CORTEX: High-Quality Cross-Domain Organization of Web-Scale Corpora through Ontological Corpus Graph

Chengtao Gan♠, Xiaoke Guo♠, Yushan Zhu♣, Zhaoyan Gong♠, Zhiqiang Liu♠, Songze Li♠, Huajun Chen♠, Wen Zhang♠†
♠Zhejiang University
♣JIUTIAN Research, Beijing, China
{chengtaogan,zhang.wen}@zju.edu.cn

Abstract

The continuous evolution of large language models drives escalating demands on data scale and quality, and as different training stages impose increasingly tailored data requirements, systematic organization of high-quality corpora becomes indispensable. Existing corpus construction pipelines confine the resulting corpora to flat, undifferentiated document collections, universally lacking systematic knowledge organization. We present **CORTEX**, to our knowledge the first framework that elevates web-scale corpus construction from flat document filtering to structured knowledge organization through an **Ontological Corpus Graph (OCG)**, a three-layer heterogeneous structure unifying a quality-refined content layer, a hierarchical lightweight ontology layer via LLM-driven automated evolution, and a cross-domain alignment layer enabling inter-domain association at arbitrary taxonomic resolution. Comprehensive experiments confirm the effectiveness of CORTEX. In particular, we leverage the OCG to synthesize CORTEXBENCH, a cross-domain search-and-reasoning benchmark whose evaluation across eight frontier LLMs validates the effectiveness of quality refinement, domain organization, and cross-domain data synthesis. We will publicly release the complete codebase, a 24.14 B-token refined corpus with its OCG, and CORTEXBENCH.

1 Introduction

Large language model (LLM) performance exhibits empirical power-law scaling with both model size and training data volume (Kaplan et al., 2020; Hoffmann et al., 2022), driving an ever-growing demand for massive training corpora. Web corpora such as Common Crawl (Common Crawl Foundation) have become the dominant source for trillion-token-scale pretraining data; however, raw web text is inherently noisy, containing substantial low-quality, redundant, or harmful content. Recent studies further demonstrate that data quality critically governs scaling efficiency, with noisy data degrading the effective contribution of each additional training token (Penedo et al., 2024; Li et al., 2024). The central challenge has thus shifted from merely accumulating data to *systematically distilling high-quality content* from massive-scale noisy web sources.

Existing corpus construction pipelines have evolved from rule-based heuristic filters (Raffel et al., 2020) to classifier-based quality scoring (Penedo et al., 2024; Li et al., 2024); however, these methods typically evaluate along a single quality dimension, whereas text quality inherently demands multi-dimensional assessment. While frontier LLMs can provide such assessment comparable to human annotators (Gilardi et al., 2023), deploying them at trillion-token scale remains prohibitively expensive. Beyond quality, existing pipelines universally lack *systematic knowledge organization*, uniformly producing flat document collections without domain taxonomy or inter-domain relationship modeling (Table 1), precluding fine-grained domain-specific extraction and cross-domain association exploitation. These gaps underscore the pressing need for frameworks that jointly address quality refinement and knowledge organization. High-quality curated corpora also remain highly imbalanced across languages (Joshi et al., 2020; Wang et al., 2024; Chen et al., 2023), with many widely spoken languages facing scarcity, further motivating language-agnostic solutions.

To jointly address these challenges, we present **CORTEX** (**C**urated **O**ntological **R**efinement and **T**iered Graph-**E**nabled Cross-domain Ne**X**us), a framework applicable to any target language that refines large-scale raw corpora and organizes the resulting high-quality content with an ***Ontological Corpus Graph*** **(OCG)**, a three-layer heterogeneous structure unifying content refinement, lightweight ontology construction, and cross-domain associ-

† Corresponding author

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 22700–22724
October 24-29, 2026 ©2026 Association for Computational Linguistics

ation modeling (Figure 1). **(1) The *High Quality Content Layer*** is constructed through knowledge distillation-based quality assessment; we design an Ordinal-Aware Regression *(OAR)* loss to distill teacher LLMs' multi-dimensional scoring capabilities into a 0.3 B-parameter student with over 1,000× compression, enabling reliable quality assessment at trillion-token scale. **(2) The *Lightweight Ontology Layer*** provides a hierarchical lightweight ontology constructed through LLM-driven automated evolution, and **(3) the *Alignment Layer*** bridges content and the Lightweight Ontology via Typicality–Fidelity–Specificity (TFS) weighted linkages, enabling inter-domain association analysis and domain-aware retrieval at arbitrary taxonomic resolution.

We validate CORTEX through three complementary experiments: (i) knowledge distillation pipeline validation with corpus and OCG distributional analysis, (ii) continual pre-training across four seed models on OCG-organized domain-specific and neighbor-chain data, and (iii) OCG-driven synthesis of CORTEXBENCH, a cross-domain search-and-reasoning QA benchmark evaluated on eight frontier LLMs under a stable protocol. Together, these experiments validate the effectiveness of quality refinement, domain organization, inter-domain association modeling, and cross-domain data synthesis.

Our main contributions are as follows:

- We propose **CORTEX**, to our knowledge the first web-scale corpus construction framework to unify content quality refinement, lightweight ontology construction, and cross-domain association modeling via an ***Ontological Corpus Graph*** **(OCG)**.
- We validate CORTEX through continual pre-training across four seed models and further design an OCG-driven data synthesis method and apply it to construct CORTEXBENCH; evaluation on eight frontier LLMs confirms the high difficulty and discriminative power of the synthesized instances, jointly validating the corpus quality, OCG-based data organization, and synthesis methodology.
- We will publicly release the complete CORTEX codebase, the trained scoring models, a 24.14 B-token refined corpus with its accompanying OCG, and CORTEXBENCH.

2 Related Work

Corpus Quality Early pipelines such as C4 (Raffel et al., 2020) rely on rule-based heuristic filters with quality signals confined to syntactic properties (Henriksson et al., 2025). Subsequent work introduces model-based quality classification (Penedo et al., 2024; Li et al., 2024), and Nemotron-CC (Su et al., 2025) deploys classifiers for quality, topic, and synthetic content detection. QuRating (Wettig et al., 2024) collects pairwise LLM judgments across four quality criteria. While frontier LLMs have demonstrated assessment quality comparable to human annotators (Gilardi et al., 2023), deploying them directly at trillion-token scale remains prohibitively expensive. Knowledge distillation (Hinton et al., 2015) offers a pathway to compress such capabilities into lightweight models.

Knowledge Organization Existing large-scale corpus construction methods focus predominantly on filtering mechanism design (Penedo et al., 2024; Li et al., 2024; Penedo et al., 2023; Su et al., 2025), producing flat document collections without knowledge organization (Table 1). Automated taxonomy construction has evolved from corpus-driven term clustering (Zhang et al., 2018) to LLM-based methods such as Chain-of-Layer (Zeng et al., 2024) and TaxoAdapt (Kargupta et al., 2025), but these remain limited to domain-specific settings.

Cross-Domain Synthesis LLM-based data synthesis is effective for training and evaluation resource construction (Wang et al., 2023; Gunasekar et al., 2023), but quality critically depends on source documents (Lupidi et al., 2024). Existing methods primarily operate on individual documents without exploiting cross-domain relationships. Cross-document reasoning benchmarks such as HotpotQA (Yang et al., 2018) and MuSiQue (Trivedi et al., 2022) advance multi-passage reasoning but lack cross-domain customization, and web search evaluation (Wei et al., 2025) lacks the stability required for controlled assessment (Chen et al., 2025).

3 Methodology

We present CORTEX (**C**urated **O**ntological **R**efinement and **T**iered Graph-**E**nabled Cross-domain Ne**X**us), a framework that distills high-quality corpora from web-scale corpora and organizes them with an ***Ontological Corpus Graph*** **(OCG)**. As shown in Figure 1, the OCG

is a three-layer heterogeneous structure formally defined as

$$\mathcal{G} = \langle \mathcal{L}_C,\ \mathcal{L}_O,\ \mathcal{L}_A \rangle. \tag{1}$$

The **High Quality Content Layer** $\mathcal{L}_C = \mathcal{D}_{\text{refined}}$ (Stage III (a) in Figure 1; §3.1) is a quality-refined document corpus constructed through multi-stage preprocessing and knowledge distillation-based quality assessment (Stages I–II). The **Lightweight Ontology Layer** $\mathcal{L}_O = \mathcal{C}$ (Stage III (b) in Figure 1; §3.2) is a hierarchical Concept Chain Set constructed via LLM-driven automated evolution. The **Alignment Layer** $\mathcal{L}_A = (\mathcal{K},\ \Phi,\ W)$ (Stage III (c) in Figure 1; §3.3) bridges $\mathcal{L}_C$ and $\mathcal{L}_O$, enabling inter-domain association analysis at arbitrary taxonomic resolution. Table 1 compares CORTEX with representative pipelines.

Procedure	CORTEX	FineW	DCLM	C4	Ne-CC
Preprocessing					
HTML-to-Text	✓	✓	✓	○	✓
Body extraction	✓	✓	✓	✗	✓
Metadata extr.	✓	✗	✗	✗	✗
Language ID	✓	✓	✓	✓	✓
Deduplication	✓	✓	✓	✓	✓
Quality Assessment					
Causal reasoning	✓	✗	✗	✗	✗
Q&A / explan.	✓	✗	○	✗	○
Factual accuracy	✓	○	✗	✗	○
Expression qua.	✓	○	✗	○	○
Info. density	✓	○	✗	✗	○
Harmful content	✓	○	○	✓	○
Temporal	✓	✗	✗	✗	✗
Cross-domain Modeling					
Domain class.	OCG	✗	✗	✗	✗
Inter-domain ass.	OCG	✗	✗	✗	✗

✓ explicitly addressed; ○ partially captured; ✗ not addressed.

Table 1: Comparison of CORTEX with representative web corpus pipelines. Unabbreviated version with full row descriptions is in Appendix A.

3.1 High Quality Content Layer

The High Quality Content Layer $\mathcal{L}_C$ is constructed through two stages. Stage I (§3.1.1) converts raw WARC records from Common Crawl into a preprocessed corpus $\mathcal{D}_0$. Stage II refines $\mathcal{D}_0$ into $\mathcal{L}_C = \mathcal{D}_{\text{refined}}$ via knowledge distillation-based quality assessment (§3.1.2–3.1.4).

3.1.1 Data Acquisition and Preprocessing

As shown in Stage I of Figure 1, this stage converts raw WARC records into the preprocessed corpus $\mathcal{D}_0$ through three standard steps with two key distinctions from prior pipelines. First, in addition to Resiliparse (Bevendorff et al., 2018, 2021) for high-throughput body text extraction (Li et al., 2024), we employ Trafilatura (Barbaresi, 2021) for structured metadata extraction (e.g., publication date), which is not retained by existing pipelines but is leveraged by our temporal relevance classifier (§3.1.2). Second, our three-stage coarse-to-fine language filtering incorporates the document-level quality scorer (§3.1.4) as an implicit final language purity check. Global exact deduplication follows snapshot-wide practices (Penedo et al., 2024; Su et al., 2025). Full preprocessing details are in Appendices C–D.

3.1.2 Multi-Teacher Annotation

For each scoring task, a stratified sample $\mathcal{D}_{\text{anno}} \subset \mathcal{D}_0$ is drawn and annotated by an ensemble of frontier LLMs $\mathcal{M} = \{\mathcal{M}_1, \ldots, \mathcal{M}_{|\mathcal{M}|}\}$ serving as teachers (specific model selection are detailed in Appendix E). For each sample $x_n \in \mathcal{D}_{\text{anno}}$, each teacher $\mathcal{M}_m$ independently produces a quality score, and the consensus soft label is $y_n = \frac{1}{|\mathcal{M}|}\sum_{m=1}^{|\mathcal{M}|} \mathcal{M}_m(x_n) \in [0, 1]$. This continuous score is then mapped to a hard label via fixed thresholds: LOWQ for $y_n \in [0,\ 0.4]$, MIDQ for $y_n \in (0.4,\ 0.7)$, and HIGHQ for $y_n \in [0.7,\ 1.0]$.

In contrast to existing pipelines that assess quality along a single axis (Penedo et al., 2024), our rubric integrates multiple quality dimensions, including causal reasoning, factual accuracy, expression quality, and harmful content filtering, into a unified All-in-One assessment (Table 1; full rubric in Appendix E.2). Teachers additionally produce a binary *temporal relevance* label (TIME-REL or TIME-IRREL), indicating whether the content is time-sensitive or time-invariant.

3.1.3 Ordinal-Aware Regression (OAR) Loss

Training the student to predict quality scores $\hat{y}_n \in [0, 1]$ requires simultaneously learning the teacher's continuous scoring standard and its discrete classification boundaries, as purely regression-based or classification-based training each sacrifices one of these objectives. Inspired by ordinal regression methods (Cao et al., 2020; Shi et al., 2023), we propose the **Ordinal-Aware Regression (OAR)** loss, which jointly optimizes a robust regression objective and a soft ordinal boundary objective:

$$\mathcal{L}_{\text{OAR}} = \lambda_{\text{reg}}\, \mathcal{L}_{\text{reg}} + \lambda_{\text{ord}}\, \mathcal{L}_{\text{ord}}. \tag{2}$$

Both loss components incorporate adaptive sample weights w_n that emphasize boundary-adjacent

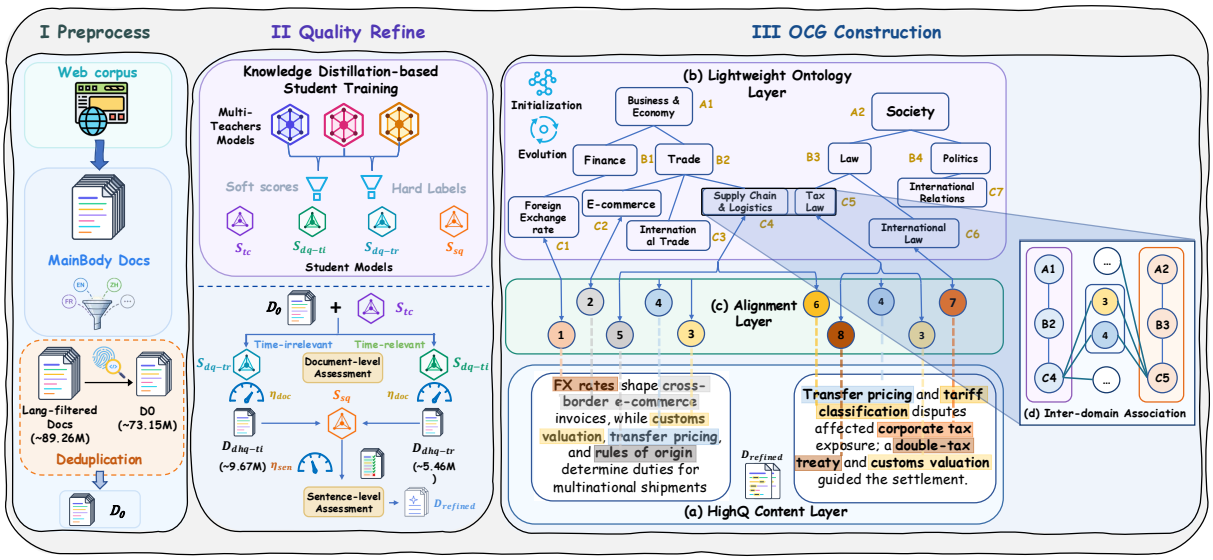

Figure 1: Overview of the CORTEX framework.

and target-class samples; the full weighting scheme is detailed in Appendix F.

