

H²Table: Hierarchical Hypergraph-Enhanced Large Language Models for Complex Table Reasoning

Jia Ling¹, Yangfan Wang¹, Chen Tang², Haoming Tan¹,
Yang Yang³, Yi Guan¹, Jingchi Jiang^{1,4,*}

¹Harbin Institute of Technology, ²AI Research Center, Midea Group (Shanghai) Co., Ltd.

³Changchun University of Science and Technology

⁴State Key Laboratory of Smart Farm Technologies and Systems

{2022112476, yf.wang, 2022111514}@stu.hit.edu.cn

travistang@foxmail.com, yangyang_hit_wi@163.com

{guanyi, jiangjingchi}@hit.edu.cn

Abstract

Tables are ubiquitous across diverse domains, yet reasoning over them remains a significant challenge for modern large language models (LLMs). Current approaches typically linearize tables into sequences, inherently overlooking their intrinsic two-dimensional and hierarchical structure. To address this, we propose **H²Table** (Hierarchical Hypergraph-Enhanced Table Reasoning), a novel framework that represents complex tables as hierarchical nested hypergraphs. To process this representation, we design a tailored hypergraph encoder to facilitate message passing between hyperedges (headers) and nodes (cells), thereby perceiving the semantic entailment relationships between them within complex tables. Furthermore, we introduce a set of learnable query vectors acting as a lightweight bridge to extract representative structural embeddings from the encoder into the LLM. Experimental results demonstrate that our approach effectively handles complex table question answering tasks with hierarchical nested headers. Notably, on the HiTab dataset, H²Table achieves an average improvement of 22.88% over state-of-the-art baselines on highly complex tables with a nesting depth of four. Our code is available at: <https://github.com/lila120/h2table>.

1 Introduction

Table understanding and reasoning are critical for enabling diverse downstream applications, such as table question answering (TableQA) (Pasupat and Liang, 2015) and Text-to-SQL (Yu et al., 2018), to process complex, real-world data. Consequently, leveraging the powerful semantic synthesis and reasoning capabilities of large language models (LLMs) has become essential in this domain. Driven by the need to decode structured

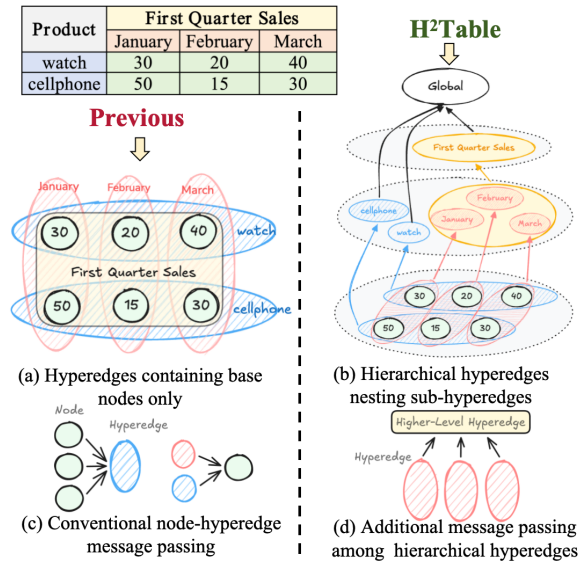


Figure 1: Comparison of modeling approaches and corresponding message passing mechanisms for complex tables between the proposed H²Table and other leading baselines.

data, existing approaches have evolved significantly. Recently, the paradigm has shifted toward LLM-based methods. These approaches primarily rely on 1D text-centric serialization to harness pre-trained knowledge, or construct massive instruction datasets for full-parameter supervised fine-tuning (SFT) (e.g., TableLlama Zhang et al., 2024, TableGPT Li et al., 2024). Furthermore, to explicitly capture internal structural information, some works represent tables as graphs. For instance, HeGTa (Jin et al., 2025) models tables as heterogeneous graphs, while HYTREL (Chen et al., 2023) utilizes hypergraphs. Building upon this structural perspective, TAMO (Li et al., 2026) integrates the hypergraph encoder from HYTREL with LLMs, achieving the current state-of-the-art performance.

However, existing methods exhibit several critical limitations. LLM paradigms relying on 1D se-

rialization and full-parameter fine-tuning demand massive amounts of high-quality data and incur prohibitive computational costs. Alternatively, representing tables as standard graphs fails to capture their intrinsic structural features. Since edges in standard graphs are strictly pairwise, they inherently contradict the one-to-many relationships between headers and cells. Existing hypergraph-based methods like HYTREL overcome this bottleneck but are restricted to simple flat tables. Furthermore, lacking LLM integration, they are incapable of handling generative tasks like TableQA. While TAMO addresses this by coupling a hypergraph encoder with an LLM, its encoding mechanism fundamentally overlooks the nested hierarchy of complex headers. As illustrated in Figures 1(a) and (c), when processing complex tables with hierarchical headers, TAMO flattens all hyperedges into a single level by directly connecting high-level headers to their associated cell nodes. This approach neglects the hierarchical relationships among hyperedges, leading to a loss of multi-level dependencies and table semantic implications. Consequently, the subsequent message passing occurs solely between these flattened hyperedges and cell nodes, ignoring the essential interactions across different header levels.

