu, Lei Hou, et al. Awaking the slides: a tuning-free and knowledge-regulated ai tutoring system via language model coordination. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 2872–2883, 2025.

Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities. *World Wide Web*, 27(5):58, 2024.

CONTENTS

1 Introduction **1**

2 S2M-Bench **3**

2.1 Problem Definition 3

2.2 Dataset Construction and Quality Control 3

2.3 Evaluation Metrics 4

2.3.1 Ground-Truth (GT)-based Comparison 4

2.3.2 Quantitative Structure Conformity 4

2.3.3 VLM-as-a-Judge 5

3 AutoMindMap **5**

3.1 Skeleton Laying 5

3.2 Iterative Knowledge Integration 6

3.3 Dual-Stage Refinement 7

4 Experiments **7**

4.1 Experiment Setup 7

4.1.1 Baselines. 7

4.1.2 Implementation. 8

4.2 Experiment Results 8

4.2.1 Main Experiments. 8

4.2.2 Ablation Study. 9

4.2.3 Further Analysis. 9

5 Conclusion **9**

A Related Work **15**

B Limitations and Future Work **16**

C Dataset Details **18**

D More Experiment Settings **18**

E Evaluation Metrics **18**

E.1 Procedure for Concept Nodes Match 18

E.2 Calculation of GT-based Metrics 19

E.3 Detailed Criteria for VLM-Judge 20

F More Experiments **21**

F.1 Performance on Other LLMs 21

F.2 Performance Across Different Course Categories 21
F.3 Cost Analysis 22
F.4 Stability Analysis 23
F.5 Detailed Results of Ablation Study 23
F.6 Detailed Results of GT-based Metrics 23

G Evaluation Reliability Analysis **24**
G.1 Human-Model Evaluation Agreement 24
G.2 VLM-Judge with Other Models 24
G.3 Correlation Between Different Metrics 24
G.4 Analysis on GT-based Metrics 25

H More Structural Comparison **27**

I Mind Map Visualization Case Study and Failure Analysis **28**
I.1 Analysis for Main Comparison Experiments 28
I.2 Analysis for Ablation Study 29

J Tool Definitions for Mind Map Modification **29**

K Important Cognitive Science Theories for Reference **29**

L Prompts **34**
L.1 Prompts of AutoMindMap Framework 34
L.2 Prompts of Baselines 39
L.3 Prompts of Evaluation 40

A Related Work

Educational Knowledge Extraction and Mind Map. Extracting structured knowledge from pedagogical materials facilitates numerous intelligent education applications (Yu et al., 2020), including cognitive diagnosis (Wang et al., 2024b), recommendation (Fabbri et al., 2018), question answering (Xiao & Zhang, 2025), lecture planning (Zhang-Li et al., 2025), and visual learning (Dhindsa & Roger Anderson, 2011). Prior studies have explored prerequisite relations (Li et al., 2019), concept maps (Rawat, 2023), mind maps (Zhang et al., 2024), educational knowledge graphs (Niu et al., 2025), and concept expansion (Yu et al., 2019), etc., and the development of recent large language models (LLMs) further facilitates these tasks (Zhang-Li et al., 2025; Han & Choi, 2025; Abu-Rasheed et al., 2025). For mind map generation, previous studies are few and focus on the identification of concept dominance relationships from simplified or well-structured text; the evaluations are limited to accuracy-based or downstream task-oriented metrics (Zhang et al., 2024; Hu et al., 2021; Ferreira-Mello et al., 2019; Ding et al., 2024). These studies present limited generalization and scalability, especially in more open-ended generation and evaluation. Although the Human-Computer Interaction field has more related studies, they only use LLMs as end-to-end mind map generators and are limited to statistical evaluation on user experiments (Gonnermann-Müller et al., 2026; Han & Choi, 2025). In conclusion, there is still a lack of focus on automated mind map generation in large-scale slide scenarios, and more comprehensive evaluation methods considering knowledge faithfulness, structure compliance, task openness, and cognitive efficiency. These gaps remain critical in current educational scenarios where slides dominate.

Table 3: Comparisons among different lecture slides datasets.

Datasets	Decks	Pages	Tasks
LectureBank (Li et al., 2019)	1,352	51,939	Prerequisite Graph Construction
SlideVQA (Tanaka et al., 2023)	2,619	52,380	Visual Question Answering
LPM (Lee et al., 2023)	334	9,031	Image Retrieval
M^{3}AV (Chen et al., 2024b)	1,113	24,936	Multimodal Video Understanding
LecSlides-370K (Zhang et al., 2025a)	25,542	370,078	Visual Question Answering & Summary
S2M-Bench	267	12,774	Mind Map Construction

Lecture Slides Datasets. Several datasets have been proposed for lecture slide-based tasks, which mainly involve pedagogical applications, such as structured knowledge extraction (Li et al., 2019), multimodal content understanding (Lee et al., 2023), and question answering (Tanaka et al., 2023). Among representative studies, LectureBank (Li et al., 2019) aims at prerequisite graph learning; SlideVQA (Tanaka et al., 2023) and LecSlides-370K (Zhang et al., 2025a) target visual question answering and summarization over thousands of slides; LPM (Lee et al., 2023) and M^{3}AV (Chen et al., 2024b) introduce other multimodal data sources like videos, audio, textbooks, etc., to support multimodal retrieval and understanding. Table 3 summarizes the comparisons among these datasets. These datasets treat slides as a corpus or visual sources for retrieval, instead of framing knowledge structuring as a standalone generation task. S2M-Bench fills this gap by providing expert-annotated mind maps as a structured knowledge abstraction over each entire course, which is the first benchmark that systematically evaluates **automated mind map generation**, a human visual-learning-targeted hierarchical knowledge reconstruction task from authentic, large-scale slide collections.

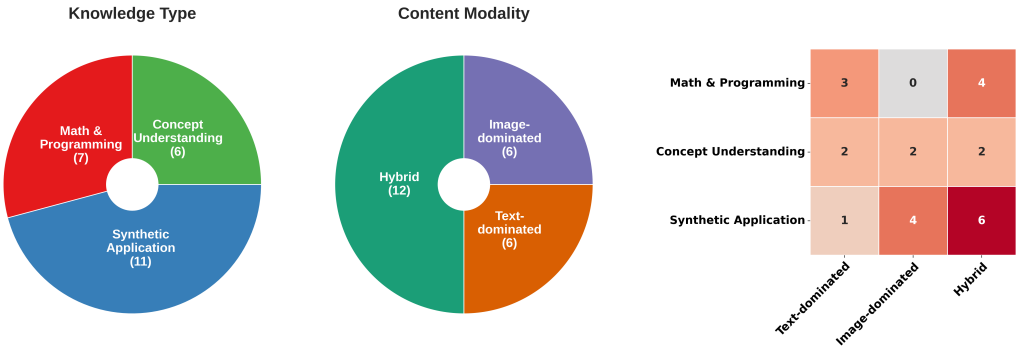
Document Hierarchy Indexing. Recent methods for table-of-contents (TOCs) style document hierarchy indexing (DHI), such as Tree-KG (Niu et al., 2025), PageIndex (Zhang et al., 2025c), and BookRAG (Wang et al., 2025), preserve the native hierarchy of well-structured documents by exploiting explicit formatting cues. While these methods can produce structures visually reminiscent of mind maps, their reliance on formatting signals renders them brittle when applied to slides, due to the systematic misalignment between surface document organization and the intrinsic concept hierarchical structure. Thus, retrieval-oriented DHI methods are ill-suited for pedagogical mind map generation. A fundamentally different generation paradigm is required.

B Limitations and Future Work

Benchmark Comprehensiveness. Due to the high cost of manual labeling and the limited access to multi-disciplinary educational resources, S2M-Bench only includes a modest-scale dataset with a focus on computer science and mathematics. Future work can further expand the dataset to more subjects. However, we still categorize the dataset based on multiple dimensions, which alleviates the data insufficiency problem. Besides, given the high cost and potential ethical concerns, we haven't conducted real-learner intervention experiments to verify the "cognitive efficiency", only break down this goal into several rubrics that are easy to quantify or assess based on cognitive science theories, to conduct evaluation. We recognize that real user interaction experiments can better demonstrate the practical application effect of the method. Therefore, we hope that researchers in the field of human-computer interaction can conduct further studies.

Generation Efficiency. AutoMindMap does not show a particularly high advantage in cost efficiency (e.g., token consumption and running time), while also focusing only on individual course generation. Future work can further propose more token-saving and faster generation methods, and introduce self-evolution mechanisms to allow agents to reuse the generation experience in different courses.

The Openness of the Task Definition. In the educational field, the visual design effect of mind maps is also a key evaluation indicator. However, Slides2MindMap is only defined within concept structure organization, and excludes visual rendering modules (i.e., visual rendering is implemented with external tools instead of an internal module). Future work can consider building mind maps

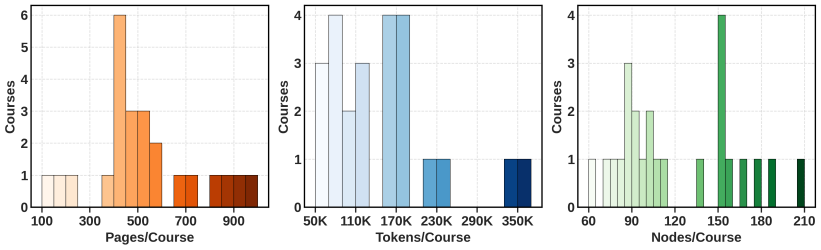


(a) Course categories.

(b) Category overlaps.

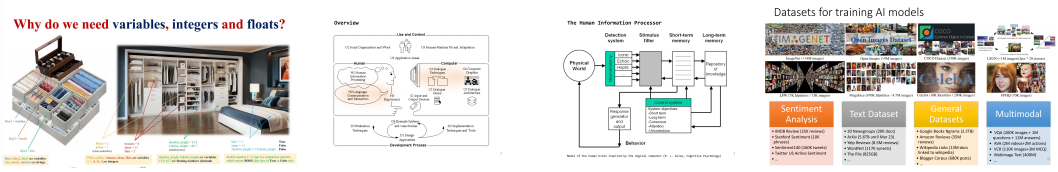


(c) Word cloud of the concepts in S2M-Bench.

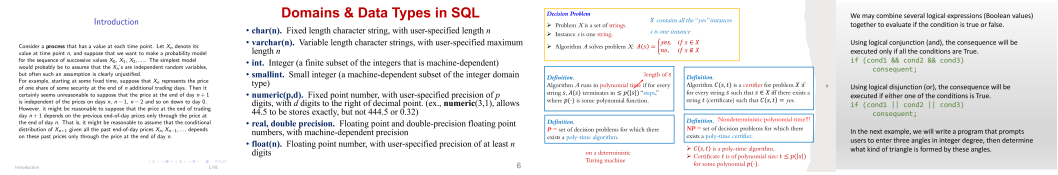


(d) Distribution of pages, tokens, and concept nodes per course.