Weighted Huber Regression Loss. To robustly fit the teacher's continuous scores, we employ a weighted Huber loss (Huber, 1992): $\mathcal{L}_{\text{reg}} = \frac{1}{N}\sum_{n=1}^{N} w_n \,\text{Huber}_\delta(\hat{y}_n - y_n)$, where $\text{Huber}_\delta(e) = \frac{1}{2}e^2$ if $|e| \le \delta$ and $\delta(|e| - \frac{1}{2}\delta)$ otherwise.

Soft-Threshold Ordinal Boundary Loss. Let $t_1 < t_2$ denote the two ordinal boundaries. We define a temperature-scaled boundary sigmoid $c_j(s;\tau') = \sigma\big((t_j - s)/\tau'\big)$ for $j \in \{1,2\}$, where $\tau'{>}0$ controls boundary sharpness, inducing a smooth three-class distribution with components $q_{\text{L}}{=}c_1,\ q_{\text{M}}{=}c_2{-}c_1,\ q_{\text{H}}{=}1{-}c_2$. We construct the *predicted* distribution $\hat{\mathbf{p}}_n = [\hat{c}_1,\ \hat{c}_2{-}\hat{c}_1,\ 1{-}\hat{c}_2]$ with $\hat{c}_j = c_j(\hat{y}_n;\tau)$, and the *soft target* $\mathbf{p}^*_n = [c^*_1,\ c^*_2{-}c^*_1,\ 1{-}c^*_2]$ with $c^*_j = c_j(y_n;\tau^*)$. The ordinal boundary loss is:

$$\mathcal{L}_{\text{ord}} = -\frac{1}{N}\sum_{n=1}^{N} w_n \sum_{c\in\{\text{L,M,H}\}} p^*_{n,c}\log \hat{p}_{n,c}. \qquad (3)$$

Further details are in Appendix G.

3.1.4 Knowledge Distillation-Based Student Training and Assessment

We train lightweight student models $\mathcal{S}_\theta$ to replicate the teacher ensemble's scoring behavior, producing $\hat{y}_n = \mathcal{S}_\theta(x_n) \in [0,1]$ (Appendix H). Given the teacher-annotated set $\{(x_n, y_n)\}_{n=1}^{N}$, the student is optimized by:

$$\theta^* = \arg\min_\theta \frac{1}{N}\sum_{n=1}^{N} \ell\big(\mathcal{S}_\theta(x_n),\, y_n\big), \qquad (4)$$

where $\ell = \mathcal{L}_{\text{OAR}}$ (Eq. 2) for quality scoring and $\ell = \mathcal{L}_{\text{BCE}}$ (binary cross-entropy) for temporal classification. We train four models: a temporal classifier $\mathcal{S}^*_{\text{tc}}$, document-level quality scorers $\mathcal{S}^*_{\text{dq-tr}}$ (TIME-REL) and $\mathcal{S}^*_{\text{dq-ti}}$ (TIME-IRREL), and a sentence-level scorer $\mathcal{S}^*_{\text{sq}}$. Post-hoc decision thresholds η_{doc} and η_{sen} are calibrated on held-out sets (Appendix J). Further details are in Appendix I.

The trained models are applied sequentially to $\mathcal{D}_0$ (Stage II in Figure 1). $\mathcal{S}^*_{\text{tc}}$ first partitions documents into $\mathcal{D}_0^{\text{TR}}$ and $\mathcal{D}_0^{\text{TI}}$. Each subset is scored by its dedicated quality scorer; documents exceeding η_{doc} form $\mathcal{D}_{\text{dhq}}$. Each $d \in \mathcal{D}_{\text{dhq}}$ is segmented into sentences $\{s_1, \ldots, s_{N_d}\}$ (Appendix K); $\mathcal{S}^*_{\text{sq}}$ retains the subset $\mathcal{S}^+_d = \{s_i \mid \hat{y}_{\text{sq}}(s_i) \ge \eta_{\text{sen}}\}$, whose sentences are concatenated in original order to produce $\tilde{d}$. The refined corpus $\mathcal{D}_{\text{refined}} = \{\tilde{d} \mid d \in \mathcal{D}_{\text{dhq}},\ \mathcal{S}^+_d \ne \emptyset\}$ completes $\mathcal{L}_C$.

3.2 Lightweight Ontology Layer

The Lightweight Ontology Layer $\mathcal{L}_O = \mathcal{C}$ provides a multi-resolution hierarchical domain taxonomy for organizing the refined corpus (Stage III (b) in Figure 1). further details are in Appendix N.1.

3.2.1 Initialization and Evolution

Concept Chain Set Initialization. The initial Concept Chain Set $\mathcal{C}^{(0)}$ is bootstrapped from

Wikipedia's portal and manually reviewed (Appendix N.1). It comprises 196 leaf-level concept chains spanning 12 top-level domains.

LLM-Driven Automated Evolution. To maximize the Concept Chain Set's coverage of the domains present in $\mathcal{D}_{\text{refined}}$, we introduce an automated evolution procedure. A small random subset $\mathcal{D}_{\text{evo}} \subset \mathcal{D}_{\text{refined}}$ is sampled to drive the evolution. Documents in $\mathcal{D}_{\text{evo}}$ are processed sequentially; at each step t, a frontier LLM $\mathcal{M}_{\text{evo}}$ takes document d_t and the current Concept Chain Set $\mathcal{C}^{(t-1)}$ as input:

$$\mathcal{M}_{\text{evo}}\big(d_t,\, \mathcal{C}^{(t-1)}\big) \;\rightarrow\; \big\{(c_i,\, \gamma_i)\big\}_{i=1}^{n_t}, \quad (5)$$

producing $n_t \in \{1, 2, 3\}$ chain–confidence pairs with $\gamma_i \in [0, 1]$. Two outcomes are possible: **(a)** when at least one existing chain achieves confidence $\gamma_i \geq \tau_{\text{conf}}$, all returned chains satisfy $c_i \in \mathcal{C}^{(t-1)}$ and the set remains unchanged: $\mathcal{C}^{(t)} = \mathcal{C}^{(t-1)}$; **(b)** when no existing chain reaches τ_{conf}, the LLM proposes a candidate new chain $c_{\text{new}} \notin \mathcal{C}^{(t-1)}$ subject to structural constraints (Appendix N.3), which is *directly incorporated* into the set: $\mathcal{C}^{(t)} = \mathcal{C}^{(t-1)} \cup \{c_{\text{new}}\}$. Newly added chains immediately become available for subsequent matching, enabling the taxonomy to grow incrementally throughout the evolution process.

3.2.2 Convergence Monitoring and Post-Evolution Admission Control

Convergence is monitored via a sliding-window update rate (Appendix N.5); evolution terminates when no proposal occurs for P consecutive documents. A post-hoc *admission control* step then filters candidate chains $\mathcal{C}_{\text{cand}} = \mathcal{C}^{(T_{\text{stop}})} \setminus \mathcal{C}^{(0)}$, where T_{stop} is the termination step. Each candidate $c \in \mathcal{C}_{\text{cand}}$ is scored as $\text{Score}(c) = \log_{10}(N_{\text{hit}}{+}1) \cdot (w_1\bar{C} + w_2 P_{\text{high}})$, where N_{hit} is the number of documents matched to c, $\bar{C}$ the average matching confidence, P_{high} the fraction with confidence $> \tau_{\text{conf}}$, and w_1, w_2 are balancing weights. The finalized Lightweight Ontology Layer is $\mathcal{L}_O = \mathcal{C} = \mathcal{C}^{(0)} \cup \{c \in \mathcal{C}_{\text{cand}} \mid \text{Score}(c) \geq \tau_{\text{admit}}\}$, containing **403 leaf-level concept chains** (Appendix L).

3.3 Alignment Layer

The Alignment Layer $\mathcal{L}_A = (\mathcal{K},\, \Phi,\, W)$ bridges $\mathcal{L}_C$ and $\mathcal{L}_O$ by establishing keyword-mediated linkages between documents and concept chains (Stage III (c) in Figure 1).

3.3.1 Document–Concept Association

Each document in $\mathcal{D}_{\text{refined}}$ is associated with the most relevant concept chains from $\mathcal{C}$. Each chain c is first expanded into a natural-language description $\mathcal{E}(c)$ (Appendix N.4). Let $\mathbf{v}_d$ and $\mathbf{v}_c$ denote the embeddings of document d and $\mathcal{E}(c)$, respectively. The association confidence is $\alpha(d, c) = \cos(\mathbf{v}_d,\ \mathbf{v}_c)$, and we retain the top-$M$ ($M{=}3$) chains for each document: $\mathcal{C}_d = \text{top-}M_{c\in\mathcal{C}}\,\alpha(d, c)$. Further details are in Appendix L.

3.3.2 Keyword Extraction and Linking

For each document in $\mathcal{D}_{\text{refined}}$, we extract the top-$T$ ($T{=}5$) keywords (Appendix O). Each keyword k in document d is then linked to its single most relevant concept chain among $\mathcal{C}_d$: $\phi_d(k) = \arg\max_{c\in\mathcal{C}_d} \cos(\mathbf{v}_k,\ \mathbf{v}_c)$, where $\mathbf{v}_k$ is the embedding of keyword k. The linkage functions $\Phi = \{\phi_d\}_{d\in\mathcal{L}_C}$ constitute the second component of $\mathcal{L}_A$. The linking confidence is $\text{conf}_d(k) = \cos(\mathbf{v}_k,\, \mathbf{v}_{\phi_d(k)})$.

3.3.3 TFS Weighting and Inter-domain Association

The Alignment Layer encodes inter-domain relationships through keyword-mediated connections between concept chains. Let $\mathcal{D} := \mathcal{D}_{\text{refined}}$ denote the document set, $\mathcal{C}$ the Concept Chain Set ($|\mathcal{C}|{=}403$), $\mathcal{K}$ the global keyword vocabulary, and $\mathcal{C}_d$ the top-M chains of document d.

TFS Global Connection Weight. The TFS weight $W_{c,k}$ between a concept chain c and a keyword k integrates three factors, *Typicality*, *Fidelity*, and *Specificity*:

$$W_{c,k} = P(k\,|\,c) \cdot \text{AC}(c, k) \cdot \text{IDF}_{\text{chain}}(k). \quad (6)$$

Let $\mathcal{D}_c = \{d \in \mathcal{D} \mid c \in \mathcal{C}_d\}$ and $\text{F}(c, k) = \sum_{d\in\mathcal{D}_c} \mathbb{I}[\phi_d(k){=}c]$, where the indicator contributes 1 only when k is among the top-T keywords of d and is linked to c. *Typicality* $P(k \mid c) = \text{F}(c, k)/|\mathcal{D}_c|$ measures how frequently k is linked to c. *Fidelity* $\text{AC}(c, k){=}\sum_{d\in\mathcal{D}_c} \text{conf}_d(k)\, \mathbb{I}[\phi_d(k) = c] \,/\, \text{F}(c, k)$ captures the average linking confidence when k is linked to c. *Specificity* penalizes keywords linked to many chains via a *chain-level* inverse document frequency: $\text{IDF}_{\text{chain}}(k) = \log \frac{|\mathcal{C}|+1}{|\mathcal{C}_k|}$, where $\mathcal{C}_k{=}\{c{\in}\mathcal{C} \mid \text{F}(c, k) > 0\}$.

Inter-domain Association. As illustrated in Stage III (d) of Figure 1, the OCG captures inter-domain relationships through association metrics

computable at *any level* of the taxonomy. Each leaf chain c has a TFS weight vector $\mathbf{w}_c \in \mathbb{R}^{|\mathcal{K}|}$ with components $[\mathbf{w}_c]_k = W_{c,k}$. For any hierarchical node p, let $\mathcal{C}_p^{\downarrow} = \{c \in \mathcal{C} \mid c \text{ starts with } p\}$ be its descendant leaf chains. We define the accumulated signal vector $\mathbf{s}_p = \sum_{c \in \mathcal{C}_p^{\downarrow}} \mathbf{w}_c$, whose component $S_{p,k} = \sum_{c \in \mathcal{C}_p^{\downarrow}} W_{c,k}$ aggregates keyword weights across all descendant leaves. When p is a leaf chain, $\mathcal{C}_p^{\downarrow}=\{p\}$ and $\mathbf{s}_p=\mathbf{w}_p$. The association between any two nodes p_i and p_j is:

$$\mathrm{Assoc}(p_i, p_j) = \frac{\sum_{k \in \mathcal{K}} S_{p_i,k}\, S_{p_j,k}}{\|\mathbf{s}_{p_i}\|\ \|\mathbf{s}_{p_j}\|}. \tag{7}$$

Further details are in Appendix P.

Concept Chain-to-Document Inverted Index. We construct an inverted index mapping each concept chain to its associated documents, supporting efficient retrieval at any level of the hierarchy. Together with the association metrics, the completed OCG $\mathcal{G} = \langle \mathcal{L}_C, \mathcal{L}_O, \mathcal{L}_A \rangle$ (Eq. 1) provides a unified, queryable structure for domain-aware corpus organization, enabling flexible extraction of high-quality data at arbitrary domain granularity and cross-domain scope.

4 Experiments

We instantiate CORTEX on Chinese, motivated by both the substantial demand and the pronounced scarcity of high-quality Chinese web corpora (Wang et al., 2024; Chen et al., 2023), and evaluate along three research questions: **RQ1**: Can the knowledge distillation-based refinement pipeline effectively distill high-quality content from web-scale corpora, and what distribution does the resulting corpus and OCG exhibit? **RQ2**: Is the refined corpus high-quality, and does the OCG accurately organize and retrieve domain-specific content while capturing latent inter-domain associations? **RQ3**: Can the high-quality refined corpus and the constructed OCG effectively support cross-domain synthesis of high-value data?

4.1 Pipeline Validation and Analysis (RQ1)

To address RQ1, we validate the effectiveness of the knowledge distillation-based refinement pipeline and characterize the distributional properties of the corpus and OCG. (Appendix I.4).

