Below the Noise Floor: Bimodal Seed Collapse and Distinct Failure Modes in Small-Model Knowledge Distillation

Anonymous ACL submission

Abstract

Function routing — selecting the correct API call from a fixed catalog given a natural-language request — is a deployment problem where small students are attractive but knowledge distillation gains are typically reported single-seed, at scales where seed variance is unknown. On a 740-instance healthcare API routing task with a 1.5B Qwen student and a 20B teacher, we compare eight KD variants against supervised cross-entropy, using three to six seeds for key configurations. We find: (i) per-seed standard deviation ranges from 2.8 to 48.7 percentage points, swallowing every claimed KD gain below five points; (ii) three of seven KD variants exhibit **bimodal collapse**, with at least one in three to five seeds falling below 55% accuracy while the others train normally, and a fourth showing elevated variance; (iii) collapse has distinct modes — wrong-function selection for `ce_Knowledge Distillation` and `ce_paraphrase`, and a previously undocumented *output-truncation* mode for `reasoning_kd`, where the model emits reasoning but terminates before producing a function name (0.9% accuracy); (iv) only `progressive_kd` and `rank_kd` avoid collapse across observed seeds, with $\sigma \leq 3.9$ pp; (v) a naive cross-split $+3.78$ pp gain from input enrichment reverses to -2.70 pp under controlled within-split multi-seed re-testing. Single-seed evaluation is therefore unable to detect central failure modes in small-model KD.

1 Introduction

Deploying small language models for function routing is attractive in latency- and cost-constrained settings, but these models frequently degrade under paraphrase variation and other forms of linguistic shift. The standard remedy is knowledge distillation (Hinton et al., 2015): a growing literature proposes increasingly sophisticated objectives (confidence-gated KD, selective distillation, contrastive distillation, progressive curricula), each reporting improvements of 1–5 percentage points over a supervised baseline. **We argue these gains are, in large part, artifacts of single-seed evaluation, and that the field is systematically missing two more interesting phenomena: KD-induced bimodal collapse, and qualitatively distinct collapse modes across seemingly related KD losses.**

On a 740-instance healthcare API routing task, we distill a 20B-parameter teacher into a 1.5B Qwen student and compare eight KD variants against a supervised cross-entropy baseline (`ce_only`). Five conditions are run with four to six seeds; the remaining KD variants are reported with three seeds. We make five findings.

(i) Per-seed standard deviation at this scale is large (σ=2.8–48.7 percentage points (pp) across conditions), comparable to or larger than the gaps between methods that the literature treats as meaningful.

(ii) Three of the seven multi-seed KD configurations exhibit **bimodal collapse**: under identical data, model, and hyperparameters, at least one in three to five random seeds drives the student to below 55% accuracy while the other seeds train to within a point or two of the baseline. A fourth (`gated_kd`) shows elevated variance ($\sigma = 10.5$ pp) without crossing the 55% threshold in our sample. This is a qualitatively different failure mode from gradual degradation, and it is invisible to single-seed evaluation.

(iii) Collapse is not a single phenomenon. We document at least two qualitatively distinct modes. `ce_Knowledge Distillation` and `ce_paraphrase` collapses produce *wrong-function selection*: the model still outputs valid function names but selects the wrong one at high confidence. `reasoning_kd` on seed 7 instead produces *output truncation*: the model emits a structurally valid reasoning chain but terminates on a learned-substitute EOS token (a single CJK character, U+622A, meaning "cut/intercept") before producing a function

name, yielding 0.9% accuracy because it never commits to a prediction. The two modes have different implications for what is going wrong during training.

(iv) Only `progressive_kd` (delayed-onset KD, $\sigma = 2.8$ pp over four seeds) and `rank_kd` (order-only KD, $\sigma = 3.9$ pp over four seeds) avoid collapse across all observed seeds. Both interventions share a property: they remove the full-strength KL gradient from the start of training, either by delaying it or by replacing it with a weaker pairwise margin signal.

(v) A cross-split comparison appears to show a strong $+3.78$ pp gain from adding function descriptions to the prompt. A controlled within-split re-test with three seeds reverses this finding ($\Delta = -2.70$ pp, $\sigma = 6$–11 pp), exposing it as a subset-selection artifact and providing a worked example of why single-seed cross-split evaluations produce unreliable headlines.

Our major contributions in this paper are as follows.