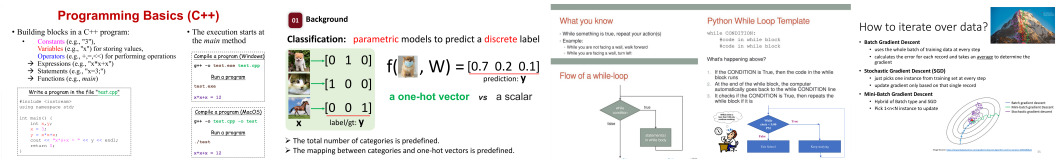
Figure 6: Further Statistic of S2M-Bench.



(a) Image-dominated Slides.



(b) Text-dominated Slides.



(c) Hybrid Slides.

Figure 7: Examples of different categories in terms of content modality.

with a more open form that involves images, annotations, links, and typography design, etc. The task can be further modeled as a text-to-image generation task.

Dependence on Document Parsing Tools. The generation quality of AutoMindMap depends on the parsing results of MinerU, which is an inevitable drawback of this work. However, parsing the layout of PDF documents is a separate task with its own challenges. We still hope that document-native end-to-end frameworks can be proposed.

Table 4: Description of each course category.

Category	Description
Math & Programming	Encompasses pure programming and foundational mathematics, featuring numerous examples and exercises, e.g., Linear Algebra, Calculus, C++ Programming, Python Programming.
Concept Understanding	Centers on the memorization and comprehension of concepts and principles, with minimal computation or programming, e.g., Operating System, Computer Organization and Architecture.
Synthetic Application	Integrates conceptual understanding and computational tasks, oriented toward specific application scenarios, e.g., Stochastic Processes for Investment, Machine Learning.
Image-dominated	Primarily visual, with images (including formulas and charts) as the main elements; text appears only as captions or short phrases.
Text-dominated	Comprises extensive textual paragraphs or long sentences across most slides.
Hybrid	Contains both substantial text blocks and significant image elements in comparable proportions.

C Dataset Details

We further provide statistics on the dataset collected in S2M-Bench. S2M-Bench includes a total of 24 courses, 267 decks, 12,774 pages, and 3,571,497 tokens. The ground truth (GT) mind maps are constructed by human annotators for each course, with a total of 2,913 concept nodes. In Figure 6, we visualize the course category distribution, category overlaps, word cloud, and the distribution of pages/course, tokens/course, and concept-nodes/course in human-labeled ground truth. In Figure 7, we select some examples for each category in terms of the content modality. We also illustrate the principle of categorization in Table 4. The categorization of content modality is based on the overall style of all the slides of a course, rather than a single slide.

D More Experiment Settings

For the baselines, the original implementation of BookRAG and PageIndex only uses the original titles as the index names and neglects the name conciseness, which is unfair in our evaluation method. Thus, we further rephrase the node names of their results into concise concept representations through LLMs (i.e., the backbone models).

We conduct experiments on a server with 64 Intel(R) Xeon(R) Gold 6426Y CPUs and 8 NVIDIA GeForce RTX 4090 GPUs. All the LLM calls are through the official API of OpenAI and DeepSeek. All text embeddings used in our experiments are based on **text-embedding-3-small**, and all image rendering, including the visual refinement stage and VLM-Judge evaluation, are based on **Graphviz**. The output DPI is set to 35 (for refinement) and 100 (for evaluation), respectively. In cross-branch retrieval, $K = 3$ and $\gamma_{sim} = 0.6$. The refinement is set to two rounds. The LLM temperature is 0.1 during generation, and 0 during evaluation. All the prompts used for AutoMindMap, baselines, and evaluation are provided in the Appendix.

E Evaluation Metrics

E.1 Procedure for Concept Nodes Match

A prerequisite for GT-based metrics is to establish a matched node-pair set $\mathcal{C} \subseteq V^{\text{pred}} \times V^{\text{gt}}$ that aligns concepts with semantically equivalent meanings despite surface-form variations. To accommodate the inherent diversity in concept naming (e.g., abbreviations, synonyms, and paraphrases), we design a multi-stage matching algorithm that progressively refines matches while preserving precision. The algorithm initializes two candidate pools: $\mathcal{P}_{\text{gt}} = V^{\text{gt}}$ and $\mathcal{P}_{\text{pred}} = V^{\text{pred}}$. At each

successful match, the matched pair is added to $\mathcal{C}$, and both nodes are removed from their respective pools (i.e., each node can be matched only once).

The matching proceeds in three stages: **(1) Exact match.** We perform case-insensitive, singular/plural-normalized string equality between nodes in $\mathcal{P}_{\text{gt}}$ and $\mathcal{P}_{\text{pred}}$. This captures identical concept names with minor inflectional variants. **(2) Abbreviations match.** For each remaining ground-truth node $v^{\text{gt}} \in \mathcal{P}_{\text{gt}}$, we invoke an LLM to examine its name and parent/child context. The LLM is prompted to infer potential full forms or common abbreviations (e.g., "ML" and "Machine Learning") and decide on a match. This stage handles semantic equivalence that is not captured by surface similarity. **(3) LLM-assisted similarity match.** For the remaining $v^{\text{gt}} \in \mathcal{P}_{\text{gt}}$, we compute two complementary similarity scores against every remaining predicted node $v^{\text{pred}} \in \mathcal{P}_{\text{pred}}$: **word-level Jaccard similarity**, and **cosine similarity of text embeddings**. For each similarity measure, we retain the top-k predicted nodes (with $k = 3$) whose score exceeds a threshold $\tau = 0.6$, forming a candidate pool of at most $2k$ candidates. The LLM then reviews these candidates along with their structural neighbors (parent/child nodes) to select the most appropriate match or reject all. If a match is accepted, the pair is added to $\mathcal{C}$ and removed from the pools; otherwise, the ground-truth node remains unmatched and is ultimately discarded. The LLM prompts are provided in Appendix L.3.

E.2 Calculation of GT-based Metrics

HN-F1. Let $\mathcal{C} \subseteq V^{\text{pred}} \times V^{\text{gt}}$ be the matched node-pair set. For each pair $(v_i^{\text{gt}}, v_i^{\text{pred}}) \in \mathcal{C}$, define

$$w(v_i^{\text{gt}}, v_i^{\text{pred}}) = 1 - \left| \frac{d(v_i^{\text{gt}})}{d^{\text{gt}}} - \frac{d(v_i^{\text{pred}})}{d^{\text{pred}}} \right|, \tag{7}$$

where $d(\cdot)$ is the distance from the root and $d^{\text{gt}} = \max_{v \in V^{\text{gt}}} d(v)$, $d^{\text{pred}} = \max_{v \in V^{\text{pred}}} d(v)$. Then the corresponding precision, recall, and the F1-score are calculated by

$$\text{HN-P} = \frac{\sum_{(v_i^{\text{gt}}, v_i^{\text{pred}}) \in \mathcal{C}} w(v_i^{\text{gt}}, v_i^{\text{pred}})}{|V^{\text{pred}}|}, \tag{8}$$

$$\text{HN-R} = \frac{\sum_{(v_i^{\text{gt}}, v_i^{\text{pred}}) \in \mathcal{C}} w(v_i^{\text{gt}}, v_i^{\text{pred}})}{|V^{\text{gt}}|}, \tag{9}$$

$$\text{HN-F1} = 2 \cdot \frac{\text{HN-P} \cdot \text{HN-R}}{\text{HN-P} + \text{HN-R}}. \tag{10}$$

EC-F1. An edge $(v_i^{\text{gt}}, v_j^{\text{gt}}) \in E^{\text{gt}}$ is *covered* if both endpoints are matched in $\mathcal{C}$ and there exists a directed path from v_i^{pred} to v_j^{pred} in $\mathcal{M}^{\text{pred}}$. Coverage for $(v_i^{\text{pred}}, v_j^{\text{pred}}) \in E^{\text{pred}}$ is defined symmetrically. Then:

$$\text{EC-P} = \frac{|\{e \in E^{\text{pred}} \mid e \text{ is covered}\}|}{|E^{\text{pred}}|}, \tag{11}$$

$$\text{EC-R} = \frac{|\{e \in E^{\text{gt}} \mid e \text{ is covered}\}|}{|E^{\text{gt}}|}, \tag{12}$$

$$\text{EC-F1} = 2 \cdot \frac{\text{EC-P} \cdot \text{EC-R}}{\text{EC-P} + \text{EC-R}}. \tag{13}$$

MEC-F1. Let $E^{\text{gt}}_{\text{map}} = \{(u, v) \in E^{\text{gt}} \mid u, v \text{ are matched in } \mathcal{C}\}$ and define $E^{\text{pred}}_{\text{map}}$ analogously. Then:

$$\text{MEC-P} = \frac{|\{e \in E^{\text{pred}}_{\text{map}} \mid e \text{ is covered}\}|}{|E^{\text{pred}}_{\text{map}}|}, \tag{14}$$

$$\text{MEC-R} = \frac{|\{e \in E^{\text{gt}}_{\text{map}} \mid e \text{ is covered}\}|}{|E^{\text{gt}}_{\text{map}}|}, \tag{15}$$

$$\text{MEC-F1} = 2 \cdot \frac{\text{MEC-P} \cdot \text{MEC-R}}{\text{MEC-P} + \text{MEC-R}}. \tag{16}$$

E.3 Detailed Criteria for VLM-Judge

The detailed scoring criteria for the three dimensions of VLM-Judge are listed in Table 5. All VLM-Judge evaluations, as well as human evaluations (Appendix G.1), are based on the criteria and Likert scales.

Table 5: Detailed criteria for VLM-Judge. These criteria are based on cognitive science theories in Appendix K.

Dimensions	Criteria

Content Accuracy	– All extracted concept names should be concise and accurate, generally proper nouns/noun phrases. The following situations should be avoided:
	  – Single abbreviations and broad words (e.g., introduction, motivation, application, example) without referents;
	  – Uninformative modifier words/section indexes;
	  – Long sentences, questions, and expressions without summarization or compression;
	  – One-phrase concatenation of multiple concepts.
	– Reasonable hierarchy and affiliation relation between parent and child concept, without confusing relation between parent-child/siblings.
	– Concepts should be representative and recapitulatory, rather than lots of unimportant, course-irrelevant, noisy entities or mutually overlapped/similar concepts.

