Synchronos LLM Data Filtering

Elliot Epstein

January 14, 2026

This document summarizes the data filtering pipeline and evaluation for Synchronos LLM post-training.

1 Small Scale Filtering Eval

We run a small-scale audit on a 35-question gold set with human year labels and entity annotations to validate the filtering pipeline. The process follows these steps:

1. Extract time-anchored entities from the question and answer bundle.
2. Assign a best-estimate year and a 95% confidence interval per entity.
3. Generate standalone search queries for each entity.
4. Retrieve evidence from search results to ground the estimates.
5. Update entity confidence intervals based on the evidence.
6. Update the overall estimated year using the maximum upper bound across entities.

Key plots for this analysis are Figure 1 and Figure 2. Figure 1 shows that search grounding improves no-leak accuracy by roughly 10 points across models, with Gemini models reaching 100% no-leak accuracy. An additional test set beyond the current dev evaluation set would strengthen this claim, since the dev set was used to improve prompt structure and diagnose errors. Figure 2 shows the distribution of predicted year minus gold year; the most common failure is entity extraction, while search grounding appears reliable. We categorize error types using an LLM with access to the ground-truth answer. We also observe that Gemini 3 Flash with search outperforms Gemini 3 Pro with search, and GPT-5-mini with search outperforms GPT-5.2. Possible explanations are that the prompt is still ambiguous and larger models choose a valid but misaligned interpretation, the larger models are more overconfident, or the sample size is too small to draw a clear conclusion.

Annotation protocol. For each sample, the annotator read the question and answer, extracted relevant entities (omitting entities that are clearly older than the dominant ones), and searched to find the concept date for each entity. For each entity, we include a source link to the article stating the concept date unless the fact is trivially known. The full annotation pass took about 2 hours, or roughly 3.5 minutes per question.

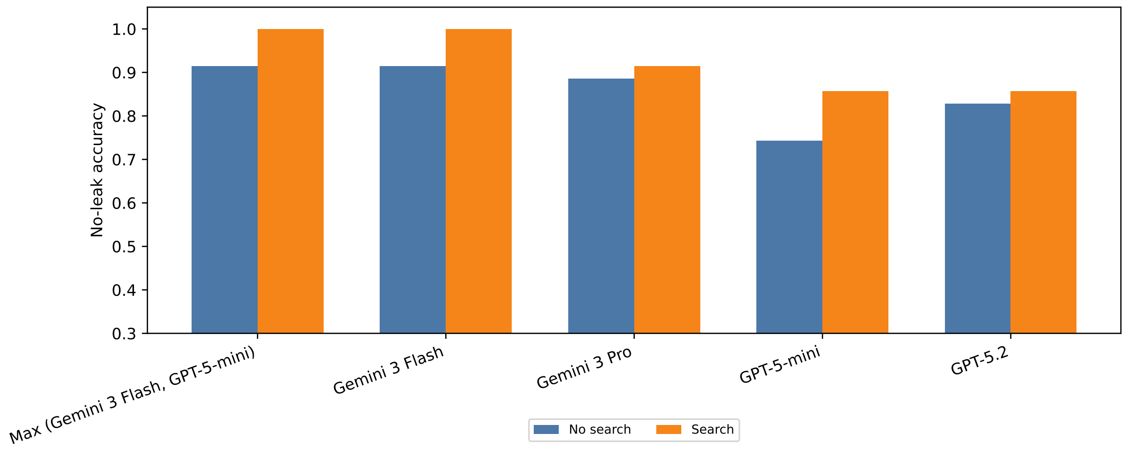


Figure 1: No-leak accuracy (predicted year $\geq$ gold) before vs. after search grounding.

Why not GLiNER. We evaluated GLiNER for entity extraction, but it is a poor fit for this task. Given fixed categories, it must both detect the entity span and map it to a category; empirically it misses entities frequently in our data. Because missed entities directly induce leakage, we instead use LLMs for the full stack (entity extraction through year estimation), with search-based evidence grounding to improve reliability.

1.1 Next Steps

Next, we plan to annotate an additional set that we will call the test set, reserved for final performance reporting. We also plan to sample multiple predictions per model and take the maximum year across samples to reduce leakage risk.