- Eight KD variants versus `ce_only` in a single controlled study, with three to six seeds per condition.
- Identification of **bimodal seed collapse** as a failure mode of auxiliary-loss KD at small student scale, observed in three independent loss configurations (`ce_Knowledge Distillation` (cross-entropy plus KL-divergence knowledge distillation), `ce_paraphrase` (cross-entropy plus paraphrase-consistency loss), `reasoning_kd`), with a fourth (`gated_kd`) showing elevated variance but no sub-55% seed in our sample.
- Identification of a previously undocumented *output-truncation* collapse mode in `reasoning_kd`, qualitatively distinct from the wrong-function mode in `ce_Knowledge Distillation`.
- A paired worked example: naive cross-split shows $+3.78$ pp from input enrichment; controlled within-split shows -2.70 pp. The reversal demonstrates concretely why multi-seed reporting is needed.

2 Related Work

Knowledge distillation. The standard formulation (Hinton et al., 2015) trains a student to match the temperature-softened softmax distribution of a larger teacher, combining a hard cross-entropy term with a soft KL-divergence term. Extensions to language models include DistilBERT (Sanh et al., 2019) and sequence-level distillation (Kim and Rush, 2016). More recent work distills reasoning capabilities via chain-of-thought rationales (Magister et al., 2023; Hsieh et al., 2023). Stanton et al. (2021) showed that students trained with KL-divergence often produce predictions that diverge from the teacher despite low loss, suggesting that the standard objective measures the wrong thing. Our work extends this skeptical line: we show that in small-data structured classification, KD-augmented objectives exhibit bimodal seed behavior — normal training in most seeds, qualitatively distinct collapses in others — that single-seed reporting cannot detect.

A separate line studies how to mitigate harmful teacher signal. Confidence-based gating modulates KD weight by teacher entropy; selective distillation (Wang et al., 2021) uses only high-confidence or high-agreement teacher predictions. Progressive or curriculum-style KD schedules (Jafari et al., 2021) delay or anneal the teacher signal. We test variants from each of these families and find that the curriculum approach (delayed onset) is alone in eliminating collapse without sacrificing accuracy.

Function calling and tool use. Recent work has emphasized scaling the function catalog and reasoning about multi-step invocation. Gorilla (Patil et al., 2023) and ToolLLM (Qin et al., 2024) target catalogs of thousands of APIs; xLAM (Liu et al., 2024) releases function-calling fine-tunes; the Berkeley Function Calling Leaderboard (Yan et al., 2024) has become a standard evaluation. Most of this work focuses on catalog breadth. We study the complementary regime: a fixed, modest catalog where the bottleneck is per-query routing accuracy under linguistic variation. Chen and Li (2024) and Erdogan et al. (2024) demonstrate that sub-2B models can achieve strong function-calling accuracy. Our methodological contribution is orthogonal: we ask whether the reported *differences* between fine-tuning strategies at this scale are reliable.

Seed variance and small-data evaluation. Dodge et al. (2020) showed that BERT fine-tuning

on small datasets exhibits seed-to-seed standard deviations of several points on GLUE benchmarks and that random restarts substantially change conclusions. Mosbach et al. (2021) and Reimers and Gurevych (2017) make analogous arguments for sentence-level classifiers and sequence labeling. To our knowledge, no published comparison of small-model knowledge distillation variants has reported the bimodal collapse phenomenon we document, the qualitatively distinct collapse modes we identify, or per-seed variance in the magnitude we measure (σ up to 48.7 pp).

3 Setup and Methods

3.1 Task definition

We study **candidate function selection**: given a natural-language user instruction x and a candidate set of K function names $\{f_1, \ldots, f_K\}$, predict the single correct function $f^* \in \{f_1, \ldots, f_K\}$. We do not study argument generation or multi-step tool use. K ranges from 2 to 4 (mean 3.02).

3.2 Data

A routing benchmark constructed by the authors from publicly documented API schemas in the healthcare, insurance, and fitness domains. Instructions were generated by the authors as synthetic natural-language paraphrases of canonical intents covered by each API, with no real user queries or patient data involved (see also the Ethics Statement). The benchmark is partitioned by **paraphrase family**: distinct phrasings of the same intent are placed in the same split, so the test set contains rephrasings never seen during training. Two split variants are used: V1 splits (515 train / 110 dev / 111 test), and V2 splits (362 train / 81 dev / 78 test), where V2 is a subset of V1 restricted to instructions whose candidate functions all have catalog descriptions available.

3.3 Student and teacher

Student. A 1.5B Qwen2.5-Instruct (Yang et al., 2025) fine-tuned with QLoRA (Dettmers et al., 2023) ($r = 32$, $\alpha = 64$, dropout 0.05) targeting all linear projections. The base model is frozen at 4-bit precision; only LoRA adapters and a small ranking head are trained.

Teacher. A 20B-parameter open-weight Mixture-of-Experts language model with native function-calling pretraining, fine-tuned on our training set. Teacher annotations are generated once and stored offline. The teacher achieves 83.1% agreement with gold on V1 training samples.

