RICoT: Recurrent Interleaved Chain-of-Thought for Page Stream Segmentation

Anonymous ACL submission

Abstract

Page Stream Segmentation (PSS)—identifying document boundaries within multi-page files—has been approached with LLMs primarily through pairwise classification, which restricts context to adjacent pages and redundantly encodes each page twice. We introduce **RICoT** (Recurrent Interleaved Chain-of-Thought), a prompting framework that combines (a) interleaved chain-of-thought reasoning, alternating per-page analysis with structured boundary decisions inside a sliding window, and (b) recurrent state propagation across windows, carrying reasoning history and an active-segment summary to preserve long-range context for arbitrarily long streams. Across commercial and open-source LLMs, RICoT reaches DocF1 of 0.847 on TABME, outperforming all prior zero-shot baselines, and PageF1 of 0.987 on Tobacco800, matching prior SOTA. We further fine-tune a 2B-parameter model under the same framework via *schema-mixed distillation*: teacher outputs are augmented with two probabilistic transformations—pagewise reasoning dropout and decision-field dropout—yielding a student robust across schema configurations. Trained only on a general PDF corpus with no TABME-specific data, the 2B student reaches DocF1 0.770 on the full TABME test set, remaining competitive with prompted models $\sim 15\times$ larger. Ablations isolate the contributions of interleaved page-level reasoning, recurrent state propagation, and the structured decision schema, with interleaved page-level reasoning emerging as the dominant factor.

1 Introduction

Document-level information retrieval typically assumes predefined document boundaries. However, real-world "page streams"—sequences of scanned or digital pages—often bundle multiple documents into a single file. Page Stream Segmentation (PSS) partitions these streams into their constituent documents (Figure 1), a task critical for finance automation (AP/AR), where batched invoices, purchase orders, and receipts are processed together; failing to segment them leads to merged field values, duplicated line items, and failed reconciliation.

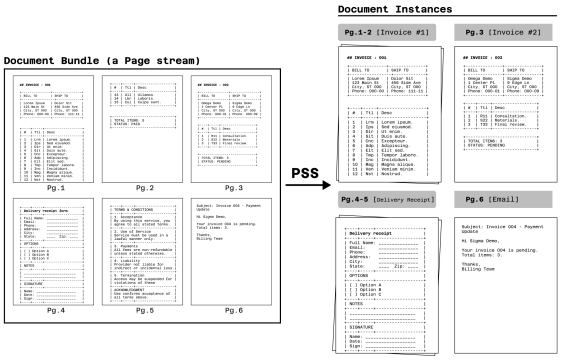

Figure 1: Page Stream Segmentation task. A stream of six pages is segmented into four documents: two invoices (pages 1–2 and 3–4), a delivery receipt (page 5), and an email (page 6). Each document may span one or more consecutive pages.

Traditional PSS relies on **pairwise classification** (Wiedemann and Heyer, 2018; Mungmeeprued et al., 2022; Heidenreich et al., 2024), deciding if page P_i starts a new document based on P_{i-1}. This paradigm suffers from two primary limitations:

- **Contextual Isolation:** Decisions often require long-range dependencies. For example, a remittance advice on page 11 may only be identifiable as part of a page 1 invoice via headers that a pairwise classifier cannot see.
- **Computational Redundancy:** Encoding each page twice—once as context and once as a target—doubles token costs without improving long-range reasoning.

Conversely, **full-stream processing** attempts to process entire files at once but falters as sequence length increases. Large Language Models (LLMs)

suffer from reasoning degradation in long contexts (Liu et al., 2024), making this approach impractical for production. While **windowed classification** offers a middle ground, it remains limited by fixed window boundaries.

Our contributions are:

- **The RICoT framework** (Section 3): a sliding-window PSS framework that combines interleaved per-page chain-of-thought reasoning within each window with recurrent state propagation across windows—a short reasoning history and an active-segment summary that maintain document-level coherence for arbitrarily long streams.
- **A distilled 2B student via schema-mixed distillation** (Section 4): the same framework is used to fine-tune a small Qwen-3.5-2B model on outputs from a strong teacher (Kimi-K2-Instruct) over a general PDF corpus. *Schema-mixed distillation* applies two probabilistic transformations to every training sample—pagewise reasoning dropout and decision-field dropout—producing a student robust across deployment-time schema configurations. Trained without exposure to any benchmark training data, the 2B student is competitive with prompted models $\sim 15\times$ larger.
- **Extensive component ablations** (Sections 6.4 and 6.5) trace how each load-bearing element of RICoT—pagewise interleaved reasoning, recurrent state propagation, the structured decision schema, and the deployment-time *expand* post-processing—contributes to the framework's accuracy.

2 Related Work

Page stream segmentation. PSS has progressed from visual CNNs classifying adjacent page pairs (Wiedemann and Heyer, 2018) through joint text-layout encoders (Mungmeeprued et al., 2022) to layout-feature embedding schemes (Drumond et al., 2023) and efficient image-only pipelines (Braz et al., 2021) on the Tobacco800 benchmark (Lewis et al., 2006; University of California, San Francisco, 2007; Agam et al., 2006). Heidenreich et al. (2024) adapt the paradigm to LLMs by prompting instruction-tuned models on adjacent page pairs and obtain the strongest zero-shot decoder baseline on TABME prior to our work; Heidenreich et al. (2025) extend the LLM treatment with new metrics and stronger fine-tuned baselines. PSS has also been formalized in adjacent media, e.g. multimodal Transformers on comic-book page streams (Serra Ortega et al., 2025). A common thread across this prior work is the *pairwise* assumption: boundary decisions are made independently for each page given only the immediately preceding page, which limits context to the local pair. RICoT removes this assumption by processing a windowed span at once and propagating state across windows.

Interleaved chain-of-thought reasoning. Chain-of-thought prompting (Wei et al., 2022) elicits intermediate reasoning steps before the final answer. The closest precedent for the mechanism we use is ReAct (Yao et al., 2023), which interleaves *thought* and *action* steps for agentic tasks; RICoT adapts the same interleaving principle to sequential *classification*, alternating per-page reasoning and per-page decision within a window. Interleaved reasoning has recently been explored as a training paradigm for LLMs (Xie et al., 2026); our contribution is at inference time, integrated with windowed processing and recurrent state for multi-page sequential classification.

Long-context state propagation. Long-context use is limited by the lost-in-the-middle effect (Liu et al., 2024) and by the cost of extending context windows. Architectural mitigations (Beltagy et al., 2020; Zaheer et al., 2020) and recursive decomposition (Zhang et al., 2025) change the underlying inference procedure; explicit memory-management approaches such as MemGPT (Packer et al., 2023) pass state across turns through structured summaries. RICoT shares the explicit-state philosophy, instantiated for PSS as a recurrent active-segment summary plus a short reasoning history carried across overlapping windows.