To address the aforementioned challenges, we propose H²Table (**H**ierarchical **H**ypergraph-**E**nhanced **T**able Reasoning). We first introduce a tailored transformation that maps complex tables into hierarchical nested hypergraphs. Rather than flattening these structures, our method preserves intrinsic tabular semantics, particularly the semantic entailment flowing from multi-level headers down to specific cells, by employing higher-level hyperedges that connect parent headers exclusively to their direct sub-headers. To encode these semantic dependencies, we develop a hypergraph encoder equipped with a four-stage hierarchical interaction module (V2E, C2P, P2C, and E2V). Beyond conventional vertex-hyperedge interactions, we specifically introduce the C2P and P2C stages to facilitate information exchange across different hyperedge levels, as depicted in Figures 1(d). Subsequently, we introduce a compact set of learnable query vectors to extract representative tabular structures from the encoder, optimized continuously through end-to-end fine-tuning. These vectors act as a Soft Structure Prompt (Lester et al., 2021) prepended to the serialized text, naturally infusing the LLM with ro-

bust structural awareness. Remarkably, rather than relying on costly full-parameter tuning, H²Table applies lightweight LoRA (Hu et al., 2022) to the base LLM, achieving performance comparable or even superior to full fine-tuning while incurring minimal computational overhead.

To demonstrate that our structural modeling yields superior reasoning and generalization capabilities, we conduct extensive experiments on multiple complex table benchmarks. Our main contributions are summarized as follows:

- **Hierarchical Nested Hypergraph Modeling for Complex Tables:** We propose a hierarchical nested hypergraph to preserve the intrinsic semantic entailment in multi-level tables. Furthermore, we develop a tailored, hierarchy-aware encoder that facilitates dynamic message passing to ensure comprehensive cross-level interactions.
- **Learnable Query Based Structural Alignment:** We introduce a query-based Soft Structure Prompt mechanism powered by a compact set of learnable query vectors. Through end-to-end fine-tuning, this mechanism progressively learns to extract task-beneficial structural features from tables, thereby boosting reasoning capabilities for question answering.
- **Comprehensive Superiority and Structural Robustness:** Empirical results show that H²Table consistently outperforms competing baselines across multiple datasets. Crucially, it demonstrates exceptional robustness against increasing structural complexity, achieving an average relative improvement of 22.88% on deep hierarchical tables and enabling fine-tuned medium-sized models to eclipse hundred-billion parameter models.

2 Related Work

2.1 LLM-based Table Reasoning

Currently, flattening two-dimensional tables into one-dimensional sequences is the prevailing approach for LLMs to process tabular data (Lu et al., 2025). To enhance the comprehension capabilities of LLMs on serialized tables, previous studies have typically employed instruction tuning (e.g., TableGPT, Li et al., 2024; TableLlama, Zhang

et al., 2024) or prompt engineering (e.g., Chain-of-Table, Wang et al., 2024) for optimization. However, these optimization strategies exhibit distinct limitations: instruction tuning demands substantial amounts of data and incurs prohibitive computational costs while prompt engineering is heavily constrained by the context window limits. More importantly, forcibly flattening a table inherently disrupts its intrinsic topological structure, a critical flaw that becomes especially evident when dealing with complex or nested tables.

To mitigate the structural information loss caused by serialization, several studies (e.g., TabPrompt, Jin et al., 2023; HeGTa, Jin et al., 2025) have modeled tables as graphs to capture their intrinsic two-dimensional and higher-order features. However, since each edge in a conventional graph typically denotes only pairwise relationships, it struggles to accurately characterize the inherent one-to-many higher-order correlations between a row (or column) header and its constituent cells. Consequently, HYTREL (Chen et al., 2023) represents tables as hypergraphs, where a single hyperedge can connect multiple nodes. Although effective, it is restricted to simple flat tables and lacks integration with LLMs, making it incapable of handling generative downstream tasks such as TableQA. Building upon this, TAMO (Li et al., 2026) integrates the HYTREL encoder with LLMs, treating tabular data as an independent continuous modality. While effective for flat tables, TAMO’s encoding mechanism fundamentally flattens complex nested headers by directly grouping all associated cells under a single, coarse-grained hyperedge. This design inherently overlooks the multi-level dependencies and internal topological structures among the headers themselves. To bridge the gap, H²Table advances hypergraph-based table reasoning by explicitly modeling these hierarchical relationships. By designing nested hyperedges and a tailored multi-stage message-passing mechanism, our framework systematically captures fine-grained structural semantics while maintaining exceptional parameter efficiency.

2.2 Cross-Modal Alignment between Graph Structures and Semantic Spaces

The aforementioned graph-based tabular reasoning methods inherently rely on fusing the structural topology of graphs with the semantic comprehension capabilities of LLMs. Fundamentally,

this constitutes a challenge of cross-modal feature alignment. Existing strategies generally fall into two paradigms. The first involves projection-based mapping, taking cues from large multimodal models (e.g., LLaVA, Liu et al., 2023a). For instance, GraphGPT (Tang et al., 2024) employs an MLP to project node and subgraph embeddings, extracted by a pre-trained GNN, directly into the continuous input space of the LLM. Similarly, both HeGTa (utilizing heterogeneous graphs) and TAMO (leveraging hypergraphs) adopt analogous linear projectors for feature transformation. Alternatively, the second paradigm focuses on serialization-based alignment. InstructGLM (Ye et al., 2024) heuristically flattens the graph’s topological structure into an extended text sequence, facilitating subsequent fine-tuning strictly within the textual semantic domain.

However, simple linear projectors often struggle to efficiently compress highly complex structural information, while serialization inevitably leads to excessively long contexts and topological loss. In contrast to these conventional methods, we draw inspiration from the architecture of Q-Former (Li et al., 2023). Specifically, we introduce learnable query vectors to perform cross-attention with graph embeddings, thereby actively extracting and compressing global topological and semantic features via end-to-end generative fine-tuning. This design not only alleviates the computational overhead but also significantly bolsters the model’s feature aggregation capacity when reasoning over complex nested tables.

3 Method

3.1 Modeling Complex Tables as Hierarchical Nested Hypergraphs

To effectively capture the structural and semantic dependencies within complex tables, we model the table as a Hierarchical Nested Hypergraph $\mathcal{H} = (V, E, R, I)$.