Training. Effective batch size 16 (4×4 gradient accumulation), AdamW (β_1=0.9, β_2=0.999, ϵ=10^{-8}) with cosine schedule and 5% linear warmup, peak LR 2×10^{-4}, weight decay 0.01, gradient clipping at 1.0, 10 epochs with early stopping (patience 3) on dev loss. Mixed precision (bfloat16) with gradient checkpointing on the frozen base model. The ranking head is a 2-layer MLP (hidden 512, ReLU, dropout 0.1) over mean-pooled final hidden states. Each training run took 35–50 minutes on a single RTX A6000 or comparable 16 GB consumer card; the full multi-seed sweep across eight conditions consumed approximately 30 GPU-hours.

Seeding. The reported seed controls (a) numpy and PyTorch RNGs, (b) the LoRA adapter initialization, (c) the ranking-head initialization, and (d) the per-epoch data shuffle order. The same seed reproduces identical results bit-for-bit on the same hardware. Tokenization, the candidate-set order, and the teacher annotations are seed-independent (computed once offline).

3.4 Student input representations

We used two input formats: **names-only** (candidate function names listed) and **descriptions** (each candidate annotated with a one-line description and typed parameter list). The main multi-seed comparison uses names-only on V1. Section 4.5 runs a controlled within-split test toggling only this format.

3.5 Teacher annotation methods

We used two annotation procedures, both producing per-candidate soft scores. **Method A: JSON self-report** (used in the main conditions). The teacher outputs a JSON object containing `candidate_scores`, a `chosen_function`, and a 3-step `reasoning` chain. Mean entropy 0.385 nats. **Method B: log-probability harvest with temperature** (used only in the `logit_kd` reference). Mean entropy 0.692 nats. We report `logit_kd` as a cross-system reference rather than as a controlled comparison.

3.6 Loss formulations

All ten loss configurations we evaluate share a common skeleton $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{LM}} + \mathcal{L}_{\text{select}}$, where $\mathcal{L}_{\text{LM}}$ is causal-LM cross-entropy on the target string and

$\mathcal{L}_{\text{select}}$ varies across configurations. The student includes a 2-layer MLP candidate ranking head over pooled hidden states. We compare ten configurations; $\mathbf{p}_T$ is the teacher's candidate distribution, $\mathbf{z}_S$ the student's candidate logits.

ce_only. $\text{CE}(\mathbf{z}_S, y^*)$. Primary baseline.

ce_Knowledge Distillation. $\text{CE} + \lambda_{KD} T^2 \,\text{KL}(\mathbf{p}_T \| \text{softmax}(\mathbf{z}_S/T))$; λ_{KD}=1.0, T=2.

ce_paraphrase. $\text{CE} + \lambda_{\text{para}}(1 - \cos(\mathbf{h}_i, \mathbf{h}_j))$ on in-batch paraphrase pairs; λ_{para}=0.5.

full. $\text{CE} + \lambda_{KD}\,\text{KL}_T + \lambda_{\text{para}}(1 - \cos)$.

rank_kd. CE plus pairwise margin over teacher-preferred candidate pairs: $\frac{1}{|P|}\sum_{(a,b)\in P} \max(0,\, m - (z_{S,a} - z_{S,b}))$, m=0.5. Only teacher ranking order matters.

gated_kd. CE plus KL weighted by teacher confidence: $(1 - \tilde{H}(\mathbf{p}_T)) \cdot \lambda_{KD} T^2 \,\text{KL}_T$.

progressive_kd. CE-only for epochs 1–4; KD weight linearly ramped epochs 5–7; full KD epochs 8–10.

reasoning_kd. KL applied only on samples where the teacher's argmax matches gold; CE only on disagreements.

contrastive_kd. InfoNCE (van den Oord et al., 2019) with paraphrase as positive, in-batch samples as negatives (τ=0.07), plus CE on the candidate head.

logit_kd. Uses Method B annotations and a different ranking head. Loss: $0.5\,\mathcal{L}_{\text{LM}} + 0.5(\alpha\,\text{KL}_T + \beta\,\text{CE})$, α=0.7, β=0.3.

3.7 Evaluation

On the held-out test split, we report three metrics: **Accuracy** (argmax of $\mathbf{z}_S$ vs gold); **Macro F1**; and **Paraphrase consistency (PC)**, the fraction of paraphrase families on which the model predicts the identical function name across all family members. The test split contains 14 such families. PC is strict (all-same) and noisy at this sample size; Section 4.1 flags one case where PC reaches 1.0 vacuously.

4 Results

4.1 Multi-seed V1 results: bimodal collapse

Table 1 reports the eight conditions for which we ran three or more seeds. Per-seed values are in Appendix A.