3 RICoT Framework

3.1 Task definition

Let $\mathbf{X} = (x_1, \ldots, x_N)$ be a page stream of N pages, each represented by OCR-extracted text and layout features. PSS maps $\mathbf{X}$ to binary boundary labels $\mathbf{y} = (y_1, \ldots, y_N)$ with $y_i = 1$ iff page x_i starts a new document; by convention $y_1 = 1$. The labels induce a partitioning into K contiguous documents $\{D_1, \ldots, D_K\}$ where $D_k = (x_{s_k}, \ldots, x_{e_k})$ and $\{s_1 < \cdots < s_K\} =$

$\{i \mid y_i = 1\}$.

3.2 Sliding windows and the two-axis decision

RICoT (Figure 2) processes $\mathbf{X}$ with *token-budgeted* windows. Each window W_c is packed greedily with consecutive pages until the running token count would exceed a budget B or a page cap $w_{\max}$ is reached, with a one-page overlap o into the previous window.[1] This adapts window size to page density—short pages pack densely up to $w_{\max}$, long pages truncate early—so context utilization is consistent across stream geometries.

Two-axis decision. For each page in the window the model emits

$$d_i \;=\; (y_i,\, z_i,\, \ell_i,\, g_i,\, c_i), \qquad (1)$$

where the *segment axis* $y_i \in \{\mathrm{S}, \mathrm{C}\}$ marks a new segment (S) or continuation (C); on continuations, the *sub-document axis* $z_i \in \{\mathrm{D}, \varnothing\}$ optionally marks a nested document instance (e.g. an invoice inside a statement of account). ℓ_i is a document-class label, g_i a grounding fingerprint (page identifier + short evidence quote), and c_i a categorical confidence (HIGH$\mapsto$9, MOD$\mapsto$6, LOW$\mapsto$3). Keeping segment-start and sub-document-start as separate axes at decision time preserves both signals for downstream control (Section 3.4).

3.3 Interleaved reasoning and recurrent state

Within a window, RICoT generates per-page reasoning (`<think>`) interleaved with the per-page decision (`<decision>`) before moving to the next page. We refer to this as *pagewise thinking* (PwT); the alternative of emitting one block of reasoning followed by all decisions is the *windowed thinking* variant we ablate in Section 6.4. Interleaving grounds each decision in evidence local to its target page.

Two recurrent state objects, $(\mathcal{H}^{(c)}, \mathcal{A}^{(c)})$, are propagated across windows. $\mathcal{H}^{(c)}$ retains the last k thought–decision pairs (short-term rationale). $\mathcal{A}^{(c)}$ describes the currently open segment in two parts: (i) deterministic metadata $(s, \ell_s, \text{PAGES-COVERED})$ derived by walking back to the most recent S decision, and (ii) a model-emitted `<active_segment_carry_forward>` block $\kappa^{(c)}$ summarizing anticipated content and progress for the next window. The stream is wrapped with `<stream>`/`</stream>` tags so the model suppresses κ on the terminal window where no successor exists.

Robust decoding. A decision cache $\mathcal{C}$ stores the most recent valid d_i per page. After parsing, any missing, malformed, or duplicate page is filled from $\mathcal{C}$ first; if the page will recur in the next overlapping window the correction is deferred; remaining gaps are repaired by a correction call, and finally by rule-based fallbacks (continuation by default, low-confidence UNKNOWN otherwise). The full procedure is summarized in Algorithm 1 (Appendix O.).

3.4 Expand: deployment-time controllability

Because the two-axis decision keeps sub-document starts as a separate signal even when their parent segment is open, downstream consumers can re-threshold what counts as a segment without re-running inference. We use this for one specific operating choice on FinancePSS-205, which we call *expand*: when a predicted sub-document of a priority class (`invoice`, `statement`, `purchase_order`, `packing_slip`, `bill_of_lading`, `commercial_invoice`, `packing_list`) appears inside a parent segment, the sub-document's start is promoted to a true segment start in post-processing. Expand is a deployment-time control that exposes the two-axis decision to the user; we report its quantitative effect alongside the no-expand variant throughout Section 6 so its contribution is visible rather than absorbed into the headline number.

4 Schema-Mixed Distillation

RICoT can be run zero-shot with a strong prompted model, but a small student model that operates under the same framework is more attractive for deployment. We describe *schema-mixed distillation*, a procedure that combines standard teacher–student distillation with two probabilistic schema transformations applied per training sample. The resulting Qwen-3.5-2B student is competitive with prompted LLMs many times its size (Section 6.1). Training corpus, transformations, and training setup are described below.

4.1 Training corpus from FinePDF

We construct training data from FinePDF (Kydlíček et al., 2025), a publicly available PDF-sourced corpus ($\sim$3T tokens, $\sim$475M documents). We restrict to the English subset ($\sim$206M documents) and

[1] We use $B = 3600$ tokens, $w_{\max} = 6$, $o = 1$ across all experiments; values were fixed *a priori*, not tuned.

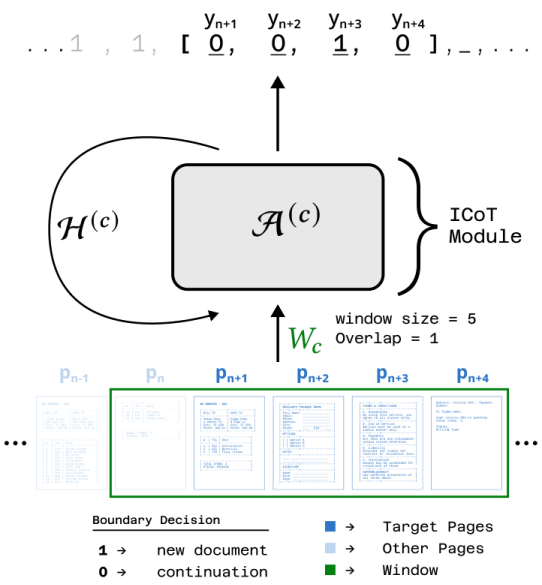

Figure 2: RICoT architecture. Window W_c processes its pages with interleaved per-page reasoning and decisions, receiving reasoning history $\mathcal{H}^{(c)}$ and active-segment summary $\mathcal{A}^{(c)}$ from preceding windows. Recurrent state and the per-page two-axis decision (segment + sub-document) extend effective context beyond the window boundary while keeping per-window inference cost bounded.

isolate finance content through a two-stage filter: Stage 1 is a high-recall keyword pass over a tiered finance lexicon, and Stage 2 is an XGBoost classifier over TF-IDF features trained on ∼100K Stage-1 documents labeled by an LLM judge for finance vs. non-finance. The combined filter yields 12.3M finance documents (∼100B tokens). Lexicon tiers, classifier details, and classification threshold tuning are in Appendix M..

From this pool we sampled 300K documents as the candidate training set. FinePDF provides only document-level OCR without page markers, but RICoT operates at page granularity, so we re-OCR each sampled PDF at the page level after fetching the source via its FinePDF Common Crawl URL. We use FinePDF's per-page language list twice—to validate that the downloaded PDF aligns with FinePDF's metadata (matching page counts), and to discard pages in scripts unsupported by our OCR model. Approximately 55% of the sampled documents survived, yielding 150,448 documents; the per-stage attrition is in Appendix J..