Specifically, the node set $V = \{v_1, v_2, \dots, v_n\}$ represents all data cells. The hyperedge set $E = \{e_1, e_2, \dots, e_m\}$ comprises all table headers along with a virtual global root representing the entire table. This set is strictly partitioned into disjoint leaf headers E_{leaf} and high-level headers E_{high} (which contains the root).

The hierarchical relation $R \subseteq E \times E$ defines the structural dependency among hyperedges across different header levels, where a directed

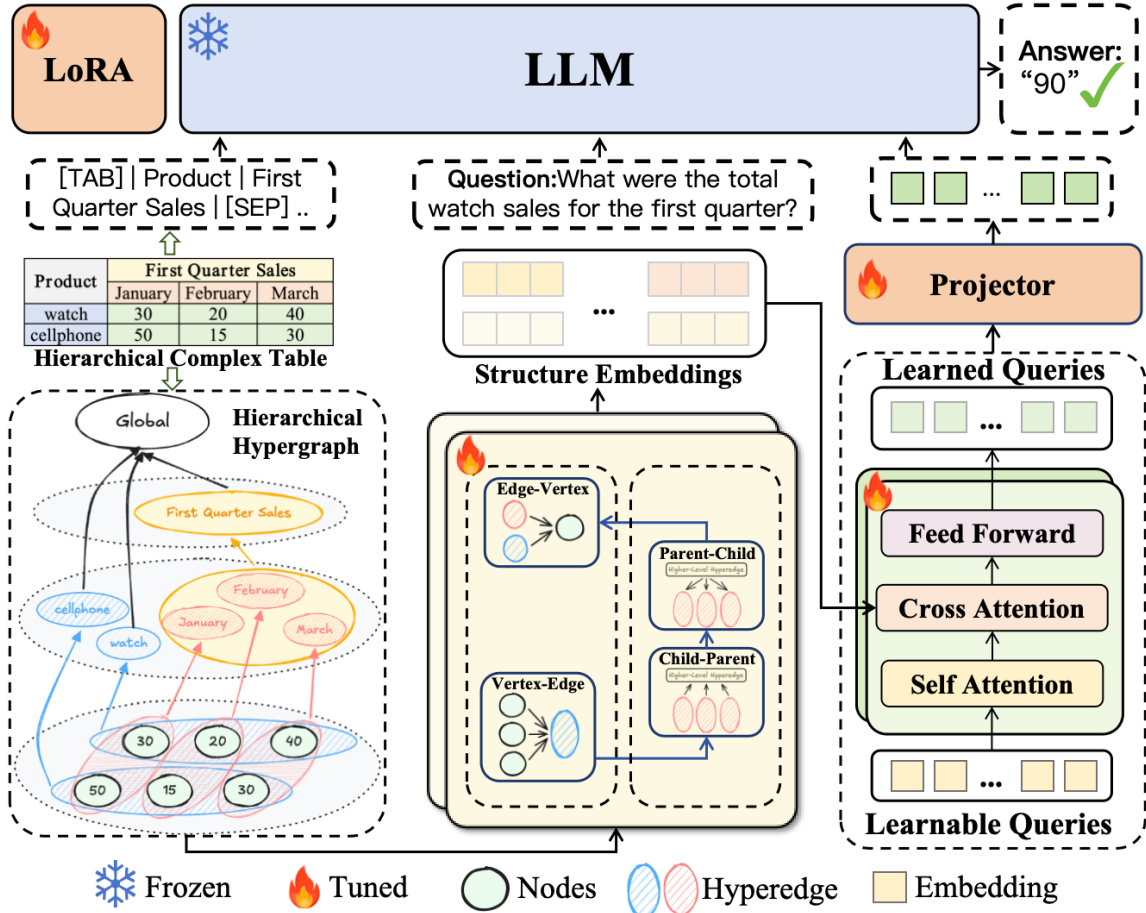


Figure 2: The overall architecture of the proposed framework. First, a complex hierarchical table is modeled as our designed hierarchical nested hypergraph, which is then processed by a hierarchical encoder featuring a four-stage message passing mechanism to derive the corresponding structural embeddings. Subsequently, a set of self-attended query vectors interacts with these structural embeddings via cross-attention. This yields a fixed number of structure-aware embeddings, which are finally combined with the serialized text and fed into the LLM to facilitate reasoning and generate the correct answer.

edge (e_i, e_j) indicates that e_i is the parent of e_j . For instance, as shown in Figure 2, in the complex table, the high-level header "First Quarter Sales" acts as the parent of the leaf header "March". The incidence relation $I \subseteq V \times E_{\text{leaf}}$ links each data cell v to its corresponding lowest-level leaf headers. For instance, as illustrated in Figure 2, the cell node "20" is connected to both the column header hyperedge "February" and the row header hyperedge "watch".

Based on these relations, the hypergraph establishes a recursively nested scope. For any leaf header $e_{\text{child}} \in E_{\text{leaf}}$, its base scope is determined by the incidence relation as $S(e_{\text{child}}) = \{v \in V \mid (v, e_{\text{child}}) \in I\}$. Conversely, the scope of any high-level header $e_{\text{parent}} \in E_{\text{high}}$ is defined as the union

of its children's scopes:

$$S(e_{\text{parent}}) = \bigcup_{(e_{\text{parent}}, e_{\text{child}}) \in R} S(e_{\text{child}})$$

Consequently, this formulation ensures that all parent hyperedges inherently perceive the scopes of their descending sub-headers all the way down to the individual cell nodes.