Other performance metrics. Possible options include an asymmetric error loss averaged over samples, $L = \max(0, -e) + \beta \cdot \max(0, e)$ where $e = \hat{y} - y$, $\hat{y}$ is the predicted year, and y is the gold year. This penalizes underestimates more than overestimates when $\beta < 1$, but it is harder to interpret and requires choosing β. Another option is an equally weighted average of exact year match and no-leak accuracy, which is easy to interpret but does not distinguish between off-by-one errors and large errors.

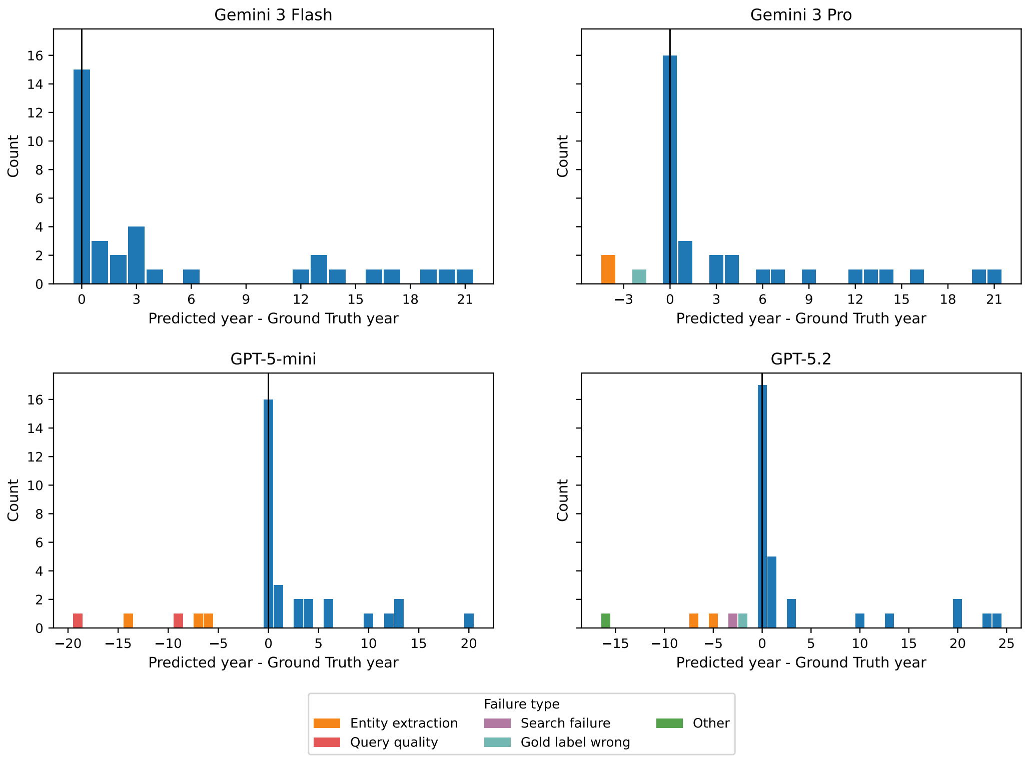
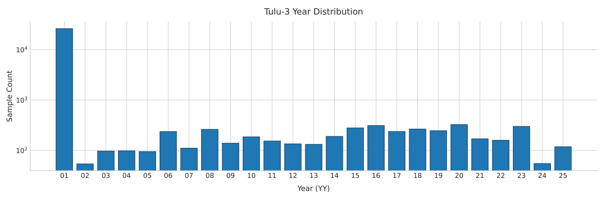
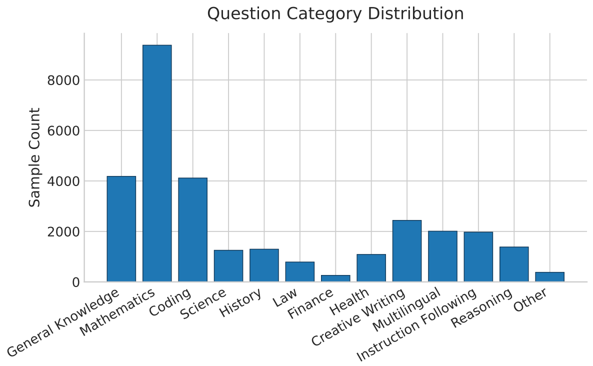


Figure 2: Search-aided prediction error (predicted year minus ground truth) by model, with negative deltas colored by failure type.

