Local Tab-Title Generation Under Product Constraints: An Eleven-Session Study of Data, Decoding, and Distillation

MD Ishtiaque Hossain
PorkiCoder Research
https://porkicoder.com/research/

August 2026

Abstract

Generating a useful title for a short interactive session is an extreme summarization problem: the output must identify intent in one to three words, remain grounded in sparse user text, satisfy a 32-character interface contract, and run locally inside a desktop application. We report an eleven-session longitudinal study covering 12.7M-, 24.5M-, 35M-, and 77M-parameter T5-family systems, extractive post-processing, source-constrained beam search, judged supervised fine-tuning, pointer heads, local selectors, and multi-teacher pseudo-labeling. On a sealed Stack Overflow proxy, changing only decoding improved an unchanged 35M model by 0.45–0.48 judge points relative to corrected greedy search and produced a 0.752-point paired advantage over a title-tuned FLAN baseline. However, the same system failed a pre-specified absolute-usefulness audit, and its proxy advantage did not transfer to real terminal tasks. On the terminal domain, a 24.5M encoder-plus-pointer system beat the 35M decoder by 0.13 points at 3.88 ms, while a domain-tuned 77M FLAN model gave the best quality among the measured candidates. Continuing that model on a leak-checked 6,219-example multi-teacher mixture improved two unseen 1,000-task holdouts by 0.21 and 0.20 points, reaching a 7.59 mean score and an 85.7% solid-title rate on the first holdout. The unchanged graph measured 29.3 ms per title in the preceding session. In contrast, 175,642 weak teacher labels and lower validation cross-entropy did not improve product quality. Across this campaign, decoding, target dialect, data curation, board separation, and checkpoint-specific composition mattered more than raw label count or training loss. We preserve negative results and distinguish within-packet evidence from cross-packet observations, offering a case study in product-constrained small-model research rather than a universal benchmark claim.

1 Introduction

An interactive coding session can contain thousands of tokens, yet the user interface may allocate only a few words to identify it. PorkiCoder, a local-first coding environment, therefore needs to map a sparse session description to a compact tab title that helps a user find the session later. A plausible output must do more than summarize. It must express the user's intent rather than merely repeat a frequent technology name; remain grounded enough to avoid inventing a stack, file, or action; fit a strict visual contract; and appear quickly enough that title generation does not become a perceptible product event.

This setting differs from conventional headline or section-title generation in several ways. First, the input can be extremely short, fragmentary, and operational: examples resemble "fix migration error," "add retry logic," or a terminal task serialized with agent and working-directory fields. Second, the title budget is one to three words and at most 32 characters, leaving little room for graceful paraphrase. Third, the target is not an existing reference title. The product criterion is whether the output will help a user recognize the tab later. Exact match is therefore neither necessary nor sufficient. Fourth, generation is local. Parameter count, packaged size, first-load cost, graph shape, and single-example latency are first-class experimental variables rather than deployment details added after model selection.

We report a longitudinal program that began with a failing 12.7M-parameter incumbent and proceeded through eleven experimental sessions. The public record consists of nine tab-title reports, with Sessions

5–8 consolidated in a techniques log, plus a separate matched-control study that motivated the campaign's claim discipline [Hossain, 2026a]. The sequence was adaptive rather than designed ex ante as a single benchmark. Each session addressed the failure exposed by the preceding one: a random-projection highlighter was combined with neural namers; a mid-training 35M checkpoint was rescued by source-constrained search; a proxy-domain win was challenged on real terminal tasks; curated judged labels were compared with much larger weak-label sets; a pointer head tested whether structured extraction could close the gap cheaply; and a domain-tuned 77M FLAN model was finally continued on a leak-checked multi-teacher mixture.

The campaign's strongest result is not a single score. It is the repeated observation that the variables easiest to dismiss as "plumbing"—decoding, label selection, board composition, post-processing compatibility, and leakage control—could dominate changes in parameter count or training loss. A four-beam decoder added no weights yet changed the ranking on a sealed 1,000-task proxy. Conversely, a training arm with 175,642 legal pseudo-labels and validation cross-entropy as low as 0.91 still lost to a curated model trained from a much smaller high-quality set. A post-processing rule that improved one checkpoint by filtering hallucinated words damaged a better domain-tuned checkpoint on an unseen confirmation set. These reversals are central evidence, not incidental failed experiments.

The work also illustrates why product benchmarks need multiple forms of validity. The source-constrained 35M system beat a title-tuned FLAN comparator on the sealed SO-board, but an independent absolute-usefulness audit estimated only 47.5% useful titles against a 60% gate. On the Terminal-board, FLAN was again clearly ahead. We therefore treat the Stack Overflow board as a public mechanism and development proxy, not as a substitute for product-domain evaluation. Similarly, a blinded LLM judge is used as a high-throughput product metric, but not claimed to be a validated replacement for human study. Candidate order is randomized and same-packet paired contrasts are emphasized because absolute scores can shift when the candidate set changes.

This paper makes four contributions:

- It defines local tab-title generation as a product-constrained short-form generation problem with explicit grounding, format, latency, and payload requirements.
- It consolidates eleven experimental sessions into a coherent account spanning heuristic extraction, compact T5 training, constrained decoding, judged SFT, pointer selection, local reranking, and multi-teacher continuation.
- It reports both successful and unsuccessful interventions, including a sealed decoding gain, a proxy-to-product reversal, a failed 176k-label scale test, and checkpoint-specific failure of a previously useful post-processor.
- It derives practical methodological lessons for small-model development: compare candidates within the same judge packet, keep domain boards separate, audit absolute usefulness independently, and optimize label dialect and selection before assuming that more labels or lower cross-entropy will improve the product metric.

The claim is deliberately narrow. Within the tested T5-family systems and the PorkiCoder tab-title task, quality was governed at least as much by inference and data-interface choices as by nominal scale. The experiments do not establish a general scaling law, a human-validated universal metric, or superiority over all alternative architectures.

2 Related Work

Text-to-text transfer and instruction tuning. The models in this study derive from the T5 formulation, which casts heterogeneous language tasks into a unified text-to-text interface [Raffel et al., 2020]. FLAN-style instruction tuning broadens task transfer by training on mixtures of prompted tasks [Chung et al., 2022]. Our results expose an important distinction between broad instruction-following ability and a narrow output dialect. Stock FLAN-T5-small performed poorly on the terminal ship board, whereas a title-tuned descendant of the same family became the strongest quality model after domain SFT. Thus, instruction tuning supplied useful capacity and initialization, but did not by itself define the desired one-to-three-word tab-title distribution.

Gap-sentence generation and title generation. The compact 35M model was pretrained on technical text and then trained with a PEGASUS-style gap-sentence-generation objective, inspired by extracting salient content for summarization [Zhang et al., 2020]. Prior section-title work has also shown that extractive selection followed by compression can produce useful short labels without a large encoder-decoder model [Gehrmann et al., 2019]. Our task is even shorter and has no canonical reference at evaluation time. This favors a hybrid view: the title can be treated partly as generation and partly as structured selection of an intent-bearing object and action.

Stack Overflow title generation. Prior software-engineering work has generated Stack Overflow question titles from code and natural-language bodies. CCBERT combines a CodeBERT encoder, Transformer decoder, and copy attention over a roughly 200,000-question corpus [Zhang et al., 2021]. FILLER adds self-improvement and post-ranking to a pretrained title generator and evaluates against both automatic metrics and a user study [Le et al., 2024]. Our use of Stack Overflow is different in purpose and output regime: it is a public proxy for naming private coding-agent sessions, the product title is limited to one to three words, and evaluation is reference-free because the original Stack Overflow title is not the desired tab label. Nevertheless, these studies motivate the use of copying, candidate ranking, and bi-modal or structured signals for technical title generation.

Constrained decoding and reranking. Lexically constrained decoding is commonly used to enforce required terms during sequence generation [Post and Vilar, 2018]. Our decoder uses a simpler task-specific constraint: enumerate safe complete one- or two-word candidates composed of visible source words, then use model sequence scores to rank them. The key experimental advantage is causal clarity: the model weights are unchanged, so gains can be attributed to search and candidate constraints rather than retraining. Centroid v2 also uses inverse-document-frequency weighting and a diversity term related to maximal marginal relevance [Carbonell and Goldstein, 1998].