Knowledge Distillation Effectiveness. Table 2 reports the performance of the four knowledge-distilled student models on their held-out natural-distribution test sets (Appendix I). All students are initialized from MacBERT-Large (∼0.3 B parameters; Appendix H) and trained to learn the consensus scoring criteria jointly established by a three-teacher LLM ensemble comprising DeepSeek-V3.1 (DeepSeek-AI, 2024), GPT-5 (OpenAI, 2026), and Doubao-Seed-1.6. All quality scorers achieve target-class recall exceeding 95%, ensuring that virtually all teacher-endorsed high-quality content is preserved during corpus-scale filtering. The Spearman correlations of 0.805–0.922 confirm strong ordinal alignment with teacher consensus soft scores, validating the OAR loss design (§3.1.3). These results demonstrate that the distillation pipeline compresses the scoring behavior of a multi-hundred-billion-parameter teacher ensemble into a single 0.3 B student (>1,000× compression; Appendix E.1), enabling reliable quality assessment at the scale of tens of billions of tokens.

Student	Task	Target	Recall	ρ
$\mathcal{S}^*_{\text{tc}}$	Temporal Classif.	82.56 ^{F_1}	—	—
$\mathcal{S}^*_{\text{dq-tr}}$	Doc Quality (TR)	80.33 ^{F_2}	98.00	.805
$\mathcal{S}^*_{\text{dq-ti}}$	Doc Quality (TI)	84.25 ^{F_2}	95.83	.922
$\mathcal{S}^*_{\text{sq}}$	Sentence Quality	96.30 ^{F_2}	98.33	.908

Table 2: Student model performance on held-out test sets. Metric definitions are in Appendix R.

Tier	Documents	Fraction	Avg Tokens
HIGHQ	15.13 M	20.69%	1,796
MIDQ	18.30 M	25.02%	1,288
LOWQ	39.72 M	54.30%	2,036
TI	36.50 M	49.91%	2,517
TR	36.65 M	50.09%	1,085
Total	73.15 M	100%	1,800

Table 3: Quality-tier and temporal distribution of $\mathcal{D}_0$. TI: TIME-IRREL; TR: TIME-REL. **Avg Tokens**: mean main-body token count per document.

Corpus Quality Distribution. Table 3 summarizes the quality-tier and temporal distributions of the preprocessed corpus $\mathcal{D}_0$ (73.15 M documents). HIGHQ documents account for 20.69%, constituting the 15.13 M-document high-quality subset $\mathcal{D}_{\text{dhq}}$ with 27.18 B tokens. Sentence-level filtering further refines this to $\mathcal{D}_{\text{refined}}$ with 24.14 B tokens (Appendix I.4). While the temporal dimension is roughly balanced in $\mathcal{D}_0$, the ratio shifts to 63.91% TI in $\mathcal{D}_{\text{dhq}}$ (Table 13), indicating that time-invariant content (e.g., tutorials) is more likely to be of higher quality. The domain distribution

across 12 top-level domains (Table 14) reveals a pronounced long-tail pattern, with five domains each contributing less than 2% of high-quality documents (Appendix S).

4.2 Corpus Quality and OCG Validation via Continual Pre-Training (RQ2)

To address RQ2, we evaluate our method through continual pre-training (CPT) (Appendix T.1).

4.2.1 Experimental Setup

Data Configurations. We select the *Finance* domain (c_{fin} = `Business & Economy > Finance`) as a case study and define two CPT data configurations using the OCG's inverted index and association metric (Eq. 7). Let $\mathcal{D}_c^{\text{TI}} = \{d \in \mathcal{D}_c \mid d \in \mathcal{D}_0^{\text{TI}}\}$ denote the TIME-IRREL subset associated with chain c, and $\mathcal{N}_K(c) = \text{top-}K_{c' \in \mathcal{C} \setminus \{c\}} \operatorname{Assoc}(c, c')$ the K nearest neighbor chains: **(i) +FIN**: $\mathcal{D}_{c_{\text{fin}}}^{\text{TI}}$, consisting of time-invariant documents from the finance chain c_{fin}; **(ii) +FIN^{++}**: $\mathcal{D}_{\text{fin}^{++}}^{\text{TI}} = \bigcup_{c' \in \{c_{\text{fin}}\} \cup \mathcal{N}_{10}(c_{\text{fin}})} \mathcal{D}_{c'}^{\text{TI}}$, additionally incorporating the top-10 neighbor chains of c_{fin} retrieved via the OCG's association metric (Table 15).

Training and Evaluation. We evaluate four seed models: Qwen2.5-0.5B, Qwen2.5-3B, Qwen2.5-7B (Yang et al., 2024), and Llama-3-8B (Llama Team, 2024). All CPT is performed via full-parameter training(Table 16); *Domain capability* is assessed on CFBenchmark (Lei et al., 2023), comprising six dimensions: Knowledge, Calculation, Explanation, Identification, Analysis, and Compliance (Appendix T).

4.2.2 Results

Table 4 presents the results across all model and data configurations.

Comparing the Base, +FIN, and +FIN^{++} rows reveals consistent improvements across all four models; for instance, Llama-3-8B improves from 13.9 to 29.5 in average score and Qwen2.5-7B from 39.4 to 50.8, with +FIN^{++} further raising Qwen2.5-7B to 52.1. These gains jointly validate the quality of the CORTEX-curated corpus, the accuracy of the OCG's domain partitioning and inverted-index retrieval, and the ability of inter-domain associations to surface complementary cross-domain training signal beyond the target domain alone. Such patterns hold consistently across models spanning 0.5 B to 8 B parameters and two model families (Appendix T).

	CFBenchmark						
Data	**Know.**	**Calc.**	**Expl.**	**Ident.**	**Anal.**	**Compl.**	**Avg**
Qwen2.5-0.5B							
Base	14.0	15.9	12.7	50.7	19.5	4.0	19.5
+FIN	14.0	23.2	28.3	46.7	19.8	4.7	22.8
+FIN^{++}	17.3	24.1	30.9	54.7	20.1	4.0	**25.2**
Qwen2.5-3B							
Base	57.3	45.4	46.7	56.0	36.0	8.0	41.6
+FIN	54.7	33.1	50.8	56.0	45.4	13.3	42.2
+FIN^{++}	58.7	26.6	55.1	56.0	47.8	13.3	**42.9**
Qwen2.5-7B							
Base	38.0	48.9	46.7	42.7	41.4	18.7	39.4
+FIN	76.0	47.9	54.9	53.3	43.4	29.3	50.8
+FIN^{++}	86.7	49.8	51.2	57.3	46.2	21.3	**52.1**
Llama-3-8B							
Base	11.3	5.4	8.2	28.0	30.7	0.0	13.9
+FIN	21.3	25.6	28.0	44.0	50.7	7.3	29.5
+FIN^{++}	22.0	28.9	35.9	57.3	36.5	6.7	**31.2**

Table 4: CPT results across all seed models and data configurations. Full dimension names are in Appendix T.4.

Data	**Standard**	**Expanded**
Base Model	19.5	19.5
$+\widetilde{\mathcal{D}}_0$	18.9	17.2
$+\widetilde{\mathcal{D}}_{\text{refined}}$	19.3	20.6
+FIN	**22.8**	**25.2**

Table 5: Average CFBenchmark scores of token-matched ablations on Qwen2.5-0.5B.

4.2.3 Ablation Studies

We conduct token-matched ablations on Qwen2.5-0.5B. Let $\mathcal{D}_{\text{refined}}^{\text{TI}}$ denote documents refined from $\mathcal{D}_0^{\text{TI}}$. We randomly sample $\widetilde{\mathcal{D}}_0, \widetilde{\mathcal{D}}_0^{++} \subset \mathcal{D}_0^{\text{TI}}$ and $\widetilde{\mathcal{D}}_{\text{refined}}, \widetilde{\mathcal{D}}_{\text{refined}}^{++} \subset \mathcal{D}_{\text{refined}}^{\text{TI}}$ without OCG-based organization. Standard and Expanded variants match the token counts of +FIN and +FIN^{++}, respectively. Other settings remain unchanged. Table 17 reports full dimension-wise results.

The refined-data configuration ($+\widetilde{\mathcal{D}}_{\text{refined}}$) outperforms the unrefined counterpart ($+\widetilde{\mathcal{D}}_0$) by 0.4 and 3.4 points for Standard and Expanded, respectively, supporting the effectiveness of quality assessment and filtering in CORTEX. OCG-organized +FIN gains 3.5 and 4.6 points over $+\widetilde{\mathcal{D}}_{\text{refined}}$, highlighting the effectiveness of OCG-based domain organization and cross-domain association.

4.3 Cross-Domain Synthesis Validation via CORTEXBENCH (RQ3)

To address RQ3, we synthesize CORTEXBENCH, a cross-domain search-and-reasoning QA benchmark, to validate the effectiveness of combining

Model	Bridge				Comparison			
	CLOSED	1-SRCH	M-SRCH	ORACLE	CLOSED	1-SRCH	M-SRCH	ORACLE
DeepSeek-V3.2	30.3	41.3 ↑11.0	52.2 ↑10.9	94.8 ↑42.6	22.6	28.8 ↑6.2	47.2 ↑18.4	94.1 ↑46.9
DeepSeek-V4-Flash	30.9	44.0 ↑13.1	54.1 ↑10.1	90.2 ↑36.1	24.8	30.3 ↑5.5	49.1 ↑18.8	92.8 ↑43.7
Gemini 3 Pro	30.9	38.1 ↑7.2	**63.0** ↑24.9	80.1 ↑17.1	28.6	31.3 ↑2.7	50.9 ↑19.6	72.2 ↑21.3
Gemini 3.1 Pro	28.1	**65.7** ↑37.6	52.5 ↓13.2	82.5 ↑30.0	24.3	**54.1** ↑29.8	40.4 ↓13.7	65.6 ↑25.2
Claude Sonnet 4.6	34.7	52.5 ↑17.8	59.1 ↑6.6	84.0 ↑24.9	25.0	41.9 ↑16.9	41.0 ↓0.9	83.1 ↑42.1
Claude Opus 4.6	35.3	50.9 ↑15.6	62.5 ↑11.6	91.9 ↑29.4	31.7	41.0 ↑9.3	**57.1** ↑16.1	**95.7** ↑38.6
GPT-4o	26.6	26.9 ↑0.3	33.8 ↑6.9	77.5 ↑43.7	21.6	23.1 ↑1.5	28.1 ↑5.0	73.4 ↑45.3
GPT-5.4	**47.2**	58.4 ↑11.2	60.9 ↑2.5	**96.1** ↑35.2	**32.3**	48.4 ↑16.1	51.6 ↑3.2	94.5 ↑42.9

Table 6: CORTEXBENCH results (Accuracy %). CLOSED: closed-book. 1-SRCH: single-round retrieval. M-SRCH: Multi-Search. ORACLE: gold evidence provided. Small values indicate absolute changes from the previous setting.

the high-quality corpus with the OCG's structured cross-domain organization for producing high-value cross-domain data. We additionally release CORTEXBENCH to support future research on cross-domain evaluation.

4.3.1 Benchmark Construction

The synthesis pipeline produces QA pairs grounded in cross-domain evidence and constructs per-instance candidate pools for controlled evaluation. It proceeds in three stages (Algorithm 1).

OCG-Guided QA Synthesis. For each instance, a keyword $k \in \mathcal{K}$ is selected, prioritizing named entities. Using $\mathcal{C}_k = \{c \in \mathcal{C} \mid \mathrm{F}(c,k) > 0\}$ (§3.3.3), we retrieve either a *high-association* chain pair $(c_1^+, c_2^+) = \arg\max_{c_i \neq c_j \in \mathcal{C}_k} \mathrm{Assoc}(c_i, c_j)$ or a *low-association* pair (analogously via $\arg\min$). For each pair (c_1, c_2), the top-N (N=40) documents per chain are retrieved via the inverted index, and an LLM-based entity extraction and matching procedure identifies an evidence pair (d_A, d_B) with $d_A \in \mathcal{D}_{c_1}$, $d_B \in \mathcal{D}_{c_2}$. A frontier LLM then synthesizes bridge and comparison QA pairs from (d_A, d_B), each paired with a candidate pool $\mathcal{S}_{\text{cand}}$ ($|\mathcal{S}_{\text{cand}}|$=6,000) (Algorithm 1; Appendix U).

Benchmark Statistics. CORTEXBENCH comprises 917 QA pairs organized along three dimensions: question type, temporal source, and chain-pair association level (Table 18; Appendix U.3). The current instantiation utilizes merely 42 of the over 1.3 million keywords in the OCG's vocabulary, a utilization ratio below 1:30,000, yet already yields 917 high-difficulty instances, demonstrating substantial scaling potential for both evaluation and training data generation.

4.3.2 Evaluation Protocol

For each instance q with gold evidence (d_A, d_B) and candidate pool $\mathcal{S}_{\text{cand}}$ ($|\mathcal{S}_{\text{cand}}|$=6,000), we define four settings: **(i)** CLOSED-BOOK: the model answers from parametric knowledge alone (lower bound); **(ii)** 1-SEARCH: the model formulates a query and retrieves chunks via $\mathrm{Search}(\mathbf{q}_{\text{kw}}, q_{\text{sum}}, m)$, a hybrid retrieval interface over $\mathcal{S}_{\text{cand}}$; **(iii)** MULTI-SEARCH: iterative query refinement via ReAct (Yao et al., 2023); **(iv)** ORACLE: gold evidence (d_A, d_B) is provided (upper bound). Unlike live web search whose dynamic environments hinder reproducibility (Chen et al., 2025), this design enables stable, fine-grained error attribution. We evaluate eight frontier LLMs using accuracy and Evidence Hit@k (Appendix U.4).

4.3.3 Results

Table 6 presents accuracy under four evaluation settings for bridge and comparison questions.

Synthesis Quality and Benchmark Difficulty. The CLOSED-BOOK setting establishes the difficulty baseline: even the best-performing model (GPT-5.4) achieves only 47.2% on bridge and 32.3% on comparison, while most models score below 35%. This confirms that the OCG-driven synthesis produces genuinely challenging cross-domain questions beyond parametric knowledge.

Retrieval Utility and Search Planning. The transition from CLOSED-BOOK to 1-SEARCH yields consistent gains ranging from 0.3 to 37.6 points on bridge. MULTI-SEARCH produces further improvements for most models (e.g., Gemini 3 Pro +24.9 to 63.0), yet some show degradation (Gemini 3.1 Pro −13.2), indicating that effective iterative search planning remains challenging for

current systems.

Benchmark Discriminative Power. Most models exceed 80% in the ORACLE setting (GPT-5.4: 96.1% on bridge), yet the large M-SRCH–ORACLE gap quantifies substantial remaining improvement space for agentic search strategies. Detailed analysis by association level, temporal source, and evidence hit rates is in Appendix U.7.

5 Conclusion

We presented CORTEX, to our knowledge the first web-scale corpus construction framework to unify content quality refinement, lightweight ontology construction, and cross-domain association modeling through an Ontological Corpus Graph (OCG). Experiments validate the effectiveness of the refinement pipeline, the OCG's domain organization and cross-domain association modeling, and the data synthesis methodology. We publicly release the complete pipeline, a 24.14 B-token refined Chinese corpus with its OCG, and CORTEXBENCH. By coupling quality refinement with structured knowledge organization, CORTEX advances corpus construction beyond flat document filtering, establishing a high-quality, well-organized ontological corpus graph at web scale.