We then ran RICoT with the teacher model (Kimi-K2-Instruct (Kimi Team, 2025)) over the 150,448 documents, producing 467,229 chunk-level samples (each chunk is one sliding-window span with the teacher's interleaved reasoning, per-page decisions, and `active_segment_carry_forward` summary). Only 20,959 (13.9%) source documents contain two or more segments; the remaining ∼80% are self-contained single-segment documents, reflecting FinePDF's web-crawled provenance.

Chunk selection. We retain every multi-segment file and every *special* chunk within it (chunks containing a segment-start, ending the file, or immediately preceding a segment-start), uniformly subsampling intervening non-special chunks up to a cap. Single-segment files are downsampled within semantic clusters of the teacher's free-form document-class labels (class strings are embedded and clustered to merge near-duplicates such as "academic paper" / "academic journal article"). A final quality gate drops truncated and single-page documents. The final corpus is 46,791 documents: 20,959 multi-segment and 25,832 single-segment. Per-track sampling caps are in Appendix I., with per-stage attrition counts in Appendix J..

4.2 Two probabilistic schema transformations

Each training sample is a triple (SYS, CTX, OUT)—system instruction, stream context, chunk output—always rewritten consistently so the schema declared, carried, and produced agree. We then apply two independent dropout transformations.

(1) Pagewise reasoning dropout (the distillation move). With probability $p_{\text{drop-think}}$ we strip the per-page `<think>` blocks for the whole chunk and remove any mention of reasoning from the instruction. The student is thus trained to emit the teacher's decisions directly rather than its rationale, making both modes available at deployment.

(2) Decision-field dropout (format robustness). We drop sub-fields of the per-page decision (document class, confidence, grounding) and of the carry-forward summary at their own probabilities, removing the corresponding instruction lines in the same step, so the student accepts reduced or extended decision schemas without retraining.

Per-field probabilities and one worked example per variant are in Appendix I. and Appendix L.; values were chosen heuristically and not tuned, owing to compute constraints.

4.3 Training setup

We fine-tune Qwen-3.5-2B (Qwen Team et al., 2025) with QLoRA (Dettmers et al., 2023) of

rank 32 (α=64, dropout 0, target modules: all linear projections) for 1 epoch on the 46,791-document training corpus, with $\sim$1,800 samples held out for validation. We use cosine learning-rate schedule (8×10^{-5}, 8% warmup), AdamW-8bit, bf16 with gradient checkpointing, effective batch size 16 (per-device 8, grad-accumulation 2). Training took $\sim$44 hours on 4×A100 80GB. The full hyperparameter set is in Appendix K.. We report a single training run; we did not perform hyperparameter sweeps over the schema-mixed transformations or LoRA rank.

5 Experimental Setup

5.1 Datasets

We evaluate on three benchmarks spanning different domains and complexity levels (Table 1).

TABME (Mungmeeprued et al., 2022). A synthetic page-stream benchmark built from the Truth Tobacco Industry Documents archive, in which $\sim$44,769 source PDFs were concatenated into synthetic folders mimicking real folder length and type distributions. Each constituent PDF is treated as a single ground-truth segment regardless of its internal structure—a convention whose implications we examine in Section 6.6. Inference cost on the full 5,000-stream test set is substantial, so we run prompting experiments on a fixed random subset of 250 streams (**TABME-250**; stream IDs in Appendix A.); our distilled model is additionally evaluated on the full 5,000-stream test set, where numbers track the subset closely (DocF1 0.756 vs. 0.770 on $\star$, PwT; Section 6.1).

Tobacco800 (Lewis et al., 2006; University of California, San Francisco, 2007; Agam et al., 2006). A standard PSS benchmark from the same archive, containing scanned letters, memos, and reports. Following Wiedemann and Heyer (2020) and Mungmeeprued et al. (2022), we evaluate on the last 259 pages from 153 documents, ordered alphabetically by filename and treated as a single continuous stream.

FinancePSS-205. A curated dataset of 205 real-world finance document streams collected from production AP/AR workflows with appropriate anonymization. Streams contain invoices, remittance advice, purchase orders, delivery receipts, and email correspondence, with characteristic bundling patterns (invoice–remittance pairs, PO–invoice matches) typical of finance automation.

	TABME-250	Tob800	FinPSS
Streams	250	1	205
Segments (docs)	2,627 (1,542)	153	685
Pages	7,271 (4,240)	259	1,772
Pages/stream	29.1 / 27 / 3 / 81	259†	8.6 / 5 / 2 / 82
Segs/stream	10.5 / 10 / 3 / 19	153†	3.3 / 2 / 1 / 44
Segment length	2.8 / 1 / 1 / 20	1.7 / 1 / 1 / 17	2.6 / 2 / 1 / 66

Table 1: Test-set statistics. TABME segment/page counts shown as *total (unique)* since synthetic concatenation reuses documents across streams (Section 6.6). Stream statistics reported as avg/med/min/max; † Tobacco800 is evaluated as a single continuous stream, so per-stream distributional statistics do not apply.

We use only the test partitions of the public datasets; no training data from any benchmark is used in this work.

5.2 Models

We evaluate RICoT in two settings: zero-shot prompting with four commercial and open-weight LLMs, and a 2B-parameter distilled student trained as described in Section 4.

Prompted LLMs. We test four prompted models spanning multiple scales and providers. **GLM-4.7-flash** (GLM Team et al., 2025) is the $\sim$30B small variant of the GLM-4.7 family, our small-model prompting baseline—roughly 15× the size of our distilled student; **GLM-4.7** (GLM Team et al., 2025) is the full GLM-4.7 model; **GLM-5** (Z.ai, 2025) is a 744B-parameter Mixture-of-Experts model released under the MIT license by Z.ai; **Claude Haiku 4.5** (Anthropic, 2025) is included as a non-GLM frontier prompted-model point of comparison. By default we run RICoT with each model's internal extended thinking *off* (Section 6.4).

Distilled student. **Qwen-3.5-2B** (Qwen Team et al., 2025) fine-tuned via schema-mixed distillation (Section 4) with Kimi-K2-Instruct (Kimi Team, 2025) as the teacher. The same base model prompted with RICoT but *without* fine-tuning is also reported, isolating the contribution of the distillation procedure from base-model capacity and framework prompts (Section 6.1).

Short labels used in tables. To keep result tables compact we use the following acronyms throughout Section 6 and the appendix: **G4.7-F** for GLM-4.7-flash, **G4.7** for GLM-4.7, **G5** for GLM-5, **C-H4.5** for Claude Haiku 4.5, and **Q3.5-2B** for the Qwen-3.5-2B base model. The distilled-student variants

are denoted **Q3.5-2B*** (no pagewise thinking) and **Q3.5-2B**$^{\star,\mathrm{PwT}}$ (with pagewise thinking).