Knowledge distillation and pseudo-labeling. Sequence-level knowledge distillation trains a smaller model on teacher-generated sequences that may be easier to model than the original target distribution [Kim and Rush, 2016]. This campaign tested several variants: copying high-scoring FLAN outputs, self-distilling best-of-K candidates, training on six-teacher judged outputs, scaling to 175,642 Flash-Lite labels, distilling the final 77M teacher into 35M, and imposing a structured schema. The mixed outcomes show that teacher identity, selection rule, and target dialect can matter more than pseudo-label count.

On-device language models. Recent work on sub-billion-parameter language models emphasizes architecture and deployment-aware optimization for local use [Liu et al., 2024]. Our parameter range is smaller, but the product constraints are analogous: a model that is marginally better but introduces a large payload or visible latency can be inferior to a structured selector on the deployment Pareto frontier. We therefore report latency and approximate bundle size alongside quality whenever the measurements are comparable.

LLM-as-a-judge evaluation. LLM judges enable scalable reference-free comparison and can agree substantially with human preferences on some benchmarks [Zheng et al., 2023]. They also exhibit position effects, benchmark-dependent rankings, and the broader distinction between reliability and validity [Norman et al., 2026]. Our protocol responds by blinding candidate identities, randomizing order, emphasizing paired same-packet deltas, and maintaining an independent absolute-usefulness audit. These controls reduce obvious artifacts, but they do not convert the judge into a human-validated ground truth.

3 Task and Experimental Protocol

3.1 Product task and title contract

Let x be a serialized session description containing fields such as agent, working directory, and task text. The output is a title $y = (w_1, \ldots, w_k)$ subject to the product contract

$$1 \le k \le 3, \qquad |y|_{\text{chars}} \le 32, \tag{1}$$

with ASCII alphanumeric tokens and spaces, Title Case formatting, no duplicate words, and a preference for two words. The semantic goal is to maximize retrospective tab recognizability rather than reproduce a reference string. Later terminal experiments further preferred an intent-object structure: an optional action or intent word, an on-page object, and an optional qualifier.

The product objective can be written schematically as

$$U(y, x) = Q(y, x) - \lambda_h H(y, x) - \lambda_f F(y) - \lambda_l L(y), \tag{2}$$

where Q is usefulness for rediscovery, H penalizes unsupported content, F penalizes format-contract violations, and L represents deployment cost. The experiments do not estimate these coefficients directly. Instead, they use a blinded usefulness judge, deterministic contract checks, and separate latency and size measurements.

Grounding is a soft product preference for the strongest models and a hard constraint for some compact systems. "On-page" means a content word can be mapped back to visible source text under the locked normalization and substring rules. Free generation can express intent more naturally, but risks hallucinated technologies or generic brand priors. Source-constrained systems sacrifice some abstraction to guarantee visible-word support.

3.2 Evaluation boards and leakage boundaries

The campaign uses two explicitly separate leaderboards.

Stack Overflow board. The SO-boardis a public proxy built from Stack Overflow task bodies. The initial clean holdout contained 2,000 newly pulled tasks from source shards not used for technical-language pretraining. Previously used title-training and validation identifiers were excluded, and the remaining tasks were split by hash sort into a reusable 1,000-task development set and a sealed 1,000-task set [Hossain, 2026c]. The board is useful for testing mechanisms because inputs are public and diverse, but its natural-language style and original-question distribution differ from PorkiCoder terminal sessions.

Terminal board. The Terminal-boardcontains real or production-shaped coding-agent tasks serialized in the app format. Task text is withheld because it can contain private paths, identifiers, or user context. Different sessions used named subsets such as terminal-dev ($n = 300$), sealed Terminal-v2 ($n = 516$), shipboard ($n = 500$), confirm-v1 ($n = 335$), and two unseen 1,000-task holdouts. These sets are not pooled after the fact. Session 11 additionally removed 1,148 source-hash overlaps from its continuation mixture before constructing a 6,219-row train set and 200-row validation set [Hossain, 2026e].

Absolute means are not compared across SO-boardand Terminal-board, nor across different blinded candidate packets on the same board. A judge score is partly a property of the task and candidate set presented in that call. The primary evidence is therefore a paired difference between candidates scored together on the same tasks.

3.3 Model and system families

Table 1 summarizes the main families. Parameter counts refer to live neural parameters when reported; size estimates include different packaging assumptions and should not be treated as a single hardware-normalized benchmark.

The 35M compact model has $d_{\text{model}} = 512$, $d_{\text{ff}} = 2048$, six encoder layers, three decoder layers, tied embeddings, and a 6,985-token vocabulary. It was pretrained on technical span corruption and continued on 4,863,657 body-to-Stack-Overflow-title pairs with a PEGASUS-style objective. The B-9500 checkpoint became the stable base for later decoding and fine-tuning experiments [Hossain, 2026f].

The FLAN comparator in early sessions was not untouched `google/flan-t5-small`; it was a title-tuned checkpoint inherited from an earlier distillation campaign. Session 10 separately included stock FLAN raw, making the distinction observable. The final Title-SFT FLAN model has 76,961,152 parameters and an untied graph. Session 11 changed its weights but not its architecture or inference graph.

Table 1: Main system families. Latencies are the best directly reported product-path measurements for the indicated system, not a cross-session hardware benchmark.

System family	Params	Reported latency	Core mechanism	Role in the campaign
Historical student	12.7M	226 ms in Session 1 research path	Small encoder-decoder distilled in an earlier campaign	Incumbent failure baseline; prone to repeated brand priors
Centroid highlighter	none beyond embedding table	0.8 ms	Random or trained subword vectors, document centroid, IDF/MMR in v2	Emergency extractive path and grounding component
Compact GSG namer	35.05M	10.1 ms for `6t_argmax` product graph	6-layer encoder, 3-layer decoder, 6,985-token vocabulary	Main compact generative line through Sessions 1–8
Dense P1 pointer	24.5M live	3.88 ms	Frozen compact encoder plus an 8 MB structured pointer head	Fast intent/object selector in Session 9–10
FLAN-T5-small line	76.96–77.0M	29.3 ms for Title-SFT graph	Instruction-tuned T5 followed by title/domain SFT	Best measured quality line in Sessions 10–11

3.4 Grounding, decoding, and composition

Centroid highlighter. For each content word u_i in the task, the system averages its subword vectors to obtain e_i and normalizes the result. The first centroid used the seed-42 random initialization of the compact model before training. The document vector was the normalized mean

$$c = \frac{\sum_i e_i}{\left\| \sum_i e_i \right\|_2}, \tag{3}$$

and words were ranked by $e_i^\top c$. This untrained table unexpectedly outperformed the historical neural incumbent. A direct cosine-training objective collapsed, so the random table survived as a filter rather than a learned semantic encoder [Hossain, 2026c]. Centroid v2 replaced the random table with a trained embedding table, added inverse-document-frequency weighting, and used an MMR-like diversity term to avoid redundant selections.

Hybrid A. Terminology changed during the campaign. In the final convention, Hybrid A refers only to a deterministic "2+1" glue rule: retain up to two distinct content words from the neural title that map to the source, then fill remaining slots from the centroid ranking. Earlier Session 1 prose used the same name for the whole compact stack. We use only the final convention in this paper.

Source-constrained beam search. The Session 3 decoder enumerates complete safe one- or two-word candidates whose words occur on the source page, maps them through the compact vocabulary, and ranks the resulting complete sequences with the unchanged B-9500 checkpoint. The important contrast is not ordinary beam search versus greedy token generation. It is complete-sequence scoring over a task-specific constrained candidate set versus a corrected greedy baseline. Beam widths 2, 4, and 8 were tested; beam 4 offered the best quality-cost balance.

Structured pointer selection. Dense P1 freezes the compact encoder and trains a pointer head to choose an optional intent verb from a small lexicon, an on-page object, and an optional qualifier. This replaces free autoregressive generation with structured selection. The head is small enough to train in under a minute

on the reported accelerator, and its product graph is materially faster than either compact autoregressive generation or FLAN.