Limitations

The current instantiation processes a single Common Crawl snapshot and is validated on Chinese; the CORTEXBENCH data synthesis utilizes only 42 of the OCG's over 1.3 million keywords. These scope choices stem from computational constraints rather than methodological limitations: each pipeline stage operates independently across snapshots, and the data synthesis procedure can readily scale to the full keyword vocabulary. This enables expansion to additional snapshots and languages, and data synthesis at substantially greater scale for both evaluation and training.

Acknowledgments

This work is founded by New Generation Artificial Intelligence-National Science and Technology Major Project 2030 (2025ZD0122800, National Natural Science Foundation of China (NSFCU23B2055/NSFC62306276), Yongjiang Talent Introduction Programme (2022A-238-G), and Fundamental Research Funds for the Central Universities (226-2023-00138).

References

Adrien Barbaresi. 2021. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL 2021 - System Demonstrations, Online, August 1-6, 2021*, pages 122–131. Association for Computational Linguistics.

Janek Bevendorff, Martin Potthast, and Benno Stein. 2021. Fastwarc: Optimizing large-scale web archive analytics. *CoRR*, abs/2112.03103.

Janek Bevendorff, Benno Stein, Matthias Hagen, and Martin Potthast. 2018. Elastic chatnoir: Search engine for the clueweb and the common crawl. In *Advances in Information Retrieval - 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings*, volume 10772 of *Lecture Notes in Computer Science*, pages 820–824. Springer.

Wenzhi Cao, Vahid Mirjalili, and Sebastian Raschka. 2020. Rank consistent ordinal regression for neural networks with application to age estimation. *Pattern Recognit. Lett.*, 140:325–331.

Jianghao Chen, Pu Jian, Tengxiao Xi, Yidong Yi, Qianlong Du, Chenglin Ding, Guibo Zhu, Chengqing Zong, Jinqiao Wang, and Jiajun Zhang. 2023. Chinesewebtext: Large-scale high-quality chinese web text extracted with effective evaluation model. *CoRR*, abs/2311.01149.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pages 2318–2335. Association for Computational Linguistics.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, Sahel Sharifymoghaddam, Yanxi Li, Haoran Hong, Xinyu Shi, Xuye Liu, Nandan Thakur, Crystina Zhang, Luyu Gao, Wenhu Chen, and Jimmy Lin. 2025. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. *CoRR*, abs/2508.06600.

Common Crawl Foundation. Common crawl corpus. `https://commoncrawl.org/`. Accessed: 2025-11-20.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for chinese natural language processing. In

Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020, volume EMNLP 2020 of *Findings of ACL*, pages 657–668. Association for Computational Linguistics.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *CoRR*, abs/2412.19437.

Rui Fang, Jian Li, Wei Chen, Bin Hu, Ying-Cong Chen, Xin Tang, and Liang Diao. 2026. Rele: A scalable system and structured benchmark for diagnosing capability anisotropy in chinese llms. *CoRR*, abs/2601.17399.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd-workers for text-annotation tasks. *CoRR*, abs/2303.15056.

goto456. 2018. Chinese stopwords corpus. https://github.com/goto456/stopwords.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks are all you need. *CoRR*, abs/2306.11644.

Erik Henriksson, Otto Tarkka, and Filip Ginter. 2025. Finerweb-10bt: Refining web data with llm-based line-level filtering. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies, NoDaLiDa/Baltic-HLT 2025, Tallinn, Estonia, March 3-4, 2025*, pages 258–268. University of Tartu Library.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. Training compute-optimal large language models. *CoRR*, abs/2203.15556.

Peter J Huber. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6282–6293. Association for Computational Linguistics.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomás Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 427–431. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Priyanka Kargupta, Nan Zhang, Yunyi Zhang, Rui Zhang, Prasenjit Mitra, and Jiawei Han. 2025. Taxoadapt: Aligning llm-based multidimensional taxonomy construction to evolving research corpora. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 29834–29850. Association for Computational Linguistics.

Yang Lei, Jiangtong Li, Ming Jiang, Junjie Hu, Dawei Cheng, Zhijun Ding, and Changjun Jiang. 2023. Cf-benchmark: Chinese financial assistant benchmark for large language model. *CoRR*, abs/2311.05812.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Kumar Guha, Sedrick Scott Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee F. Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, and 40 others. 2024. Datacomp-lm: In search of the next generation of training sets for language models. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Llama Team. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Alisia Maria Lupidi, Carlos Gemmell, Nicola Cancedda, Jane Dwivedi-Yu, Jason Weston, Jakob N. Foerster, Roberta Raileanu, and Maria Lomeli. 2024. Source2synth: Synthetic data generation and curation grounded in real data sources. *CoRR*, abs/2409.08239.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain*, pages 404–411. ACL.

OpenAI. 2026. Openai GPT-5 system card. *CoRR*, abs/2601.03267.

Guilherme Penedo, Hynek Kydlícek, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin A. Raffel, Leandro von Werra, and Thomas Wolf. 2024. The

fineweb datasets: Decanting the web for the finest text data at scale. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Hamza Alobeidli, Alessandro Cappelli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon LLM: outperforming curated corpora with web data only. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 101–108. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Xintong Shi, Wenzhi Cao, and Sebastian Raschka. 2023. Deep neural networks for rank-consistent ordinal regression based on conditional probabilities. *Pattern Anal. Appl.*, 26(3):941–955.

Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2025. Nemotron-cc: Transforming common crawl into a refined long-horizon pretraining dataset. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 2459–2475. Association for Computational Linguistics.

Martin Thoma. 2018. The wili benchmark dataset for written language identification. *CoRR*, abs/1801.07779.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Musique: Multi-hop questions via single-hop question composition. *Trans. Assoc. Comput. Linguistics*, 10:539–554.

Liangdong Wang, Bowen Zhang, Chengwei Wu, Hanyu Zhao, Xiaofeng Shi, Shuhao Gu, Jijie Li, Quanyue Ma, Tengfei Pan, and Guang Liu. 2024. CCI3.0-HQ: a large-scale chinese dataset of high quality designed for pre-training large language models. *CoRR*, abs/2410.18505.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. 2025. Browsecomp: A simple yet challenging benchmark for browsing agents. *CoRR*, abs/2504.12516.

Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. 2024. Qurating: Selecting high-quality data for training language models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 52915–52971. PMLR / OpenReview.net.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 641–649. ACM.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2369–2380. Association for Computational Linguistics.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Qingkai Zeng, Yuyang Bai, Zhaoxuan Tan, Shangbin Feng, Zhenwen Liang, Zhihan Zhang, and Meng Jiang. 2024. Chain-of-layer: Iteratively prompting large language models for taxonomy induction from limited examples. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*, pages 3093–3102. ACM.

Chao Zhang, Fangbo Tao, Xiusi Chen, Jiaming Shen, Meng Jiang, Brian M. Sadler, Michelle Vanni, and Jiawei Han. 2018. Taxogen: Unsupervised topic taxonomy construction by adaptive term embedding and clustering. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 2701–2709. ACM.

A Pipeline Comparison Table Details

Table 7 presents the full unabbreviated version of Table 1. Abbreviations used in the main-text table: **FineW**: FineWeb-Edu (Penedo et al., 2024); **Ne-CC**: Nemotron-CC (Su et al., 2025); **Metadata extr.**: Metadata extraction; **Language ID**: Language identification; **Q&A / explan.**: Q&A / explanation; **Expression qua.**: Expression quality; **Info. density**: Information density; **Temporal**: Temporal relevance; **Domain class.**: Domain classification; **Inter-domain ass.**: Inter-domain association. Each quality assessment row indicates whether the pipeline's scoring rubric or filtering mechanism explicitly accounts for that aspect of content quality. "OCG" denotes capabilities provided by the Ontological Corpus Graph.

B Data Format and Auxiliary Fields

After Stage I, each document is stored as a JSON record containing: `url`, `warc_id`, `date` (publication date via Trafilatura), `title`, `language`, `text_main` (main body text for all subsequent processing), and several auxiliary fields including full-page text, non-body text, multimedia URLs, and extracted question patterns. A complete field specification is available in our released codebase.

C Preprocessing Pipeline Details

Data License. The raw data used in this work is sourced from Common Crawl (Common Crawl Foundation), a publicly available web archive that provides its datasets free of charge for any purpose, including commercial use, under liberal terms of use. [1]

HTML Content and Metadata Extraction. We extract main body text from raw HTML using Resiliparse (Bevendorff et al., 2018, 2021), a high-performance web archive parsing toolkit. The DCLM study (Li et al., 2024) reports that Resiliparse achieves comparable downstream quality to Trafilatura (Barbaresi, 2021) while being approximately $8\times$ faster, making it well-suited for trillion-token-scale processing. Because Resiliparse does not reliably extract structured metadata (e.g., publication date), we additionally employ Trafilatura for metadata and main body text extraction. We also retain several auxiliary data fields; details are in Appendix B.

Multi-Stage Language Filtering. We adopt a three-stage coarse-to-fine strategy to identify documents in the target language: **(i)** Resiliparse's built-in language filter performs a rapid first pass, exploiting its high throughput to efficiently discard obviously off-target pages at full archive scale; **(ii)** surviving documents are re-classified with the FastText language identification model (Joulin et al., 2017), which achieves high per-language recall on the WiLI-2018 benchmark (Thoma, 2018) (e.g., 0.984 for Chinese and 1.000 for English); **(iii)** since the scoring rubric explicitly down-weights off-target-language content (§3.1.2), our document-level quality scoring model (§3.1.4) inherently penalizes such documents, serving as an implicit final language purity check.

Global Exact Deduplication. Inspired by the snapshot-wide deduplication practices of FineWeb (Penedo et al., 2024) and Nemotron-CC (Su et al., 2025), we perform global exact deduplication within a single snapshot to remove identical documents across different WARC slices. The core idea is to compute a deterministic fingerprint for each document's normalized main body text and retain only the earliest occurrence of each unique fingerprint. We implement this via a scalable three-pass *bucket–select–rewrite* algorithm; details are given in Appendix D.

D Deduplication Algorithm Details

Our three-pass deduplication operates as follows. **Pass A** scans all documents, applies a deterministic text normalization function $\mathrm{Norm}(\cdot)$—comprising NFKC Unicode normalization, whitespace/newline folding, zero-width and control character removal, and leading/trailing stripping—and computes a 128-bit fingerprint $h(d) = \texttt{xxh3_128}\big(\mathrm{Norm}(\texttt{text_main}(d))\big)$ for each document d. Fingerprint–position pairs $\langle h(d), \mathrm{pos}(d)\rangle$ are written into hash-partitioned buckets $b(d) = h(d) \bmod B$, where B is the number of buckets. Because identical texts always fall into the same

[1] https://commoncrawl.org/terms-of-use

	Pipeline Step	CORTEX	FineWeb(-Edu)	DCLM	C4	Nemotron-CC
Preprocessing	HTML-to-Text	✓	✓	✓	○	✓
	Body extraction	✓	✓	✓	✗	✓
	Metadata extraction	✓	✗	✗	✗	✗
	Language identification	✓	✓	✓	✓	✓
	Deduplication	✓	✓	✓	✓	✓
Quality Assessment	Causal reasoning	✓	✗	✗	✗	✗
	Q&A / explanation	✓	✗	○	✗	○
	Factual accuracy	✓	○	✗	✗	○
	Expression quality	✓	○	✗	○	○
	Information density	✓	○	✗	✗	○
	Harmful content	✓	○	○	✓	○
	Temporal relevance	✓	✗	✗	✗	✗
Cross-domain Modeling	Domain classification	OCG	✗	✗	✗	✗
	Inter-domain assoc.	OCG	✗	✗	✗	✗

✓ explicitly addressed; ○ partially captured; ✗ not addressed.

Table 7: Full unabbreviated version of Table 1: comparison of CORTEX with representative web corpus pipelines across preprocessing, quality assessment, and cross-domain modeling.

bucket, within-bucket deduplication is equivalent to global deduplication. **Pass B** applies a *keep-first* policy within each bucket: for each unique fingerprint f, only the document with the earliest position in lexicographic file order is retained, i.e., $d^*_f = \arg\min_{d:\, h(d)=f} \mathrm{pos}(d)$. This produces a global keep bitmap. **Pass C** rewrites only kept records to a new directory.

E Teacher Model Selection and Scoring Rubrics

E.1 Teacher Model Selection

While CORTEX can be applied to any target language without architectural changes, we instantiate it on **Chinese** in this work. The teacher models are therefore selected based on their Chinese language capabilities. We employ an ensemble of $|\mathcal{M}|=3$ frontier LLMs: DeepSeek-V3.1 (DeepSeek-AI, 2024), GPT-5 (OpenAI, 2026), and Doubao-Seed-1.6, selected based on their top rankings on the ReLE benchmark (Fang et al., 2026), a scalable evaluation system that diagnoses capability anisotropy across 7 domains and $\sim$300 fine-grained dimensions for Chinese LLMs.

Teacher Model Scale. Among the three teachers, DeepSeek-V3.1 is the only open-weight model; it inherits the DeepSeek-V3 architecture, a Mixture-of-Experts model comprising 671 B total parameters with 37 B activated per token (DeepSeek-AI, 2024). GPT-5 and Doubao-Seed-1.6 are closed-source systems whose parameter counts have not been officially disclosed. Collectively, each teacher operates at a scale that is orders of magnitude larger than the 0.3 B-parameter student, yielding a compression ratio exceeding $1{,}000\times$.

E.2 Document-Level Teacher Scoring Rubric

The document-level teacher scoring rubric instructs each teacher $\mathcal{M}_m$ to output a JSON object with `hard_label`$\in$\{HIGHQ,MIDQ,LOWQ\}, `soft_label`$\in [0,1]$, and `time_related`$\in$\{True, False\}.

The evaluation dimensions are: (1) **Causal reasoning**: presence of causal chains, argumentation, or step-by-step reasoning; (2) **Question answering / explanation**: clear answers to "why" or "how" questions; (3) **Factual definitions / descriptions**: accurate scientific, historical, or conceptual information; (4) **Expression quality**: coherence, fluency, completeness; (5) **Information accuracy**: reliability and correctness. Content exhibiting strong causal reasoning, high factual accuracy, and expressive quality receives higher scores.

Additional instructions include explicit down-weighting of adult/gambling/advertising content, temporal relevance assessment based on content nature (not metadata), and lower scores for off-target-language content. The input is constructed as `[Publication date:{date}]. {title}. {text_main}`. The complete document-level teacher scoring rubric prompt is provided in the released codebase.