Method	n	Mean $\pm\sigma$	Collapse
`ce_only`	6	82.6 ± 4.8	—
`progressive_kd`	4	82.5 ± 2.8	none
`rank_kd`	4	80.9 ± 3.9	none
`full`	4	78.2 ± 6.6	none
`ce_paraphrase`	5	75.7 ± 13.4	1/5
`gated_kd`	4	73.9 ± 10.5	none
`ce_Knowledge Distillation`	5	61.1 ± 22.1	2/5
`reasoning_kd`	3	57.1 ± 48.7	1/3†

Table 1: Multi-seed V1 test accuracies (%), names-only prompts, ranked by mean. Collapse = number of seeds with test accuracy $< 55\%$. Three of seven KD configurations exhibit collapse; a fourth (`gated_kd`) shows elevated variance without a sub-55% seed. Only `progressive_kd` and `rank_kd` are stable across all observed seeds. Per-seed values in Appendix A. †`reasoning_kd` seed 7 collapses to 0.9% via a qualitatively distinct *output-truncation* mode (Section 4.2), not wrong-function selection.

Bimodal collapse in auxiliary-loss KD. `ce_Knowledge Distillation` collapses on 2 of 5 seeds (31.5%, 48.7%), with the remaining three seeds (62.2, 81.1, 82.0) clustered near or slightly below baseline. Its standard deviation, $\sigma = 22.1$ pp, is more than four times the baseline's. `ce_paraphrase` shows the same pattern at lower frequency: one seed at 52.3%, four seeds between 77 and 86%. `reasoning_kd` produces the most extreme collapse, with one of three seeds at 0.9% — a qualitatively different failure mode we examine separately in Section 4.2. The other auxiliary-loss configurations (`gated_kd`, `full`) show elevated variance (σ = 10.5 and 6.6) without sub-55% seeds in our sample, suggesting milder versions of the same instability. This is invisible to single-seed evaluation: depending on which seed is drawn, the same training recipe would be reported as "ties with baseline" or "catastrophic failure."

Two methods stable across all seeds. `progressive_kd` (delayed KD onset) achieves σ = 2.8 pp across four seeds (79.3–85.6), with no seed below 79% accuracy. `rank_kd` (order-only KD) achieves $\sigma = 3.9$ pp across four seeds (77.5–84.7). Both interventions share a structural property: they remove the full-strength KL-divergence gradient from the start of training, either by delaying it (`progressive_kd`) or by replacing it with a weaker pairwise margin signal that uses only teacher rankings, not magnitudes (`rank_kd`). Neither significantly beats the baseline

on mean accuracy (-0.1 and -1.7 pp), but both achieve sub-baseline variance.

Method ranking depends on the seed. Under seed 42, the top three are `ce_only`/`full`/`progressive_kd` clustered at 85.6%. Under seed 123, `ce_Knowledge Distillation` (82.0) and `ce_paraphrase` (80.2) outrank `ce_only` (75.7). Under seed 7, `reasoning_kd` drops to 0.9% while `rank_kd` reaches 83.8%. No two seeds produce the same ranking. Single-seed comparison of these methods is, by direct measurement, uninformative about which is better.

4.2 Distinct collapse modes

The collapses in Table 1 are not a single phenomenon. We identify at least two qualitatively distinct modes by inspecting raw outputs from the collapsed checkpoints (Figure 1).

Wrong-function selection (`ce_Knowledge Distillation`, `ce_paraphrase`). On collapsed seeds, the trained student emits structurally valid outputs — a reasoning chain followed by a function name — but selects the wrong function with high confidence. Per-class confusion matrices for the worst `ce_Knowledge Distillation` seed (31.5%) show errors concentrated on a small subset of intent families: `return_equipment` predicted as `get_equipment_list`, `view_medical_records` as `search_patients`, `add_remedy` as `search_remedies`. The student has learned a degenerate mapping that aliases related-but-distinct intents.

Output truncation (`reasoning_kd`). On seed 7, `reasoning_kd` collapses to a different mode entirely. The model produces 1 correct prediction out of 111. Inspection of the raw outputs reveals that 107 of the 110 wrong predictions are `None` — the regular expression that extracts the function name from the reasoning chain returns no match. The model outputs a structurally valid two-step reasoning chain (`reasoning_coverage`=1.0, `avg_reasoning_steps`=2.07) and then terminates on a single CJK character (Unicode codepoint U+622A, meaning "cut" or "intercept") before producing the required "Chosen function:" line. The character acts as a learned EOS substitute: the optimizer has converged on a fixed point that produces fluent-looking reasoning but never commits to a prediction. We note that `paraphrase_consistency` for this collapsed seed is 1.0 — vacuously, since every paraphrase variant hits the same truncation pattern.