5.3 Evaluation metrics

We report four metrics spanning page, segment, and stream granularities. **PageF1** (Heidenreich et al., 2025, 2024) is page-level boundary-classification F1. **DocF1** represents each document as the tuple of its constituent page indices: a true positive is a tuple appearing in both prediction and ground truth, exact-match. MNDD_{μ} (per-stream mean of Minimum Number of Drag-and-Drops) is a user-centric metric counting the minimum drag-and-drop operations needed to reconcile a prediction with the ground truth in a folder-management UI; lower is better. **STP%** (Straight-Through Pass) is the percentage of streams with zero boundary errors.

We treat MNDD_{μ} as a first-class metric because it captures repair cost rather than aggregate error counts: a prediction with three small misalignments and a prediction with one large catastrophic split can have similar PageF1 but very different downstream impact. DocF1 says "how often correct end-to-end"; MNDD_{μ} says "how expensive when wrong."

5.4 Baselines

We compare RICoT against the two standard LLM-PSS paradigms, both re-run on our infrastructure with the same prompted backbones used for RICoT.

Pairwise. Each adjacent page pair (p_{i-1}, p_i) is classified independently for whether p_i starts a new document (Heidenreich et al., 2024, 2025); window size w=2, overlap o=1. Internal extended thinking enabled (its strongest configuration per Section 6.4).

Windowed. A fixed-size window of w=5 pages is shown to the LLM with a single decision prompt asking for boundary labels across the window; windows overlap by o=1. Internal extended thinking enabled.

5.5 Implementation details

The RICoT sliding window is sized by a constant token budget (B=3600), a per-window page cap ($w_{\max}$=6), and a one-page overlap (o=1); these were fixed *a priori* and held constant across all experiments. Pages whose tokenized length exceeds a fixed per-page threshold are truncated using a head-and-tail rule (first 500 + last 300 tokens, with an explicit truncation marker between), since segmentation signal concentrates at document openings and closings. All inference uses temperature 0 and greedy decoding for both prompting and the distilled model. Full inference configuration and prompt templates are in Appendix N..

6 Results

We report results across the three benchmarks of Section 5.1 along five axes: (i) headline cross-dataset performance, (ii) comparison against prior published TABME baselines, (iii) head-to-head against the standard pairwise and windowed LLM-PSS paradigms, (iv) the pagewise-vs-internal-thinking ablation that justifies our default operating point, and (v) component ablations across the rest of the framework. Model abbreviations, baseline configurations, and the FinancePSS with/without-expand convention (0.832 (0.795)) follow Sections 5.2 and 5.4; superscripts iT, PwT, $\star$ denote internal-thinking, pagewise-thinking, and fine-tuned runs respectively, and em-dashes mark cells we did not run. Distilled-model FinancePSS and Tobacco800 numbers are means of 5 inference runs.

6.1 Headline cross-dataset results

Table 2 reports RICoT across all four prompted models and two distilled-model variants on the three benchmarks. The strongest prompted model (GLM-5) reaches DocF1 0.967 on Tobacco800 and 0.847 on TABME-250, and the strongest prompted-model FinancePSS configuration (GLM-4.7) reaches 0.832 with expand. The distilled 2B model (Qwen-3.5-2B$^{\star,\mathrm{PwT}}$) reaches 0.785 on FinancePSS-205 and 0.770 on the full 5,000-stream TABME test set without exposure to TABME training data—a per-dataset gap of roughly 0.04–0.08 DocF1 to the strongest frontier prompted model, at a small fraction of the inference cost. The same base Qwen-3.5-2B prompted with RICoT but *without* fine-tuning reaches only 0.445 / 0.396 / 0.281 on the three datasets, isolating the contribution of schema-mixed distillation from base-model capacity and framework prompts.

6.2 Comparison to prior published TABME baselines

Table 3 situates RICoT against the published TABME baselines collected in prior LLM-PSS work (Heidenreich et al., 2025). RICoT prompting with GLM-5 (DocF1 0.847) beats the strongest published zero-shot decoder (GPT-4o-ZS) by +0.144 DocF1 and +33 STP percentage points,

Model	FinancePSS-205			Tobacco800			TABME-250		
	DocF1	PageF1	MNDD_μ	DocF1	PageF1	MNDD_μ	DocF1	PageF1	MNDD_μ
G4.7-F	0.762 (0.741)	0.898 (0.884)	1.005 (1.225)	0.867	0.949	48.0	0.683	0.829	5.65
G4.7	**0.832** (0.795)	**0.916** (0.884)	**0.848** (1.255)	0.961	**0.987**	7.0	0.794	0.892	3.31
C-H4.5	0.776 (0.739)	0.880 (0.831)	1.172 (1.647)	0.957	0.984	8.0	0.816	0.910	2.88
G5	0.826 (0.769)	0.906 (0.864)	0.975 (1.422)	**0.967**	**0.987**	**3.0**	**0.847**	**0.922**	**2.40**
Q3.5-2B (base)	0.445	0.688	3.103	0.396	0.664	167.0	0.281	0.563	14.62
Q3.5-2B*	0.773 (0.763)	0.904 (0.892)	0.897 (1.079)	0.914	0.954	26.6	0.753 / 0.774^{F}	0.873 / 0.894^{F}	4.56 / 4.21^{F}
Q3.5-2B$^{\star,\mathrm{PwT}}$	0.785 (0.779)	0.901 (0.897)	1.044 (1.073)	0.927	0.965	18.6	0.756 / 0.770^{F}	0.885 / 0.896^{F}	4.20 / 4.30^{F}

Table 2: Headline cross-dataset results for RICoT. DocF1 / PageF1 higher is better; MNDD_μ lower. FinancePSS numbers show with-expand outside parentheses and without-expand inside; TABME-250 / TABME-full reported separately for the distilled model. Bold marks the column best. * fine-tuned rank-32 QLoRA; ^{PwT} pagewise thinking enabled. ^{F} Full 5,000-stream TABME test set (vs. 250-stream prompting subset).

Model	Type	DocF1	STP%	MNDD
XGBoost	FT	0.642	7.4	10.85
DiT	FT	0.623	6.6	10.48
Phi-3-mini	ZS	0.060	0.0	14.79
Mistral-7B	ZS	0.368	1.4	21.15
GPT-4o^{a}	ZS	0.703	9.4	7.79
Phi-3-mini	FT	0.933	63.7	1.76
Mistral-7B	FT	**0.967**	**80.0**	**0.81**
G5^{250} (RICoT)	ZS	<u>0.847</u>	42.0	2.40
Q3.5-2B$^{\star,\mathrm{PwT}\,\mathrm{F}}$ (RICoT)	ZS	0.770	22.0	4.30

Table 3: TABME comparison against published baselines (Heidenreich et al., 2025). ZS / FT: zero-shot vs. fine-tuned on TABME training split. Bold marks the overall best; underline marks the best zero-shot result. ^{a} GPT-4o-ZS evaluated on a 10% subset; ^{250} our prompting runs use TABME-250 (Appendix A.); ^{F} our distilled model uses the full 5,000-stream test set. Rule groups separate classical/encoder, zero-shot LLM, fine-tuned LLM, and RICoT runs.