E.3 Sentence-Level Teacher Scoring Rubric

The sentence-level teacher scoring rubric instructs each teacher $\mathcal{M}_m$ to output a JSON object with `hard_label`$\in$\{HIGHQ_SEN, MIDQ_SEN,

LOWQ_SEN} and `soft_label`$\in [0, 1]$, following the same threshold mapping as the document level: $[0, 0.4]$ corresponds to LOWQ_SEN, $(0.4, 0.7)$ to MIDQ_SEN, and $[0.7, 1.0]$ to HIGHQ_SEN. No temporal relevance judgment is included, as this is resolved at the document level (§3.1.4).

The rubric inherits the core quality dimensions from the document-level rubric and additionally emphasizes: (1) **Logical coherence markers**: sentences containing explicit numbering or structural markers (e.g., sequential markers such as *firstly*/*secondly*, numbered lists such as *1.*/*2.*, or enumeration symbols) receive a scoring boost, as they preserve the overall coherence of the merged paragraph in downstream processing; (2) **Causal connectives**: sentences with explicit causal connectives (e.g., *because*, *therefore*, *hence*, *as a result*, *thus it can be seen*) in valid reasoning receive a significant score increase; (3) **Information density**: substantive, information-rich sentences are preferred over low-density guiding content; (4) **Paragraph coherence**: transitional and summarizing sentences that link preceding and following context are scored favorably, balancing intrinsic quality and structural contribution. The complete sentence-level teacher scoring rubric prompt is provided in the released codebase.

F Adaptive Sample Weighting

Samples near the critical decision boundary and those belonging to the target positive class may carry more discriminative information. We apply adjustable weights $w_n = w_0 \cdot \big(1 + \lambda_+ \cdot \mathbb{I}[y_n \in \mathcal{Y}_+]\big) \cdot \big(1 + \lambda_{\text{bd}} \cdot \mathbb{I}[|y_n - t_*| \leq \epsilon]\big)$, where w_0 is the base weight, $\mathcal{Y}_+$ is the target positive class, t_* is the critical decision boundary, λ_+ and λ_{bd} are amplification coefficients, and ϵ is the boundary bandwidth. For document-level quality scoring, $\mathcal{Y}_+ = \{y : y \geq t_2\}$ (i.e., HIGHQ) and $t_* = t_2$. For sentence-level scoring, $\mathcal{Y}_+ = \{y : y > t_1\}$ (i.e., the "Keep" class) and $t_* = t_1$. Hyperparameter values are listed in Appendix L.

G OAR Temperature Behavior

Ordinal Boundary Correspondence. The two ordinal boundaries in $\mathcal{L}_{\text{ord}}$ correspond to the LOWQ/MIDQ threshold (t_1) and the MIDQ/HIGHQ threshold (t_2). Non-negativity of all three class probabilities $(q_{\text{L}}, q_{\text{M}}, q_{\text{H}})$ is guaranteed since $t_2 > t_1$ and σ is monotonically increasing, ensuring $c_2 \geq c_1$ for any input score s.

In the ordinal boundary loss $\mathcal{L}_{\text{ord}}$ (§3.1.3), the temperatures τ and τ^* are separately tunable for the predicted and target distributions, respectively. As $\tau \to 0$, the predicted distribution $\hat{\mathbf{p}}_n$ approaches a hard one-hot vector concentrated on the class containing $\hat{y}_n$, recovering standard classification. Larger τ values produce smoother distributions that permit non-zero gradients even when $\hat{y}_n$ is far from the class boundaries, facilitating optimization in early training stages. An analogous trade-off applies to τ^* for the target distribution. In practice, we set $\tau{=}\tau^*{=}0.05$ (Table 10), which provides sharp but differentiable boundary signals.

H Student Model Architecture

All student models share a unified architecture $\mathcal{S}_\theta$, parameterized by θ, initialized from MacBERT-Large (Cui et al., 2020) ($\sim$0.3 B parameters). A single linear projection head maps the encoded representation to a scalar logit, and a sigmoid function produces the predicted score $\hat{y} = \mathcal{S}_\theta(x) = \sigma(\tilde{z}) = 1/(1 + e^{-\tilde{z}})$. The output interpretation differs by task: the temporal classifier produces a binary probability, while quality scorers produce a continuous score in $[0, 1]$ mapped to ordinal labels through a post-hoc decision threshold. Since document-level tasks require processing texts far exceeding MacBERT's 512-token context window, we introduce a *chunk-then-aggregate* architecture; sentence-level scoring uses the encoder directly without chunking.

Chunk Encoding. Given a document with main body text $\mathbf{x}$, we prepend the publication date and apply a sliding window of length $L{=}512$ with stride $S{=}256$ to produce chunks $\{x_1, \ldots, x_J\}$, retaining only the first $J_{\max}$ chunks (head selection). Each chunk is independently encoded by MacBERT; its pooler output $\mathbf{h}_j$ is projected to a scalar logit $z_j = \mathbf{w}^\top \text{Dropout}(\mathbf{h}_j) + b$ for $j = 1, \ldots, J$.

Document-Level Aggregation. We aggregate chunk logits into a single document logit $\tilde{z}$ via one of two strategies depending on the task. For *temporal relevance classification*, we use the **mean**: $\tilde{z}^{\text{mean}} = \frac{1}{J}\sum_{j=1}^{J} z_j$. For *quality scoring*, we use a **proportional top-k** (top-k_ρ) strategy that focuses on the most informative segments: given a ratio $\rho \in (0, 1]$, the number of aggregated chunks is $K = \min(J,\ \max(1, \lceil \rho J \rceil))$, and the document logit is the mean of the K highest-valued chunk log-

Prompt Template for Concept Chain Expansion
 # System Prompt
You are a content expansion expert in Chinese academic writing and knowledge graph domains. Your task is to expand a "concept chain" into a short paragraph for semantic vector matching.
Requirements: preserve the semantics of the original concept chain, do not introduce irrelevant content, and do not include named entities such as specific works, brands, or people.
The output should be a natural Chinese paragraph of approximately 300–500 Chinese characters. It should cover the definition or core topic, the scope of subtopics, common discussion dimensions or scenarios, and related keywords, which should be naturally integrated into the text rather than presented as a list.
Output only the expanded text, without any prefix, suffix, quotation marks, numbering, or JSON.
 # User Prompt
Concept chain: `{chain}`
Please expand it according to the above requirements.

Table 8: Prompt template for concept chain expansion. The prompt guides the LLM to expand a concept chain into a Chinese paragraph suitable for semantic vector matching while preserving the original chain semantics and avoiding named entities.

its: $\tilde{z}^{\text{topk}} = \frac{1}{K}\sum_{i=1}^{K} z_{\pi(i)}$, where π sorts $\{z_j\}_{j=1}^{J}$ in descending order.

I Training Protocol and Data Allocation

I.1 Training Protocol

For all student models, a natural-distribution test set is held out from the teacher-annotated data before constructing balanced training sets. All models are trained with 5-fold stratified cross-validation as a *model selection* mechanism (not an ensemble): the fold yielding the best validation metric is selected as the final model. Early stopping monitors validation macro-F_1 for the temporal classifier and the target-class F_2 for quality scorers, with a patience of 2 epochs.

I.2 Data Allocation

Document-Level. The teacher-annotated set $\mathcal{D}_{\text{anno}}$ contains 60,626 documents at the document level. A 5% natural-distribution test set (3,031 samples) is held out first. From the remaining 57,595 samples, three balanced subsets are drawn: the *temporal classifier* set contains 8,220 samples (4,110 TIME-REL and 4,110 TIME-IRREL; within each temporal category, samples are uniformly distributed across the three quality tiers—HIGHQ, MIDQ, and LOWQ—at 1,370 per tier, to prevent confusion between temporal and quality signals); the TIME-REL quality scorer $\mathcal{S}^*_{\text{dq-tr}}$ uses 4,110 samples exclusively from TIME-REL documents (1,370 per quality tier: HIGHQ, MIDQ, LOWQ); the TIME-IRREL quality scorer $\mathcal{S}^*_{\text{dq-ti}}$ uses 9,939 samples exclusively from TIME-IRREL documents (3,313 per quality tier: HIGHQ, MIDQ, LOWQ).

Sentence-Level. At the sentence level, 57,024 sentences from selected documents in $\mathcal{D}_0$ are annotated by $\mathcal{M}$, from which a 5% natural-distribution test set (2,852 sentences) is held out. The sentence-level quality scorer $\mathcal{S}^*_{\text{sq}}$ is trained on 22,230 balanced samples (7,410 per quality tier: HIGHQ_SEN, MIDQ_SEN, and LOWQ_SEN). Note that sentence-level quality scoring does not involve temporal categorization, as temporal relevance is resolved at the document level (§3.1.4).

I.3 Evaluation Metrics

Since the document-level pipeline retains only HIGHQ documents, we prioritize recall of the HIGHQ class and adopt F_2 as the primary evaluation metric, which weights recall twice as heavily as precision. For the temporal classifier, we use macro-F_1—the unweighted average of per-class F_1 scores across TIME-REL and TIME-IRREL—to ensure balanced evaluation of both precision and recall across both temporal categories. For sentence-level scoring, the target is the Keep class (MIDQ_SEN∪HIGHQ_SEN), and we similarly use Keep-F_2.

I.4 Pipeline Data Flow

We instantiate CORTEX on Chinese using the Common Crawl (Common Crawl Foundation) snapshot CC-MAIN-2025-43 (∼2.61 B raw pages). Table 9 traces data volume through each stage, following the notation in §3.

Released Resources. We release the complete CORTEX pipeline, the trained student scoring models, the 24.14 B-token refined corpus with its OCG, and CORTEXBENCH, providing the research community with a reproducible end-to-end pipeline, quality-refined training data with structured knowledge organization, and a rigorous cross-domain evaluation benchmark.

Stage	Volume
Raw CC snapshot (pages)	∼2.61 B
After coarse language filter	118.94 M
After fine language filter (zh)	89.26 M
$\mathcal{D}_0$ (after global exact dedup)	73.15 M
$\mathcal{D}_{\text{dhq}}$ (after doc-level filter)	15.13 M
TIME-IRREL	9.67 M
TIME-REL	5.46 M
Token counts (main body):	
$\mathcal{D}_{\text{dhq}}$	27.18 B
$\mathcal{D}_{\text{refined}}$ (after sentence refinement)	**24.14 B**
TIME-IRREL	15.05 B
TIME-REL	9.09 B

Table 9: Data volume at each pipeline stage (Chinese instantiation on Common Crawl (Common Crawl Foundation) snapshot CC-MAIN-2025-43). Symbols follow §3.

J Threshold Calibration Protocol

Structural vs. Decision Thresholds. The ordinal boundaries t_1, t_2 in $\mathcal{L}_{\text{OAR}}$ are *fixed structural parameters* governing the training objective. At inference time, separate *post-hoc decision thresholds* η_{doc} and η_{sen} are calibrated on held-out natural-distribution sets for document-level and sentence-level scoring, respectively, to control the precision–recall trade-off. This decoupling allows the same trained model to be deployed with different thresholds for different applications.

Each natural-distribution test set is split into a *calibration* (40%) and *final test* (60%) partition. On the calibration set, we search for the task-specific decision threshold (η_{doc} for document-level quality scorers, η_{sen} for the sentence-level scorer) that maximizes the target-class F_β score:

$$F_\beta(\theta) = \frac{(1+\beta^2)\, P(\theta)\, R(\theta)}{\beta^2 P(\theta) + R(\theta)}, \tag{8}$$

where P and R are precision and recall at threshold θ. For document-level quality scorers, the target class is HIGHQ with β=2; for sentence-level scoring, the target class is Keep (MIDQ_SEN∪HIGHQ_SEN) with β=2. The temporal classifier uses a fixed threshold of 0.5 without calibration. The selected thresholds are frozen for final-test evaluation and full-corpus inference.

K Corpus-Scale Inference Details

Temporal-Specific Scoring. The document-level quality score $\hat{y}_{\text{dq}}(d)$ is produced by the task-appropriate scorer: $\hat{y}_{\text{dq}}(d) = \mathcal{S}^*_{\text{dq-tr}}(d)$ for $d \in \mathcal{D}_0^{\text{TR}}$ and $\hat{y}_{\text{dq}}(d) = \mathcal{S}^*_{\text{dq-ti}}(d)$ for $d \in \mathcal{D}_0^{\text{TI}}$. This bifurcation accounts for the differing quality criteria between time-sensitive and time-invariant content.

Sentence Segmentation. Each document $d \in \mathcal{D}_{\text{dhq}}$ is segmented into sentences using Stanza (Qi et al., 2020), a neural pipeline supporting tokenization, POS tagging, and syntactic parsing for over 70 languages.

Parameter	Temp.	DQ-TR	DQ-TI	SQ
Backbone	MacBERT-Large (∼0.3B)			
Optimizer	AdamW (lr=2e-5, wd=0.01)			
Max seq len	512	512	512	512
Stride	256	256	256	—
Chunking	Yes	Yes	Yes	No
Chunk repr.	pooler	pooler	pooler	pooler
$J_{\max}$ (train/infer)	14	14	14	—
Aggregation	mean	top-k_ρ	top-k_ρ	—
ρ	—	0.2	0.2	—
Loss	BCE	OAR	OAR	OAR
$\lambda_{\text{reg}}/\lambda_{\text{ord}}$	—	1.0/1.0	1.0/1.0	1.0/1.0
τ/τ^*	—	.05/.05	.05/.05	.05/.05
Huber δ	—	0.1	0.1	0.1
λ_+	—	1.0	1.0	1.0
λ_{bd}	—	1.0	1.0	1.0
ϵ	—	0.05	0.05	0.05
Epochs	8	10	10	10
Batch / Grad accum	8/8	4/8	4/8	4/8
Precision	bf16	bf16	bf16	bf16
Warmup ratio	0.06	0.06	0.06	0.06
Dropout	0.1	0.1	0.1	0.1
Early stop metric	macro-F_1	HighQ-F_2		Keep-F_2
Early stop patience	2			
Cross-validation	5-fold stratified (model selection)			

Table 10: Student model hyperparameters. **Temp.**: temporal relevance classifier $\mathcal{S}^*_{\text{tc}}$; **DQ-TR/DQ-TI**: document-level quality scorer $\mathcal{S}^*_{\text{dq-tr}}/\mathcal{S}^*_{\text{dq-ti}}$; **SQ**: sentence-level quality scorer $\mathcal{S}^*_{\text{sq}}$.

L Hyperparameter Settings

Table 10 summarizes student model hyperparameters.

OCG Construction. Embedding model: BGE-M3 (Chen et al., 2024) (0.56B, multilingual). Keyword fusion: α_s=β_s=0.5; top-T=5 per document. Document–concept: top-M=3 chains. Keyword–chain: top-1 per keyword. Evolution LLM $\mathcal{M}_{\text{evo}}$: Doubao-Seed-2.0-Pro (`doubao-seed-2-0-pro-260215`). Evolution: W=2,000; patience P=300; τ_{conf}=0.80; τ_{admit}=0.90; w_1=0.6, w_2=0.4.