Method	Acc	F1	PC
`ce_only` (baseline)	88.5	79.2	90.9
`ce_Knowledge Distillation`	85.9	72.9	72.7
`ce_paraphrase`	84.6	69.4	81.8
`full`	69.2	52.5	45.5

Table 2: V2 condition results, single seed ($n = 78$ test). Given the seed variance documented for V1, single-point V2 numbers should be read as samples from high-variance distributions rather than stable per-method effects.

Why this matters. The two modes have different mechanistic implications. Wrong-function selection (panel a) is consistent with the model finding a degenerate minimum where teacher and gold signals partially cancel, producing high-confidence aliasing on intent-family clusters. Output truncation (panel b) is consistent with the optimizer learning to short-circuit the output sequence to avoid penalty entirely. Notably, `reasoning_kd` — which applies KL *only* on teacher-gold-agreeing samples, and is therefore the variant with the least noisy KD signal in our study — produces the most extreme failure. This refutes a simple "teacher noise amplification" explanation: it is not the disagreement-driven 16.9% of training samples that destabilizes training, since removing them entirely makes things worse. We return to mechanism in Section 5.1, with the caveat that our data identify which interventions correlate with stability (delaying KD onset, replacing KL with a pairwise margin) but do not isolate a single causal mechanism.

4.3 V2 results (descriptions prompt, single seed)

Table 2 reports the four V2 single-seed runs. The 19 pp drop of `full` on V2 is consistent with a single-seed draw from the high-variance distribution characterized on V1 ($\sigma = 6.6$, range 70.3–85.6); we do not interpret it as a stable per-method effect.

4.4 The apparent input-enrichment effect

A naive cross-split comparison of `ce_only` suggests that adding function descriptions to the student prompt produces a $+3.78$ pp gain, larger than any KD-design mean effect we measured. The comparison is confounded along two axes: prompt

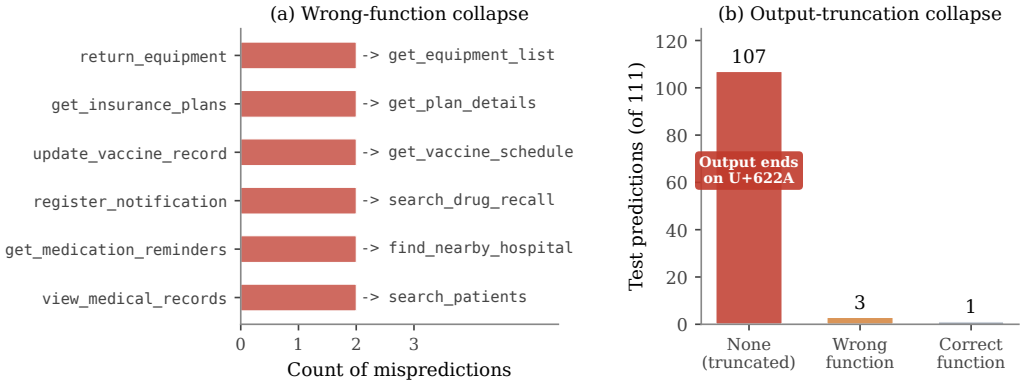

Figure 1: Two qualitatively distinct collapse modes. (a) Wrong-function collapse in `ce_Knowledge Distillation` seed 456 (31.5% accuracy): the model emits valid function names but consistently aliases related intents (each row: gold → predicted, two test instances per row). (b) Output-truncation collapse in `reasoning_kd` seed 7 (0.9% accuracy): 107 of 111 test predictions return `None` because the output terminates on a learned single-character EOS surrogate (Unicode codepoint U+622A, meaning "cut/intercept") before the model commits to a function name.

Condition	Acc	F1	PC
V1 `ce_only` (names)	84.7	74.0	78.6
V2 `ce_only` (descriptions)	88.5	79.2	90.9
Δ (V2 − V1), naive	**+3.78**	**+5.20**	**+12.34**

Table 3: Naive cross-split comparison. The apparent +3.78 pp gain confounds prompt format with subset selection (V2 is a description-available subset of V1).

Arm	Acc (mean $\pm\sigma$)	Δ
V1, names-only	82.9 ± 6.3	—
V1, descriptions	80.2 ± 10.9	-2.70

Table 4: Controlled within-split prompt-format comparison, 3 seeds per arm (per-seed values in Appendix A). Descriptions show a -2.70 pp mean difference, with both arms exhibiting standard deviations larger than the difference. The naive $+3.78$ pp finding (Table 3) does not survive.

format and data subset. We initially read it as a real effect, then ran the controlled experiment below, and the apparent effect did not survive.

4.5 Controlled within-split test

We fix the data split to V1 and vary only the prompt format. Three seeds (42, 123, 7) per arm, `ce_only` loss, all other settings identical.