Structure Quality	– Poor structures should be avoided, including:
	  – Too dense and disordered relations without clear organization;
	  – Overload child nodes under one parent without hierarchy design, or cascaded long one-child chains;
	  – Too deep (with more than 7 levels) or shallow flat hierarchy (with only 2–3 levels);
	  – Extremely unbalanced branch scale/depth;
	  – Multiple roots, ring/bi-directional edges, isolated nodes.
	– All local concepts and relations are organized into an overall coherent logic, with meaningful taxonomy and progressive hierarchy.
	– Allow for proper structural flexibility and scale difference between branches, rather than a completely neat structure.

Pedagogical Efficiency	– Proper overall concept scale with clear visual effect on hierarchy, rather than chaotic, dense graphs, or just superficial concepts without delving. – Generalizable core topics and framework are presented obviously, can be captured quickly from the mind map, rather than a laborious search. – Reasonable cross-links across branches that promote concept association, rather than a pure tree. The proportion is proper rather than dense. – Core knowledge should be emphasized structurally (e.g., include more and deeper nodes) rather than all concepts being equally important. – Overall organization is consistent with human cognitive habits, leading to less confusion.

Table 6: Experimental results of different methods, *GPT-5.4-mini* serves as both the backbone model and VLM for refinement in AutoMindMap. Best results are **bolded**, second-best results are underlined.

Methods	GT-based			Structure Conformity		VLM-Judge			
	HN-F1	EC-F1	MEC-F1	Pass@1	SLF↓	CA	SQ	PE	Avg.
Direct	22.18	11.55	47.24	72.73	54.32	2.86	2.36	2.09	2.44
Chunk-Merge	18.20	11.26	57.66	66.67	20.85	3.29	3.00	2.63	2.97
Highlights-only	26.93	15.09	53.15	95.83	45.17	3.21	2.92	2.92	3.01
BookRAG	23.58	9.78	42.44	82.60	22.55	1.74	1.48	1.35	1.52
PageIndex	25.85	8.54	36.25	**100.00**	40.52	2.33	2.00	1.71	2.01
AutoMindMap	**35.39**	**19.59**	**68.48**	**100.00**	**3.29**	**3.54**	**3.67**	**3.63**	**3.61**

Table 7: Performances of different models on the Direct method and AutoMindMap. The best result in the Direct method is underlined, and the global optimum is in **bold**. "DS" abbreviates "DeepSeek".

Methods	Models	GT-based			Structure Conformity		VLM-Judge			
		HN-F1	EC-F1	MEC-F1	Pass@1	SLF↓	CA	SQ	PE	Avg.
Direct	GPT-5.4-mini	22.18	11.55	47.24	72.73	54.32	2.86	2.36	2.09	2.44
	DS-V4-Flash	33.43	20.67	64.50	79.16	20.22	3.17	3.00	2.88	3.01
	DS-V4-Pro	34.98	18.71	68.61	95.83	4.30	3.38	3.38	3.21	3.32
AutoMindMap	GPT-5.4-mini	35.39	19.59	68.48	**100.00**	3.29	3.54	3.67	3.63	3.61
	DS-V4-Flash + GPT-4.1-mini	37.80	22.96	74.01	**100.00**	**1.82**	3.58	3.63	**3.71**	3.64
	DS-V4-Flash + GPT-5.4-mini	**39.88**	**25.12**	**74.34**	**100.00**	1.93	**3.83**	**3.71**	3.63	**3.72**

F More Experiments

We discuss more experimental results, including the performance of AutoMindMap with different LLMs and on different course categories, the cost and stability, the ablation study table, and the detailed results of GT-based metrics. **If not mentioned otherwise, the model setting is DeepSeek-V4-Flash (baselines, and backbone of AutoMindMap) + GPT-4.1-mini (VLM for refinement in AutoMindMap).**

F.1 Performance on Other LLMs

To show the performance of AutoMindMap on different models, we first conduct the comparison experiments on pure GPT-5.4-mini (i.e., used as both the backbone model and VLM-based refinement model); the result is shown in Table 6. Besides, to examine the end-to-end generation performance of the recent advanced reasoning models on the Slides2MindMap task, we run the Direct method with DeepSeek-V4-Pro. The performances of different models in the Direct method and AutoMindMap are summarized in Table 7.

The results show that AutoMindMap still outperforms baseline methods with GPT-5.4-mini, but the performance of GPT-5.4-mini is worse than DeepSeek-V4-Flash when serving as the backbone LLM. A possible reason is that GPT-5.4-mini has limited capabilities in long context scenarios. In addition, more advanced reasoning models like DeepSeek-V4-Pro can significantly outperform other models in the Direct method, but there is still a gap with AutoMindMap (with DeepSeek-V4-Flash), especially in VLM-Judge and structure conformity metrics. Stronger LLMs can extract more faithful knowledge, but fail in further structural organization and pedagogy-oriented generation. The result shows the effectiveness of AutoMindMap.

F.2 Performance Across Different Course Categories

For the different course categories in S2M-Bench, we conduct comparisons in terms of the knowledge types and content modality. The results are shown in Table 8 and Table 9.

AutoMindMap outperforms the Direct method and shows eligible generalization across different categories. However, the performance differences in different categories still exist.

Both the Direct method and AutoMindMap show significant differences in performance on different types of courses. In Table 8, the Direct approach performs best in *math & programming* courses, while performing worst in concept understanding, but AutoMindMap performs best in concept understanding, which shows concept-intensive courses inherently have a more complex conceptual hierarchical structure for direct identification, and AutoMindMap enhances the capacity to identify complex hierarchy.

In Table 9, the differences are comparatively smaller. The Direct method shows close performance across three categories, and AutoMindMap performs better in *image-dominated* slides. The performance might benefit from the parsing effect of MinerU on visual information.

Table 8: Performance comparisons on knowledge type-based categories.

Methods	Categories	GT-based			VLM-Judge			
		HN-F1	EC-F1	MEC-F1	CA	SQ	PE	Avg.
Direct	Math & Programming	**38.92**	20.92	**65.83**	**3.29**	**3.14**	**3.14**	**3.19**
	Concept Understanding	28.42	15.78	63.68	3.00	2.67	2.50	2.72
	Synthetic Application	32.67	**23.17**	64.10	3.18	3.09	2.91	3.06
	Total	33.43	20.67	64.50	3.17	3.00	2.88	3.01
AutoMindMap	Math & Programming	**40.68**	23.85	69.71	3.57	3.57	3.86	3.67
	Concept Understanding	40.23	**25.33**	**76.17**	**3.83**	**4.00**	**4.00**	**3.94**
	Synthetic Application	34.64	21.10	75.57	3.45	3.45	3.45	3.45
	Total	37.80	22.96	74.01	3.58	3.63	3.71	3.64

Table 9: Performance comparisons on content modality-based categories.

Methods	Categories	GT-based			VLM-Judge			
		HN-F1	EC-F1	MEC-F1	CA	SQ	PE	Avg.
Direct	Image-dominated	**40.34**	**23.31**	63.25	**3.17**	2.83	2.67	2.89
	Text-dominated	30.44	19.72	**66.86**	**3.17**	3.00	**3.00**	**3.06**
	Hybrid	31.47	19.82	63.95	**3.17**	**3.08**	2.92	**3.06**
	Total	33.43	20.67	64.50	3.17	3.00	2.88	3.01
AutoMindMap	Image-dominated	**43.33**	**25.54**	73.95	**3.83**	**3.83**	**3.83**	**3.83**
	Text-dominated	34.49	19.51	**75.11**	3.67	3.67	3.67	3.67
	Hybrid	36.69	23.40	73.49	3.42	3.50	3.67	3.53
	Total	37.80	22.96	74.01	3.58	3.63	3.71	3.64

F.3 Cost Analysis

Table 16 lists prompt tokens, completion tokens, and runtime averaged per course. AutoMindMap incurs higher cost than single-pass methods (Direct, Highlights-only) due to its iterative agentic architecture, thus reducing the probability of illegal structure generation to 0 (with pass@1 $= 100\%$), and obviously narrowing the gap to human expert (e.g., average VLM-Judge metrics gap decreases from 0.88 to 0.17 in Table 2, and GT-based metrics gap also decreases in the sampled courses in Table 18). When compared against multi-step iteration methods (e.g., PageIndex), AutoMindMap is substantially more efficient. Besides, the runtime of AutoMindMap is sufficient for offline generation scenarios.

Nevertheless, we should admit that the cost efficiency is not a strength of AutoMindMap, and the quality improvement comes at a non-negligible cost, which is a trade-off inherent to the complexity of the Slides2MindMap task. Developing more cost-efficient generation algorithms remains an important future direction.

Table 10: Cost Analysis. The values are averaged by course. *P* and *C* abbreviate *Prompt* and *Completion*, respectively.

Methods	Tokens (P)	Tokens (C)	Runtime(s)
Direct	160.14K	8.87K	82
Chunk-Merge	178.22K	50.00K	461
Highlights-only	7.60K	14.22K	119
BookRAG	48.63K	17.52K	398
PageIndex	560.36K	114.51K	1482
AutoMindMap	316.84K	64.22K	679

Table 11: Stability analysis for AutoMindMap. The experiments were run 10 times on 3 selected courses at temperature = 0.5.

Indicators	GT-based			Structure Conformity		VLM-Judge			
	HN-F1	EC-F1	MEC-F1	Pass@1	SLF↓	CA	SQ	PE	Avg.
Range	[33.53, 38.78]	[24.55, 28.30]	[80.15, 85.40]	100.00	[1.81, 4.59]	[3, 4]	[3, 4]	[3, 3.67]	[3, 3.67]
Mean	36.72	25.86	82.81	100.00	2.99	3.40	3.40	3.13	3.31
Std.	1.93	1.53	2.26	0.00	0.99	0.14	0.21	0.16	0.14

Table 12: Ablation study results. Best results are **bolded**, second-best are underlined.

Methods	GT-based			Structure Conformity		VLM-Judge			
	HN-F1	EC-F1	MEC-F1	Pass@1	SLF↓	CA	SQ	PE	Avg.
w/o vision	34.69	20.99	73.50	95.83	**1.80**	3.50	3.50	3.38	3.46
w/o tools	30.85	16.15	70.21	**100.00**	3.76	3.29	3.25	3.25	3.26
w/o refinement	35.09	18.40	64.20	**100.00**	2.34	3.17	3.25	2.96	3.13
w/o summary	31.46	18.44	**76.66**	**100.00**	2.02	3.38	3.17	3.00	3.18
w/o skeleton	35.30	16.15	66.56	91.67	2.99	3.08	3.08	3.04	3.07
AutoMindMap	**37.80**	**22.96**	74.01	**100.00**	1.82	**3.58**	**3.63**	**3.71**	**3.64**

F.4 Stability Analysis

To assess generation stability, we select three courses spanning different categories and run AutoMindMap 10 times per course with temperature raised to 0.5 to increase randomness. Table 11 reports the range, mean, and standard deviation of the results. **Range** averages the best and the worst values of each course, rather than the values averaged over a single run of all courses. The results show that multiple runs can still maintain eligible generation robustness even with increased randomness. In particular, Pass@1 shows reliable structure quality.