3.5 Blind judging and statistical comparisons

The main product metric is a blinded Gemini 3.5 Flash-Lite score from 1 to 10. For each task, the judge sees the task and anonymous candidate titles in randomized order, with reference labels hidden. The rubric asks whether the title would help the user find the tab again. Early SO-boardreports defined "useful" as score ≥ 4 and "fail" as score ≤ 2. Later terminal reports use a stricter "solid-title" threshold of score ≥ 6; some also report score ≥ 7. These rates are not interchangeable.

For a pair of systems a and b scored on the same n tasks, the primary statistic is

$$\widehat{\Delta}_{a,b} = \frac{1}{n} \sum_{i=1}^{n} (s_{i,a} - s_{i,b}) \,. \tag{4}$$

Where published, 95% percentile bootstrap intervals resample tasks with replacement [Efron and Tibshirani, 1994]. Win/loss/tie counts are also reported because a mean can conceal broad low-magnitude gains or a small number of large corrections.

The protocol has two safeguards and one major limitation. Blinding and randomized candidate order reduce identity and position artifacts. Same-packet comparisons reduce score drift from candidate composition. However, the judge remains a model-based proxy. Session 3 therefore used an independent reference-free absolute-usefulness audit on a sealed 40-task subset with a pre-specified 60% gate, followed by a larger 200-task development audit. The audit and product judge disagreed enough to prevent a strong absolute-quality claim.

3.6 Longitudinal, adaptive design

The eleven sessions are a sequential product research program, not eleven independent confirmatory trials. Decisions were adaptive: failures selected the next intervention, and some hyperparameters were explored on reusable development packets. We preserve this chronology to avoid presenting exploratory choices as preregistered. The strongest claims come from sealed or unseen sets, matched candidate packets, explicit leakage checks, and replicated direction across two holdouts.

This caution was informed by an immediately preceding study of the 12.7M incumbent. A fixed model pair suggested that a literary embedding initialization reduced repeated titles, but a deterministic 31-run matched control found no stable advantage for literary sequence order: ordered Verne averaged 14.75% repeated titles versus 13.79% for frequency-matched shuffled controls [Hossain, 2026j]. That seed-lottery result established the campaign's preference for matched controls, negative-result publication, and narrow mechanism claims.

4 Eleven-Session Experimental Program

Figure 1 presents the trajectory. Sessions 1–3 used the public SO-boardto establish baselines and isolate decoding. Sessions 4–8 moved to the Terminal-boardand explored judged SFT, objectives, post-processing, and label scale. Session 9 introduced a structured pointer model. Sessions 10–11 compared the quality-latency frontier and finalized the domain-tuned FLAN line.

4.1 Sessions 1–3: heuristic grounding, compact generation, and search

4.1.1 Session 1: two device recipes and a surprising random highlighter

Session 1 compared seven systems on the same blinded 1,000-task SO-board packet [Hossain, 2026c]. Table 2 reproduces the central results. The historical 12.7M model scored 3.01 with 30.5% useful titles and a 53.9% fail rate. The 16 MiB centroid-only path scored 4.21 at 0.8 ms despite using an untrained random embedding table. The early 35M-plus-centroid stack reached 4.72 and 64.0% useful. The title-tuned FLAN model scored 5.22 raw and 5.43 after the same grounding glue, with 77.4% useful and 9.9% fail.

The result established three themes that persisted. First, the incumbent's problem was not merely small size; its outputs collapsed toward frequent technology priors. Second, grounding reduced catastrophic failures

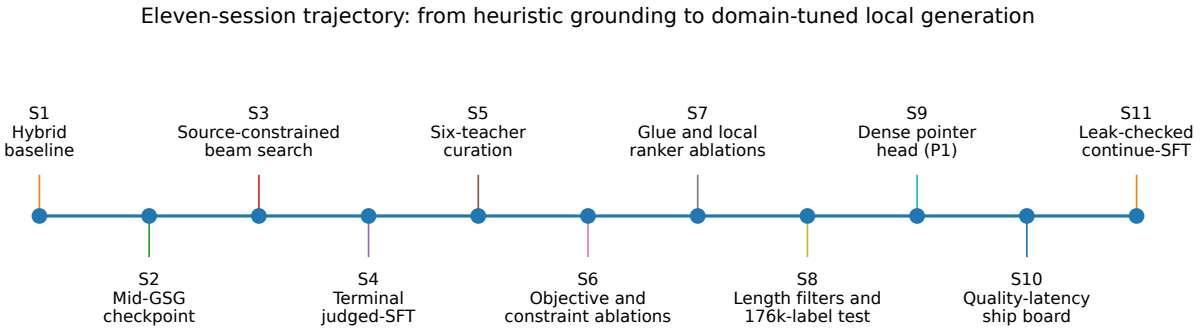

Figure 1: The eleven sessions form an adaptive sequence. Sessions 5–8 are documented together in a public techniques log; the separate Sniff Test predates this sequence and informs its methodology.

Table 2: Session 1 SO-board results on one blinded $n = 1{,}000$ packet. "Useful" is score ≥ 4 and "fail" is score ≤ 2. Latency used the Session 1 research path and is not comparable to later ONNX measurements.

System	Size	ms	Mean	Useful	Fail
Historical 12.7M	28.5 MiB	226	3.01	30.5%	53.9%
Centroid only	16 MiB	0.8	4.21	55.1%	22.6%
Early 35M + centroid	∼150 MiB slim est.	252	4.72	64.0%	16.6%
12.7M + 35M + centroid	300 MiB	477	4.70	64.3%	18.5%
Title-tuned FLAN raw	296 MiB	31	5.22	67.7%	18.3%
Title-tuned FLAN + centroid	312 MiB	32	5.43	77.4%	9.9%

even when it slightly reduced the rate of very high scores. Third, a simple extractor could dominate a learned model on latency and robustness. However, the initial compact model often learned the syntax of Stack Overflow titles without identifying the task's gist, producing fragments such as "How to a . . . " rather than product-ready labels.

4.1.2 Session 2: checkpoint selection closes most of the gap

The same 35M architecture was trained from scratch on 4,863,657 body-to-title pairs for 12,000 steps with batch size 256. A mid-training checkpoint, B-9500, was substantially better than the earlier stopping point [Hossain, 2026f]. Under the older centroid packet, B-9500 plus glue scored 5.65 with 78.6% useful titles, slightly above raw title-tuned FLAN at 5.64 but below FLAN plus centroid at 5.81. When centroid v2 changed the candidate packet, FLAN plus v2 scored 6.11 and B-9500 scored 5.96, a gap of 0.15.

This session supplied an early warning about evaluation packets: even when task rows are unchanged, adding or changing candidates can shift absolute judge means. The robust statement is that B-9500 improved over the older compact checkpoint and came close to the title-tuned FLAN stack within the same packet. It did not establish a stable cross-packet absolute ranking.

4.1.3 Session 3: four beams change the ranking without changing weights

Session 3 held B-9500 fixed and changed only decoding [Hossain, 2026g]. On three clean-development comparisons, source-constrained beam 4 improved the compact system over the title-tuned FLAN comparator by 0.644, 0.613, and 0.601 points under slightly different packet and bounded-decoding conditions. On the sealed 1,000-task test, the compact beam-4 stack scored 6.20 versus 5.45 for FLAN, a paired difference of 0.752 with a 95% interval of $[0.611, 0.891]$. Against a corrected compact greedy baseline, beam 4 improved by

0.481 and 0.453 points in two clean-development packets. Beam 2 helped; beam 8 tied beam 4 while costing more. The decoder added 8.6–9.0% latency on the reported accelerator and no model weights.

The contract audit was strong: every generated title contained exactly two distinct visible mapped words, the maximum length was 31 or 32 characters depending on the packet, and no contract violations were found. Yet the absolute-usefulness audit failed. Only 19 of 40 titles, or 47.5%, passed the sealed gate against a pre-specified 60% requirement. A subsequent 200-task development audit estimated 40.0% usefulness with a Wilson interval of 33.5–46.9%. The system therefore earned a narrow claim—constrained search improved the model and beat the comparator on the proxy packet—but not the claim that it had reached sufficient product usefulness.