3.2 Hierarchy-Aware Hypergraph Encoder

Tailored to the proposed hierarchical nested hypergraph, we design a hierarchy-aware hypergraph encoder to facilitate comprehensive information propagation, encompassing both the interactions between data cells and hyperedges, and the hierarchical message passing among hyperedges across different levels. Within this encoder, we innovatively introduce a Hierarchical

Hyperedge Interaction Module, which integrates a GAT into the hierarchical message-passing mechanism of hypergraphs. This design facilitates directed, weighted message propagation dynamically among hyperedges, thereby enabling the model to autonomously learn the underlying semantic correlations between table headers. As shown in Figure 2, once a complex table is modeled as the hierarchical hypergraph detailed in Section 3.1 and initial embeddings are generated, these embeddings undergo iterative information interaction through this encoder. The encoder’s single-layer forward propagation is meticulously structured into four distinct stages:

- **V2E (Vertex-to-Edge):** Guided by the incidence relation I , this stage utilizes a set attention mechanism (Lee et al., 2019; Chen et al., 2023) to aggregate data cell features into their corresponding leaf headers, thereby establishing the base representations for the lowest-level hyperedges.
- **C2P (Child-to-Parent):** Along the directed edges of the hierarchical relation R , a GAT dynamically weights and propagates information from child headers to their parents in a bottom-up manner. This enables high-level headers and ultimately the global root to capture the underlying data distribution and structural hierarchy.
- **P2C (Parent-to-Child):** Operating symmetrically top-down along the reverse direction of R , this phase distributes global context. The GAT mechanism adaptively balances the broader semantic information passed from the parent with the preservation of the child’s own local features.
- **E2V (Edge-to-Vertex):** Finally, the hierarchically enriched hyperedge features are returned to the data cells via set attention. Residual connections are applied here to mitigate gradient vanishing, completing one full hierarchy-aware message-passing cycle.

3.3 Query-Based Parameter-Efficient Feature Alignment

While the hypergraph encoder captures rich topological representations, bridging the gap between continuous graph structural representations and the semantic reasoning space of LLMs remains a

non-trivial challenge; naively mapping extensive graph embeddings via linear projection underutilizes structural priors and overloads the LLM’s context. To address this, we introduce a parameter-efficient cross-modal alignment module to seamlessly bridge these distinct spaces.

Inspired by Q-Former (Li et al., 2023), we design a query-based mechanism for structural feature extraction and alignment. Specifically, we initialize a compact set of learnable query vectors designed to distill salient global topological representations from the dense node and hyperedge features yielded by the encoder. As illustrated in Figure 2, the alignment module is instantiated via a stack of Transformer decoder layers (Vaswani et al., 2017).

In each layer, the query vectors first undergo self-attention to model their intra-query dependencies. During the subsequent cross-attention phase, these queries attend to the comprehensive structural embeddings acting as keys and values generated by the hypergraph encoder, thereby facilitating deep cross-modal fusion (Li et al., 2023). Finally, the fused representations are projected through a feed-forward network (FFN).

Formally, let H denote the structural graph features and $Q^{(l-1)}$ denote the queries at layer $l - 1$; the core update process at the l -th layer can be abstracted as:

$$Q^{(l)} = \text{FFN}\left(\text{MCA}\left(\text{MSA}(Q^{(l-1)}), H, H\right)\right)$$

where MSA and MCA represent multi-head self-attention and cross-attention, respectively. For brevity, the standard residual connections and layer normalization applied within each sub-module are omitted from the notation.

Optimized continuously through end-to-end fine-tuning, the queries learn to extract the most representative structural features of the table. After multi-layer alignment, these highly distilled query embeddings function as soft structural prompts that are prepended to the serialized table text, constructing a cohesive dual-stream semantic-structural input that seamlessly guides the downstream LLM’s reasoning (Liu et al., 2023b).

Irrespective of table complexity, the module condenses structural information into a fixed number of tokens, alleviating the sequence explosion common in direct graph inputs and minimizing the LLM’s contextual burden. Moreover, as this module manages the cross-modal trans-

Dataset	HiTab					TATQA			
Depth	I	II	III	IV	Avg.	I	II	III	Avg.
Llama3.1-8B	0.3143	0.3609	0.2977	0.2286	0.3004	0.1866	0.1693	0.1603	0.1721
+pure text (LoRA)	0.6571	0.7190	0.6458	0.5714	0.6483	0.3528	0.3964	0.3462	0.3651
+TAMO (LoRA)	0.7143	<u>0.7960</u>	<u>0.7355</u>	<u>0.6286</u>	0.7186	<u>0.4461</u>	0.4819	<u>0.4295</u>	<u>0.4525</u>
+H ² Table (LoRA)	<u>0.6857</u>	0.8046	0.7614	0.7143	0.7415	0.4606	0.4819	0.4359	0.4595
Gemma2-9B	<u>0.6000</u>	0.6733	0.5818	0.5143	0.5924	0.3965	0.4136	0.3974	0.4025
+pure text (LoRA)	0.7143	0.7946	0.7343	<u>0.7143</u>	<u>0.7394</u>	<u>0.4781</u>	<u>0.5328</u>	0.5128	<u>0.5079</u>
+TAMO (LoRA)	0.7143	<u>0.8031</u>	<u>0.7540</u>	0.5714	0.7107	0.4723	0.5311	0.4936	0.4990
+H ² Table (LoRA)	0.7143	0.8260	0.7552	0.7429	0.7596	0.4927	0.5354	<u>0.5000</u>	0.5094
Llama2-7B	0.2000	0.1997	0.1242	0.1429	0.1667	0.1283	0.1364	0.1410	0.1352
+pure text (LoRA)	0.5429	<u>0.5649</u>	0.4379	0.4000	0.4864	0.1953	0.2409	0.1859	0.2074
+TAMO (LoRA)	<u>0.6000</u>	0.6862	<u>0.5461</u>	<u>0.4571</u>	<u>0.5724</u>	<u>0.3324</u>	0.3575	0.3718	<u>0.3539</u>
+H ² Table (LoRA)	0.6571	0.6862	0.5720	0.5714	0.6217	0.3615	<u>0.3549</u>	<u>0.3654</u>	0.3606
TableLlama	0.5714	0.6933	0.5966	0.5429	0.6011	0.1574	0.1451	0.1282	0.1436
GPT-4o	0.7143	0.7418	0.6913	0.6571	0.7011	0.5335	0.5337	0.5064	0.5245
DeepseekV3	0.6571	0.7247	0.6531	0.6286	0.6659	0.4606	0.4491	0.4551	0.4549