Method	G4.7-F	G4.7	G5	C-H4.5
Tobacco800				
Pairwise^{iT}	0.886	0.888	0.872	0.908
Windowed^{iT}	0.823	0.867	0.958	0.876
RICoT	0.867	**0.961**	**0.967**	**0.957**
FinancePSS-205				
Pairwise^{iT}	0.506	0.763	0.739	0.749
Windowed^{iT}	0.608	0.757	0.769	0.755
RICoT	**0.762**	**0.832**	**0.826**	**0.776**
RICoT (no exp.)	(0.741)	(0.795)	(0.769)	(0.739)

Table 4: DocF1 of RICoT vs. standard LLM-PSS baselines. Baselines run with internal thinking enabled (their best general configuration); RICoT with internal thinking off (Section 6.4). Bold marks the column best.

while the distilled 2B model evaluated on the full TABME test set (0.770) beats every traditional and encoder baseline and beats GPT-4o-ZS by +0.067 DocF1. Neither approaches the in-domain fine-tuned baselines, which were trained on TABME's training split; the gap is analyzed in Section 6.6 as a segment-definition mismatch, not a capability gap.

6.3 RICoT vs. pairwise and windowed paradigms

Table 4 reports DocF1 head-to-head against pairwise and windowed LLM-PSS baselines on the two datasets where the distinction matters (Tobacco800 has only one stream, but the single-stream DocF1 still differentiates configurations; TABME-250 comparisons are in Appendix H.). Across both datasets, RICoT is the best configuration for every model on which we ran it. On FinancePSS the margins are large for every model (+0.054 to +0.154 over the best baseline per column); on Tobacco800 the margins narrow into the saturation band ($\leq$ 0.10 DocF1) but RICoT still wins or ties on three of four models. For the larger models on FinancePSS, the headline gain depends materially on the expand post-processing step (Section 3.4); we therefore report both with-expand (headline) and without-expand (in parentheses) for the RICoT rows. We return to this honestly in Section 6.6.

6.4 Pagewise vs. internal extended thinking

Modern LLMs expose internal extended thinking as a deployment-time setting, raising the question of whether RICoT's gains overlap with what that setting already provides. We run a 2×2 ablation on GLM-4.7-flash crossing pagewise interleaved thinking with internal extended thinking across all three datasets (Table 5). Pagewise yields a positive DocF1 delta on every dataset; internal-thinking alone is positive only on FinancePSS and neutral-to-negative elsewhere; the two are roughly additive when combined. Per-page output-token costs differ

Configuration	FinPSS	Tob.	TABME	Out toks
no-think (baseline)	0.688	0.822	0.661	172
internal only	+0.028	−0.012	−0.008	823
pagewise only	+0.074	+0.045	+0.022	232
both	**+0.118**	**+0.057**	**+0.046**	722

Table 5: 2×2 DocF1 deltas vs. no-think baseline on GLM-4.7-flash. Last column reports per-page output tokens on FinancePSS-flash.

Configuration	G4.7-F	G4.7	G5
RICoT (full)	0.762	0.832	0.826
− pagewise thinking	0.688 (−.074)	0.784 (−.048)	0.738 (−.088)
− context propagation	0.752 (−.010)	0.837 (+.005)	0.778 (−.048)
− carry-forward only	0.736 (−.026)	0.841 (+.009)	0.795 (−.031)
− grounding	0.695 (−.067)	0.831 (−.001)	0.755 (−.071)
− expand (no-expand)	0.741 (−.021)	0.795 (−.037)	0.769 (−.057)

Table 6: Component ablations on FinancePSS-205 DocF1. Each row removes one component from the full RICoT (with-expand) configuration; Δ shown in parentheses.

by an order of magnitude: on FinancePSS-flash, pagewise adds ∼60 output tokens to gain +0.074 DocF1, while internal thinking adds ∼650 tokens for +0.028. The per-channel cost breakdown is in Appendix D.. We therefore default to internal-thinking-off for all RICoT runs in this paper.

6.5 Component ablations

Table 6 ablates each load-bearing component of RICoT on FinancePSS-205 across the three prompted GLM models by removing one component at a time and re-running. *Pagewise thinking* is the largest single contributor on all three models; removing it costs −0.074 to −0.088 DocF1. *Context propagation* (carrying reasoning history $\mathcal{H}^{(c)}$ and active-segment summary $\mathcal{A}^{(c)}$ across windows) is the second-largest, with the cost increasing with model size (−0.010 flash, −0.048 GLM-5). *Grounding* (page-identifier fingerprint + evidence quote) is reliably positive for the smaller and largest models. *Carry-forward* alone is model-specific and mixed—it can be removed with a small gain on GLM-4.7—so the state-propagation story rests on the cognitive trace rather than carry-forward specifically. *Expand* contributes +0.021 to +0.057 to DocF1 depending on the model; the gain is largest on GLM-5 (Section 6.6). Per-dataset full ablation tables, including TABME and Tobacco800, are in Appendix E., Appendix F., and Appendix G..

6.6 Discussion

Segment-definition divergence on TABME. RICoT's TABME gap to fine-tuned baselines (GLM-5+RICoT 0.847 vs. Mistral-7B-FT 0.967) is partly a definitional one. TABME treats each source PDF as a single segment regardless of internal composition, while RICoT segments by functional scope: a bundled PDF containing, say, a quote, a purchase order, and an invoice is one TABME segment but three RICoT segments. The mismatch is one-directional—we match or split further but never under-segment. The empirical signature is visible in the page-level metrics: recall is ≥ 0.955 on every prompted model while precision spans 0.733–0.873. Fine-tuning on TABME's training split internalizes its construction convention, which closes the gap by alignment rather than capability. Per-model precision/recall and four representative false-positive cases are in Appendix B. and Appendix C..

Expand on FinancePSS. On FinancePSS, the expand step (Section 3.4) drives a meaningful share of the headline gain for the larger models: without expand, RICoT-GLM-5 ties the windowed baseline (0.769 vs. 0.769) and RICoT-GLM-4.7 leads by only +0.038; with expand the gains become +0.057 and +0.075. The flash model wins regardless (+0.154 without expand). We therefore report both with- and without-expand throughout Section 6 so the mechanism's contribution stays visible. Expand itself is a deployment-time control over what counts as a segment, enabled by the two-axis decision.

Limitations

Our evaluation is restricted to English documents; multilingual streams—common in global enterprises—remain untested. FinancePSS covers a single domain, so generalization to legal or medical content requires further validation. The framework assumes that the first page of a stream begins a document and treats each page image as one logical unit, with no granularity for multiple pages combined into a single scan. It relies on sequential continuity and cannot reassociate non-contiguous or interleaved pages into one document entity. Finally, recurrent state propagation makes within-stream processing strictly causal: multiple streams can be processed in parallel, but a single long stream must be processed sequentially.

Ethical Considerations

The FinancePSS dataset was collected from production workflows under appropriate data-handling agreements and anonymized to remove personally identifiable information (names, addresses, account numbers); document structure and formatting were preserved while sensitive content was replaced with realistic synthetic values. No human annotators beyond the research team were employed for dataset labeling. AI writing assistance was used for portions of manuscript drafting; all scientific content, experimental design, results, and conclusions are entirely the author's own.