A follow-up training experiment sharpened the distinction between token prediction and sequence ranking. Continuing B-9500 on cleaned short targets reduced cross-entropy and improved greedy copying, but damaged beam-ranked title quality by 0.534–0.647 points. Even 500 steps at a learning rate of 5×10^{-6} lost 0.601 points. The likely interpretation is not that cross-entropy is intrinsically harmful, but that the continuation changed the score geometry over complete candidate pairs in a way that the constrained decoder relied on.

4.2 Sessions 4–8: domain shift, judged SFT, and negative ablations

4.2.1 Session 4: the proxy winner does not transfer

The first direct domain comparison revealed a large shift [Hossain, 2026h]. On the 1,000-task SO-board packet, FLAN plus centroid v2 scored 6.11 and the compact pin scored 5.96, a gap of 0.15. On a 300-task terminal packet, FLAN plus v2 scored 7.11 and the compact pin scored 5.94, a gap of 1.17. The source-constrained compact system's relative proxy strength did not imply product-domain parity.

Session 4 then trained an r3 compact checkpoint on 971 FLAN-generated titles that were both spellable from the page and judged at least 6. In a terminal confirmation packet, r3 scored 5.67 versus 5.56 for the pin, a paired gain of 0.12 with interval $[0.04, 0.20]$; an earlier packet showed a smaller gain of 0.07 with interval $[0.01, 0.14]$. Self-distillation from the compact model's own best-of-8 and best-of-24 candidates was flat. The successful labels were not merely high-scoring strings: they reflected a teacher with a more consistent terminal-title dialect.

The gain was domain-specific. On the reusable 200-task SO-board audit, r3 regressed by 0.17 points with interval $[-0.32, -0.02]$. Session 4 also exposed a scoring bug in which the wrong centroid implementation omitted the third word; correcting it restored the pin to its expected baseline. Both observations reinforced the need to separate boards and freeze the inference composition before interpreting training changes.

4.2.2 Session 5: curated multi-teacher SFT

Session 5 trained `6t_argmax` from six teacher sources with judged selection [Hossain, 2026k]. On a 300-task terminal packet, it scored 5.80 versus 5.68 for the pin, a gain of 0.12 with interval $[0.04, 0.20]$. On a larger 1,000-task packet, the gain increased to 0.19: 5.99 versus 5.80, interval $[0.14, 0.25]$. A four-teacher variant was weaker at 5.77, suggesting that source diversity or the additional selected rows helped, but the study was not factorial enough to isolate which teachers were responsible.

Length-based training filters did not improve the model. Training only on tasks with at least ten words tied `6t_argmax` at 6.00, with a paired difference of 0.005 and interval $[-0.017, 0.028]$. A five-word filter also tied. The operational conclusion was not to retrain around a length cutoff, but to keep the previous title at runtime for prompts shorter than ten words, where all systems were noisier.

4.2.3 Sessions 6–7: objectives, freedom, glue, and local selectors

Sessions 6 and 7 tested several plausible interventions that were flat or negative [Hossain, 2026k]. A contrastive hinge objective gained only 0.04 over `6t_argmax` in the reported packet. Allowing free off-page generation lost 0.89 points, confirming that hallucination control remained essential for the compact model. Replacing the global centroid with terminal-specific IDF features lost 0.19. A local neural ranker lost 0.13, and a local proposer also failed to improve the ship candidate. Self-distillation remained approximately zero.

These results are informative because they reject a simple narrative in which every added learned component improves the system. The compact line benefited when model scores were used to rank a carefully

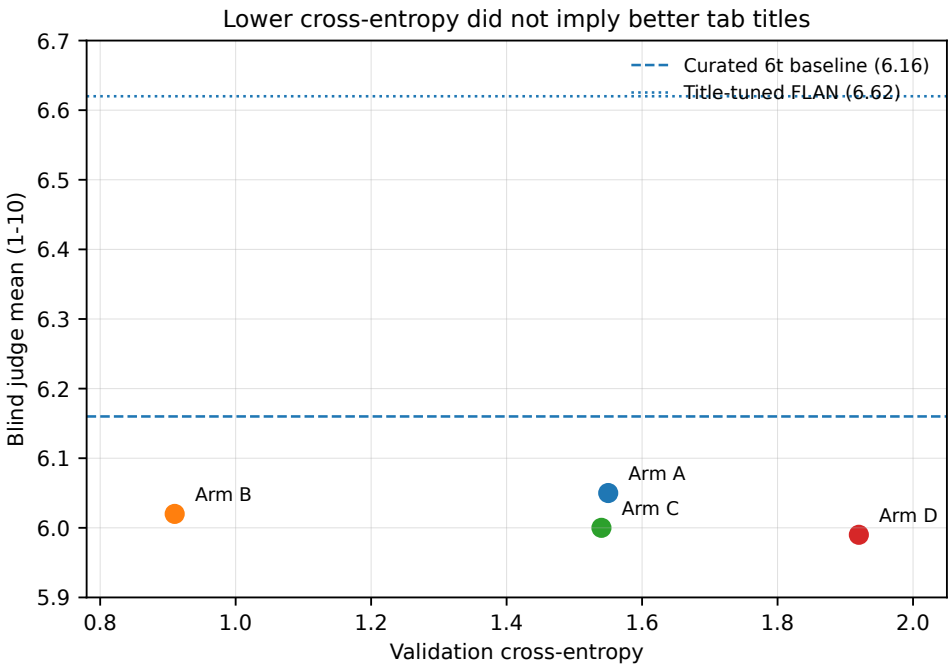

Figure 2: Session 8 scale test on one 300-task terminal packet. All four 175,642-label continuation arms remained below the curated `6t_argmax` baseline and the title-tuned FLAN comparator. The plot is descriptive; with four adaptively chosen arms it is not a statistical estimate of the general relation between cross-entropy and usefulness.

bounded candidate set. It did not benefit from relaxing the grounding constraint or from stacking weak local selectors whose errors were correlated with the generator's.

4.2.4 Session 8: 175,642 weak labels do not beat curated data

The scale test began with 219,800 Flash-Lite pseudo-labels, of which 175,642 were legal under the compact vocabulary and title contract [Hossain, 2026b]. Four 35M continuation arms were trained for 8,000 steps at different learning rates. The legal label count was roughly 86 times the 2,040 reward-at-least-6 rows used by the curated `6t_argmax` recipe, but the labels were much less selective.

On packet 2, title-tuned FLAN scored 6.62 and `6t_argmax` scored 6.16. The four scale arms scored 6.05, 6.02, 6.00, and 5.99. Arm A's paired difference from `6t_argmax` was -0.112. Arm B achieved the lowest reported validation cross-entropy, 0.91, yet scored only 6.02. Figure 2 shows the absence of a useful monotonic relation between validation cross-entropy and the product judge across these four arms.

The result does not show that large pseudo-labeled datasets are generally ineffective. It shows that these weak, broadly accepted labels did not transfer the teacher's product "taste." The selection function was part of the target distribution. More rows optimized token prediction without reproducing the short, intent-centered title dialect rewarded by the judge.

4.3 Session 9: structured selection reaches a useful Pareto point

Dense P1 reframed the compact problem as structured selection rather than autoregressive title generation [Hossain, 2026i]. It froze the `6t_argmax` encoder and added an approximately 8 MB pointer head over a schema containing an optional intent verb, an on-page object, and an optional qualifier. The head trained for three epochs in 59 seconds on one reported accelerator; label generation cost was approximately US$5.26.

On sealed Terminal-v2 ($n = 516$), P1 plus Hybrid A scored 6.43 with 68.2% of titles at least 6. The `6t_argmax` stack scored 6.30 and 65.3%, giving P1 a paired gain of 0.13 with interval $[0.04, 0.22]$. Title-tuned

Table 3: Session 10 ship-board results on one blinded $n = 500$ packet. The same-packet table supports comparisons within rows, not across earlier sessions.