M Computational Resources

All continual pre-training experiments (§4.2) are conducted on a cluster equipped with 48*NVIDIA

H100 80 GB HBM3 GPUs, Intel Xeon Platinum 8468 CPUs, running Rocky Linux 8.6 (Green Obsidian). All other pipeline stages, including data preprocessing, knowledge distillation training and corpus-scale inference, OCG construction, and CORTEXBENCH synthesis and evaluation, are performed on a separate environment with three NVIDIA A800 80 GB PCIe GPUs, Intel Xeon Gold 6326 CPUs at 2.90 GHz, running Ubuntu 20.04.6 LTS.

N Concept Chain Evolution Details

N.1 Initial Concept Chain Set Structure

The initial Concept Chain Set $\mathcal{C}^{(0)}$ is organized as a multi-level domain hierarchy, where each concept chain represents a root-to-leaf path. Chains may have varying depths (typically 2–4 levels), reflecting different domains' natural granularity. For example, `Natural Science>Physics>Optics` has three levels, while `Sports>Basketball` has two. The hierarchy spans 12 top-level domains (Table 11), containing 196 leaf-level concept chains in total.

A defining property of this structure is that every node at every level of the hierarchy—from the root through intermediate nodes to leaf chains—constitutes a valid domain identifier. Leaf-level nodes represent the finest-grained domains, while higher-level nodes represent progressively broader domains at coarser granularity. For instance, given the chain `Natural Science>Physics>Optics`, the leaf node `Optics` identifies a fine-grained domain, the intermediate node `Natural Science>Physics` a broader one, and the root `Natural Science` the broadest. This multi-resolution property is directly leveraged in the inter-domain association computation (§3.3.3), enabling domain relationship analysis at arbitrary resolution without any structural modification.

N.2 Matching and Evolution Protocol

The evolution procedure instructs $\mathcal{M}_{\text{evo}}$ to: (1) extract 1–3 core domain topics from each document d_t; (2) match each to existing chains in the current Concept Chain Set $\mathcal{C}^{(t-1)}$ (using full root-to-leaf paths), with a quality baseline of confidence $\geq \tau_{\text{conf}}$ for normal matching; (3) propose new leaf-level chains *only when no existing chain reaches* τ_{conf}, subject to the structural constraints specified in §N.3; (4) output JSON with `topics` and `results` (each containing `chain`, `confidence`, `is_new`, `reason`). The current complete Concept Chain Set $\mathcal{C}^{(t-1)}$ is provided to $\mathcal{M}_{\text{evo}}$ for matching and judgment. A five-step internal workflow—semantic extraction, chain matching, quality assessment, new chain addition (triggered only when needed), and ranked output—guides $\mathcal{M}_{\text{evo}}$ through a structured reasoning process. Matching existing chains is always the priority, governed by principle **P1** (Existing Chain Priority), with new chain proposals following the six generation rules of principle **P2** (Appendix N.3). Newly added chains are immediately available in subsequent evolution steps. The complete evolution prompt is provided in the released codebase.

N.3 Full Structural Constraints

Candidate new chains must satisfy six constraints: **(i)** *Domain abstractness* (highest weight): chains represent abstract domains generalizing a class of things or a knowledge domain, and must not refer to any specific named entity, including person names, place names, organization names, brand names, product models, or specific event names; **(ii)** *Upper-level stability*: the top two levels of the hierarchy remain unchanged unless absolutely necessary to accommodate a fundamentally new domain; **(iii)** *Leaf-level addition only*: new chains must append a new leaf-level concept under an existing parent path, forming a complete root-to-leaf path; it is strictly forbidden to use a path truncated at an intermediate level as a new chain; **(iv)** *Clear parent attachment*: the parent path must be unambiguous, with the new chain belonging to exactly one logically clear existing parent node; **(v)** *Sibling distinguishability*: clear semantic boundaries must exist between the new chain and its sibling concepts at the same level; synonymous or near-synonymous concepts are forbidden; **(vi)** *Granularity alignment*: the abstraction level of the new chain must match that of its sibling nodes, being neither too coarse nor too fine.

N.4 Concept Expansion Protocol

Each concept chain is expanded by an LLM into a natural-language Chinese paragraph of approximately 300–500 characters, covering definitions or core topics, the scope of subtopics, common discussion dimensions or scenarios, and related keywords. The system prompt explicitly forbids named entities (specific works, brands, people) and requires keywords to be naturally integrated into the text rather than presented as a list. The complete con-

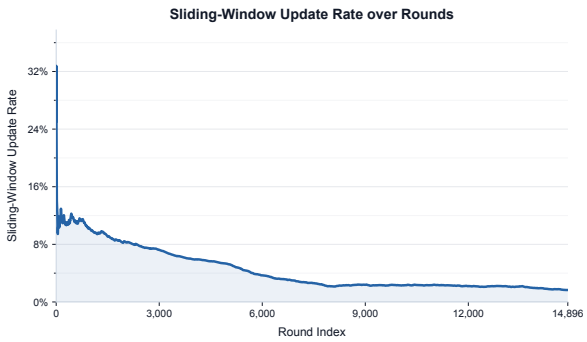

Figure 2: Sliding-window update rate r_t (Eq. 9) during concept chain evolution.

cept expansion prompt is presented in Table 8.

N.5 Convergence Monitoring

Convergence is monitored via the sliding-window update rate r_t, defined as the fraction of the most recent W documents that triggered a new chain proposal at evolution step t:

$$r_t = \frac{1}{\min(t, W)} \sum_{j=\max(1,\, t-W+1)}^{t} \mathbb{I}\big[\mathcal{C}^{(j)} \neq \mathcal{C}^{(j-1)}\big]. \tag{9}$$

We set W=5,000. Evolution terminates under an early stopping criterion: when no proposal occurs for P consecutive documents (i.e., r_t=0 persists for a window of size P), the Concept Chain Set is considered stabilized. Figure 2 plots r_t over the evolution process; the update rate exhibits an overall decreasing trend, confirming progressive stabilization of the taxonomy.

O Keyword Extraction Details

Given a document $d \in \mathcal{D}_{\text{refined}}$, keyword extraction proceeds through four stages: preprocessing, independent scoring by two methods, score fusion, and top-T selection.

Preprocessing. Traditional Chinese text is first converted to Simplified Chinese. Documents undergo word segmentation and part-of-speech (POS) tagging via jieba, a dictionary-based Chinese segmentation toolkit that combines prefix dictionary scanning with an HMM-based model for unknown word recognition. Stopwords are removed using a consolidated Chinese stopword list (goto456, 2018) (merging Baidu, HIT, and SCU resources).

POS Filtering. Let $\mathcal{W}_d^{\text{raw}} = (w_1, w_2, \ldots, w_{L_d})$ denote the segmented token sequence of document d with corresponding POS tags $(p_1, p_2, \ldots, p_{L_d})$. We retain only tokens whose POS tag belongs to the allowed set $\mathcal{P}_{\text{allow}} = \{$`n`, `nr`, `nrt`, `ns`, `nt`, `nz`, `ng`, `vn`, `an`, `eng`, `i`, `l`$\}$, producing the filtered token sequence $\mathcal{W}_d = (w_{i_1}, w_{i_2}, \ldots, w_{i_{L'_d}})$ where $p_{i_j} \in \mathcal{P}_{\text{allow}}$ for all j, and L'_d is the filtered length.

TF-IDF Scoring. For each unique keyword k in $\mathcal{W}_d$, let $f_{k,d} = |\{j : w_{i_j} = k\}|$ denote its frequency in the filtered sequence. The raw TF-IDF score is:

$$s_{\text{tfidf}}^{\text{raw}}(k, d) = \underbrace{\frac{f_{k,d}}{L'_d}}_{\text{TF}(k,d)} \cdot \underbrace{\log \frac{|\mathcal{D}_{\text{idf}}| + 1}{1 + |\{d' \in \mathcal{D}_{\text{idf}} : k \in d'\}|}}_{\text{IDF}(k)}, \tag{10}$$

where $\mathcal{D}_{\text{idf}}$ is a reference corpus from which IDF values are pre-computed.

TextRank Scoring. Following TextRank (Mihalcea and Tarau, 2004), we construct an undirected co-occurrence graph $G_d = (V_d, E_d)$ from $\mathcal{W}_d$: V_d contains all unique keywords in $\mathcal{W}_d$, and an edge $(k_i, k_j) \in E_d$ exists if k_i and k_j co-occur within a sliding window of size ω=5 tokens in $\mathcal{W}_d$. The TextRank score is computed iteratively:

$$\text{TR}^{(l+1)}(k_i) = (1-\alpha) + \alpha \sum_{k_j \in \mathcal{N}(k_i)} \frac{1}{|\mathcal{N}(k_j)|} \text{TR}^{(l)}(k_j), \tag{11}$$

where α=0.85 is the damping factor, $\mathcal{N}(k_i)$ denotes the neighbor set of k_i in G_d, and $\text{TR}^{(0)}(k_i) = 1$ for all k_i. Iteration proceeds until convergence ($\max_i |\text{TR}^{(l+1)}(k_i) - \text{TR}^{(l)}(k_i)| < 10^{-4}$), yielding $s_{\text{tr}}^{\text{raw}}(k, d) = \text{TR}(k)$.

Score Fusion and Selection. Each scoring method independently produces a ranked keyword list. We apply per-document min-max normalization to each method:

$$s_{\text{tfidf}}(k) = \frac{s_{\text{tfidf}}^{\text{raw}}(k) - \min_{k'} s_{\text{tfidf}}^{\text{raw}}(k')}{\max_{k'} s_{\text{tfidf}}^{\text{raw}}(k') - \min_{k'} s_{\text{tfidf}}^{\text{raw}}(k')}, \tag{12}$$

and analogously for TextRank, yielding $s_{\text{tr}}(k) \in [0, 1]$. The fused score is:

$$s_{\text{comb}}(k) = \alpha_s \, s_{\text{tfidf}}(k) + \beta_s \, s_{\text{tr}}(k), \tag{13}$$

with α_s=β_s=0.5. The final keyword set of document d is obtained by selecting the top-T keywords

by fused score:

$$\mathcal{K}_d = \text{top-}T_{k \in V_d}\, s_{\text{comb}}(k), \tag{14}$$

with T=5. These keywords are then linked to concept chains via the keyword–concept linking formula (§3.3.2).

P Properties of Assoc

Summation vs. Averaging. The accumulated signal vector $\mathbf{s}_p$ uses summation rather than averaging over descendant leaves: a keyword connected to many descendants naturally receives a stronger aggregated signal, reflecting its broader relevance to the node's overall domain. If averaging were used instead, a keyword appearing in a single descendant would receive the same weight as one appearing across all descendants, obscuring the breadth of its domain relevance.

Neighbor Definition. Two nodes p_i and p_j are *neighbors* if their descendant leaves share at least one active keyword, i.e., $\sum_k S_{p_i,k}\, S_{p_j,k} > 0$.

P.1 Leaf-Level Specialization and Degeneracy Consistency

When both nodes p_i=c_i and p_j=c_j are leaf-level concept chains, $\mathcal{C}^{\downarrow}_{c_i} = \{c_i\}$, hence $\mathbf{s}_{c_i} = \mathbf{w}_{c_i}$ and $S_{c_i,k} = W_{c_i,k}$ for all k. The generalized association (Eq. 7) reduces to:

$$\text{Assoc}(c_i, c_j) = \frac{\sum_{k \in \mathcal{K}_{i,j}} W_{c_i,k}\, W_{c_j,k}}{\|\mathbf{w}_{c_i}\|\, \|\mathbf{w}_{c_j}\|}. \tag{15}$$

Two leaf chains are *neighbors* if $\mathcal{K}_{i,j} \neq \emptyset$.

Proof. Since $\mathcal{C}^{\downarrow}_{c_i} = \{c_i\}$, substituting into Eq. (7) gives $\text{Assoc}(c_i, c_j) = \frac{\sum_{k \in \mathcal{K}} W_{c_i,k}\, W_{c_j,k}}{\|\mathbf{w}_{c_i}\|\, \|\mathbf{w}_{c_j}\|}$. Since $W_{c_i,k}\, W_{c_j,k} = 0$ for all $k \notin \mathcal{K}_{i,j}$, the sum over $\mathcal{K}$ collapses to the sum over $\mathcal{K}_{i,j}$, yielding Eq. (15). □

P.2 Pairwise Decomposition

Proposition 1. *The numerator of* $\text{Assoc}(p_i, p_j)$ *decomposes as*

$$\sum_{k \in \mathcal{K}} S_{p_i,k}\, S_{p_j,k} = \sum_{c \in \mathcal{C}^{\downarrow}_{p_i}} \sum_{c' \in \mathcal{C}^{\downarrow}_{p_j}} \sum_{k \in \mathcal{K}_c \cap \mathcal{K}_{c'}} W_{c,k}\, W_{c',k}. \tag{16}$$

Proof. By expanding the definitions: $\sum_k S_{p_i,k}\, S_{p_j,k} = \sum_k \big(\sum_{c \in \mathcal{C}^{\downarrow}_{p_i}} W_{c,k}\big)\big(\sum_{c' \in \mathcal{C}^{\downarrow}_{p_j}} W_{c',k}\big) = \sum_c \sum_{c'} \sum_k W_{c,k}\, W_{c',k}$, where the last equality follows from distributivity of multiplication over addition. Only terms with $k \in \mathcal{K}_c \cap \mathcal{K}_{c'}$ are non-zero, since $W_{c,k} = 0$ for $k \notin \mathcal{K}_c$. □

Each leaf-pair contribution $\sum_{k \in \mathcal{K}_c \cap \mathcal{K}_{c'}} W_{c,k}\, W_{c',k}$ enters the numerator without compression, ensuring full signal fidelity.

P.3 Worked Example

Consider computing the association between two intermediate-level nodes p_i and p_j, where:

p_i = Natural Science > Physics,
p_j = Technology&Engineering > Engineering.

Suppose p_i has two descendant leaf chains and p_j likewise:

c_1 = Natural Science > Physics > Optics,
c_2 = Natural Science > Physics > Mechanics,
c_3 = Technology&Engineering > Engineering > Civil Engineering,
c_4 = Technology&Engineering > Engineering > Electrical Engineering.

The accumulated signal vectors are $\mathbf{s}_{p_i}$=$\mathbf{w}_{c_1}$+$\mathbf{w}_{c_2}$ and $\mathbf{s}_{p_j}$=$\mathbf{w}_{c_3}$+$\mathbf{w}_{c_4}$. By the pairwise decomposition (Appendix P.2), the numerator of $\text{Assoc}(p_i, p_j)$ decomposes into four leaf-pair dot products: $\mathbf{w}_{c_1}\cdot\mathbf{w}_{c_3}$, $\mathbf{w}_{c_1}\cdot\mathbf{w}_{c_4}$, $\mathbf{w}_{c_2}\cdot\mathbf{w}_{c_3}$, and $\mathbf{w}_{c_2}\cdot\mathbf{w}_{c_4}$.