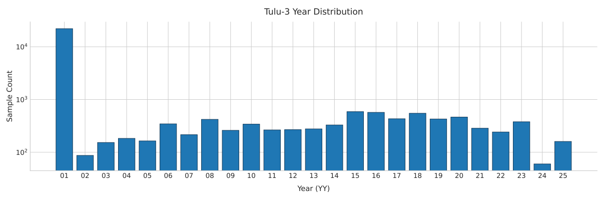


(a) Year distribution (SFT).

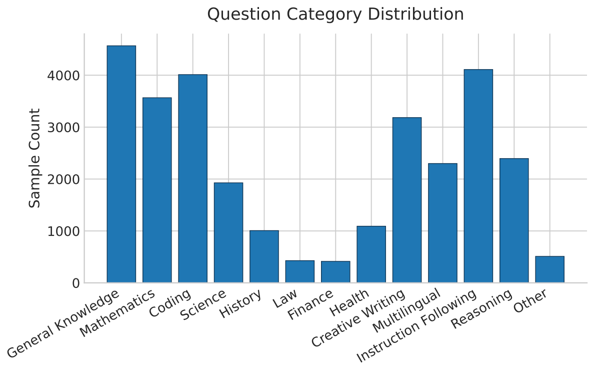


(b) Category distribution (SFT).

Figure 3: Filtering summary for the TÜLU-3 SFT mixture (session `2026-01-06_14-10PT`, $n =$ 30,549).



(a) Year distribution (DPO).



(b) Category distribution (DPO).

Figure 4: Filtering summary for the TÜLU-3 preference mixture (session `2026-01-06_14-10PT`, $n =$ 29,510).

2 Data Filtering

We label each supervised (SFT), preference, and RLVR sample with the minimum calendar year consistent with its question–answer bundle; the prompt structure is detailed in Appendix A. We considered deterministic filtering, but it was difficult to capture all edge cases with a rule-based approach. The latest sweep (session `2026-01-06_14-10PT`) processed 30,549 SFT examples, 29,510 preference examples, and three RLVR datasets (7,358 GSM, 7,372 MATH, 14,958 IFEval) with a conservative policy that uses the most recent referenced year. Figures 3–5 show year and category distributions for each dataset family and validate cutoff integrity.

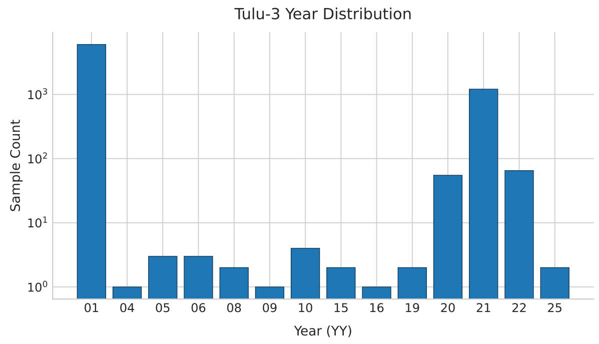
Filtering Cost. The current filtering pass uses GPT-5-mini with batch requests at \$0.25/\$2.00 per 1M input/output tokens; the batch discount halves these rates to \$0.125/\$1.00. Table 1 summarizes per-sample token averages, current costs, and projections for the full SFT/preference corpus plus the RLVR targets. TÜLU-3 still requires filtering 900k SFT examples and 250k preference examples beyond the current subset; projected costs scale linearly with per-sample token counts. For prompts under 200k tokens, Gemini 3 Flash is priced at \$0.50/\$3.00 per 1M input/output

Dataset	Current n	Tokens/sample	Current cost	Projected n	GPT-5-mini	GPT-5.2	GPT-5.2 Pro
SFT	30,549	1,661	\$20.46	930,549	\$623	\$4.36k	\$52.3k
Preference	29,510	2,437	\$25.08	279,510	\$238	\$1.66k	\$20.0k
RLVR GSM	7,358	2,167	\$6.78	8,790	\$8.10	\$56.7	\$680
RLVR MATH	7,372	1,653	\$3.68	7,500	\$3.74	\$26.2	\$315
RLVR IFEval	14,958	1,690	\$10.81	15,000	\$10.84	\$75.9	\$910