Two findings, methodologically central. First, the +3.78 pp apparent effect from the naive cross-split comparison *reverses sign* under controlled within-split conditions (-2.70 pp; Figure 2), suggesting that the V2 subset is intrinsically easier — not that the prompt format itself helps. Second, the per-seed variance is large: $\sigma = 6.3$ for names-only and $\sigma = 10.9$ for descriptions, with individual seed runs spanning 67.6% to 87.4% on the same 111-instance test split. The 2.70 pp arm-mean difference is well within both arms' standard deviations; we do not claim descriptions hurt either. We claim only that the data do not support a description effect of either sign at this scale.

4.6 Summary of findings

Five findings emerge: (i) per-seed variance dominates between-method differences at this scale (Figure 4); (ii) auxiliary-loss KD variants exhibit bimodal collapse on 1-in-3 to 2-in-5 seeds; (iii) collapse has at least two qualitatively distinct modes (wrong-function selection and output truncation; Figure 1); (iv) only delayed-onset and order-only KD avoid collapse across all observed seeds; (v) a naive cross-split prompt-format gain reverses sign under controlled within-split re-testing (Figure 2).

5 Discussion

5.1 What the collapse modes suggest about mechanism

The most informative comparison in our data is between `ce_Knowledge Distillation` and `reasoning_kd`. Both apply full-strength KL-divergence between teacher and student distributions; they differ only in *which* training examples receive the KD signal. `ce_Knowledge Distillation` applies KL to every example, including the 16.9% where the teacher disagrees with gold. `reasoning_kd` applies KL only on the 83.1% where teacher and gold agree — so by construction, the KD signal it receives is denoised.

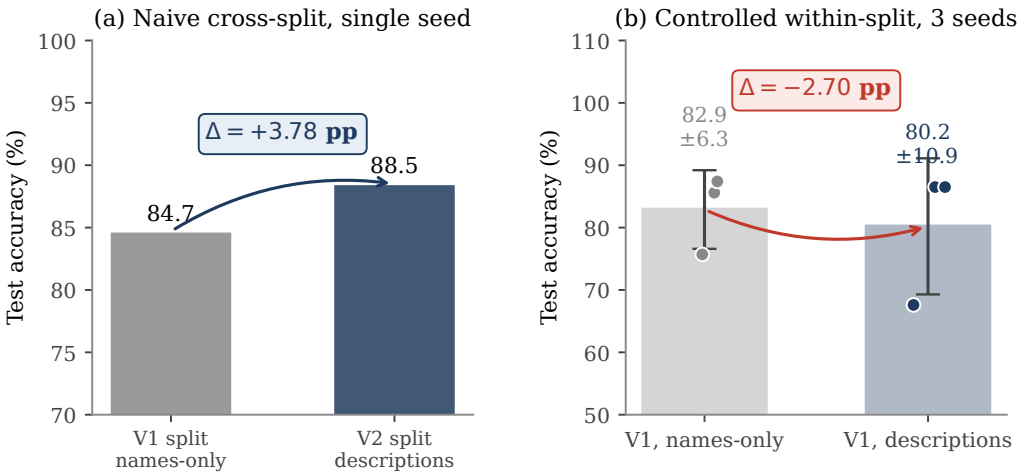

Figure 2: The cross-split reversal. (a) A naive single-seed comparison across two splits suggests a $+3.78$ pp gain from adding function descriptions to the student prompt. (b) The same intervention re-tested under a controlled within-split protocol with three seeds per arm reverses sign (-2.70 pp). Dots are individual seed runs; bars and error bars show arm mean $\pm$ one standard deviation. Both arms' standard deviations are larger than the difference between their means: the apparent effect was a subset-selection artifact, and the within-split data do not support a description effect of either sign at this scale.

If teacher noise were the primary driver of collapse, `reasoning_kd` should be *more* stable than `ce_Knowledge Distillation`. Empirically, it is not: `reasoning_kd` produces the most extreme collapse we observe (0.9% accuracy, σ=48.7), and into a different failure mode than `ce_Knowledge Distillation`. Teacher noise alone is not sufficient to explain the collapse pattern.

A second comparison provides positive evidence. `progressive_kd` delays the KL gradient until epoch 5; `rank_kd` replaces the KL gradient with a much weaker pairwise margin signal. Both avoid collapse entirely across all observed seeds. Both interventions share a structural property: they alter the timing or strength of the KL gradient at training onset. Combined with the `reasoning_kd` null result, the natural hypothesis is that bimodal collapse is driven by early-training interaction between the full-strength KL gradient and the randomly-initialized ranking head, rather than by teacher noise per se.