References

Gady Agam, Shlomo Argamon, Ophir Frieder, David Grossman, and David D. Lewis. 2006. The complex document image processing (cdip) test collection. Technical report, Illinois Institute of Technology.

Anthropic. 2025. Introducing Claude Haiku 4.5. Anthropic blog post. https://www.anthropic.com/news/claude-haiku-4-5.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Fabricio Ataides Braz, Nilton Correia da Silva, and Jonathan Alis Salgado Lima. 2021. Leveraging effectiveness and efficiency in Page Stream Deep Segmentation. *Engineering Applications of Artificial Intelligence*, 105:104394.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. *Advances in Neural Information Processing Systems*, 36.

Patricia Medyna Lauritzen de Lucena Drumond, Lindeberg Pessoa Leite, Teofilo E. de Campos, and Fabricio Ataides Braz. 2023. LayoutQT—Layout Quadrant Tags to embed visual features for document analysis. *Engineering Applications of Artificial Intelligence*, 122:106091.

GLM Team, Aohan Zeng, and 1 others. 2025. GLM-4.5: Agentic, reasoning, and coding (ARC) foundation models. *arXiv preprint arXiv:2508.06471*. Z.ai. The GLM-4.7 family used in our experiments is a subsequent release in the same line; see https://github.com/zai-org/GLM-4.5 for model cards.

Hunter Heidenreich, Ratish Dalvi, Rohith Mukku, Nikhil Verma, and Neven Pičuljan. 2024. Large language models for page stream segmentation. *Preprint*, arXiv:2408.11981.

Hunter Heidenreich, Ratish Dalvi, Nikhil Verma, and Yosheb Getachew. 2025. Page stream segmentation with LLMs: Challenges and applications in insurance document automation. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 305–317, Abu Dhabi, UAE. Association for Computational Linguistics.

Kimi Team. 2025. Kimi K2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*. Base and post-trained checkpoints at https://huggingface.co/moonshotai/Kimi-K2-Instruct.

Hynek Kydlíček, Guilherme Penedo, and Leandro von Werra. 2025. Finepdfs. https://huggingface.co/datasets/HuggingFaceFW/finepdfs.

David D. Lewis, Gady Agam, Shlomo Argamon, Ophir Frieder, David Grossman, and John Heard. 2006. Building a test collection for complex document information processing. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*, pages 665–666. ACM.

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

Thisanaporn Mungmeeprued, Yuxin Ma, Nisarg Mehta, and Aldo Lipani. 2022. Tab this folder of documents: page stream segmentation of business documents. In *Proceedings of the 22nd ACM Symposium on Document Engineering*, DocEng '22, New York, NY, USA. Association for Computing Machinery.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*.

Qwen Team, An Yang, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Marc Serra Ortega, Emanuele Vivoli, Artemis Llabrés, and Dimosthenis Karatzas. 2025. CoSMo: A multimodal transformer for page stream segmentation in comic books.

University of California, San Francisco. 2007. The legacy tobacco document library (ltdl). http://legacy.library.ucsf.edu/.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Gregor Wiedemann and Gerhard Heyer. 2018. Multimodal page stream segmentation with convolutional neural networks. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1735–1738. ACM.

Gregor Wiedemann and Gerhard Heyer. 2020. Multimodal page stream segmentation with convolutional neural networks. *Language Resources and Evaluation*, 54(4):867–888.

Roy Xie, David Qiu, Deepak Gopinath, Dong Lin, Yanchao Sun, Chong Wang, Saloni Potdar, and Bhuwan Dhingra. 2026. Interleaved reasoning for large language models via reinforcement learning. *Preprint*, arXiv:2505.19640.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and 1 others. 2020. Big bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems*, volume 33, pages 17283–17297.

Z.ai. 2025. GLM-5: 744b-parameter mixture-of-experts foundation model. Open-weight model release under MIT license. https://z.ai/ / https://github.com/zai-org/. 40B active parameters per token; technical report forthcoming.

Alex L. Zhang, Tim Kraska, and Omar Khattab. 2025. Recursive language models. *Preprint*, arXiv:2512.24601.

Appendix A. TABME-250 Stream IDs

The 250 TABME stream IDs used for all prompting experiments were sampled uniformly at random from TABME's 5,000-stream test split with a fixed seed and held constant across runs:

2492, 1410, 1806, 1699, 2082, 2914, 1944, 3341, 4049, 4073, 2664, 2227, 1672, 2658, 2375, 2024, 2099, 758, 1374, 2002, 3178, 4357, 3601, 1295, 4428, 134, 3494, 2527, 4925, 176, 1965, 4676, 701, 1492, 2347, 4195, 2997, 4865, 225, 1994, 4981, 3060, 2037, 575, 3342, 1517, 3121, 705, 2449, 204, 3614, 440, 2233, 4297, 2162, 4230, 1863, 3107, 2826, 3273, 1089, 3823, 4807, 409, 2605, 2794, 2499, 2845, 1320, 4997, 638, 3191, 1464, 601, 1223, 2791, 3144, 3678, 3659, 650, 2810, 3403, 3540, 2984, 1685, 2817, 4879, 1943, 3412, 4797, 1695, 554, 2866, 813, 3749, 367, 1325, 1638, 1494, 1970, 1509, 42, 2060, 1381, 1760, 4188, 619, 172, 331, 3695, 1399, 3172, 946, 3794, 4975, 1175, 3756, 3687, 1383, 2917, 3688, 4758, 714, 2665, 566, 226, 4821, 4661, 1645, 3663, 4370, 1642, 3498, 3055, 3543, 2606, 3203, 2136, 4858, 989, 1969, 4098, 4362, 2895, 1713, 2278, 1352, 4967, 3934, 1789, 3929, 2532, 2941, 1807, 4933, 2830, 3536, 2859, 3185, 4783, 1395, 402, 2964, 2715, 2117, 1892, 3175, 3503, 3522, 1006, 2581, 4912, 3889, 2334, 4482, 3186, 2422, 2106, 1487, 1237, 950, 1773, 1157, 2870, 3070, 4806, 4825, 3149, 4882, 1179, 3857, 3210, 4991, 844, 2153, 2858, 1061, 4378, 642, 4574, 4533, 471, 2766, 1723, 4900, 786, 746, 3501, 1362, 4893, 4376, 4107, 858, 3825, 1598, 4911, 2088, 2955, 3285, 308, 1465, 1840, 4090, 4270, 485, 4192, 4265, 4598, 242, 3747, 3572, 2772, 2103, 1756, 3923, 2398, 4604, 1143, 2785, 3097, 2860.

Our distilled 2B model is additionally evaluated on the full 5,000-stream test set (TABME-full); the subset / full numbers track closely (DocF1 0.756 vs. 0.770 for Qwen-3.5-2B$^{\star,\mathrm{PwT}}$), supporting the 250-stream subset as representative of the full distribution.