F.5 Detailed Results of Ablation Study

The results of the ablation study are detailed in Table 12, corresponding to Figure 4.

F.6 Detailed Results of GT-based Metrics

As GT-based metrics HN-F1, EC-F1, and MEC-F1 cannot fully reflect the recall and precision of concept nodes and edges, we further list the full metrics in terms of comparison with the GT data in Table 13 (comparison experiment) and Table 14 (ablation study). N-P, N-R, and N-F1 are the node precision, recall, and F1-score without hierarchy-aware weights.

The two tables show a trade-off between precision and recall in terms of both nodes and edges. A detailed empirical statistic can be found in Figure 8b. The tables also indicate that the baselines can achieve high knowledge recall or precision, but cannot strike a balance and maintain global coherence. Some methods (e.g., Chunk-Merge) focus on too fine-grained knowledge, which may lead to lower precision, and overly conservative methods (e.g., Highlights-only) may lead to lower recall. In the ablation study, the trade-off also exists. Specifically, the node recall drops after refinement, but other metrics increase, which means a better balance between precision and recall, but also erroneously removes some nodes to streamline the structure.

Besides, AutoMindMap cannot achieve the best performance in all the detailed metrics, but remains top-2 in most of the metrics, and can maintain a good balance between recall and precision. Compared with N-P, N-R, and N-F1, AutoMindMap performs even better than baselines in the metrics with hierarchy-aware weights, which further confirms that AutoMindMap can not only extract appropriate concepts from slides but also identify the hierarchies.

Table 13: Full GT-based metrics in comparison experiment. The model combination is DeepSeek-V4-Flash + GPT-5.4-mini.

Methods	N-P	N-R	N-F1	HN-P	HN-R	HN-F1	EC-P	EC-R	EC-F1	MEC-P	MEC-R	MEC-F1
Direct	**44.77**	49.96	46.06	32.62	36.19	33.43	**24.36**	19.55	20.67	75.02	58.93	64.50
Chunk-Merge	26.45	59.64	34.22	19.43	43.33	25.03	12.80	29.17	16.21	76.31	62.83	68.10
Highlights-only	39.63	30.16	32.03	29.31	22.23	23.63	23.61	11.92	14.50	**84.10**	57.50	66.25
BookRAG	26.03	52.51	32.20	20.39	40.41	25.12	9.47	13.68	10.12	73.01	37.74	47.85
PageIndex	23.45	60.08	29.88	18.98	48.17	24.15	11.08	20.30	11.66	81.50	45.83	57.11
AutoMindMap	41.31	**61.21**	**47.69**	**34.65**	**50.94**	**39.88**	21.32	**34.70**	**25.12**	74.37	**75.61**	**74.34**

Table 14: Full GT-based metrics in ablation study. The model combination is DeepSeek-V4-Flash + GPT-4.1-mini.

Methods	N-P	N-R	N-F1	HN-P	HN-R	HN-F1	EC-P	EC-R	EC-F1	MEC-P	MEC-R	MEC-F1
w/o vision	**42.92**	44.61	42.01	**35.67**	36.57	34.69	**21.39**	23.25	20.99	75.86	72.37	73.50
w/o tools	40.52	36.17	37.38	33.39	29.87	30.85	17.75	16.16	16.15	74.33	68.35	70.21
w/o refinement	35.80	**63.31**	43.89	28.74	**50.36**	35.09	14.80	28.62	18.40	69.33	61.65	64.20
w/o summary	37.19	42.43	37.98	30.93	34.94	31.46	17.42	21.42	18.44	**79.76**	**75.30**	**76.66**
w/o skeleton	38.87	49.65	42.50	32.28	41.24	35.30	13.95	21.58	16.15	71.37	63.29	66.56
AutoMindMap	40.48	57.14	**45.59**	33.68	47.20	**37.80**	19.61	**31.18**	**22.96**	76.66	72.42	74.01

G Evaluation Reliability Analysis

G.1 Human-Model Evaluation Agreement

To demonstrate the evaluation effectiveness of VLM-Judge, we invite two human annotators to score 48 generated results for 8 courses. Human annotators and models are provided with the same criteria, which emphasize the quality of intrinsic structure, content, and pedagogy efficiency, ignoring pure visual rendering aesthetics. The scores are based on the 5-level Likert scale.

For the scoring results, we use the **Quadratic Weighted Kappa Coefficient** (Viera & Garrett, 2005) to measure the consistency. Table 15 shows the results of human-human consistency and human-model consistency. The results demonstrate an eligible evaluation quality of VLM-Judge. Although the potential biases of VLM-Judge (e.g., aesthetic preferences, same-family model preferences) cannot be completely removed, the result already demonstrates effective mitigation. Besides, human-human consistency is slightly higher than human-model consistency. SQ shows the lowest consistency for both categories, indicating greater subjective variability.

G.2 VLM-Judge with Other Models

To further demonstrate the effectiveness of the VLM-Judge and eliminate same-family model preferences, we re-evaluate the results of the main experiment using Qwen3.5-Plus. Although there are small numerical differences, the overall ranking is still highly consistent with the main experiment (Table 2). Human experts still achieve the best results.

G.3 Correlation Between Different Metrics

To analyze the potential correlations among different metrics, we perform empirical statistical analysis on all the data of the main comparison experiment with DeepSeek-V4-Flash + GPT-4.1-mini.

The correlations between GT-based metrics and VLM-Judge metrics are low, and the mutual correlations within the VLM-Judge metrics are larger. Although the overly high correlation may indicate the influence of VLM's bias, the discretization and the too few levels of the scoring, and the inherent consistency of the experimental methods on these metrics could also be potential reasons. We further measure the correlation with **Kendall's Tau coefficient** (Sen, 1968) in Table 17, which indicates that the scoring mechanism is one of the factors leading to high correlation. Besides, the three dimensions target distinct pedagogical aspects (factual, structural, and cognitive), justifying

Table 15: Quadratic Weighted Kappa for consistency checking for VLM-as-a-Judge.

Categories	CA	SQ	PE	Avg.
Human and VLM	0.811	0.774	0.835	0.807
Human and Human	0.862	0.790	0.883	0.845

Table 16: VLM-Judge with *Qwen3.5-Plus* on the main experiment results. **Bold** and underlined results indicate the global optimum and the baseline optimum (except for Human), respectively.

Methods	CA	SQ	PE	Avg.
Human	4.04	3.75	3.88	3.83
Direct	3.50	2.93	3.06	3.16
Chunk-Merge	3.18	2.75	2.69	2.87
Highlights-only	3.38	3.00	3.13	3.17
BookRAG	2.13	1.81	1.81	1.92
PageIndex	2.13	1.94	1.94	2.00
AutoMindMap				
+ *GPT-4.1-mini*	3.80	3.23	3.23	3.42
+ *GPT-5.4-mini*	**3.94**	**3.31**	**3.56**	**3.60**

keeping them separate for fine-grained diagnosis. The high *quadratic weighted Kappa* with human evaluation also confirms that the bias has been effectively mitigated.

Among GT-based metrics, HN-F1 and MEC-F1 show the largest correlation but HN-F1 and MEC-F1 have the weakest. Specifically, HN-F1 and MEC-F1 have an obvious trade-off in the high-value interval of HN-F1. The result is consistent with previous experiments (especially in the ablation study): organizing hierarchies for fewer matching nodes naturally has lower difficulty and leads to higher MEC-F1. The trade-off makes MEC-F1 at risk of being **hacked**, but doesn't mean complete invalidity. MEC-F1 can still reflect the capacity of hierarchy identification more accurately than EC-F1 when the number of matching nodes is similar. The correlation between N-R and N-P is also visualized in Figure 8b; the trade-off is consistent with Appendix F.6. Besides, a single metric is easy to hack (e.g., some baselines achieve high MEC-F1 but have poor overall quality), but multi-facet metrics with low correlation increase the evaluation robustness.

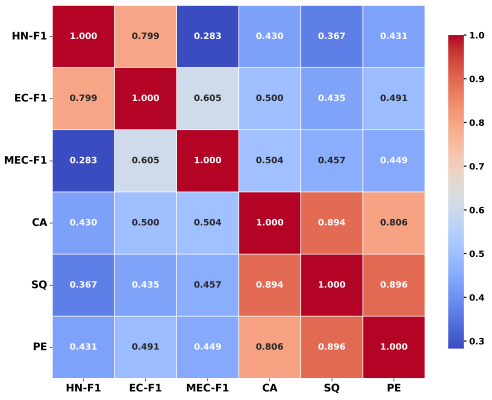
G.4 Analysis on GT-based Metrics

Despite achieving the best performance among automated methods, the absolute GT-based metrics of AutoMindMap remain relatively low. For further analysis, we sample three courses from different categories, and invite new annotators to reconstruct mind maps without referencing the original ground truths. Evaluating these re-annotations against the original GT (Table 18) reveals substantial variability even among human experts, underscoring the inherent open-ended nature of the Slides2MindMap task. Additionally, the strict tuple-level matching rule may penalize semantically valid but structurally divergent hierarchies, further suppressing absolute scores. This analysis confirms that the comparison with ground truth is insufficient, justifying our multi-faceted evaluation framework.

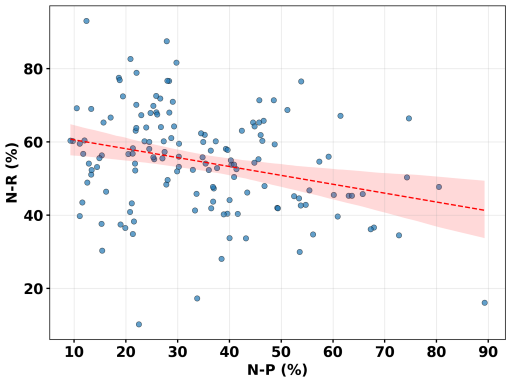
However, the following conclusions are still tenable: (1) Human re-annotations consistently outperform all automated methods and align with the results of VLM-Judge, confirming that GT-based metrics might be influenced by the task openness, but still retain discriminative power for knowledge faithfulness, particularly in the low-score regime. GT-based metrics also prevent other metrics

Table 17: Comparison of Pearson and Kendall's Tau coefficients on VLM-Judge metrics.