If, for instance, c_2 (Mechanics) and c_3 (Civil Engineering) share keywords that carry high TFS weights on both chains—i.e., keywords such as "structural analysis," "materials," and "stress" that are strongly linked to both domains via large $W_{c_2,k}$ and $W_{c_3,k}$ values—their dot product $\mathbf{w}_{c_2}\cdot\mathbf{w}_{c_3} = \sum_{k \in \mathcal{K}_{c_2} \cap \mathcal{K}_{c_3}} W_{c_2,k}\, W_{c_3,k}$ will dominate the sum. This demonstrates that inter-domain association is determined not merely by the number of shared keywords, but by the strength of each keyword's association with both chains (reflected in the TFS weight products). The resulting high $\text{Assoc}(p_i, p_j)$ value is thus primarily driven by the Mechanics–Civil Engineering overlap, providing directly interpretable inter-domain semantics.

At the leaf level (Appendix P.1), the association between any two leaf chains reduces to the cosine similarity between their TFS weight vectors.

Top-Level Domain	Leaf Chains
Culture & Life	34
Arts & Entertainment	37
Sports	11
Travel & Geography	7
Health & Medicine	28
History	8
Natural Science	28
Philosophy & Religion	12
Society	83
Business & Economy	44
Technology & Engineering	100
People & Biography	11
Total	**403**

Table 11: Leaf chain distribution across the 12 top-level domains after evolution and admission control.

Q Concept Chain Taxonomy

Table 11 summarizes the distribution of the 403 leaf-level concept chains across top-level domains after evolution and admission control.

The full initial Concept Chain Set $\mathcal{C}^{(0)}$ (196 leaf-level chains) and the post-evolution set $\mathcal{C}$ (403 leaf-level chains) are released in our codebase and data package, in both hierarchical and enumerated representations.

R Student Model Evaluation Details

Metric Selection. For the temporal classifier $\mathcal{S}^*_{\text{tc}}$, macro-$F_1$ is used as the primary metric to ensure balanced evaluation across both TIME-REL and TIME-IRREL. For the two document-level quality scorers $\mathcal{S}^*_{\text{dq-tr}}$ and $\mathcal{S}^*_{\text{dq-ti}}$, the target class is HIGHQ; we use F_2 (weighting recall twice as heavily as precision) because the pipeline's goal is to retain as many high-quality documents as possible. For the sentence-level quality scorer $\mathcal{S}^*_{\text{sq}}$, the target class is KEEP (HIGHQ_SEN $\cup$ MIDQ_SEN); we likewise use Keep-F_2 to maximize retention of informative sentences. Spearman rank correlation (ρ) between student predictions and teacher consensus soft scores is reported for the three quality scoring tasks to assess whether the student preserves the fine-grained ordinal ranking produced by the teacher ensemble. The consistently high ρ values (0.805, 0.922, and 0.908) confirm that the OAR loss (§3.1.3) effectively trains the student to reproduce both the teacher's classification decisions and the underlying continuous scoring standard. The calibrated decision thresholds (Appendix J) are then applied to these well-ordered scores for binary retention decisions during corpus-scale inference.

Qual.	Temp.	Docs	Doc%	Tokens	Tok%	Avg Tok.
HIGHQ	TI	9.67 M	13.22%	17.19 B	13.06%	1,778
	TR	5.46 M	7.47%	9.99 B	7.59%	1,828
MIDQ	TI	9.14 M	12.50%	12.22 B	9.29%	1,337
	TR	9.15 M	12.52%	11.34 B	8.62%	1,239
LOWQ	TI	17.69 M	24.19%	62.46 B	47.46%	3,530
	TR	22.02 M	30.11%	18.41 B	13.99%	836
Total		73.15 M	100%	131.63 B	100%	1,800

Table 12: Joint distribution of quality tiers and temporal categories in $\mathcal{D}_0$ 73.15 M documents, 131.63 B tokens. TI: TIME-IRREL; TR: TIME-REL. Qual.: Quality; Temp.: Temporal; Docs: Documents; Tok%: Token percentage; Avg Tok.: Average tokens.

Temporal	Docs	Doc%	Tokens	Tok%	Avg Tok
TI	9.67 M	63.91%	17.19 B	63.24%	1,778
TR	5.46 M	36.09%	9.99 B	36.76%	1,828
Total	15.13 M	100%	27.18 B	100%	1,796

Table 13: Temporal distribution of $\mathcal{D}_{\text{dhq}}$ (high-quality subset). TI/TR: TIME-IRREL / TIME-REL.

S Corpus Distribution Analysis

S.1 Quality $\times$ Temporal Cross Distribution

Table 12 presents the joint distribution of quality tiers and temporal categories across the full preprocessed corpus $\mathcal{D}_0$.

Notably, LOWQ documents exhibit the highest average token length (2,036 overall; Table 3), indicating that document length alone is a poor proxy for quality. Several patterns are noteworthy. Time-invariant LOWQ documents are disproportionately long (avg 3,530 tokens), likely reflecting verbose, low-information-density web pages such as repetitive template-generated content or promotional material. Conversely, time-related LOWQ content is short (avg 836 tokens), consistent with low-quality news snippets or brief social media reposts. Within the HIGHQ tier, both temporal categories show similar average lengths ($\sim$1,800 tokens), suggesting that quality assessment is largely independent of temporal characteristics once the content passes the quality threshold.

S.2 High-Quality Corpus Temporal Distribution

Table 13 summarizes the temporal breakdown of the high-quality corpus $\mathcal{D}_{\text{dhq}}$.

While the temporal dimension is nearly balanced in $\mathcal{D}_0$ (49.91% TI; Table 3), it shifts to 63.91% TI

in $\mathcal{D}_{\text{dhq}}$, indicating that time-invariant content accounts for a relatively higher proportion among high-quality documents compared to the overall corpus. Within $\mathcal{D}_{\text{dhq}}$, both temporal categories exhibit comparable average lengths (1,778 tokens for TI vs. 1,828 tokens for TR), indicating that the quality threshold is content-driven rather than length-driven regardless of temporal category.

S.3 Domain Distribution of High-Quality Documents

Table 14 shows the distribution of document–concept associations across the 12 top-level domains for $\mathcal{D}_{\text{dhq}}$. Since each document is associated with top-M=3 concept chains (§3.3.1), the total count (∼45.04 M) is approximately $3 \times |\mathcal{D}_{\text{dhq}}|$.

The temporal composition varies dramatically across domains. Knowledge-intensive domains such as Natural Science (85.07% TI), History (90.22%), and Philosophy & Religion (90.45%) are overwhelmingly time-invariant, while event-driven domains such as Sports (67.04% TR) and Society (56.42% TR) are predominantly time-related. This validates the pipeline's temporal classification and confirms that the OCG's domain organization captures meaningful structural differences in web content.

T Continual Pre-Training Details

T.1 CPT Validation Design

Concretely, the CPT experiments are designed to validate three aspects: (a) corpus quality, OCG domain organization, and domain-specific retrieval accuracy through domain-specific performance improvements; (b) the OCG's inter-domain association modeling through the incremental benefit of neighbor-domain data; and (c) robustness of these benefits across model families and scales.

All CPT experiments exclusively use TIME-IRREL data, as time-invariant content provides durable domain knowledge well-suited for continual pre-training.

Scaling and Cross-Family Robustness. The four seed models spanning 0.5 B to 8 B parameters and two model families (Qwen vs. Llama) enable analysis of scaling behavior (0.5 B to 8 B) and cross-family robustness (Qwen vs. Llama), verifying that the quality signal is intrinsic to the CORTEX-curated data rather than an artifact of specific model-data interactions. Consistent improvement patterns across scales and architectures further confirm the robustness of the pipeline's quality filtering and the OCG's domain organization.

T.2 OCG Neighbor Chains for Finance

Table 15 lists the top-10 OCG neighbor chains for c_{fin} (`Business & Economy > Finance`), ranked by $\text{Assoc}(c_{\text{fin}}, \cdot)$ (Eq. 7).

T.3 CPT Training Hyperparameters

Table 16 summarizes the CPT training configuration.

T.4 Domain Capability Benchmark

The domain capability benchmark referred to as "CFBenchmark" in the main text corresponds to CFBenchmark-OpenFinData (Lei et al., 2023), which evaluates financial domain capabilities across six dimensions. The column abbreviations used in Tables 4 and 17 are: **Know.**: Knowledge, **Calc.**: Calculation, **Expl.**: Explanation, **Ident.**: Identification, **Anal.**: Analysis, and **Compl.**: Compliance. Each dimension targets a distinct aspect of financial language understanding and reasoning. All evaluations are conducted using the official CFBenchmark evaluation toolkit.

Detailed Ablation Results. Table 17 reports the complete dimension-wise results corresponding to the average scores in Table 5.

U CORTEXBENCH Details

U.1 Benchmark Design Rationale

CORTEXBENCH is a cross-domain search-and-reasoning QA dataset in which each instance requires integrating evidence from two documents belonging to different yet OCG-associated domains. Each instance is paired with a dedicated candidate document pool ($|\mathcal{S}_{\text{cand}}|$=6,000) and a deterministic hybrid search interface. Unlike settings that rely on live web search APIs, where dynamic and opaque retrieval environments hinder fair comparisons and reproducibility, and make it difficult to isolate specific system components or attribute the sources of model failures (Chen et al., 2025), this design provides a stable, transparent interaction environment that enables reproducible evaluation and fine-grained attribution of model weaknesses to specific capabilities such as query formulation, evidence retrieval, or cross-domain reasoning. Concretely, this enables attribution of weaknesses to specific

Rk	Top-Level Domain	Assoc.	%	TI	TI%	TR	TR%
1	Technology & Engineering	15.67 M	34.80%	11.79 M	75.20%	3.89 M	24.80%
2	Society	6.89 M	15.29%	3.00 M	43.58%	3.89 M	56.42%
3	Business & Economy	5.88 M	13.05%	2.66 M	45.26%	3.22 M	54.74%
4	Health & Medicine	3.95 M	8.77%	3.08 M	78.06%	0.87 M	21.94%
5	Culture & Life	3.77 M	8.36%	2.86 M	76.05%	0.90 M	23.95%
6	Arts & Entertainment	3.23 M	7.17%	2.33 M	72.22%	0.90 M	27.78%
7	Sports	3.02 M	6.70%	0.99 M	32.96%	2.02 M	67.04%
8	Natural Science	0.79 M	1.76%	0.67 M	85.07%	0.12 M	14.93%
9	Philosophy & Religion	0.62 M	1.38%	0.56 M	90.45%	59.3 K	9.55%
10	People & Biography	0.48 M	1.07%	0.25 M	52.23%	0.23 M	47.77%
11	History	0.48 M	1.07%	0.43 M	90.22%	46.9 K	9.78%
12	Travel & Geography	0.26 M	0.58%	0.13 M	49.60%	0.13 M	50.40%
	Total	45.04 M	100%	28.78 M	63.88%	16.27 M	36.12%

Table 14: Domain distribution of high-quality documents ($\mathcal{D}_{\text{dhq}}$) across 12 top-level domains. **Assoc.**: document–concept association count. **TI/TR**: TIME-IRREL / TIME-REL breakdown within each domain.

Rank	Neighbor Chain
1	Business & Economy > Corporate Mgmt.
2	Business & Economy > Industrial Econ.
3	Society > Law
4	Business & Economy > Consumer & Retail
5	Business & Economy > Real Estate
6	Tech. & Eng. > Computer & IT
7	People & Biography > Entrepreneurs
8	Business & Economy > Trade
9	Society > Career & Employment
10	Business & Economy > Theoretical Econ.

Table 15: Top-10 OCG neighbor chains for c_{fin}, spanning 4 distinct top-level domains. This confirms that the OCG effectively models latent associations across different domains.

capabilities such as query formulation, evidence retrieval, or cross-domain reasoning.

The candidate pool comprises four equal-sized partitions: documents from the two evidence chains c_1 and c_2 (including the gold evidence d_A and d_B), and documents from two randomly selected chains $c_{r_1}, c_{r_2} \in \mathcal{C} \setminus \{c_1, c_2\}$. This design creates a controlled yet challenging retrieval setting: the gold evidence documents are embedded among same-domain distractors that share topical similarity, while cross-domain and random distractors test the model's retrieval precision and noise robustness.

U.2 Synthesis Pipeline

Algorithm 1 formalizes the OCG-driven evidence pair selection, QA synthesis, and per-QA-instance candidate set construction described in §4.3.1. Let $\mathcal{M}_{\text{ext}}$ denote the LLM-based entity extraction model, $\mathcal{E}_{\text{m3}}$ the BGE-M3 embedding model (Chen et al., 2024), and $\mathcal{M}_{\text{syn}}$ the QA synthesis LLM.

Parameter	Value
Training mode	Full-parameter
Learning rate	2×10^{-6}
LR scheduler	cosine
Warmup ratio	0.05
Batch size (eff.)	128
Precision	bf16
Max seq length	4,096
Inference engine	vLLM
Eval (CFBenchmark)	Official toolkit

Table 16: CPT training configuration (shared across all seed models and data configurations).

	CFBenchmark						
Data	Know.	Calc.	Expl.	Ident.	Anal.	Compl.	Avg
Base Model	14.0	15.9	12.7	50.7	19.5	4.0	19.5
$+\widetilde{\mathcal{D}}_0$	17.3	18.7	15.8	34.7	20.0	7.3	18.9
$+\widetilde{\mathcal{D}}_0^{++}$	10.7	9.0	23.5	36.0	19.9	4.0	17.2
$+\widetilde{\mathcal{D}}_{\text{refined}}$	10.7	16.7	23.5	45.3	16.7	2.7	19.3
$+\widetilde{\mathcal{D}}_{\text{refined}}^{++}$	7.3	20.0	27.2	49.3	17.9	2.0	20.6
+FIN	14.0	23.2	28.3	46.7	19.8	4.7	<u>22.8</u>
+FIN^{++}	17.3	24.1	30.9	54.7	20.1	4.0	**25.2**

Table 17: Full CFBenchmark results of token-matched ablations on Qwen2.5-0.5B.

Both $\mathcal{M}_{\text{ext}}$ and $\mathcal{M}_{\text{syn}}$ are implemented using GPT-5.4.