Table 1: Performance comparison of H²Table against baseline models on the TableQA task across different table nesting depths (I–IV for HiTab, and I–III for TATQA). Models without indentation (including backbone baselines and the bottom group) are evaluated under the zero-shot setting, while indented rows represent LoRA fine-tuned variants. “Avg.” denotes the arithmetic mean across depths. **Bold** and underlined values indicate the best and second-best results within each model group, respectively.

lation, it circumvents costly full-parameter fine-tuning. By updating merely 1% of the parameters, this paradigm achieves performance comparable to full fine-tuning. Consequently, it drastically reduces computational costs while mitigating catastrophic forgetting and preserving the LLMs inherent reasoning capabilities.

4 Experiments

4.1 Experimental setup

Datasets. We evaluate H²Table on several representative benchmarks featuring complex hierarchical structures, namely HiTab (Cheng et al., 2022) and TATQA (Zhu et al., 2021). The primary downstream task across all selected datasets is TableQA, with Accuracy serving as the core evaluation metric. The distribution of tables within these datasets, categorized by header nesting depth, is illustrated in Figure 3. Notably, because the formatted TATQA dataset includes a mere 12 tables with four-level headers, a quantity insufficient for statistical significance, these instances were excluded from our main experimental evaluation.

Baselines. We select TableLlama (Zhang et al., 2024) as a primary baseline, as it achieved state-of-the-art performance across numerous table rea-

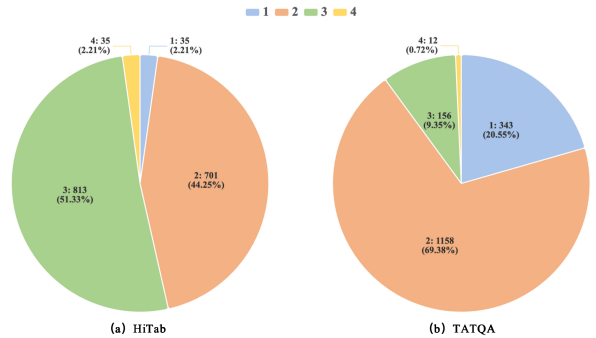


Figure 3: Distribution of table nesting depth for HiTab and TATQA

soning tasks through extensive SFT on diverse tabular datasets. Additionally, we compare our approach with TAMO (Li et al., 2026), a contemporary method that treats tables as an independent modality and has demonstrated superior performance over TableLlama on several benchmarks. To ensure a fair and direct comparison with these representative models, we adopt Llama2-7B (Touvron et al., 2023) as our primary base model. Furthermore, to evaluate the efficacy of our method on more advanced architectures, we extend our experiments to Llama3.1-8B (Grattafiori et al., 2024) and Gemma2-9B (Team et al., 2024). Our eval-

uation also includes comparisons against several training configurations for the base models, including zero-shot inference, and LoRA fine tuning using pure-text representations. Detailed experimental results are reported in Table 1.

Implementation Details For the initialization of our hypergraph, node embeddings are extracted from the final layer of a pre-trained RoBERTa-large model (Liu et al., 2019), and the hidden size of our hypergraph encoder is strictly set to 1024 to maintain dimensional consistency. For a fair comparison, both the baseline models and our proposed framework are fine-tuned end-to-end using LoRA with a rank of 8, with binary cross-entropy as the loss function. More detailed hyperparameter settings and training configurations are provided in Appendix A.

4.2 Main Results

Table 1 presents the performance comparison on the HiTab and TATQA datasets. To evaluate the capability of handling complex tables, the results are further categorized by table nesting depth. Overall, our proposed H²Table consistently outperforms all baseline representations across different base models and depth levels, demonstrating its effectiveness in TableQA tasks especially when encountering hierarchical complex tables.

When integrated with various open-source LLMs such as Llama3.1-8B, Gemma2-9B, and Llama2-7B, the H²Table representation consistently yields the highest average accuracy within each base model group. For instance, fine-tuning Gemma2-9B with H²Table achieves an average accuracy of 0.7596 on HiTab, substantially surpassing both the pure text and TAMO representations. This indicates that H²Table serves as a general and robust paradigm to improve the table reasoning capabilities of LLMs.

Aligning with our primary motivation to tackle complex tables, H²Table exhibits remarkable superiority on tables with high nesting depths (i.e., Depth-3 and Depth-4). While the performance of traditional baseline methods drops precipitously as structural complexity increases, H²Table remains highly robust. On the HiTab dataset at Depth-4, Llama3.1-8B with H²Table achieves an accuracy of 0.7143, yielding a significant absolute improvement over the pure text (0.5714) and TAMO (0.6286) baselines. Similar trends are consistently observed across other base models, vali-

dating that our method effectively preserves structural integrity and captures deep hierarchical semantics.

Remarkably, fine-tuning medium-sized models (7B–9B parameters) with H²Table enables them to rival or even surpass state-of-the-art closed-source and massively large LLMs in complex table reasoning. On the HiTab dataset, by only updating approximately 1% of the total parameters, Gemma2-9B equipped with H²Table (0.7596) significantly outperforms the hundred-billion parameter DeepseekV3 (0.6659), and the powerful GPT-4o (0.7011). These results compellingly demonstrate that providing a superior structural representation empowers smaller models to achieve highly competitive reasoning capabilities on complex tables.

4.3 Ablation Study

To thoroughly investigate the individual contributions and architectural synergies of key components in H²Table, we conduct an extensive ablation study on the HiTab dataset based on Llama3.1-8B.