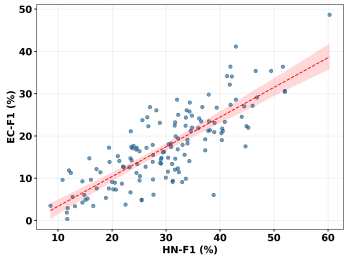
Metrics	CA & SQ	CA & PE	SQ & PE
Pearson	0.894	0.806	0.896
Kendall's Tau	0.843	0.750	0.854



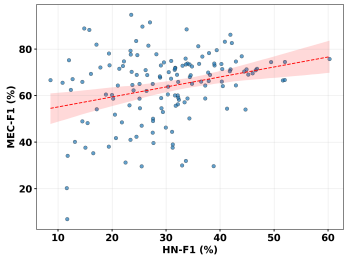
(a) Pearson correlation matrix between GT-based metrics and VLM-Judge metrics.



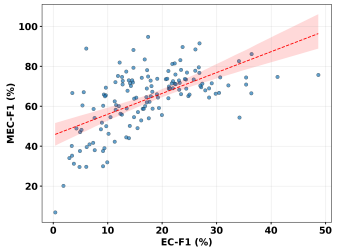
(b) Correlation between N-R and N-P, with a Pearson correlation coefficient of -0.2913.



(c) Correlation between EC-F1 and HN-F1.



(d) Correlation between MEC-F1 and HN-F1.



(e) Correlation between MEC-F1 and EC-F1.

Figure 8: Visualization of correlation among main evaluation metrics, and scatter diagrams.

Table 18: GT-based metrics of human re-annotation, compared with *AutoMindMap* and *Direct*, on the sampled courses. The differences between automated methods and human re-annotation are attached. The model setting is DeepSeek-V4-Flash + GPT-5.4-mini.

Methods	HN-F1	EC-F1	MEC-F1
Direct	40.71 (-20.93)	17.54 (-37.19)	54.03 (-29.64)
AutoMindMap	51.64 (-10.00)	33.45 (-21.28)	66.30 (-17.37)
Human	61.64	54.73	83.67

from being hacked. (2) AutoMindMap substantially narrows the gap to the expert, demonstrating meaningful progress toward expert-level performance in this inherently open-ended task. This gap also confirms the inescapable challenge of the Slides2Mindmap task. There is still much room for improvement in future work.

H More Structural Comparison

We employ more structural indicators to analyze the experimental results (with DeepSeek-V4-Flash + GPT-5.4-mini). These indicators represent the core structural characteristics of the generated mind maps but don't absolutely represent the quality of mind maps; thus, these extra structural comparisons only serve as soft guidelines and a quantified representation of the **SQ** metric for reference.

First, for N-R in Table 13, we count the N-R of each layer's node in the ground truth (with the root node as layer-1, L_1) in Table 19. The N-R values of all methods decrease as the layers become deeper. The deeper nodes in the mind map tend to be less important (Ausubel, 1960), which indicates that all the existing methods can achieve higher knowledge recall on more core concepts, and the Chunk-Merge method has the slowest N-R decline, which is caused by its mechanism that leads to many fine-grained concept extractions.

Grounded by the principle of cognitive balance and the criteria of VLM-Judge, to quantify the balance of branch depth and node distribution, we define a **branch imbalance coefficient** based on the complement of **normalized entropy**. For a node v, let $\mathcal{E}(v) = \{u \mid (v,u) \in E\}$ be the set of direct child. Each $u \in \mathcal{E}(v)$ is described by a scalar indicator $\phi(u)$, including **sub-tree height** $h(u) = 1 + \max_{w \in V(u)}(d(w) - d(u))$ and **sub-tree size** $s(u) = |V(u)|$, where $V(u)$ is the node set of the sub-tree rooted at u. After normalizing the children values into $p_u = \phi(u) / \sum_{w \in \mathcal{E}(v)} \phi(w)$, the local imbalance at v is

$$\eta_\phi(v) = 1 - \frac{-\sum_{u \in \mathcal{E}(v)} p_u \log p_u}{\log |\mathcal{E}(v)|}, \tag{17}$$

to aggregate across the hierarchy, we fix a depth limit L (the number of levels counted from the root). Recall that the root r has $d(r) = 0$, so the node set at level k is $V_k = \{v \in V \mid d(v) = k\}$. The mean imbalance at level k is $\bar{\eta}_\phi^{(k)} = \frac{1}{|V_k|} \sum_{v \in V_k} \eta_\phi(v)$. The global branch imbalance coefficient is the average of the first L level-wise means:

$$\Psi_\phi^{(L)} = \frac{1}{L} \sum_{k=0}^{L-1} \bar{\eta}_\phi^{(k)}. \tag{18}$$

Plugging in $\phi = h$ yields the **branch depth imbalance** $\Psi_{\text{depth}}^{(L)}$; $\phi = s$ yields the **branch size imbalance** $\Psi_{\text{size}}^{(L)}$. $\Psi_\phi^{(L)}$ captures structural skew that average depth/child fan-out alone cannot reveal.

The two coefficients, as well as the node size, cross-link ratio, and the average leaf-node layers in each mind map, are listed in Table 20. Regarding human-annotated data as a benchmark, the balance of all methods is in a reasonable interval (as branches cannot be fully balanced due to the heterogeneity of knowledge organization), while Direct, Highlights-only, and AutoMindMap show stronger balance as a whole. The above three methods also produce significantly more concise node scales that are closer to human data, and AutoMindMap is comparatively worse. Although the mind

Table 19: N-R for different layers.

Methods	Overall	L_2	L_3	L_4	$\geq L_5$
Direct	49.96	63.12	58.59	48.60	42.97
Chunk-Merge	59.64	68.59	64.01	**59.38**	**53.73**
Highlights-only	30.16	50.93	41.72	26.63	21.44
BookRAG	52.21	72.28	58.26	52.57	39.77
PageIndex	60.08	69.98	**68.22**	58.60	50.07
AutoMindMap	**61.21**	**72.31**	58.97	54.35	40.27

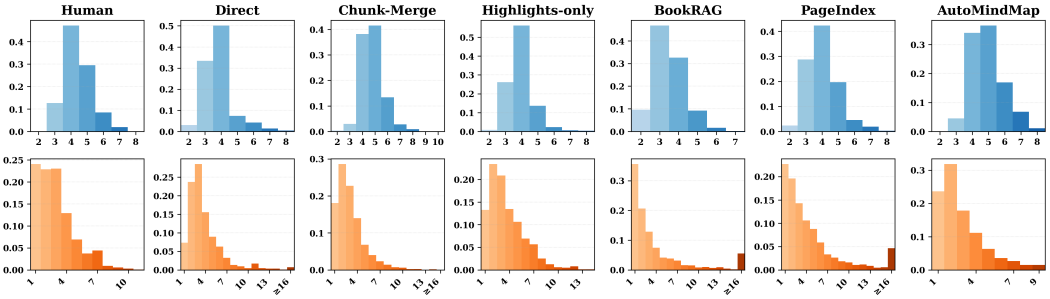


Figure 9: Distributions of leaf-node layers (top) and direct child-node fan-out (bottom) in mind maps generated by different methods. The vertical axis represents the proportion.

map directly generated by LLMs may lack knowledge faithfulness, the structural **balance** and **scale** are closer to human preferences. However, in terms of cross-link ratio and layer number, the preferences of these two methods are farther from human. AutoMindMap performs more conservatively in cross-link ratio, and has the closest performance to human in hierarchical depth.

As average counting cannot reveal extreme values and overall distribution, we further count the distribution of leaf-node depth and child-node fan-out in Figure 9. More intuitively, AutoMindMap shows more similar distributions to Human in both dimensions, while avoiding overly deep and overly many child-node fan-outs. Comparatively, many baselines generate an excess of child nodes (e.g., BookRAG and PageIndex) or depth (e.g., Chunk-Merge), or overly flat layers (e.g., Direct), which indicate poor structures. The result is consistent with the evaluation involving structure quality in the main experiment.

I Mind Map Visualization Case Study and Failure Analysis

We select a course named "Big Data Analytics" as a case and render the results of each method, with the help of Xmind[1]. This course has a total of 12 decks and 449 slide pages. The visualization results of human annotation, baseline methods, and AutoMindMap are shown in Figure 10, and the results of different ablation study settings are shown in Figure 12.

I.1 Analysis for Main Comparison Experiments

The result of AutoMindMap is mostly similar to human annotation, having a clear hierarchical structure, accurate concept description and proper cross-links. The overall map is concise and clear with prominent focuses, which helps to reduce the cognitive load of learners. But shortcomings still exist in these two results. Although human-annotated data maintains good knowledge faithfulness, it is poor in branch balance, which indicates that ground truth is still imperfect, making multifaceted evaluations necessary. AutoMindMap also neglects some key concepts, especially in deeper layers.

The Direct method shows excellent overall structure and logic, but still has too many level-2 branches, the omission of some key concepts, and fewer layers, which are limitations of end-to-end

[1] https://xmind.cn/online-map/

Table 20: More structural characteristics analysis. We set $L = 2$. All the indicators are averaged by courses. Branch imbalance coefficients and cross-link ratio are scaled to 0%-100%.

Methods	$\Psi^{(2)}_{\text{depth}}$	$\Psi^{(2)}_{\text{size}}$	Size	Cross-link Ratio	Layers
Human	0.71	12.17	121.38	9.26	4.40
Direct	0.64	7.16	133.13	27.00	3.82
Chunk-Merge	1.07	11.80	295.08	10.00	4.77
Highlights-only	0.91	9.42	130.67	18.26	3.93
BookRAG	1.08	15.86	294.33	12.51	3.47
PageIndex	0.80	11.84	376.63	14.00	4.02
AutoMindMap	0.91	10.14	185.75	5.53	4.33

long document understanding. Providing only important elements also weakens the understanding of the overall knowledge hierarchy, which leads to the disordered structure and logic of the Highlights-only method. Chunk-Merge, BookRAG, and PageIndex show even worse results. These three methods pay much attention to the local knowledge or mere retrieval-oriented document hierarchy, ignoring the coherence and progressiveness of the overall knowledge logic, resulting in excessive density of child nodes fan-out and cross-links, excessive flattening of the level, and a large number of factual errors, ambiguity, and noise in concept representation. For example, many concepts have "exercise" as their common child in Figure 11c, and lots of titles are directly used as node names without rephrasing.

The above baseline methods may be of high quality in the face of clearly structured textbooks, and BookRAG and PageIndex are efficient in building indexes for RAG systems, but all the above baselines perform poorly in Slides2MindMap, which presents the uniqueness of AutoMindMap: slide-specific document structure adaptation and the global-local trade-off of mind map generation.