System	Mean	Solid ≥ 6	Median ms
Title-SFT FLAN + Hybrid A	7.41	89.2%	29.3
Title-SFT FLAN keep rule	7.35	–	29.3
Title-SFT FLAN raw	7.28	87.8%	29.3
P1 $\cup$ `6t_argmax` oracle	7.04	–	–
Neural ranker	6.83	80.6%	17.5
CPU picker	6.81	–	11.72
P1 + Hybrid A	6.79	–	3.88
`6t_argmax` + Hybrid A	6.65	–	10.1
One-minute agent selector	6.50	–	–
`6t_argmax` free decode	6.39	–	–
Stock FLAN raw	3.12	15.8%	25.7

FLAN scored 7.02 and 80.2%, leaving P1 0.60 points behind with interval $[-0.70, -0.49]$. P1 used 24.5M live parameters versus 35M for `6t_argmax` and 77M for FLAN. In the later product graph it ran in 3.88 ms.

This result isolates the remaining gap. The pointer system improved object selection and latency, but its bounded verb inventory and extractive schema could not fully express the terminal-title dialect. It became a credible low-cost fallback rather than the best-quality system.

4.4 Session 10: every namer versus the clock

Session 10 placed the leading systems in one 500-task blinded ship-board packet and measured product-path latency [Hossain, 2026d]. Table 3 and Figure 3 summarize the result. Raw Title-SFT FLAN scored 7.28 with 87.8% solid titles at 29.3 ms. Adding Hybrid A raised the same-packet mean to 7.41, but this composition later failed confirmation. P1 plus Hybrid A scored 6.79 at 3.88 ms; the CPU picker and neural ranker reached 6.81 and 6.83 at 11.72 and 17.5 ms. Stock FLAN raw scored only 3.12 despite similar architecture and 25.7 ms latency.

The same-packet paired differences were large. Raw Title-SFT FLAN beat stock FLAN by 4.17 points with interval $[3.95, 4.38]$, P1 by 0.50 with interval $[0.36, 0.64]$, and `6t_argmax` by 0.64 with interval $[0.50, 0.78]$. P1 beat `6t_argmax` by 0.14 with interval $[0.05, 0.23]$. The contrast between stock FLAN and Title-SFT FLAN is particularly important: broad instruction tuning supplied a capable base, but domain title training accounted for most of the product usefulness.

The confirmation packet reversed the post-processing conclusion. On unseen confirm-v1 ($n = 335$), raw Title-SFT FLAN scored 6.55 with 71.3% solid titles. P1 and the CPU picker both scored 6.41; the neural ranker scored 6.40; and `6t_argmax` scored 6.24. Applying Hybrid A to Title-SFT FLAN reduced it to 5.78 and 51.9% solid. The rule had been helpful when a model produced unsupported or malformed tokens, but it destroyed valid abstractions from the better domain-tuned checkpoint. Raw Title-SFT FLAN therefore became the Session 10 ship candidate.

4.5 Session 11: leak-checked continuation produces the final ship

Session 11 continued the Session 10 77M untied FLAN checkpoint for four epochs at a flat learning rate of 10^{-4} and seed 1 [Hossain, 2026e]. The historical mixture contained 7,623 rows: 1,530 gold repetitions and 6,093 pseudo-labels from Grok 4.5, Claude Opus 5, and Codex. After dropping 1,148 source-hash overlaps with protected evaluation material, 6,419 rows remained; 6,219 were used for training and 200 for validation. The continuation ran for 1,556 steps with batch size 16. No Hybrid A post-processing was applied.

On unseen holdout1000, `flat1e4` scored 7.59 with 85.7% at least 6 and 80.1% at least 7, compared with 7.38, 83.5%, and 74.6% for the Session 10 ship. The mean gain was 0.21, with 296 wins, 551 ties, and 153 losses. On unseen holdout1000b, `flat1e4` scored 7.55 with 84.5% at least 6 and 78.4% at least 7, compared

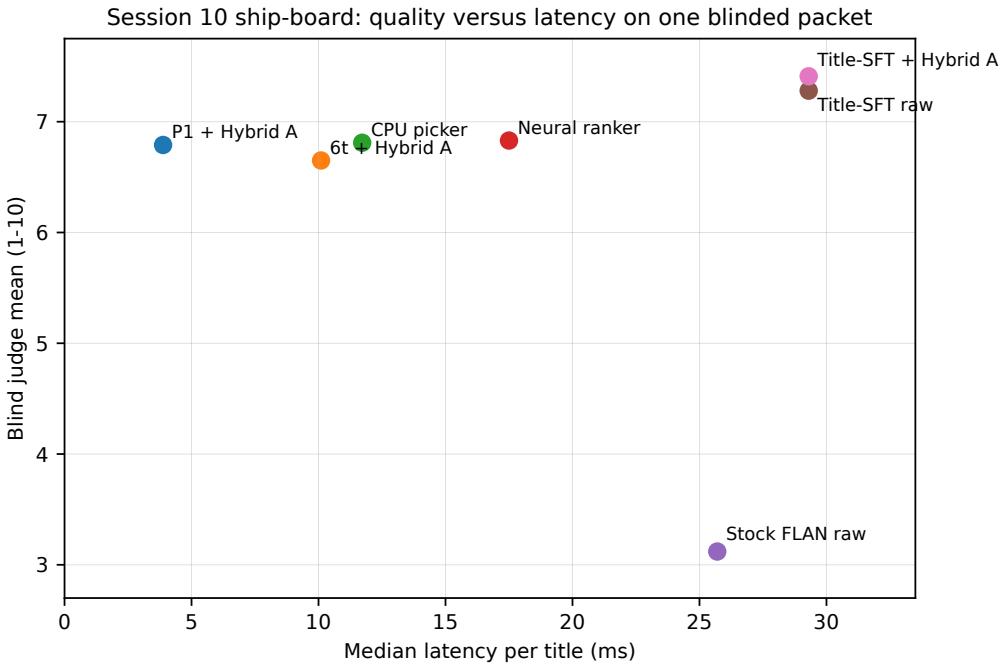

Figure 3: Session 10 quality-latency plane. All points come from the same blinded ship-board quality packet; latency is the reported product-path median. The Title-SFT FLAN plus Hybrid A point is shown because it won this packet, but raw Title-SFT FLAN became the ship after confirmation.

with 7.34, 82.5%, and 74.4%. The mean gain was 0.20, with 281 wins, 570 ties, and 149 losses. Figure 4 displays the replicated mean improvement.

The graph was unchanged from Session 10, so the paper carries forward the prior 29.3 ms latency and approximately 193 MB int8 bundle estimate rather than claiming a fresh clock. Two smaller alternatives failed. A 35M cross-entropy distillation learned fluent English but remained 1.82 points behind its teacher. Continuing the structured Session 9 schema produced the wrong output dialect and lost, with one reported comparison at 6.70 versus 7.37. The recommended next training move was therefore new high-quality rows, not another learning-rate or seed sweep.

4.6 Major same-packet comparisons

Table 4 collects the most informative paired contrasts. It intentionally omits cross-packet rankings. A positive difference favors the first system.

5 Cross-Session Findings

5.1 Inference search was a first-order intervention

The cleanest causal intervention in the campaign was Session 3. The model checkpoint, vocabulary, and post-processing were held fixed while the decoder changed. The resulting 0.45–0.48 point gains over corrected greedy search and 0.752 point sealed advantage over the title-tuned FLAN packet were comparable to, or larger than, many training interventions. This matters because small-model studies often treat decoding as a fixed implementation detail and attribute all quality differences to weights.

The gain was task-specific. Exhaustive safe search was feasible because the output contract was tiny and candidates could be constructed from visible source words. The result should not be generalized to unrestricted generation. It does show that when output length and lexical support are tightly bounded, evaluating complete sequences can reveal competence hidden by locally greedy token decisions.