Appendix B. TABME Page-Level Precision/Recall

Table 7 reports the page-level precision/recall gap referenced in Section 6.6. Recall is $\geq$ 0.955 on every prompted model; precision spans 0.733–0.873. The pattern is consistent with the segment-definition divergence argument.

Model	PageP	PageR	Gap
G4.7-F	0.733	0.955	+0.222
G4.7	0.832	0.962	+0.130
C-H4.5	0.861	0.965	+0.104
G5	0.873	0.978	+0.105

Table 7: TABME-250 page-level precision/recall under RICoT prompting.

Appendix C. Representative TABME False-Positive Cases

Four illustrative cases where RICoT (GLM-5) identified a functional-scope boundary that TABME's PDF-identity ground truth treats as internal to a single segment.

Case A: multi-part research report (stream 0134, page 38). A research report split into Part 1 (pages 34–37) and Part 2 (pages 38–39), with independent pagination. TABME labels the whole PDF as one segment. RICoT reasons: "*New document begins: 'Segmentation Learning – Part 2' from Burrows & Associates, same date 02/11/97. Header shows 'Part 2 Page 1' and reference code 51679 3320; distinct part of the study with its own page numbering starting at 1.*" The explicit part-marker plus pagination reset is a functional-scope boundary by our definition; an FP under TABME's.

Case B: sample request after technical-info bundle (stream 0242, page 45). Pages 40–45 are a single TABME segment: Henkel technical information (p. 40), an MSDS (p. 41–44), and a sample-

request form (p. 45). RICoT reasons: "*Active segment is the Henkel technical information (p. 40) which explicitly referenced the MSDS (now complete on p. 44) but did not declare or index this sample-request form. No fulfillment evidence applies.*" The active segment's anticipated scope is fulfilled and the new page's initiation evidence is not satisfied by that scope, so by our definition the sample request is a distinct functional document.

Case C: fax cover sheet at end of sales-report bundle (stream 0172, page 9). Pages 5–9 form a single TABME segment: a sales performance report (pages 5–7), an attachment (page 8), and a fax cover sheet (page 9). RICoT reasons that the active segment's anticipated scope (report plus attachment) had been fulfilled on page 8, and that the fax cover sheet introduces a distinct transmission document with its own header and metadata. By our functional-scope definition the fax cover is a separate segment; by TABME's PDF-identity definition it is internal.

Case D: legislative session calendar appended to newsletter (stream 0308, final page). A tobacco-industry newsletter ends with a state legislative session calendar that substantively supports the newsletter's legislative coverage. RICoT identified the format and content shift as a boundary and treated the calendar as a distinct document. This case is genuinely ambiguous: the calendar plausibly continues the newsletter as a topical appendix; we flag it as borderline rather than as a clean over-segmentation.

Appendix D. Token-Cost Breakdown

Table 8 expands the cost figures in Section 6.4 with per-channel breakdowns, all on GLM-4.7-flash with FinancePSS-205 input. Input is approximately constant across modes; output variation is dominated by the internal-thinking budget, which constitutes 73%–82% of total output in internal modes. When internal thinking is on, external pagewise-thinking tokens drop from 72 to 44 ($\sim$39% reduction), evidence of partial substitution between the two reasoning channels.

Appendix E. Interleaved-Thinking Ablation (Full)

Table 9 expands the pagewise-thinking ablation across all three datasets and all three prompted models. The *think* row is the full RICoT configuration with pagewise thinking; *windowed_think* replaces the interleaved per-page reasoning with a single block of reasoning emitted before the window-level decisions; *wo_think* disables structured reasoning entirely. FinancePSS uses with-expand.

Mode	In	Out	Int.	Ext.	Dec.	C-fwd
no-think	752	172	0	0	142	21
pagewise	797	232	0	72	126	20
windowed	772	251	0	83	128	20
int.+pagewise	767	722	528	44	115	16
int. only	736	823	674	0	110	17

Table 8: Per-page output-token cost on GLM-4.7-flash, FinancePSS-205, broken out by component. Input is roughly constant across modes. *Int.* = internal reasoning tokens; *Ext.* = external (pagewise) reasoning tokens.

Dataset	Configuration	G4.7-F	G4.7	G5
FinancePSS	think (full)	0.762	0.832	0.826
	windowed_think	0.754	0.814	0.746
	wo_think	0.688	0.784	0.738
Tobacco800	think (full)	0.867	0.961	0.967
	windowed_think	0.846	0.968	0.981
	wo_think	0.822	0.942	0.981
TABME-250	think (full)	0.683	0.794	0.847
	windowed_think	0.677	0.805	0.847
	wo_think	0.661	0.762	0.829

Table 9: Interleaved-thinking ablation, DocF1 by dataset and model.

Appendix F. State-Propagation Ablation (Full)

Table 10 expands the state-propagation ablation across all three datasets and three models. The full RICoT configuration carries both the reasoning history $\mathcal{H}^{(c)}$ ("cognitive trace") and the active-segment summary $\mathcal{A}^{(c)}$ (which includes the model-emitted `active_segment_carry_forward` block) across windows. *only_cognitive_trace* keeps only $\mathcal{H}^{(c)}$; *only_active_segment* keeps only $\mathcal{A}^{(c)}$; *no_carry_forward* drops the model-emitted carry-forward block but keeps the deterministic summary; *no_context_propagation* disables all cross-window state.

The cognitive trace is consistently load-bearing (removing it hurts noticeably), while the carry-forward block is mixed across models—supporting the framing in Section 6.5 that the state-propagation story rests on the cognitive trace rather than carry-forward specifically.

Dataset	Configuration	G4.7-F	G4.7	G5
FinancePSS	full	0.762	0.832	0.826
	only_cognitive_trace	0.748	0.824	0.766
	only_active_segment	0.749	0.801	0.786
	no_carry_forward	0.736	0.841	0.795
	no_context_prop.	0.752	0.837	0.778
Tobacco800	full	0.867	0.961	0.967
	only_cognitive_trace	0.887	0.971	0.981
	only_active_segment	0.892	0.935	0.968
	no_carry_forward	0.910	0.968	0.967
	no_context_prop.	0.892	0.961	0.977
TABME-250	full	0.683	0.794	0.847
	only_cognitive_trace	0.692	0.799	0.854
	only_active_segment	0.668	0.789	0.818
	no_carry_forward	0.677	0.816	0.855
	no_context_prop.	0.670	0.751	0.796

Table 10: State-propagation ablation, DocF1.

Appendix G. Decision-Block Ablation (Full)

Table 11 ablates the per-page decision schema. *barebone* drops everything except y_i and ℓ_i; *no_auxiliary* drops auxiliary tags (⟨page_role⟩, ⟨sub_document_start⟩); *no_grounding* drops the fingerprint and evidence quote; *no_nested* drops the sub-document axis z_i. The pattern inverts with model size: smaller models often pay a capacity tax for schema overhead, while larger models are roughly invariant or benefit from the structure.