I.2 Analysis for Ablation Study

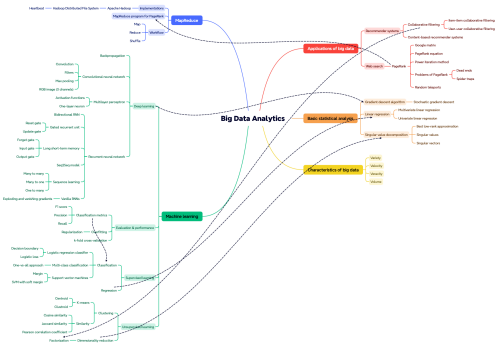
Figure 12 shows that the *refinement mechanism* of AutoMindMap can significantly rebalance branches and prune fine-grained nodes to keep conciseness, and avoid mutual deconstruction of local-global structures. The result without the *visual information* in refinement may lead to a misunderstanding of the mind map structures, resulting in inappropriate modification tool-use and collapsed structure. The result of *w/o summary* shows logic breaks in some branches, and neglects key concept hierarchy extraction due to loss of context-associative bridge. AutoMindMap *w/o skeleton* shows unnatural hierarchical progression, indicating the necessity to understand the overall content and plan the preliminary structure before the formal construction. AutoMindMap *w/o tools* leads to confusing and collapsed local structures, which shows that the tool can promote structural constraints and understanding, and reduce the hallucinations generated by the pure-text way.

J Tool Definitions for Mind Map Modification

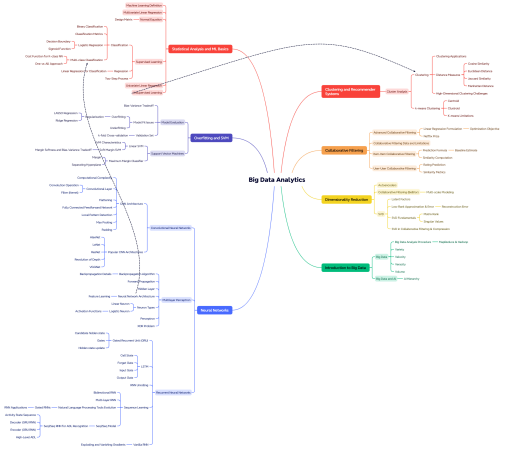
The descriptions of all atomic tools for mind map modification are listed in Table 21. Specs for registered tools are appended to the prompts in AutoMindMap.

K Important Cognitive Science Theories for Reference

Relationship Between Mind Maps and Concept Maps (Davies, 2011). Mind maps and concept maps present similar features; both are widely used in educational scenarios to promote knowledge understanding and organize concepts into divergent structures around a specific center. However, concept maps have stricter and more complex relational constraints than mind maps (i.e., the various relations between concepts should be identified explicitly), which contain more detailed knowledge association, but may cause higher cognitive costs, while mind maps are more flexible in organization and more obvious in hierarchy, facilitating more efficient knowledge overview. Under certain conditions, mind maps can be compatible with the features of concept maps. In our work, although



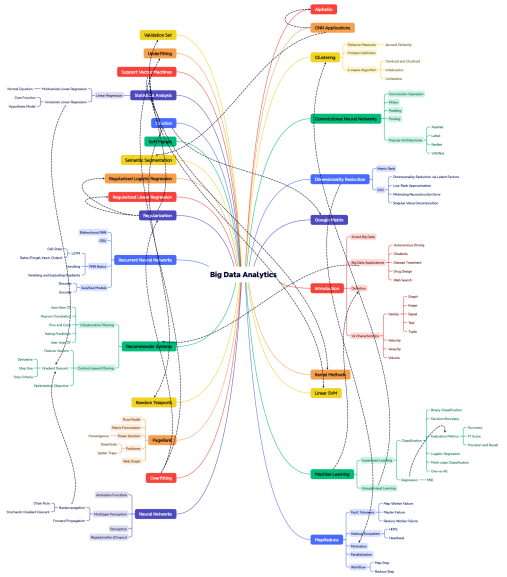
(a) Human.



(b) AutoMindMap.

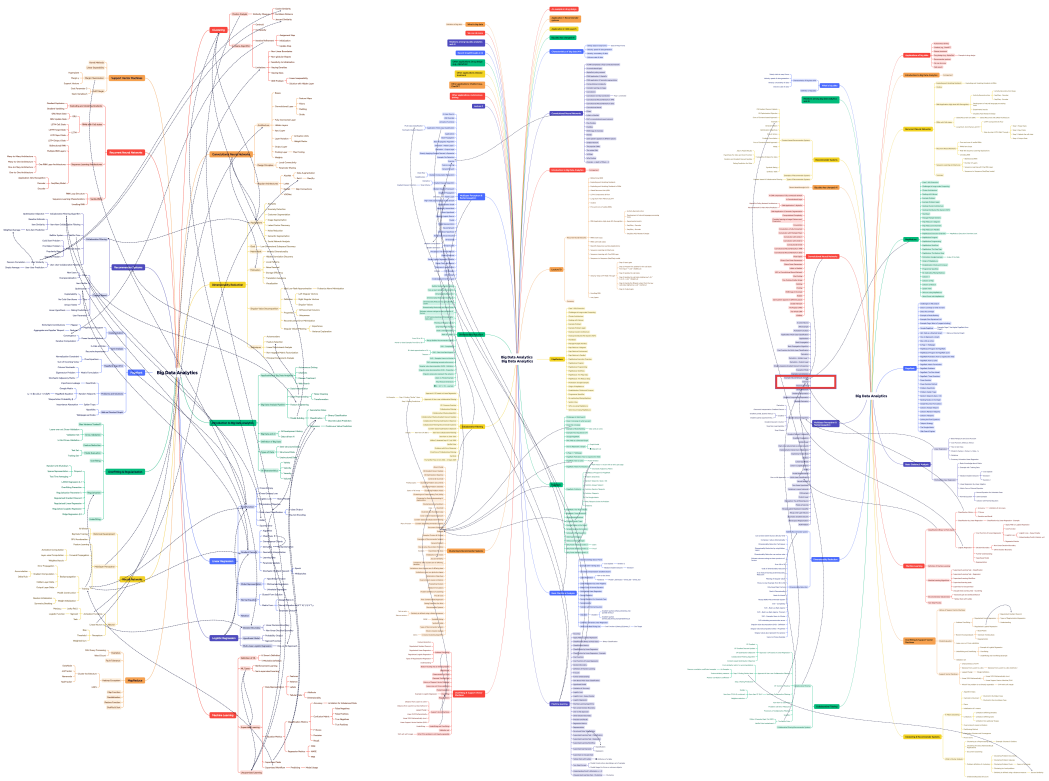


(c) Direct.



(d) Highlights-only.

Figure 10: Visualization of the mind maps generated by different baselines.

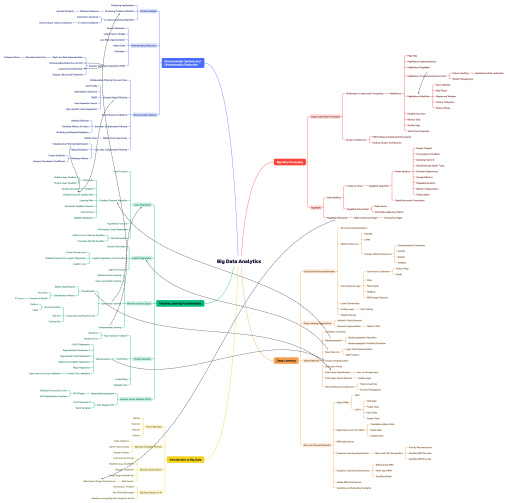


(a) Chunk-Merge. (b) BookRAG. (c) PageIndex.

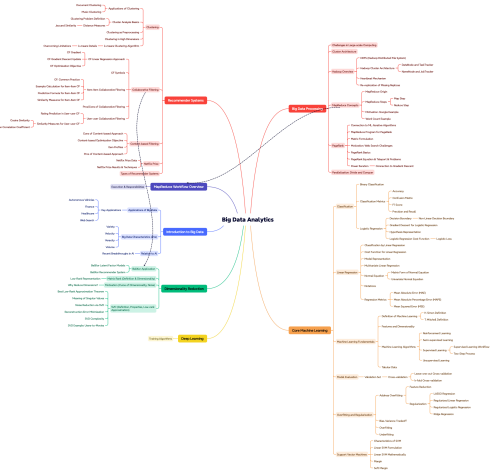
Figure 11: (Continued) Visualization of the mind maps generated by different baselines.

Table 21: Tool specifications for mind map operations.

Tool Name	**Parameters**	**Description**
`rename_node`	`old_name` *(string, required)* `new_name` *(string, required)*	Rename an existing node.
`add_node`	`parent_name` *(string, required)* `new_node_name` *(string, required)* `new_node_description` *(string, optional)*	Add a new node under an existing parent. A brief description incorporating the hierarchical context is encouraged.
`add_edge`	`parent_name` *(string, required)* `child_name` *(string, required)*	Add a new directed edge between two existing nodes.
`delete_edge`	`parent_name` *(string, required)* `child_name` *(string, required)*	Delete an existing directed edge without removing the associated nodes. Can be combined with `add_edge` to reorganize local structure; carries the risk of creating isolated nodes.
`delete_node`	`node_name` *(string, required)*	Delete an existing node along with its incident edges. Avoid use unless the target is a leaf node or other tools are used together, to prevent orphaned child nodes.
`merge_nodes`	`node_names` *(array<string>, required)* `merged_name` *(string, required)*	Merge N existing nodes into a single node while preserving all original edges.



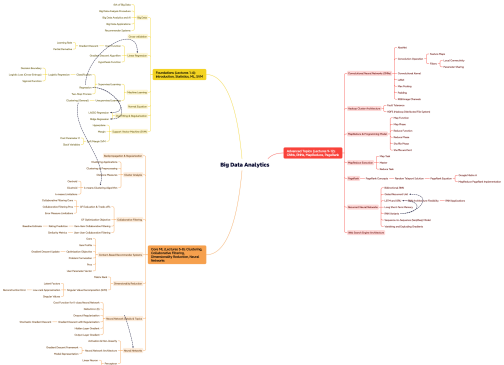
(a) AutoMindMap w/o refinement.



(b) AutoMindMap w/o summary.



(c) AutoMindMap w/o vision.



(d) AutoMindMap w/o skeleton.



(e) AutoMindMap w/o tools.

Figure 12: Visualization of the mind maps generated in the ablation study.

the constructed mind maps have lots of features of concept maps, we focus on broader hierarchies instead of more concrete relations, which is still closer to the category of mind maps.

Structure Building Framework (SBF) (Gernsbacher, 2013). The theory characterizes human discourse comprehension as the construction of coherent mental representations through three core processes: laying a foundation, mapping, and shifting. The human learners first lay a foundation using initial input segments, then map subsequent coherent information onto the developing structure. When incoming information is less coherent or related, they shift to initiate a new substructure, resulting in multi-branching mental representations. This framework directly inspires the design of AutoMindMap.