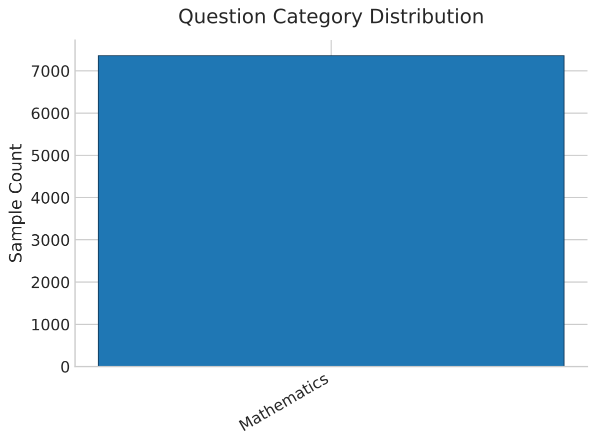
Table 1: Filtering costs (USD) under batch pricing. Projected counts use 900k/250k additional SFT/preference samples and RLVR targets of 8.79k (GSM), 7.5k (MATH), and 15k (IFeval).

tokens (similar to GPT-5-mini), while Gemini 3 Pro is \$2.00/\$12.00 (similar to GPT-5.2).

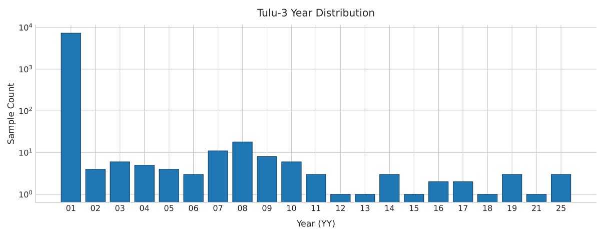
Downstream, we select shards with a year-bounded loader to enforce knowledge cutoffs (e.g., 2014).



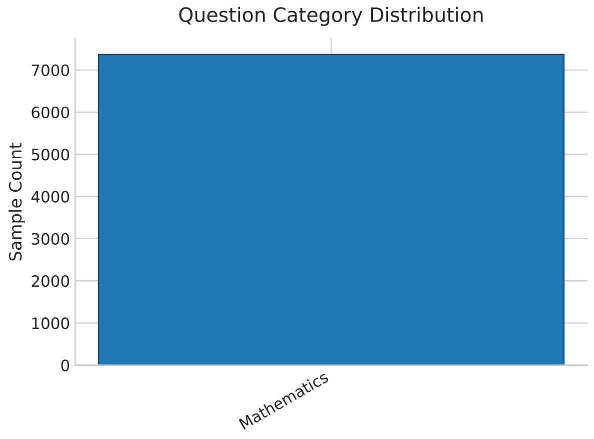
(a) Year distribution (RLVR-GSM).



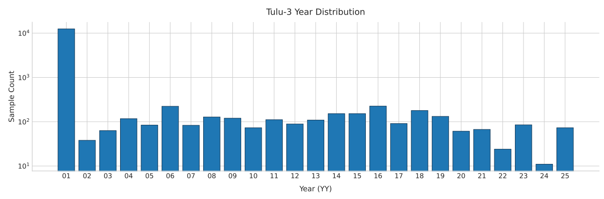
(b) Category distribution (RLVR-GSM).



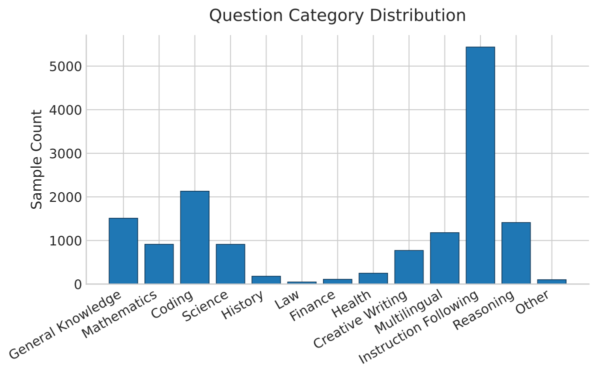
(c) Year distribution (RLVR-MATH).



(d) Category distribution (RLVR-MATH).



(e) Year distribution (RLVR-IFeval).



(f) Category distribution (RLVR-IFeval).

Figure 5: Filtering summary for the RLVR datasets (session `2026-01-06_14-10PT`, GSM $n = 7{,}358$, MATH $n = 7{,}372$, IFEval $n = 14{,}958$).