Dataset	Configuration	G4.7-F	G4.7	G5
FinancePSS	full	0.762	0.832	0.826
	barebone	0.708	0.772	0.760
	no_auxiliary	0.775	0.809	0.770
	no_grounding	0.695	0.831	0.755
	no_nested	0.740	0.812	0.746
Tobacco800	full	0.867	0.961	0.967
	barebone	0.939	0.961	0.987
	no_auxiliary	0.910	0.971	0.971
	no_grounding	0.851	0.967	0.977
	no_nested	0.824	0.968	0.977
TABME-250	full	0.683	0.794	0.847
	barebone	0.661	0.812	0.851
	no_auxiliary	0.672	0.804	0.849
	no_grounding	0.647	0.791	0.841
	no_nested	0.656	0.802	0.845

Table 11: Decision-block ablation, DocF1.

Appendix H. Per-Dataset Full Result Dumps

Complete per-dataset result tables—all metrics (DocF1, PageF1, PageP, PageR, MNDD_μ, STP%); all backbone×method×ablation combinations; both with-expand and without-expand variants on FinancePSS—are included in our supplementary material as `finepss_results.md`, `tobacco_results.md`, and `tabme_results.md`.

Appendix I. Schema-Mixed Distillation: Dropout Configuration

Per-field dropout probabilities for Transformation 2 of Section 4.2 are listed in Table 12. Values were chosen heuristically on a small held-out validation set and not subjected to hyperparameter sweeps owing to compute constraints. The pagewise reasoning dropout (Transformation 1) is applied independently at the chunk-level probability shown in the first row.

Field	p_{drop}
Pagewise reasoning blocks (chunk-level)	0.40
Per-page: `document_class`	0.20
Per-page: `confidence`	0.20
Per-page: `grounding` (fingerprint+evidence)	0.20
Per-page: `sub_document` nested instance	0.15
Carry-forward: `progress` field	0.30

Table 12: Dropout probabilities used in schema-mixed distillation.

Note. The probabilities above are working values used in the single training run reported in the paper. We did not sweep them; the dropout-vs-performance landscape remains unexplored.

Appendix J. Training Corpus Attrition Breakdown

Table 13 summarizes the per-stage attrition of the FinePDF-derived training corpus. Of the 300K candidate documents sampled from the 12.3M-document finance subset, roughly 55% survive download and language-alignment filtering. The main attrition stages are: failed Common Crawl fetch (cache miss / dead URL), page-count mismatch between the downloaded PDF and FinePDF metadata (indicating unreliable OCR alignment), and detection of pages in scripts unsupported by our OCR model. The surviving 150,448 documents feed into teacher labeling.

Appendix K. Training Hyperparameters

Full QLoRA hyperparameters used for Qwen-3.5-2B fine-tuning (Section 4.3):

- LoRA rank 32; $\alpha = 64$; dropout 0; target modules `q_proj`, `k_proj`,

Stage	Count
FinePDF (English subset)	∼206M docs
Finance filter (stages 1+2)	12.3M docs
Sampled for training pool	300K docs
Post download + re-OCR + language filter	150,448 docs
Teacher labeling (chunk-level samples)	467,229 chunks
Final training corpus	46,791 docs
multi-segment	20,959
single-segment	25,832

Table 13: Training corpus construction pipeline.

`v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj`.

- Optimizer: AdamW-8bit; weight decay 0.01; max grad norm 1.0.
- Learning rate: 8×10^{-5} with cosine schedule and 8% warmup ratio.
- Batch size: per-device 8, gradient-accumulation 2 (effective 16); evaluation batch size 32.
- Epochs: 1; validation split 1% ($\sim$1,800 samples).
- Precision: bf16; gradient checkpointing on.
- Hardware: $4\times$NVIDIA A100 80GB; wall-clock $\sim$44 hours.

Appendix L. Worked Training-Sample Example

A complete example showing the same source chunk rendered into a training sample (i) with all schema fields present and pagewise reasoning emitted, (ii) with pagewise reasoning dropped (Transformation 1), and (iii) with both pagewise reasoning and several decision sub-fields dropped (Transformation 2 stacked on 1) is included in our supplementary material.

Appendix M. FinePDF Finance-Filter Classifier

The Stage 1 finance lexicon contains three tiers of keyword patterns of increasing specificity (high-confidence finance terms, moderate-confidence terms, and disambiguation terms used to distinguish between finance vs. accounting vs. marketing contexts). The Stage 2 XGBoost classifier was trained on $\sim$100K Stage-1-surviving documents labeled by an LLM judge (prompted to apply binary finance vs. non-finance labels using a detailed rubric); we used TF-IDF features with a 50K-term vocabulary capped at unigrams and bigrams. The decision threshold was tuned on a 5K-document held-out set to maximize $F_{0.5}$ (precision-weighted). Lexicon source files and classifier weights are included in our supplementary material.

Algorithm 1 RICoT page-stream segmentation

Require: Stream $\mathbf{X}$, token budget B, page cap $w_{\max}$, overlap $o=1$, history size k
Ensure: Decisions $\mathbf{d} = (d_1, \ldots, d_N)$
$\mathcal{H}\leftarrow\emptyset$; $\mathcal{A}\leftarrow\emptyset$; $\kappa\leftarrow\emptyset$; $\mathcal{C}\leftarrow\emptyset$; $c\leftarrow1$
while pages remain **do**
  $W_c \leftarrow$ PACKWINDOW$(\mathbf{X};\ B, w_{\max}, o)$ // greedy until budget or cap
  tag W_c with `<stream>`/`</stream>` if first/last
  $(\mathbf{T}_c, \mathbf{d}_c, \kappa') \leftarrow$ ICoT$(W_c, \mathcal{H}, \mathcal{A}\cup\kappa)$
  $\mathbf{d}_c \leftarrow$ VALIDATE$(\mathbf{d}_c, \mathcal{C})$ // cache-fill, retry, fallback
  $\mathcal{C}\leftarrow\mathcal{C}\cup\{d : \text{valid}\}$; append $\mathbf{d}_c$ to $\mathbf{d}$
  $\mathcal{H}\leftarrow$ last k pairs of $(\mathbf{T}_c, \mathbf{d}_c)$
  if $\exists\, d \in \mathbf{d}_c : y=\text{S}$ **then**
    $\mathcal{A}\leftarrow$ metadata from new segment
  else
    $\mathcal{A}\leftarrow\mathcal{A}\cup\mathbf{d}_c$
  end if
  $\kappa\leftarrow\kappa'$ ($\emptyset$ if final window)
  $c\leftarrow c+1$
end while
return $\mathbf{d}$

Appendix N. Prompt Templates and Inference Configuration

All RICoT and baseline prompt templates, including the system instruction, stream-context format, and per-page decision schema, are included in our supplementary material. All inference runs use temperature 0 and greedy decoding. For prompted models, we use each provider's default sampling configuration with internal extended thinking explicitly disabled in the RICoT runs.

Appendix O. Pseudocode for the RICoT Inference Loop

Algorithm 1 gives the full inference-time procedure referenced in Section 3.3.