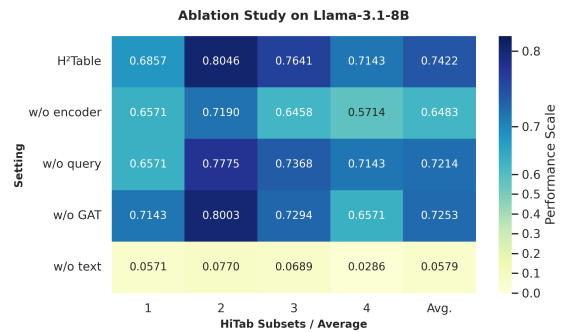


Figure 4: The heatmap of the ablation study. Deep blue hues indicate superior performance, while lighter shades represent degraded capabilities.

As visualized in Figure 4, removing the textual feature module (w/o text) causes a catastrophic performance collapse (represented by the blank row), underscoring that textual semantics remain the foundation of table understanding. Crucially, omitting the complete hierarchical hypergraph encoder (w/o encoder) also induces severe performance degradation, which validates that explicit structural modeling is indispensable for breaking the table-comprehension bottleneck of plain LLMs.

Furthermore, removing the GAT-based message passing (denoted as w/o GAT, where the hierar-

Training Set	HiTab			TATQA		
Test Set	HiTab	TATQA	AITQA	TATQA	HiTab	AITQA
Llama3.1-8B	0.3245	0.1708	0.5146	-	-	-
+Prompt tuning	0.0322	0.0298	0.1023	0.0639	0.0463	0.0498
+pure text (LoRA)	0.6774	0.2548	0.8330	0.3815	0.5696	0.7540
+TAMO (LoRA)	<u>0.7643</u>	<u>0.3156</u>	<u>0.8634</u>	<u>0.4721</u>	<u>0.6086</u>	<u>0.7974</u>
+H²Table (LoRA)	0.7713	0.3225	0.8673	0.4745	0.6313	0.8220

Table 2: Performance comparison of various models on standard and cross-domain TableQA tasks. **Bold** and underlined values indicate the best and second-best performance among the fine-tuned variants for each base model, respectively. All results are averaged over three independent training and evaluation runs.

chical edge aggregation is replaced by a standard Set Attention mechanism) or omitting the cross-attention alignment (w/o query) leads to a slight yet consistent performance decline across most subsets. Although these performance gaps may appear modest, such stable improvements are highly meaningful given the extremely challenging generative setting of complex hierarchical TableQA.

Interestingly, the w/o GAT variant yields highly competitive results and even achieves the top score on the Depth-1 subset. As Depth-1 tables are essentially flat, they do not heavily rely on complex hierarchical message passing; thus, the strong performance of this variant is expected. However, the performance gap becomes significantly more pronounced on deeper hierarchical tables (e.g., Depth-3 and Depth-4), which is exactly where the GAT-based hierarchy-aware propagation demonstrates its efficacy. Overall, the full H²Table architecture achieves the best average performance, validating the necessity and strong synergistic effects of our integrated modules.

4.4 Out-of-Distribution Generalization

To evaluate the out-of-distribution (OOD) generalization capability of our method, we conduct cross-domain experiments by incorporating AITQA (Katsis et al., 2022), another complex hierarchical table benchmark, as an unseen evaluation set.

As shown in Table 2, the vanilla Llama3.1-8B and its prompt-tuning variant exhibit poor robustness across domains, with prompt tuning suffering from catastrophic performance drops. Incorporating table-specific structural modeling significantly boosts generalization. For instance, when trained on HiTab, the pure text baseline (LoRA) achieves an OOD score of 0.8330 on AITQA, whereas the

structure-aware TAMO improves it to 0.8634.

Most importantly, our proposed H²Table consistently achieves the best performance across all settings, outperforming all baselines by a clear margin. Specifically, whether trained on HiTab or TATQA, H²Table uniformly surpasses TAMO on both in-domain and all cross-domain test sets. These results demonstrate that by effectively encoding hierarchical table structures, H²Table learns more robust, domain-invariant representations for complex TableQA tasks.

4.5 Analysis on Number of Query Vectors

As shown in Table 3, we evaluated the impact of varying the number of query vectors on overall performance.

N	Llama2-7B	Llama3.1-8B	Gemma2-9B
1	0.6172	0.7675	<u>0.7862</u>
4	0.5905	<u>0.7677</u>	0.7847
8	0.6061	0.7672	0.7841
12	<u>0.6113</u>	0.7713	0.7860
16	0.5995	0.7603	0.7914

Table 3: Comparison of different numbers of query vectors (denoted as N). Bold and underlined values indicate the best and second-best performance, respectively.

While the optimal vector count differs slightly across model architectures, using 12 query vectors consistently yields robust and highly competitive results. Specifically, it achieves the best performance on Llama3.1-8B and maintains near-optimal scores for both Llama2-7B and Gemma2-9B. To strike a balance between performance and cross-model stability, we empirically set the number of query vectors to 12 for all experiments.

5 Conclusion

In this paper, we propose a novel framework, H²Table. Unlike conventional methods that model tables as standard hypergraphs, we design a specialized hierarchical hypergraph representation tailored for complex tables, coupled with a hypergraph encoder featuring a four-stage hierarchical message-passing mechanism. Subsequently, a learnable query vector mechanism is employed to extract the most representative structural features from the encoder’s output. The resulting structural embeddings effectively assist LLMs in table reasoning tasks. The effectiveness of our framework has been validated across multiple datasets, particularly those featuring intricate hierarchical tables, demonstrating strong performance and generalization capabilities.