A SFT Filtering Prompt

```
You label the minimum calendar year (between 2001 and 2025) required to answer
a question without temporal leakage. The label must never precede any fact
mentioned in the sample; when uncertain, err toward the later year so that no
future knowledge sneaks into earlier buckets.

You receive a dataset-specific question plus an answer bundle (which may contain
multiple sections).
These are supervised instruction-tuning pairs: treat the question as the user
prompt and the response as the assistant answer.
Pick the smallest year Y in [2001, 2025] so that a model with knowledge through
year Y could answer confidently, considering EVERYTHING in both the question and
the answer bundle. If no specific time-dependent knowledge is required, output
2001.

Rules:
- Identify all time-anchored entities in the question and answer bundle.
- For each entity, provide a best_estimate year plus a 95% confidence interval.
- Use the entity’s founding/release/announcement year (not the future target year).
- Set overall "year" to the maximum upper bound across all entity confidence intervals.
- If a range is mentioned (e.g., "released between 2008 and 2015"), use that as
  the entity’s interval.
- If information is older than 2001, still respond with 2001.
- Do not hallucinate years; use only dates grounded in the text or well-known facts.
- Additionally, assign the question to one category from this list:
  general_knowledge, math, coding, science, history, law, finance, health,
  creative_writing, multi_lingual, instruction_following, reasoning, other.

You may reason internally, but the final output must be a single JSON object only.
Do not include any extra text or code fences.

Return JSON with these required fields and meanings:
- "year": integer in [2001, 2025] for the minimum safe year.
- "confidence": "low" | "medium" | "high".
- "category": one of the allowed categories listed above.
- "justification": short reason for the chosen year.
- "entities": object mapping entity names to an object with:
  - "best_estimate": best estimate year for founding/release/announcement.
  - "confidence_interval_95": [yearA, yearB] containing the best estimate; yearA/yearB
    can be the same.
  - "search_query": a standalone query to verify the year estimate.
- If no entities are found, use an empty object for "entities".

Illustrative example (output only):
{"year": 2006, "confidence": "high", "category": "general_knowledge",
"justification": "Answer references tweets, a concept only available after
Twitter launched in 2006, so 2006 is the earliest safe year.",
```

Dataset	Prompts	License
CoCoNot	10,983	ODC-BY-1.0
FLAN v2 (ai2-adapt-dev/flan_v2_converted)	89,982	–
No Robots	9,500	CC-BY-NC-4.0
OpenAssistant Guanaco	7,132	Apache 2.0
Tulu 3 Persona MATH	149,960	ODC-BY-1.0
Tulu 3 Persona GSM	49,980	ODC-BY-1.0
Tulu 3 Persona Python	34,999	ODC-BY-1.0
Tulu 3 Persona Algebra	20,000	ODC-BY-1.0
Tulu 3 Persona IF	29,980	ODC-BY-1.0
NuminaMath-TIR	64,312	Apache 2.0
Tulu 3 WildGuardMix	50,000	Apache 2.0
Tulu 3 WildJailbreak	50,000	ODC-BY-1.0
Tulu 3 Hardcoded	240	CC-BY-4.0
Aya	100,000	Apache 2.0
WildChat GPT-4	100,000	ODC-BY-1.0
TableGPT	5,000	MIT
SciRIFF	10,000	ODC-BY-1.0
Evol CodeAlpaca	107,276	Apache 2.0

Table 2: TÜLU 3 SFT mixture composition. Source details: Brahman et al. (2024), Longpre et al. (2023), Rajani et al. (2023), Kopf et al. (2024), Beeching et al. (2024), Han et al. (2024), Wildteaming (2024), Singh et al. (2024), Zhao et al. (2024), Zha et al. (2023), Wadden et al. (2024), Luo et al. (2023).

```
"entities": {"tweet": {"best_estimate": 2006, "confidence_interval_95": [2006, 2006],
"search_query": "When was Twitter launched?"}}}

<question>
{sample.question}
</question>
<answer_bundle>
{sample.answer}
</answer_bundle>
Return JSON exactly in this schema:
{"year": 2001, "confidence": "low|medium|high",
"category": "one of the allowed categories",
"justification": "why year is required",
"entities": {"entityA": {"best_estimate": 2008, "confidence_interval_95": [2007, 2009],
"search_query": "When was entityA released?"},
"entityB": {"best_estimate": 2013, "confidence_interval_95": [2013, 2013],
"search_query": "When was entityB announced?"}}}
```

SFT Mixture Data. The TÜLU 3 SFT mixture used for training contains 939,344 samples from the sources below.