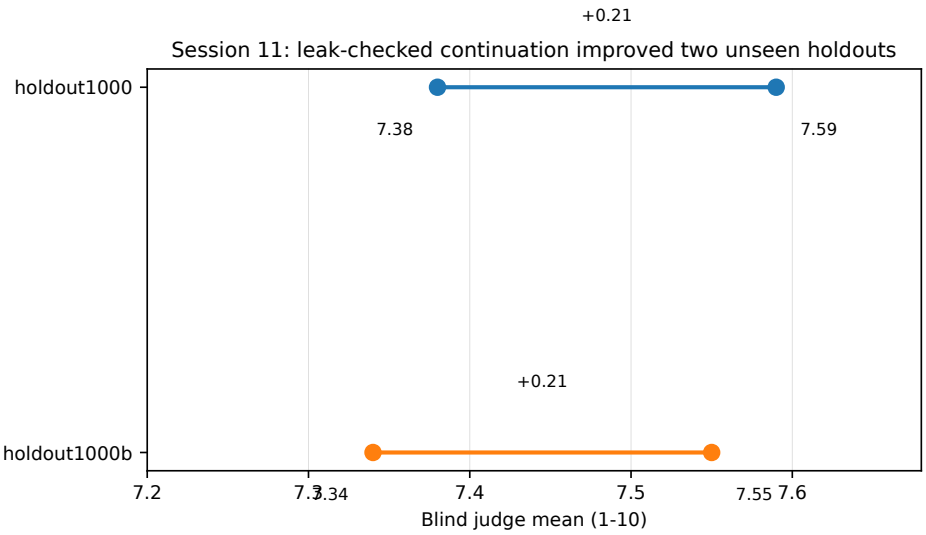

Figure 4: Session 11 same-packet mean scores on two unseen 1,000-task holdouts. Confidence intervals were not published for these contrasts; the replication across independently held-out task sets is therefore more informative than either point estimate alone.

The negative continuation experiments strengthen the point. Lower cross-entropy and improved greedy copying did not preserve the model's ranking over complete safe pairs. Thus, a model can become a better next-token predictor under one target distribution while becoming a worse scorer for a downstream constrained decoder. Training and inference must be evaluated as a composition.

5.2 The target dialect was a property of the data pipeline

Several experiments separated fluency from product usefulness. The original 35M GSG model learned the recognizable syntax of Stack Overflow titles without reliably naming the task. Stock FLAN could follow instructions broadly yet scored 3.12 on the terminal ship board. The 35M final distillation learned English but remained 1.82 points behind the teacher. The structured schema generated legal titles but lost because its action vocabulary and composition represented the wrong dialect.

Conversely, comparatively small curated sets moved the product metric. Session 4 used only 971 high-scoring, source-spellable FLAN titles and improved the compact terminal model. Session 5's judged six-teacher mixture produced a replicated gain. Session 11 improved two 1,000-task holdouts with 6,219 leak-checked continuation examples. These datasets did not merely supply more text; they defined which abstractions, action words, and levels of specificity counted as a good tab title.

This explains why the 175,642-label experiment failed. The labels were legal and abundant, and one arm achieved very low validation cross-entropy. But broad acceptance admitted titles that were easy to imitate without concentrating the intended product preference. Within this campaign, label selection was not a cleanup step after data collection. It was part of the task specification.

5.3 Domain boards could not be treated as interchangeable

The most dramatic ranking reversal occurred between the SO-board and Terminal-board. Source-constrained beam search made the compact model look stronger than title-tuned FLAN on the sealed public proxy, yet FLAN led by 1.17 points on the first terminal comparison and by 0.60 over the later P1 system on sealed Terminal-v2. The compact model had learned an extractive strategy well matched to verbose public questions. Terminal tasks were shorter, more imperative, and more dependent on intent verbs and product-specific naming conventions.

The correct response was not to discard the public board. It remained valuable for mechanism isolation,

Table 4: Major within-packet comparisons. A dash means no interval was published in the session report.

Session	Comparison and board	n	Paired mean difference	Claim supported
3	Beam-4 compact vs title-tuned FLAN, sealed SO-board	1,000	$+0.752$ $[0.611, 0.891]$	Search changed the proxy ranking without new weights
4	r3 vs compact pin, terminal confirmation	300	$+0.12$ $[0.04, 0.20]$	Selective FLAN labels improved the compact terminal model
5	`6t_argmax` vs pin, terminal packet	1,000	$+0.19$ $[0.14, 0.25]$	Curated multi-teacher SFT improved over the compact pin
8	176k Arm A vs `6t_argmax`, terminal packet 2	300	-0.112	Much larger weak-label CE training did not beat curated SFT
9	P1 + Hybrid A vs `6t_argmax` + Hybrid A, sealed Terminal-v2	516	$+0.13$ $[0.04, 0.22]$	Structured pointer selection improved the compact frontier
10	Raw Title-SFT FLAN vs `6t_argmax`, ship-board	500	$+0.64$ $[0.50, 0.78]$	Domain-tuned FLAN provided a large quality gain
11	`flat1e4` vs Session 10 ship, holdout1000	1,000	$+0.21$	Leak-checked continuation improved an unseen holdout
11	`flat1e4` vs Session 10 ship, holdout1000b	1,000	$+0.20$	The direction replicated on a second unseen holdout

contract auditing, and reproducible evidence. The response was to prevent proxy wins from being promoted into product claims. The public board answers "does this intervention improve this model under a controlled, shareable distribution?" The terminal board answers "does the candidate help on the application distribution?" Both are necessary, but they answer different questions.

5.4 Post-processing was checkpoint-specific, not universally beneficial

The centroid and Hybrid A glue initially improved robustness by replacing unsupported or malformed neural words with visible content words. On the Session 1 packet, FLAN plus centroid increased the useful rate from 67.7% to 77.4% and cut failures from 18.3% to 9.9%. Similar grounding helped the compact line.

The same rule damaged the final domain-tuned FLAN checkpoint on confirm-v1, reducing mean score from 6.55 to 5.78 and solid-title rate from 71.3% to 51.9%. The better model was using off-page abstractions productively; a filter designed for weaker checkpoints removed them. This is a general systems lesson: deterministic safety or cleanup layers encode assumptions about upstream error modes. They must be revalidated whenever the upstream distribution changes.

5.5 Structured small systems formed a real deployment frontier

The final quality winner was the 77M Title-SFT FLAN line, but the compact systems were not simply obsolete. P1 reached 6.79 on the Session 10 ship board at 3.88 ms, while the raw title-SFT model reached 7.28 at 29.3 ms. The CPU picker and neural ranker occupied intermediate latency points. For devices or contexts where 25 ms and a larger bundle matter, P1 is a defensible alternative.

The results also show why parameter count alone is a weak deployment proxy. The historical 12.7M research path took 226 ms, while the 77M FLAN product path took 29.3 ms. Graph implementation, autoregressive steps, runtime backend, tokenizer, page-in behavior, and packaging matter. A model can be smaller in weights but slower in the application.

5.6 Automated agreement did not establish absolute validity

The blinded judge produced stable enough paired rankings to guide rapid iteration, and several gains replicated across larger or unseen packets. Yet the Session 3 audit showed that a large relative judge win could coexist with inadequate absolute usefulness. The title generator satisfied every lexical contract while fewer than half of audited titles met the independent usefulness criterion.

This discordance supports a layered evaluation protocol:

1. deterministic checks for length, character set, duplication, and grounding;
2. blinded same-packet model judging for high-throughput paired iteration;
3. sealed or unseen task sets for candidate selection;
4. periodic independent absolute-usefulness audits; and
5. product telemetry or human user studies before strong claims about real utility.

The campaign reached the first four layers unevenly and did not conduct a controlled end-user study. Accordingly, the final result is best described as the strongest candidate under the campaign's product metric, not a demonstrated improvement in user task completion.

5.7 Negative results made the final interpretation possible

The public notes retain failed axes: trained cosine embeddings collapsed; self-distillation was flat; cleaned-target continuation damaged beam ranking; free decode hallucinated; terminal-specific IDF, a local ranker, and a proposer lost; length filters tied; 176k weak labels failed; compact final distillation remained far behind; schema continuation learned the wrong dialect; and Hybrid A damaged the strongest checkpoint. Without these results, the final narrative could be mistaken for a straightforward story of scaling from 12.7M to 77M.