Counterexamples that complicate the story. Two observations resist this clean reading. First, `full` applies full-strength KL from epoch 1 alongside CE and paraphrase losses, yet did not collapse in our four seeds ($\sigma = 6.6$, no sub-55% seed). This may be a sample-size effect — collapse rates of 1-in-5 to 1-in-3 in collapsing configurations leave a 30–50% probability that four seeds would all train normally even if `full` shared the same failure distribution — or it may indicate that mixing KL with other auxiliary objectives is genuinely protective. Second, `gated_kd` applies confidence-weighted KL from epoch 1 and shows high variance ($\sigma = 10.5$) without a sub-55% seed in our sample. Both observations are consistent with collapse being a probabilistic outcome whose rate depends on a configuration-specific interaction we have not isolated. The two collapse modes (Figure 1) further suggest the optimizer can be driven into more than one degenerate basin; isolating which basin a given seed-configuration combination will land in would require gradient-trajectory and loss-landscape measurements that we leave to future work.

5.2 Practical implications

For practitioners at this scale: treat `ce_only` as a strong baseline; prefer `progressive_kd` or `rank_kd` when KD is desired (variance reduction without expected-mean cost); avoid auxiliary-loss configurations that apply full-strength KL from epoch 1, including agreement-conditioned variants; run at least three seeds before reporting improvements below 5 pp; inspect raw outputs on at least one seed, not just argmax accuracy, since at least one collapse mode in our data produces null predictions invisible to standard scoring.

6 Conclusion

We compared eight knowledge distillation variants against a supervised cross-entropy baseline on a

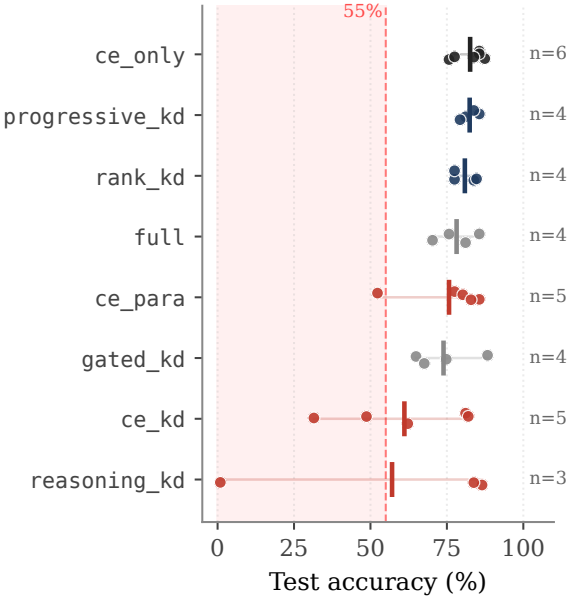

Figure 3: Per-seed test accuracy (%) for the multi-seed V1 conditions in Table 1. Dots are individual seed runs; vertical ticks show means. Sub-55% seeds (the collapse region, dashed line) appear in three of seven KD configurations; `reasoning_kd`'s 0.9% point reflects the output-truncation mode (Section 4.2), not wrong-function selection. Only `progressive_kd` and `rank_kd` keep their full seed range above the baseline's worst seed.

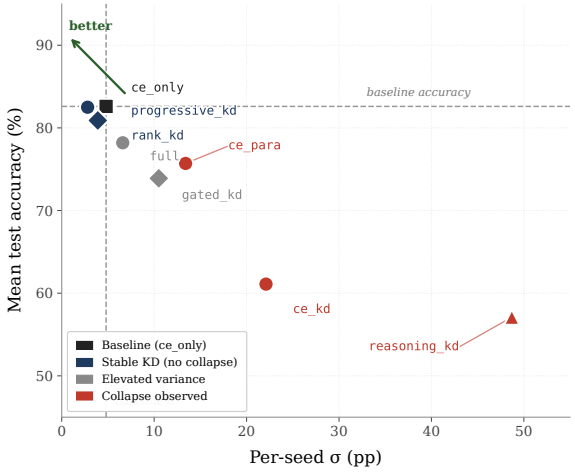

Figure 4: Mean test accuracy versus per-seed standard deviation across the eight multi-seed conditions. Dotted reference lines mark the baseline (`ce_only`). The stability-accuracy tradeoff is monotone in our data: methods that match the baseline mean (`progressive_kd`, `rank_kd`) sit at or below the baseline's variance; methods with elevated variance (`gated_kd`, `ce_paraphrase`) lose accuracy; methods with high variance lose substantially more (`ce_Knowledge Distillation`, `reasoning_kd`). No method achieves both higher mean and lower variance than the baseline.