Working Memory Capacity Theory (Miller, 1956). Miller's classic formulation posits that human working memory can hold approximately 7 ± 2 information chunks at a time. This bounded capacity implies that knowledge structures with excessive child nodes under one parent or overly deep hierarchies impose cognitive overload on learners. The principle is adopted by previous mind map rubrics. In our evaluation, the **Structural Load Factor** directly operationalizes this constraint, and the **Structure Quality** explicitly discourages overloaded structures.

Advance Organizer Theory (Ausubel, 1960). This theory emphasizes that meaningful learning occurs when new knowledge is integrated into existing cognitive structures via advance organizers, which are introductory materials presented at a higher level of abstraction. A key principle is **progressive differentiation**: knowledge should be presented from general to specific, building a stable scaffold before introducing details. Mind maps serve as visual advance organizers; in **Structure Quality**, **Pedagogical Efficiency**, and **Structural Load Factor**, we favor maps that present core topics prominently and have progressive hierarchy instead of a flat structure. The **Skeleton Laying** stage in AutoMindMap is also inspired by the theory.

Principle of Cognitive Balance. Drawing from information visualization (Johnson & Shneiderman, 1991) and cognitive load research (Sweller et al., 2019), this principle holds that highly unbalanced hierarchical structures reduce navigation efficiency and impose uneven cognitive burdens. Our evaluation incorporates this principle in the **Structure Quality** dimension and extra structural analysis in Appendix H. Notably, perfect balance is neither achievable nor desirable due to the intrinsic heterogeneity of knowledge, so our framework treats balance as a soft guideline allowing reasonable structural flexibility.

Cognitive Flexibility Theory (Spiro, 1988). Advanced knowledge acquisition in ill-structured domains requires learners to revisit concepts from multiple perspectives and construct alternative linkages among knowledge elements. Cross-links in mind maps serve as structural analogues to this principle, enabling learners to perceive interconnections beyond strict hierarchies. Our evaluation thus rewards *"reasonable cross-links across branches that promote concept association"* in the **Pedagogical Efficiency** criteria, while cautioning against excessive density that would instead induce confusion.

Memory-based Text Processing Theory (O'Brien et al., 1998). The theory emphasizes that discourse comprehension requires continuous coordination of local and global coherence, maintaining a situation model that integrates textual information across distal discourse units. This insight directly inspired the context-aware summarization mechanism in the **Iterative Knowledge Integration** module, which serves as a semantic bridge across disconnected slide groups, preventing the loss of cross-group associations in fragmented long-context processing.

Cognitive Load Theory (Sweller, 2011). Sweller's Cognitive Load Theory distinguishes intrinsic, extraneous, and germane load. A well-constructed mind map reduces *extraneous load* by eliminating the need for learners to infer conceptual relationships from fragmented sources, and manages *intrinsic load* through progressive differentiation. Our **Structural Load Factor** and **Pedagogical Efficiency** metrics are grounded in these principles, assessing whether generated maps impose unnecessary cognitive burdens or facilitate efficient knowledge acquisition.

L PROMPTS

L.1 PROMPTS OF AUTOMINDMAP FRAMEWORK

Deck summarization and page grouping

You are a planner for segmenting and summarizing course slide files. Return STRICT JSON only.

Input:
- "course_name"
- "file_name"
- "page_titles": [{"page":1,"title":""}]

Output schema:
{
"thinking": "",
"groups": [[start_page, end_page], ...],
"file_summary": ""
}

Hard constraints:
- Groups must be contiguous, non-overlapping, and jointly cover all pages.
- Each range is inclusive [start, end], 1-based index.

Planning guidance:
- Think first based on title clue. Segment by topic continuity and pedagogical transitions.
- Try to make each group either: 1) centered on one core concept, or 2) consistent in one discourse function (e.g., overview / detailed explanation / supplementary / examples or exercises).
- Keep proper length. Prefer medium-size unless title transitions strongly suggest otherwise. Make the grouping flexible.
- Keep "file_summary" concise but structurally useful for downstream parsing, and well-founded, including the main title and ALL significant concepts logically and coherently.

Skeleton Laying

You should output an initial plan for a course mind-map construction pipeline. The mind map construction should be based on several slide decks for one course. Return STRICT JSON only.

Input:
- "course_name": the root node
- "file_infos": [{"file_name","page_count","file_summary"}]

Goal:
- Propose a learning order over files.
- Build only level-2 (L2) skeleton nodes under mind map root, assign each node files and summary.

Output schema:
{
"teaching_order": ["file_a", "file_b"],
"thinking": "",
"level2_nodes": ["name": "", "summary": "", "files": ["file_a", "file_b"]]
}

Requirements:
- For learning order, based on filenames if clues exist. Otherwise, use your prior knowledge.
- L2 nodes aim to abstract a higher knowledge hierarchy over file partition. Please:
  - Cluster files with sufficient clues. Think enough first to illustrate reason.
  - Cluster is not necessary. One file can be one L2 node if the files are few or naturally with a clear knowledge partition. Don't forcibly cluster unrelated files.
  - Keep the granularity modest and each L2 branch balanced.
  - L2 node names must be concise and meaningful domain concept noun phrases, exclude meaning-

less modifiers, and section index.
- Summary will be used as a PRINCIPLE SKILL for downstream mind map expansion. Don't just summarize clustered file content. Please generate a BRIEF plan according to the file info:
  - What should be emphasized under this L2 branch, e.g., main concepts, hierarchical and branch structures plan.
  - Potential risks may lead to disorder/illogicality that should be avoided, and suggestions to make the mind map more concise and coherent.
  - Be insightful, traceable and actionable, not broad or shallow. Think outside the box.

Knowledge Extraction

You need to extract and summarize knowledge from a slide fragment "group_pages_md" to support mind map generation. Regard "group_pages_md" as visual slides instead of a text-intensive document. Return STRICT JSON only.

Output schema:
{
"reading_memory_update": " ",
"discourse_role": " ",
"concepts": [{"name": "", "description": ""}],
"local_relations": [{"parent":"", "child":"", "description":""}]
}

Goals:
- "Discourse_role": briefly summarize discourse function.
- "reading_memory" is the previous fragment's summarization. Anchor on the new fragment to update a context-aware memory. Keep key content of previous knowledge, the coherence, hierarchy/affiliation with the current fragment.
- Extract **key concepts** that prefer few but optimal, generalized ones for knowledge overview. EXCLUDE all fine-grained and noisy marginal concepts, e.g., prerequisites, intermediate transition, examples and exercises. Keep proper granularity.
- Concept names should be CONCISE domain noun phrases, better to extract from the original, avoid single broad word/abbr. without referents. Exclude meaningless modifiers (e.g., section index, uninformative words).
- Concept description includes hierarchical context relationship inferred from "reading_memory", description amount in this group, and the concept importance.
- "local_relations" must be direct parent - child relations only. Don't extract parallel/similar knowledge relations.

Visual marker in "group_pages_md":
- "[visual_summary:...]": Textual/markdown description for embedded image.
- "[image]": image exists, but no description is provided.

Knowledge Integration

Please integrate a batch of concepts into an existing course mind map's branch to provide a knowledge overview. ONLY mount general key concepts, not every detailed knowledge list. Return a STRICT JSON object only.

Input:
- "exposed_subtree": provided node information
- "branch_guideline": branch-level principle to follow
- "reading_memory_update": context-aware section summarization for reference
- "discourse_role": original discourse analysis
- "concepts", "local_relations": local knowledge extracted
- "retry_context" (if any): previous failure tool-call log for retry
- "tool_registry": tool specs

Goals:
Plan the tool-call strategy first for new concept integration, based on all context, discourse roles, and guidelines.
- Exclude too fine-grained, meaningless, similar concepts. Mount the most representative ones with proper granularity.
- Plan the most proper position in "exposed_subtree" to mount concepts by "add_node", make the mind map hierarchical and concise. Avoid mounting to the root node.
- Finally, use other tools to refine local structure if poor overload structures/wrong relations exist, but not necessary.

Tool rules:
- Check the parameters' validity. Return "tool_calls":[] if no action.
- Multiple tool-calls will be executed sequentially; the subsequent step is based on preorder modification.
- If 'retry_context' exists, rethink and regenerate a corrected FULL 'tool_calls' (NOT from the fail step).

Output schema:
{"thinking":"", "tool_calls": [{"tool":"", "args":{}}]}

Cross Branch Check

You are an agent to judge cross-branch merge and cross-link relations in a course concept mind map. Return STRICT JSON only.

Input:
- "new_node": brief of a newly added node
- "candidate_nodes": one-hop briefs from other branches
- "retry_context" (if any): previous failure tool-call log for retry

Output schema:
{"thinking":"", "tool_calls": [{"tool":"", "args":{}}]}

Goals:
- Merge similar concepts in most cases if the names are detailed phrases, despite different descriptions/contexts.
- If the candidate concept names are very broad (e.g., application/introduction/example), rename them into detailed ones.
- Add cross-links ONLY WHEN the hierarchical relation is very explicit and meaningful with sufficient evidence.

Tool rules:
- Check the parameters' validity. Return "tool_calls":[] if no action.
- Multiple tool-calls will be executed sequentially; the subsequent step is based on preorder modification.
- If "retry_context" exists, rethink and regenerate a corrected FULL "tool_calls" (NOT from the fail step).

Structure Repair

You are a structural repair agent for a course mind map. Return STRICT JSON only.

Input:
- "violation_items": local structure to be repaired
- "constraints"
- "tool_registry"
- "retry_context" (if any)

Output schema:
{
“thinking”:“”,
“tool_calls”: [{“tool”:“”, “args”:{}}]
}

Policy:
- Think first, prefer minimal local structure edits to fix the structural violation. Better reorganize hierarchical edges, avoid frequent node deletion and addition.
- Keep the mind map coherent and concise. Fix only the structure, don’t break the whole logic.

Tool rules:
- Check the parameters’ validity. Return “tool_calls”:[] if no action.
- Multiple tool-calls will be executed sequentially; the subsequent step is based on preorder modification.
- If “retry_context” exists, rethink and regenerate a corrected FULL “tool_calls” (NOT from the fail step). Simplify tool-call steps if the error is complex.

Branch Level Critic

You are a critic for improving the quality of a mind map’s Level 2 branch. Return STRICT JSON only.

Input:
- “l2 node”
- “branch_guideline”
- “l2_branch_count”: total l2-branch in mind map
- “branch_node_count”,“branch_edge_count”: for this branch
- “branch_relations”

Output schema:
{
“issues”: [{“problem_analysis”:“”,“feasible_suggestion”:“”}]
}

Policy:
- If no problem, return “issues: []”.
- Don’t focus on visual rendering problem. Focus on structural or logical flaws revealed by visual way.
- Mainly focus on the following problems (not limited to):
  - Branch violates the principle in “branch_guideline”.
  - Too dense fine-grained/meaningless nodes that can be pruned to keep proper scale.
  - Similar intermediate node/complex local structure that can be compressed into a concise one.
  - Poor local structure, e.g., child-node overload, long chains, dense cross-link. A deeper hierarchy is preferred over a flat one.
  - Unclear concept name and relation (sibling and parent-child).
- Each issue must be concrete and actionable, focusing on a specific part. You can output several problems, but be concise.
- Think outside the box. Focus on obvious, objective problems rather than trivial subjective preferences.