Instead, the evidence supports a more specific account. The campaign first extracted value from a compact model through search and grounding. It then hit a domain and dialect ceiling. A larger instruction-tuned base became useful only after title-specific and terminal-specific supervision. Additional quality came from selective, leak-checked data rather than indiscriminate label scale. The failed interventions identify the boundaries of each successful mechanism.

6 Limitations and Threats to Validity

Single product and model family. The study concerns one short-form UI task and mostly T5-family systems between 12.7M and 77M parameters. The constrained decoder depends on a tiny output space and visible-source candidates. Results may not transfer to longer summaries, conversational responses, or decoder-only architectures.

Adaptive experimentation and multiple comparisons. The sessions were sequential and exploratory. Candidate families, learning rates, thresholds, and compositions were selected after observing prior results. Reported bootstrap intervals quantify task-sampling uncertainty within a fixed comparison; they do not correct for the full adaptive search process. The two unseen Session 11 holdouts and sealed Session 3/9 sets are stronger evidence than reusable development packets, but they do not eliminate researcher degrees of freedom.

LLM judge validity. The primary quality metric is a single judge family and rubric. Blinding, candidate randomization, paired comparisons, and independent audits reduce some risks, but no large human preference study validates the 1–10 scale or the solid-title threshold. The judge model itself may change over time unless the exact endpoint and behavior are reproducibly pinned. Absolute means should therefore be treated as operational measurements, not universal psychological quantities.

Private terminal data. Production-shaped terminal tasks are withheld to protect user and project context. This prevents full external replication of the strongest results. Public Stack Overflow tasks and evidence bundles support mechanism-level inspection, while terminal aggregate tables support provenance, but the final quality claims cannot be independently recomputed without access to the private evaluation sets.

Nonuniform packets and thresholds. The campaign used different board sizes and candidate packets across sessions. Early reports used score ≥ 4 as useful; later reports used score ≥ 6 as solid. We avoid converting between them, but the evolving protocol limits a single end-to-end learning curve. Some later contrasts lack published confidence intervals.

Latency and size comparability. Session 1 used a PyTorch/MPS research path, while later sessions used optimized product graphs, including ONNX. Hardware, warm-up, page-in, quantization, and tokenizer costs were not held constant across the full campaign. Figure 3 is restricted to Session 10's reported product-path measurements. Session 11 carries forward the Session 10 clock because the graph was unchanged; it was not freshly benchmarked.

No end-user outcome study. The rubric asks whether a title would help a user find a tab later, but the campaign did not measure search time, error rate, retention, or user satisfaction in a randomized product experiment. The title metric is therefore a proxy for the final user outcome.

7 Reproducibility and Artifact Boundaries

Every session is documented in the public PorkiCoder research index, including findings later superseded [Hossain, 2026a]. The Session 3 evidence bundle publishes decoder and glue source, frozen task/output artifacts, blinded audits, score rows, environment pins, and checksums. The Sniff Test publishes aggregate matched-control outcomes, prospective stopping rules, runtime information, source revision, and provenance hashes. Other reports publish aggregate tables, candidate definitions, and selected output examples.

The public artifacts intentionally do not expose production task text, private source identifiers, or all row-level terminal predictions. The present manuscript therefore supports three levels of reproducibility:

- **Document reproducibility:** every numerical claim is traceable to a dated session report.
- **Mechanism reproducibility:** the public SO-board experiments and Session 3 decoder artifacts permit inspection of the constrained-search result.
- **Product-result auditability:** terminal reports preserve aggregate same-packet outcomes, training recipes, and leakage boundaries, but not the private evaluation corpus.

For arXiv submission, the accompanying source archive contains the LaTeX manuscript, bibliography, compiled bibliography, and all figures used in the paper. It does not present the source archive itself as an experimental data release.

8 Conclusion

Local tab-title generation is a small-output problem with a large systems surface. Across eleven sessions, the decisive improvements did not follow a simple progression from more parameters or more labels to better titles. A random-projection highlighter beat a neural incumbent. Constrained search exposed a large gain in an unchanged 35M model. That proxy-domain victory failed an absolute-usefulness gate and did not transfer to terminal tasks. Curated judged labels improved the compact line, while 175,642 weak labels and lower cross-entropy did not. A 24.5M pointer system established a fast structured fallback, but a domain-tuned 77M FLAN model ultimately provided the best measured quality. A leak-checked continuation then improved two unseen 1,000-task holdouts by 0.21 and 0.20 points.

The campaign's practical conclusion is that product-constrained generation should be optimized as a complete pipeline. Training data define a target dialect; decoding exposes or suppresses model competence; grounding rules encode checkpoint-specific error assumptions; evaluation boards define the domain claim; and

deployment graphs determine whether a nominally small model is actually fast. For this task, the final ship is raw `flat1e4` Title-SFT FLAN, with no Hybrid A glue. The next scientifically useful step is not another small learning-rate sweep, but new high-quality, leak-checked terminal examples and a controlled end-user retrieval study.

A Complete Session Chronology

Table 5: The eleven experimental sessions consolidated in this paper. Sessions 5–8 share one public techniques-log publication.

S	Date	Primary intervention	Headline same-session result	Decision or boundary
1	14 Aug 2026	Seven-way baseline: incumbent, centroid, compact GSG, title-tuned FLAN, and glue	FLAN + centroid reached 5.43 and 77.4% useful; centroid-only reached 4.21 at 0.8 ms	Two initial device recipes; later superseded
2	14 Aug 2026	Select mid-GSG B-9500 checkpoint; introduce centroid v2	B-9500 reached 5.65 in old packet and 5.96 in v2 packet, close to title-tuned FLAN	Keep B-9500 as compact base; compare only within packet
3	15 Aug 2026	Source-constrained beam search with unchanged weights	Sealed SO-board: 6.20 vs 5.45, paired +0.752; independent usefulness gate failed	Search result accepted; absolute product readiness rejected
4	16 Aug 2026	Terminal-domain judged SFT from selected FLAN outputs	r3 beat pin by +0.12 on confirmation; regressed on reusable SO-board	Domain-specific labels help; boards remain separate
5	16 Aug 2026	Six-teacher judged SFT	`6t_argmax` beat pin by +0.19 on terminal $n = 1{,}000$	`6t_argmax` becomes compact ship
6	16 Aug 2026	Contrastive objective and free/off-page decoding	Hinge approximately flat; free decode −0.89	Do not relax grounding; objective axis closed
7	16 Aug 2026	Terminal IDF, glue variants, local ranker/proposer	Terminal IDF −0.19; local ranker −0.13	Additional local components did not improve ship
8	16 Aug 2026	Prompt-length filters and 175,642-label scale test	min5/min10 tied; all four large-label arms lost to `6t_argmax`	Quantity and lower CE did not replace curated SFT
9	17 Aug 2026	Dense P1 intent/object pointer head	P1 beat `6t_argmax` by +0.13 but lost to FLAN by −0.60	P1 becomes fast fallback, not quality winner
10	17 Aug 2026	Common ship-board plus product latency; terminal Title-SFT FLAN	Raw Title-SFT 7.28 at 29.3 ms; confirm-v1 6.55; Hybrid A hurt confirmation	Accept size/latency; ship raw SFT candidate
11	17 Aug 2026	Four-epoch leak-checked continuation of Session 10 FLAN	+0.21 and +0.20 on two unseen 1,000-task holdouts	`flat1e4` ships; next move is more quality data

Table 6: Selected Session 9 results on sealed Terminal-v2, $n = 516$.

System	Mean	Score ≥ 6	Relative comparison
Title-tuned FLAN	7.02	80.2%	P1 difference -0.60 $[-0.70, -0.49]$
P1 + Hybrid A	6.43	68.2%	`6t_argmax` difference $+0.13$ $[0.04, 0.22]$
`6t_argmax` + Hybrid A	6.30	65.3%	baseline

B Detailed Terminal Results

B.1 Session 9 sealed Terminal-v2

B.2 Session 10 confirmation packet

Table 7: Selected Session 10 confirm-v1 results, $n = 335$. This packet selected the raw Title-SFT composition.