Limitations

Our current work assumes that the header structures of complex hierarchical tables can be accurately obtained beforehand. Under this setting, we treat header hierarchies as available structural annotations to isolate and evaluate the contribution of hierarchical hypergraph modeling itself. Future work could explore integrating upstream table-structure extractors with H²Table for end-to-end processing. Additionally, the proposed framework still relies on feeding the serialized tabular text into the LLM. Subsequent work will investigate how to enable the structural embeddings generated by the encoder to simultaneously capture rich semantic information.

Acknowledgments

We gratefully acknowledge the support of the Key Research and Development Program of Heilongjiang Province, China [2024ZX01A07] and the National Key Research and Development Program [2025YFE0209200].

References

Pei Chen, Soumajyoti Sarkar, Leonard Lausen, Balasubramaniam Srinivasan, Sheng Zha, Ruihong Huang, and George Karypis. 2023. Hytrel: Hypergraph-enhanced tabular data representation learning. *Advances in Neural Information Processing Systems*, 36:32173–32193.

Zhoujun Cheng, Haoyu Dong, Zhiruo Wang, Ran Jia, Jiaqi Guo, Yan Gao, Shi Han, Jian-Guang Lou, and

Dongmei Zhang. 2022. Hitab: A hierarchical table dataset for question answering and natural language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1094–1110.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Rihui Jin, Yu Li, Guilin Qi, Nan Hu, Yuan-Fang Li, Jiaoyan Chen, Jianan Wang, Yongrui Chen, Dehai Min, and Sheng Bi. 2025. Hegta: Leveraging heterogeneous graph-enhanced large language models for few-shot complex table understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 24294–24302.

Rihui Jin, Jianan Wang, Wei Tan, Yongrui Chen, Guilin Qi, and Wang Hao. 2023. Tabprompt: Graph-based pre-training and prompting for few-shot table understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7373–7383.

Yannis Katsis, Saneem Chemmengath, Vishwajeet Kumar, Samarth Bharadwaj, Mustafa Canim, Michael Glass, Alfio Gliozzo, Feifei Pan, Jaydeep Sen, Karthik Sankaranarayanan, and 1 others. 2022. Aitqa: Question answering dataset over complex tables in the airline industry. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*, pages 305–314.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiosek, Seungjin Choi, and Yee Whye Teh. 2019. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pages 3744–3753. PMLR.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 3045–3059.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Liyao Li, Chao Ye, Wentao Ye, Yifei Sun, Zhe Jiang, Haobo Wang, Jiaming Tian, Yiming Zhang, Ningtao Wang, Xing Fu, and 1 others. 2026. Table as

- a modality for large language models. *Advances in Neural Information Processing Systems*, 38:29285–29308.
- Peng Li, Yeye He, Dror Yashar, Weiwei Cui, Song Ge, Haidong Zhang, Danielle Rifinski Fainman, Dongmei Zhang, and Surajit Chaudhuri. 2024. Table-gpt: Table fine-tuned gpt for diverse table tasks. *Proceedings of the ACM on Management of Data*, 2(3):1–28.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.
- Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023b. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, 55(9):1–35.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Weizheng Lu, Jing Zhang, Ju Fan, Zihao Fu, Yueguo Chen, and Xiaoyong Du. 2025. Large language model for table processing: A survey. *Frontiers of Computer Science*, 19(2):192350.
- Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480.
- Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2024. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 491–500.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Zilong Ryan Wang, Hao Zhang, Chun-Liang Li, Julian M Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and 1 others. 2024. Chain-of-table: Evolving tables in the reasoning chain for table understanding. In *International Conference on Learning Representations*, volume 2024, pages 55587–55610.
- Ruosong Ye, Caiqi Zhang, Runhui Wang, Shuyuan Xu, and Yongfeng Zhang. 2024. Language is all a graph needs. In *Findings of the association for computational linguistics: EACL 2024*, pages 1955–1973.
- Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, and 1 others. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 3911–3921.
- Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. 2024. Tablellama: Towards open large generalist models for tables. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6024–6044.
- Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th annual meeting of the Association for Computational Linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, pages 3277–3287.

A Details of the Experiments

A.1 Training Settings.

For efficient fine-tuning, we applied LoRA ($r = 8, \alpha = 16$, dropout = 0.05). The model was optimized using AdamW ($\beta_1 = 0.9, \beta_2 = 0.95$, weight decay = 0.05) with a learning rate of 1×10^{-5} . To construct the training sequences, we concatenated the graph/query embeddings with the text prompt (comprising the description and the question) and the target label. Descriptions were truncated to 1,024 tokens, and the target generation length was bounded to 128 tokens. While the training sequences were dynamically batched, we relaxed the maximum context window to 4,096 tokens during the inference and evaluation phases.

A.2 Datasets.

In addition to HiTab, we applied specific pre-processing steps to the other two datasets. For

Field / Sex				2017	2018	2019	Percent Change 2017-19
All surveyed fields	Male			16580	17468	17980	8.4
	Female			11600	11816	12369	6.6
	Science and engineering	Science	Agricultural sciences	496	565	645	30
			Biological and biomedical sciences	8203	8250	8229	0.3
			Computer and information sciences	476	515	510	7.1
			Geosciences, atmospheric sciences, and ocean sciences	1794	2106	2177	21.3
			Mathematics and statistics	240	266	305	27.1
			Multidisciplinary and interdisciplinary studies	806	832	820	1.7
			Natural resources and conservation	364	580	582	59.9
			Physical sciences	2871	3056	3316	15.5
			Psychology	494	507	576	16.6
			Social sciences	1524	1601	1659	8.9
		Engineering	Bioengineering and biomedical engineering	415	440	492	18.6
			Chemical engineering	281	257	328	16.7
			Civil engineering	422	414	492	16.6
			engineering	557	588	637	14.4
			Engineering science, mechanics, and physics	200	220	186	-7
			Industrial and manufacturing engineering	119	105	137	15.1
			Mechanical engineering	458	489	531	15.9
			Metallurgical and materials engineering	181	215	242	33.7
			Other engineeringa	641	842	864	34.8
	Health	Clinical medicine		6448	6159	6273	-2.7
		Other health		1190	1277	1348	13.3

Question: rates of growth in the number of nfrs between 2017 and 2019 in various s&e fields of research varied widely, which broad field had the largest percentage of increasing?