In line 10, n=5 for bridge and n=10 for comparison questions. Each keyword k contributes at most 5 candidate d_A documents, and each d_A generates at most one QA pair (line 19 breaks after the first successful synthesis). The candidate pool size per chain is N=40. Within the bridge and comparison categories, a small proportion ($\sim$10%) of QA pairs have their premises deliberately modified to contain factual errors; for these instances, a model is judged correct only if it identifies the erroneous

Algorithm 1 OCG-Guided QA Synthesis and Candidate Set Construction

Require: Keyword vocabulary $\mathcal{K}$; concept chains $\mathcal{C}$; inverted index $\{\mathcal{D}_c\}$; $\mathrm{Assoc}(\cdot,\cdot)$; entity extractor $\mathcal{M}_{\text{ext}}$; embedding model $\mathcal{E}_{\text{m3}}$; QA synthesis LLM $\mathcal{M}_{\text{syn}}$; pool size s=6,000

Ensure: QA set $\{(q, a, d_A, d_B, \mathcal{S}_{\text{cand}})\}$

1: **for** each selected keyword $k \in \mathcal{K}$ **do**
2: $\quad \mathcal{C}_k \leftarrow \{c \in \mathcal{C} \mid \mathrm{F}(c, k) > 0\}$
3: $\quad$ Select (c_1, c_2) from $\mathcal{C}_k$ (high-assoc. or low-assoc.)
4: $\quad \mathcal{D}^{\text{top}}_{c_i} \leftarrow$ top- $N_{d \in \mathcal{D}_{c_i}} \alpha(d, c_i), i \in \{1, 2\}$
5: $\quad$ **for** each $d_A \in \mathrm{sample}(\mathcal{D}^{\text{top}}_{c_1}, 5)$ **do**
6: $\quad\quad \mathcal{E}_A \leftarrow \mathcal{M}_{\text{ext}}(d_A)$ ▷ Extract entities & attributes
7: $\quad\quad$ **if** $\max_{e \in \mathcal{E}_A} \mathrm{score}(e) < \tau_{\text{ent}}$ **then**
8: $\quad\quad\quad$ **continue** ▷ Skip low-salience d_A
9: $\quad\quad$ **end if**
10: $\quad\quad \mathcal{B} \leftarrow$ top- $n_{d \in \mathcal{D}^{\text{top}}_{c_2}} \cos\big(\mathcal{E}_{\text{m3}}(\mathcal{E}_A),\ \mathcal{E}_{\text{m3}}(d)\big)$
11: $\quad\quad$ **for** each $d_B \in \mathcal{B}$ **do**
12: $\quad\quad\quad (q, a) \leftarrow \mathcal{M}_{\text{syn}}(d_A, d_B, \text{type}),\ \text{type} \in \{\text{bridge}, \text{comparison}\}$
13: $\quad\quad\quad$ **if** quality filter passes **then**
14: $\quad\quad\quad\quad$ **Build candidate pool:**
15: $\quad\quad\quad\quad \mathcal{S}_{c_1} \leftarrow \{d_A\} \cup \mathrm{sample}\big(\mathcal{D}_{c_1} \setminus \{d_A\},\ s/4{-}1\big)$
16: $\quad\quad\quad\quad \mathcal{S}_{c_2} \leftarrow \{d_B\} \cup \mathrm{sample}\big(\mathcal{D}_{c_2} \setminus \{d_B\},\ s/4{-}1\big)$
17: $\quad\quad\quad\quad$ Select random $c_{r_1}, c_{r_2} \in \mathcal{C} \setminus \{c_1, c_2\}$
18: $\quad\quad\quad\quad \mathcal{S}_{\text{cand}} \leftarrow \mathcal{S}_{c_1} \cup \mathcal{S}_{c_2} \cup \mathrm{sample}(\mathcal{D}_{c_{r_1}}, s/4) \cup \mathrm{sample}(\mathcal{D}_{c_{r_2}}, s/4)$
19: $\quad\quad\quad\quad$ **emit** $(q, a, d_A, d_B, \mathcal{S}_{\text{cand}})$; **break** ▷ d_A yields $\leq$1 QA
20: $\quad\quad\quad$ **end if**
21: $\quad\quad$ **end for**
22: $\quad$ **end for**
23: **end for**

premise, further evaluating the model's reasoning capability.

QA Types and Quality Filtering. Two QA types are synthesized: *bridge* questions require chaining facts across d_A and d_B, while *comparison* questions require contrasting entities or attributes across the two documents. A quality filter rejects QA pairs whose answers cannot be traced back to the evidence documents, ensuring that every retained question is answerable given the source material.

U.3 Benchmark Statistics

The 917 QA pairs are organized along three dimensions: (i) *question type*: bridge and comparison (Appendix U.2); (ii) *temporal source*: whether evidence documents are drawn from the TIME-IRREL or TIME-REL subset of $\mathcal{D}_{\text{refined}}$; and (iii) *chain-pair association level*: whether the two evidence chains have high or low Assoc scores (§3.3.3). Table 18 summarizes the composition.

Dimension	Category	Count
QA Type	Bridge	320
	Comparison	597
Temporal Source	TI source	544
	TR source	373
Association Level	High-association pairs	512
	Low-association pairs	405
Total		**917**

Table 18: CORTEXBENCH composition across QA type, temporal source, and chain-pair association level.

U.4 Search Interface and Evaluation Protocol

Evaluated Models. We evaluate eight frontier LLMs spanning four providers: GPT-4o and GPT-5.4 (OpenAI); Claude Sonnet 4.6 and Claude Opus 4.6 (Anthropic); Gemini 3 Pro and Gemini 3.1 Pro (Google); DeepSeek-V3.2 and DeepSeek-V4-Flash (DeepSeek). Accuracy is evaluated by an LLM-based judge that assesses semantic equivalence between the predicted and gold answers (Appendix U.5). Evidence Hit@k measures whether at least one gold evidence document appears among the top-k retrieved chunks (Appendix U.6).

Candidate Pool Construction. For each QA instance $(q, a, d_A, d_B, \mathcal{S}_{\text{cand}})$, the candidate pool $\mathcal{S}_{\text{cand}}$ ($|\mathcal{S}_{\text{cand}}|$=6,000) is constructed during synthesis (Algorithm 1, lines 14–18). Each QA pair has its own dedicated candidate pool.

Document Chunking. Because the embedding model used for retrieval (BGE-Large-zh (Xiao et al., 2024)) has a maximum input length of 512 tokens, each document $d \in \mathcal{S}_{\text{cand}}$ exceeding 512 tokens is split into non-overlapping chunks of at most 512 tokens. All subsequent retrieval operates at the chunk level.

Hybrid Scoring. For a query $(\mathbf{q}_{\text{kw}}, q_{\text{sum}})$ and each chunk d_j, two scores are computed: $s_{\text{sem}}(d_j) = \cos\big(\mathcal{E}_{\text{bge}}(q_{\text{sum}}),\ \mathcal{E}_{\text{bge}}(d_j)\big)$ using BGE-Large-zh (1,024-dim, 512 max tokens), and $s_{\text{lex}}(d_j) = \mathrm{BM25}(\mathbf{q}_{\text{kw}}, d_j)$ with default parameters (k_1=1.5, b=0.75). Each score is min-max normalized within the pool, and the final score is $s(d_j) = 0.5\,\hat{s}_{\text{sem}}(d_j) + 0.5\,\hat{s}_{\text{lex}}(d_j)$. The function returns the top-m chunks ranked by s.

Caching. Chunk embeddings for each candidate pool are pre-computed and cached. BM25 inverted indices are built per pool and reused across search

Algorithm 2 CORTEXBENCH Evaluation Protocol

Require: QA instance $(q, a, d_A, d_B, \mathcal{S}_{\text{cand}})$; evaluated model $\mathcal{M}$; setting $\in$ {CLOSED, 1-SRCH, M-SRCH, ORACLE}
Ensure: Correctness $\in \{0, 1\}$

```
if setting = CLOSED then
    â ← M(q)                                   ▷ Parametric knowledge only
else if setting = ORACLE then
    â ← M(q, d_A, d_B)                        ▷ Gold evidence provided
else
    (q_kw, q_sum, m) ← M(q),   m ≤ 5          ▷ Model sees only q
    R ← Search(q_kw, q_sum, m; S_cand)
    if setting = 1-SRCH then
        â ← M(q, R)
    else                                      ▷ M-SRCH: up to 3 rounds via ReAct
        R_all ← R;  r ← 1
        while r < 3 do
            if M decides to answer then break
            end if
            (q'_kw, q'_sum, m') ← M(q, R_all),  m' ≤ 5
            R' ← Search(q'_kw, q'_sum, m'; S_cand)
            R_all ← R_all ∪ R';  r ← r + 1
        end while
        â ← M(q, R_all)
    end if
end if
return Judge(q, a, â)
```

	Bridge		Comparison	
Model	Hit@5	Hit@15	Hit@5	Hit@15
DeepSeek-V3.2	22.7	31.4	28.6	45.2
DeepSeek-V4-Flash	23.1	31.6	30.5	45.5
Gemini 3 Pro	22.5	**55.0**	24.6	**50.9**
Gemini 3.1 Pro	**38.4**	14.1	**46.0**	12.6
Claude Sonnet 4.6	20.3	41.1	29.8	35.7
Claude Opus 4.6	20.6	40.8	29.1	49.2
GPT-4o	33.4	27.9	36.6	33.9
GPT-5.4	24.4	28.6	29.6	39.5

Table 19: Evidence Hit rates on CORTEXBENCH. **Hit@5**: 1-Search top-5. **Hit@15**: Multi-Search cumulative ($\leq$3×5).

rounds within the same QA instance.

Evaluation Protocol. Algorithm 2 formalizes the evaluation procedure for CORTEXBENCH. The evaluated model $\mathcal{M}$ receives only the question q and must autonomously extract a keyword list $\mathbf{q}_{\text{kw}}$, compose a natural-language summary q_{sum}, and specify the number of chunks to retrieve m ($m \leq 5$) before each search call. In the MULTI-SEARCH setting, the model may issue up to 3 rounds of search but may also choose to generate an answer at any earlier round based on the accumulated evidence.

U.5 LLM-as-Judge Evaluation

We use GPT-5.4 as the evaluator. Given the original question q, the gold answer a, and the predicted answer $\hat{a}$, the evaluator assesses semantic equivalence rather than exact string matching, outputting a binary judgment (correct/incorrect). The evaluator prompt template is specified in our released codebase.

U.6 Evidence Hit Rate Analysis

Table 19 reports Evidence Hit@k for each model. Evidence Hit@k is the fraction of instances where at least one gold evidence document (d_A or d_B) appears among the top-k retrieved chunks.

U.7 Results by Association Level and Temporal Source

Table 20 provides accuracy breakdowns by chain-pair association level and temporal source data.

U.8 Detailed Results Analysis

We provide a systematic analysis of CORTEXBENCH results across Table 6, Table 19, and Table 20.

Cross-Setting Performance Progression. Across all eight models, the four evaluation settings reveal a consistent progression pattern. In the CLOSED-BOOK setting, bridge questions (avg 33.0%) prove slightly more accessible than comparison questions (avg 26.4%), likely because bridge questions sometimes permit partial answers from parametric knowledge about individual entities. The 1-SEARCH setting produces highly variable improvements: Gemini 3.1 Pro achieves the largest gain on bridge (+37.6 points to 65.7%), while GPT-4o gains only 0.3 points, suggesting that single-round query formulation quality varies dramatically across model families. The MULTI-SEARCH setting benefits models with strong iterative planning (Gemini 3 Pro gains 24.9 points from 1-Search to reach 63.0% on bridge), but notably degrades Gemini 3.1 Pro by 13.2 points on bridge and 13.7 on comparison, indicating that additional search rounds can introduce noise when the initial retrieval was already highly effective.

Evidence Retrieval Analysis. Table 19 reveals a strong correspondence between evidence hit rates and accuracy. Gemini 3.1 Pro achieves the highest Hit@5 (38.4% bridge, 46.0% comparison) but the lowest Hit@15 (14.1%, 12.6%), explaining its high 1-Search accuracy but Multi-Search degradation: its initial queries are highly effective, but subsequent queries fail to locate additional evidence and may introduce distracting content. Conversely,

	By Chain-Pair Association Level								By Temporal Source							
	High-Association				Low-Association				TIME-IRREL Source				TIME-REL Source			
Model	CLS.	1-S	M-S	ORC.	CLS.	1-S	M-S	ORC.	CLS.	1-S	M-S	ORC.	CLS.	1-S	M-S	ORC.
DeepSeek-V3.2	24.0	34.2	48.4	**96.0**	26.9	31.9	49.6	92.4	27.2	33.8	48.5	93.6	22.5	32.2	49.6	**95.4**
DeepSeek-V4-Flash	25.8	35.5	50.0	91.5	28.4	34.6	51.9	92.3	27.9	38.4	52.2	91.4	25.5	30.3	48.8	92.5
Gemini 3 Pro	29.3	34.6	57.4	72.7	29.6	32.6	52.8	78.2	32.9	36.4	54.4	72.3	24.4	29.8	56.7	79.2
Gemini 3.1 Pro	26.8	**61.4**	44.7	68.2	24.2	35.1	44.4	75.8	27.9	**58.3**	48.2	72.5	22.3	**59.4**	39.4	70.3
Claude Sonnet 4.6	27.7	46.5	46.7	82.8	29.1	44.4	48.1	84.2	30.0	46.7	49.3	83.2	26.0	44.0	44.5	83.7
Claude Opus 4.6	32.0	46.3	**60.2**	94.7	34.1	42.2	**57.5**	94.0	34.4	46.7	**60.5**	**95.2**	30.8	41.3	**56.8**	93.3
GPT-4o	21.3	23.8	30.8	73.0	25.9	25.2	30.6	76.9	24.6	24.8	33.3	75.3	21.4	23.9	26.9	74.0
GPT-5.4	**36.5**	53.7	55.7	95.7	**38.8**	**49.6**	53.8	**94.2**	**40.8**	54.4	55.5	**95.2**	**32.7**	48.3	53.9	94.9

Table 20: Accuracy breakdowns on CORTEXBENCH under chain-pair association level and temporal source settings (bridge and comparison combined). CLS.: Closed-book. 1-S: 1-Search. M-S: Multi-Search. ORC.: Oracle.

Gemini 3 Pro shows the opposite pattern with low Hit@5 (22.5%, 24.6%) but the highest Hit@15 (55.0%, 50.9%), explaining its large Multi-Search gains. GPT-4o achieves relatively high Hit@5 (33.4%, 36.6%) but translates this poorly into accuracy gains (+0.3 on bridge), suggesting a reasoning bottleneck rather than a retrieval bottleneck.

Association Level Effects. Table 20 shows the Oracle gap differs: high-association pairs generally yield higher Oracle accuracy (e.g., DeepSeek-V3.2: 96.0% vs. 92.4%), suggesting that high-association evidence pairs produce more coherent cross-domain reasoning chains. The Multi-Search advantage is generally preserved across both levels, confirming the robustness of the benchmark design.

Temporal Source Effects. TIME-IRREL source instances yield slightly higher closed-book accuracy than TIME-REL instances across most models (e.g., GPT-5.4: 40.8% vs. 32.7%), consistent with time-invariant content being more likely to overlap with pre-training data. The search-augmented settings largely close this gap (e.g., Claude Opus 4.6 Multi-Search: 60.5% TI vs. 56.8% TR), demonstrating that the retrieval mechanism effectively compensates for the parametric knowledge deficit on time-sensitive content. Oracle accuracy shows minimal temporal variation, confirming that both temporal categories produce questions of comparable reasoning difficulty once evidence is provided.