System	Mean	Score ≥ 6
Raw Title-SFT FLAN	6.55	71.3%
P1 + Hybrid A	6.41	64.5%
CPU picker	6.41	–
Neural ranker	6.40	–
`6t_argmax` + Hybrid A	6.24	–
Title-SFT FLAN + Hybrid A	5.78	51.9%

B.3 Session 11 unseen holdouts

Table 8: Session 11 final comparison. Both boards contain 1,000 unseen tasks.

Board	System	Mean	≥ 6	≥ 7	Wins	Ties / losses
holdout1000	`flat1e4`	7.59	85.7%	80.1%	296	551 / 153
holdout1000	Session 10 ship	7.38	83.5%	74.6%	–	–
holdout1000b	`flat1e4`	7.55	84.5%	78.4%	281	570 / 149
holdout1000b	Session 10 ship	7.34	82.5%	74.4%	–	–

C Failure Catalogue

D Title Contract and Data Recipe

The final documented title contract is:

- one to three words, preferably two;
- at most 32 characters;
- ASCII alphanumeric characters and spaces;
- Title Case;
- no duplicate words;

Table 9: Negative results retained in the public record. Magnitudes are same-packet where reported.

Intervention	Observation	Interpretation used in later sessions
Cosine-trained centroid embeddings	Embeddings collapsed; random initialization worked as a filter	Treat centroid as a projection/highlighter, not learned semantic representation
Self-distill best-of-8 / best-of-24	Approximately flat	A teacher must add a different and consistent target dialect
Cleaned short-target CE continuation	Beam quality fell 0.534–0.647; low-LR run fell 0.601	Optimize the full decoder composition, not CE alone
Free off-page compact decode	Lost 0.89	Compact model still required hard grounding
Terminal-specific IDF	Lost 0.19	More domain-specific statistics were not automatically better
Local ranker / proposer	Ranker lost 0.13; proposer did not improve ship	Weak correlated selectors did not compose into a stronger system
Prompt-length filters	min5 and min10 tied `6t_argmax`	Handle very short prompts with a runtime policy rather than retraining
175,642 weak labels	All four arms lost to curated `6t_argmax`; lowest CE did not win	Label selection and dialect mattered more than quantity
35M final-teacher distillation	Remained 1.82 points behind teacher	Capacity and/or target complexity exceeded the compact student's effective regime
Structured schema continuation	Example comparison 6.70 vs 7.37	Legal structure alone encoded the wrong terminal dialect
Hybrid A on final Title-SFT	confirm-v1 fell from 6.55 to 5.78	Post-processing assumptions must be revalidated per checkpoint

- prefer visible source content, while allowing justified intent abstractions in the strongest free-generating model.

The Session 11 continuation mixture retained 6,419 rows after source-hash de-duplication against protected evaluation material, then split 6,219/200 for train/validation. Its historical 7,623-row source mixture comprised 1,530 repeated gold rows, 2,854 Grok 4.5 pseudo-labels, 1,868 Claude Opus 5 pseudo-labels, and 1,371 Codex pseudo-labels. The final teacher mixture did not use FLAN as a teacher. These counts describe the published recipe; they do not imply equal teacher quality or a controlled teacher ablation.

E Use of AI Assistance

The author used GPT-5.6 Pro to synthesize the public session notes into a manuscript structure, edit prose, generate the plotted figures from published aggregate values, and debug LaTeX. The experimental designs, task data, model training, evaluations, and published numerical results predate this manuscript synthesis. The author remains responsible for every claim and should verify the final source against the underlying artifacts before submission.

References

Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336. Association for Computing Machinery, 1998. doi: 10.1145/290941.291025. URL `https://dl.acm.org/doi/10.1145/290941.291025`.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi

Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. URL https://arxiv.org/abs/2210.11416.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall/CRC, 1994. doi: 10.1201/9780429246593.

Sebastian Gehrmann, Steven Layne, and Franck Dernoncourt. Improving human text comprehension through semi-Markov CRF-based neural section title generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1677–1688, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1168. URL https://aclanthology.org/N19-1168/.

MD Ishtiaque Hossain. Porkicoder research: Published studies and evidence. PorkiCoder Research, August 2026a. URL https://porkicoder.com/research/. Accessed 18 August 2026.

MD Ishtiaque Hossain. 176k flash-lite CE test: Terminal-board, still shipping 6t_argmax. PorkiCoder Research, August 2026b. URL https://porkicoder.com/research/2026-08-16_g300k-flashlite-test.html. Session 8 scale test, published 16 August 2026.

MD Ishtiaque Hossain. Two recipes, one glue. PorkiCoder Research, August 2026c. URL https://porkicoder.com/research/tab-titles-flan-centroid.html. Session 1, published 14 August 2026.

MD Ishtiaque Hossain. Session 10: Every namer, quality versus milliseconds. PorkiCoder Research, August 2026d. URL https://porkicoder.com/research/session10_results.html. Published 17 August 2026.

MD Ishtiaque Hossain. Session 11: Aug-12 continue-SFT is the new ship. PorkiCoder Research, August 2026e. URL https://porkicoder.com/research/session11_results.html. Published 17 August 2026.

MD Ishtiaque Hossain. Mid-pack GSG: The 35m that beat hybrid a and tied raw FLAN. PorkiCoder Research, August 2026f. URL https://porkicoder.com/research/tab-namer-mid-gsg.html. Session 2, published 14 August 2026.

MD Ishtiaque Hossain. Four beams, no new weights: A sealed test confirmed the FLAN win but missed the final gate. PorkiCoder Research, August 2026g. URL https://porkicoder.com/research/tab-namer-four-beams.html. Session 3, published 15 August 2026.

MD Ishtiaque Hossain. Session 4: Beating our own namer, still chasing FLAN. PorkiCoder Research, August 2026h. URL https://porkicoder.com/research/session4_results.html. Published 16 August 2026.

MD Ishtiaque Hossain. Session 9: Dense P1 + hybrid a beats 6t, still chasing FLAN. PorkiCoder Research, August 2026i. URL https://porkicoder.com/research/session9_results.html. Published 17 August 2026.

MD Ishtiaque Hossain. The sniff test: A seed lottery in a 12.7m-parameter language model. PorkiCoder Research, August 2026j. URL https://porkicoder.com/research/the-sniff-test.html. Published 12 August 2026.

MD Ishtiaque Hossain. Shipping 6t_argmax: A techniques log for terminal-board. PorkiCoder Research, August 2026k. URL https://porkicoder.com/research/tab-namer-techniques.html. Sessions 5–8, published 16 August 2026.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

Duc Anh Le, Anh M. T. Bui, Phuong T. Nguyen, and Davide Di Ruscio. Good things come in three: Generating SO post titles with pre-trained models, self improvement and post ranking. *arXiv preprint arXiv:2406.15633*, 2024. URL https://arxiv.org/abs/2406.15633.

Zechun Liu, Changsheng Zhao, Forrest Iandola, Chen Lai, Yuandong Tian, Igor Fedorov, Yunyang Xiong, Ernie Chang, Yangyang Shi, Raghuraman Krishnamoorthi, et al. MobileLLM: Optimizing sub-billion parameter language models for on-device use cases. *arXiv preprint arXiv:2402.14905*, 2024. URL https://arxiv.org/abs/2402.14905.

Justin D. Norman, Michael U. Rivera, and D. Alex Hughes. Reliability without validity: A systematic, large-scale evaluation of LLM-as-a-judge models across agreement, consistency, and bias. *arXiv preprint arXiv:2606.19544*, 2026. URL https://arxiv.org/abs/2606.19544.

Matt Post and David Vilar. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1119. URL https://aclanthology.org/N18-1119/.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL https://arxiv.org/abs/1910.10683.

Fengji Zhang, Xiao Yu, Jacky Keung, Fuyang Li, Zhiwen Xie, Zhen Yang, Caoyuan Ma, and Zhimin Zhang. Improving stack overflow question title generation with copying enhanced CodeBERT model and bi-modal information. *arXiv preprint arXiv:2109.13073*, 2021. URL https://arxiv.org/abs/2109.13073.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, pages 11328–11339, 2020. URL https://arxiv.org/abs/1912.08777.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging LLM-as-a-judge with MT-Bench and chatbot arena. *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://arxiv.org/abs/2306.05685.