Global Level Critic

You are the global critic for improving the course mind-map quality. Please propose meaningful suggestions for mind map refinement. Return STRICT JSON only.

Input:
- mindmap image, dash lines are cross-links
- “mindmap_relations”
- “total_node_count”

- “total_edge_count”

Output schema:
{
“issues”: [{“problem_analysis”:“”,“feasible_suggestion”:“”}]
}

Critic policy:
- If no problem, return “issues: []”.
- Don’t focus on visual rendering problem. Focus on structural or logical flaws revealed by visual way.
- Use “total_node_count” and “total_edge_count” as hard references to judge global density.
- Focus on macro structure and logic first, including but not limited to:
  - Imbalanced branches size and depth.
  - Too dense nodes and cross-links, propose simplification, especially marginal leaf nodes.
  - Poor global structure.
  - Confusing logic design fails to present the course knowledge overview efficiently.
- Each issue must be concrete and actionable, focusing on a specific region. You can output several problems one time, but be concise.
- Think outside the box. Focus on obvious, objective problems rather than trivial subjective preferences.

Branch Level Editor

Please use suggestions from an expert critic to refine one branch (under a level-2 node) of a low-quality course concept mind map. Return STRICT JSON only.

Input:
- “branch_l2”
- “issues”
- “branch_relations”
- “retry_context” (if any): previous failure tool-call log for retry

Output schema:
{
“thinking”: “”, “tool_calls”: [{“tool”:“”, “args”:{}}],
}

Policy:
- Based on “issues” as overall principle to plan modification and tool-use steps strictly with focused local improvements. Prefer fewer and concise tool steps to make minimal, local edits.
- Priority is removing non-core nodes and cross-links if too dense. Use “delete_node” to iteratively prune the map.
- Generally, avoid creating intermediate concepts/structures. Better to reorganize within existing concepts.
- Every change should map to a specific critical issue. Do not do broad, unrelated edits.

Tool rules:
- You can refuse unreasonable suggestions. Return “tool_calls: []” if no action.
- Multiple tool-calls will be executed sequentially; the subsequent step is based on preorder modification.
- If “retry_context” exists, rethink and regenerate a corrected FULL “tool_calls” (NOT from the fail step). Simplify tool-call steps if the error is complex.

Global Level Editor

Please use suggestions from an expert critic to refine a low-quality mind map. Return STRICT JSON only.

Input:
- “issues”
- “mindmap_relations”
- mindmap image, dash lines are cross-links
- “retry_context” (if any): previous failure tool-call log for retry

Output schema:
{
“thinking”: “”, “tool_calls”: [{“tool”:“”, “args”:{}}],
}

Policy:
- Based on “issues” (as overall principle) to plan modification and tool-use steps strictly with focused local improvements. Prefer fewer and concise tool steps to make minimal, local edits.
- Priority is removing non-core nodes and cross-links if too dense. Use “delete_node” to iteratively prune the map.
- Generally, avoid creating intermediate concepts/structures. Better to reorganize within existing concepts.
- Pay less attention to rename and non-leaf node deletion (may influence overall structure), and avoid modifying the high-level skeleton.
- Every change should map to a specific critical issue. Do not do broad, unrelated edits.

Tool rules:
- You can refuse unreasonable suggestions. Return “tool_calls: []” if no action.
- Multiple tool-calls will be executed sequentially; the subsequent step is based on preorder modification.
- If “retry_context” exists, rethink and regenerate a corrected FULL “tool_calls” (NOT from the fail step). Simplify tool-call steps if the error is complex.

L.2 Prompts of Baselines

For Direct, Chunk-Merge, and Highlights-only, we design unified prompts that include the key mind map construction principles. The Chunk-Merge method has its own prompt for merging different branches, and the Highlights-only method has its own task-specific prompts as supplements. For BookRAG and PageIndex, we use the prompts in their source codes.

Shared Prompt for Direct, Chunk-Merge, and Highlights-only

Please construct a **Course Concept Mind map** based on the provided information.

Task Description
- **Input:** Multiple Markdown files parsed from slides for one specific course (or only parts of content in slides)
- **Output:** A well-organized educational mind-map

Mind-Map Schema
- Radiates around from a central node (course name, will be given first). Each node represents a **concept** in the course.
- Edges between nodes represent the hierarchical relation, without any attributes.

Principles and Constraints
- **Well-Organized**: A fully connected directed acyclic graph (DAG), cross-links (multiple parent concepts for one concept, instead of a pure tree) are encouraged but not dense.
- **Identifies Core Concepts**: Don’t extract all knowledge concepts, exclude too-fine-grained/marginal ones. Keep the overall concept scale concise, less confusing.
- **Concise node name**: Name of each node should be a course-related concept Noun phrase (or word), exclude (or rephrase) any long sentence/questions/section index/abstract expressions. Better extracted from the original and to retain the semantically indispensable words.
- **Coherent Hierarchy and Branch**: Keep clear, coherent hierarchy and branch structure. Build hierarchy from global to parts, abstract to concrete logically, and design branches unambiguously and

consistently.
- **Meaningful in Education**: Your mind map should be understood quickly and efficiently, and reduce students' cognitive loads. While remaining knowledge faithfulness, keep the map brief and structurally balanced. Don't add too many fan-out nodes under one, and too deep layers. Avoid too flat hierarchies and single-node long chains.

Output Format (Strictly)
1. (Optional but encouraged) Output your analysis, thinking process, and reasons for your answer, enclosed with ¡think¿...¡/think¿ tags.
2. Output your final answer, enclosed with ¡result¿...¡/result¿ tags. The output format is as follows:
[concept_1, sub-concept_2],
[concept_3, sub-concept_4],
...
3. Always use the course name as the ONLY root node. Don't output any content outside the tag pair, and don't add other tags by yourself.

Merge Prompt in Chunk-Merge

There are several partial mind-map branches of this course. Please merge them into one.
- **Merge same/similar nodes and edges**: You can identify these nodes/edges then merge them.
- **Check the structure constraints and refine/prune the mind-map**: If the whole merged mind map doesn't satisfy the structure requirements (e.g., many fanout nodes under one, too deep hierarchy, not a fully-connected DAG expanded from one center node), please fix the mind map, but ensure the knowledge faithfulness, don't lose the core knowledge structure.
- **Be coherent and logical**: Final check: let the merged mind map have a coherent logic and hierarchy, and not too dense (to reduce the user's cognitive load).

You can still edit local structure (e.g., prune the dense structure or re-organize hierarchy/branch) where unreasonable, to make the mind map more perfect, concise, and less confusing.

Extra Prompt for Highlights-only

You will be provided with the highlighted titles of every page in all slide files under this course.
You need to reorganize them into a hierarchical mind-map based on just those titles and your knowledge.
- Note that not all titles are important core concepts or can be used directly. Try to make filtering, denoise, and rephrase, to keep the core concepts, to meet the requirements.
- Try to refine the mind map into a coherent one, organize a clear and hierarchical mind map from the provided information.

L.3 Prompts of Evaluation

VLM Judge

You are a strict expert VLM judge to score the quality of a course's mind maps. Return STRICT JSON only.

Based on:
- a rendered mindmap image (rendered by an external interface, so don't evaluate rendering and typography structures). Images only help to better understand.
- textual relations in "parent-¿ child" format.

Detailed rubric:
Content accuracy
- All concept names are concise and accurate, generally proper nouns/noun phrases with 2-3 words. Poor names are avoided, including:
  - Single abbr., broad word (e.g., introduction / motivation / application / example) without referents,

or uninformative modifier words/section indexes.
  - Sentences/questions/expressions, any confusing and meaningless symbols/characters.
  - Concatenation of multiple concepts into one node.
- Reasonable hierarchy and affiliation relation between parent and child concept, without confusing relation between parent-child/siblings.
- Concepts should be representative and recapitulatory, rather than lots of unimportant, course-irrelevant, noisy entities or mutually overlapped/similar concepts.

Structure quality
- Poor structures should be avoided, including but not limited to:
  - Too dense and disordered relation without clear organization.
  - Overload child nodes under one parent without hierarchy design, or cascaded long one-child chains.
  - Too deep (with more than 7 levels) or shallow flat hierarchy (with only 2-3 levels).
  - Very unbalanced branch scale/depth.
  - Shallow leaf nodes under the root.
  - Multiple root, ring/bi-direct edge, isolated nodes.
- All local concepts and relations are organized into an overall coherent logic, with meaningful taxonomy, progressive hierarchy.
- Allow for proper structural flexibility and scale difference between branches, rather than a completely neat structure.

Pedagogical efficiency
- Proper overall scale with clear visual effect on hierarchy, rather than chaotic dense graphs, or just superficial concepts without delving.
- Generalizable core topics and framework are presented obviously, can be captured quickly from the mind map rather than a laborious search.
- Reasonable cross-links across branches that promote concept association, rather than a pure tree. The proportion is proper rather than dense.
- Core knowledge should be emphasized structurally (e.g., include more and deeper nodes) rather than all concepts being equally important.
- Overall organization is consistent with human cognitive habits, leading to less confusion.

Follow **Likert scale** to grade:
- 5: very perfect, almost no flaws
- 4: obvious but non-critical flaws won't affect the overall quality
- 3: mixed quality with notable problems, but still obey most overall principles. 3 is a pass line
- 2: severely problematic, only meet a small number of requirements
- 1: meet no requirement, extremely poor quality

For each dimension, think and analyze sufficiently to have more evidence, then grade.
Be strict, try to grade lower scores.

Output schema:
{
"Content accuracy":{"think":"", "score":1},
"Overall structure":{"think":"", "score":1},
"Practicality and efficiency":{"think":"", "score":1}
}

Concept Matching

You are a strict concept matcher for course mindmaps. For each query, pick at most one most relevant true match.
- Reject when semantic scope differs (e.g., "application" vs "AI application"). But you can use the parent context to narrow broad scope.
- Minor form variants are allowed (e.g., "machine learning" vs "machine learning algorithms"), more detailed name is also allowed.
- Abbreviation/full-name variants may match only if they denote the same canonical concept. Use parent context as a tie-breaker ONLY WHEN ambiguous, instead of being more strict.
- If the candidate one includes index info (e.g., section 1/lecture 1), very long sentences (not concise

concept), or confusing symbols, reject it. DO NOT be too lenient with academic proper nouns.

Return JSON only:
{decisions:[gt_node:“”,accept:true/false, pred:candidate_or_empty]}