- **(H²Table)Answer:** natural resources and conservation 
- **(TAMO)Answer:** geosciences, atmospheric sciences, and ocean sciences 

Figure 5: An example of complex hierarchical tables which has a depth of 4 and corresponding question.

AITQA, since the original dataset provides annotations for all parent headers of each leaf header, it was straightforward to convert it into the HiTab format, specifically, comprising a row header tree, a column header tree, and a data matrix. For TATQA, we employed LLM to transform the original flat matrix tables into the aforementioned hierarchical format. Consequently, this approach inevitably introduces some conversion errors, which partially explains why the model trained on the TATQA dataset exhibits less stable performance compared to the model trained on HiTab. A detailed analysis of this noise is provided below.

While converting flat TATQA tables into hierarchical structures via LLMs inherently introduces a degree of structural noise, all evaluated models (both baselines and H²Table) are trained and tested on this identical augmented dataset. Consequently, the relative performance comparisons remain strictly fair.

To quantitatively assess the impact of this conversion noise, we manually reviewed a random sample of 100 tables from the TATQA test set (comprising 600 QA pairs, averaging 6 questions

per table). The manual inspection revealed that 91 tables were converted accurately, whereas 9 exhibited noticeable structural errors. This low error rate (9%) suggests that TATQA tables possess predominantly simple, shallow hierarchies, making the LLM-based structural extraction largely reliable for this benchmark.

To further investigate how this structural noise affects downstream reasoning, we evaluated H²Table (using the Gemma-2-9B backbone) on this sampled subset. The results reveal a clear performance disparity between correctly and incorrectly converted tables:

Subset	Accuracy
Correct (546 QA pairs)	52.20%
Wrong (54 QA pairs)	37.04%
Overall (Full TATQA Test Set)	52.07%

Table 4: Performance disparity between correctly and incorrectly converted tables. The overall evaluation encompasses the full TATQA test set of 1,669 QA pairs.

As detailed above, the model achieves a 52.20%

accuracy on the correctly converted subset, which is 15.16% higher than its performance on the subset with structural errors (37.04%). This discrepancy confirms that structural conversion quality directly influences downstream TableQA performance. However, because the overall conversion error rate remains low, the accuracy on the full TATQA test set (52.07%) aligns closely with the performance on the accurately converted subset, thereby demonstrating the robustness of our evaluation on this benchmark.

A.3 Evaluation.

Regarding the evaluation metrics, rather than employing a strict Exact Match (EM), we applied necessary data cleaning and format normalization to the model’s predictions, such as stripping currency symbols (e.g., \$). We clarify that all compared methods (including closed-source models, fine-tuned baselines, and H²Table) are evaluated under exactly the same answer-cleaning and format-normalization criteria. The reported accuracy is therefore computed under a unified relaxed-match protocol after shared post-processing, ensuring fair comparability.

B Case Study

In this section, we present a case study to demonstrate our model’s capability in handling complex table structures. As illustrated in Figure 5, our model successfully yields the correct answer to the given question, whereas a strong baseline, TAMO, fails to do so. This superior performance is attributed to our specifically designed modeling and hierarchical message-passing mechanism, which enable us to process such intricate tables more effectively.

C Discussion

C.1 Compatibility and Generalization over Flat Hypergraphs

A natural theoretical question is how H²Table behaves when applied to standard flat tables lacking nested structures. In such degenerate cases, where all headers reside at a single depth, the hierarchical relations naturally collapse, and message passing reduces strictly to the leaf-level vertex-hyperedge interactions (i.e., V2E and E2V).

We regard this property as an intrinsic strength of our design: H²Table remains mathematically

well-defined and seamlessly applicable to flat tables, without imposing any artificial or redundant hierarchical priors. Crucially, even under this flat degeneration, H²Table is fundamentally distinct from prior hypergraph-based approaches such as TAMO in how structural knowledge is transferred to the LLM. Rather than relying on simple average-pooling over encoder embeddings, our query-based cross-attention module employs learnable queries to selectively attend over encoder representations, effectively distilling high-density structural signals into a compact soft prompt. Furthermore, once multi-level nested headers are present, the hierarchical message-passing stages (C2P and P2C) are dynamically activated, empowering the model to capture cross-level semantic entailments that flat hypergraph formulations fundamentally overlook. Consequently, H²Table generalizes rather than merely replaces flat hypergraph modeling—it preserves full backward compatibility with flat tables, provides a more selective encoder-LLM alignment mechanism, and explicitly incorporates hierarchical inductive bias precisely when nested headers matter.

C.2 Statistical Significance Analysis on Deep Hierarchy Tables

To ensure a robust statistical evaluation and address the limited sample size of the isolated Depth-4 subset ($N = 35$), we aggregate the Depth-3 ($N = 813$) and Depth-4 tables into a unified "Deep Hierarchy" subset, yielding a more substantial total sample size of $N = 848$. Evaluated using the Llama-3.1-8B backbone, our proposed H²Table architecture demonstrates a stable improvement over the strongest baseline, TAMO, on this combined challenging subset.

To formally validate the reliability of these performance gains, we conduct a paired McNemar’s test comparing the exact match predictions of H²Table against TAMO. The test confirms that the improvements introduced by our model are statistically significant, yielding an exact McNemar p -value of 0.0079 and a continuity-corrected χ^2 p -value of 0.0083 ($p < 0.01$). These statistical results substantiate that the advantages of H²Table on highly complex hierarchical tables are robust and not an artifact of small sample variance.