740-instance function routing task, with three to six seeds per condition. We find that auxiliary-loss KD variants exhibit **bimodal seed collapse** — one in three to five random seeds drives the student below 55% accuracy while the others train normally on identical data. Collapse is not a single phenomenon: `ce_Knowledge Distillation` and `ce_paraphrase` produce wrong-function selection, while `reasoning_kd` produces a previously undocumented *output-truncation* mode in which the trained model emits fluent reasoning but never commits to a function. Only `progressive_kd` (delayed KD onset) and `rank_kd` (order-only KD) avoid collapse across all observed seeds. The common property of the methods that succeed — removing the full-strength KL gradient from the start of training — and the failure of agreement-conditioning to prevent collapse together suggest that early-training KL interaction with the randomly-initialized ranking head, not teacher noise, drives the instability. An apparent $+3.78$ pp gain from a non-KD intervention reverses to -2.70 pp under controlled multi-seed re-testing. For the field, multi-seed reporting at small scale is not optional; bimodal collapse, including silent output-truncation modes, is a failure mode worth looking for in any small-model KD result.

Limitations

Uneven seed counts across conditions. Seed counts are not equal across configurations (Table 1: $n = 3$ to $n = 6$). Experiments were run incrementally as compute became available over a four-day window, with seed allocation prioritized by the information each new run was expected to add: more seeds for the most-controlled comparisons (`ce_only`, `progressive_kd`), fewer for ablations that turned out to be qualitatively decisive on their first observation (`reasoning_kd`, where seed 7's 0.9% collapse arrived in the final batch). All seeds we ran are reported; none were dropped. The bimodal collapse and output-truncation findings are robust to the variation in n — they would survive any reasonable reweighting of the seeds we collected — but precise σ estimates for conditions with $n = 3$, in particular `reasoning_kd`'s $\sigma = 48.7$, should be read as preliminary indicators of high variance rather than as tight population estimates. We recommend that follow-up work treat our $n = 3$ values as load-bearing for the qualitative

finding (collapse mode exists, distinct from wrong-function mode) but not for quantitative ranking against other conditions.

Three seeds is a low bar. Even our best-sampled conditions ($n = 6$) are below the seed counts Dodge et al. (2020) and Mosbach et al. (2021) recommend for stable BERT-fine-tuning variance estimates ($n \geq 10$). Our results therefore document the existence and qualitative shape of bimodal collapse but do not pin down the precise collapse-rate distribution for any single configuration.

Scope of claims. A single healthcare/insurance/fitness domain with $\sim$700 training examples and a 1.5B student. Bimodal collapse may not transfer to larger catalogs, other domains, or datasets with $\sim 10^5$+ examples. Our teacher has 16.9% disagreement with gold; a more capable teacher could shift the comparison.

Mechanism is inferred from ablations, not measured directly. The early-training KL-interaction hypothesis in Section 5.1 is supported by two converging ablations (`progressive_kd`, `rank_kd`) and one informative null result (`reasoning_kd`), but is complicated by two counterexamples (`full` and `gated_kd`) and is not established by direct measurement of gradient trajectories or loss-landscape geometry.

PC metric. Paraphrase consistency is computed over only 14 families and can reach 1.0 vacuously, as in the `reasoning_kd` truncation case. Individual PC values should not be over-interpreted.

Output inspection is partial. We manually inspected raw outputs only for the most extreme collapsed checkpoints. Other multi-seed conditions reporting wrong-function collapse may also contain subtler output pathologies we did not catch.

Ethics Statement

This work studies training stability of small language model students on a function routing task over public-style healthcare, insurance, and fitness API schemas. We do not collect new human data; the dataset consists of synthetic instruction-tool pairs without personal information. The healthcare-themed API schemas are illustrative templates, not connected to any real patient records or clinical systems, and our trained models are not intended for deployment in clinical or patient-facing settings without independent safety evaluation. All model and dataset artifacts were used in accordance with their stated licenses, terms of use, and redistribution restrictions

Our central finding — that small-model knowledge distillation exhibits bimodal seed collapse including silent output-truncation failures — has a direct safety implication: a single-seed deployment of any of the auxiliary-loss KD variants we test could, with non-trivial probability, produce a model that emits fluent reasoning but fails to commit to a function call, or that confidently routes user requests to the wrong API. In settings where the routed function has real-world consequences (medication look-up, appointment booking, claim submission), this is a deployment risk worth flagging. We recommend practitioners run multi-seed evaluation and inspect raw outputs — not just argmax accuracy — before shipping any KD-trained small student in a function-routing role.

Compute usage for the full multi-seed sweep is approximately 30 GPU-hours on a single consumer-grade 16 GB card. We release the training and evaluation code, per-seed result logs, and prompt templates with the paper to enable independent replication.

We used AI assistants (Claude) for writing assistance and coding support during the preparation of this work.

References

Wei Chen and Zhiyuan Li. 2024. Octopus v2: On-device language model for super agent. *arXiv preprint arXiv:2404.01744*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. *arXiv preprint arXiv:2305.14314*. NeurIPS 2023.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *Preprint*, arXiv:2002.06305.

Lutfi Eren Erdogan, Nicholas Lee, Siddharth Jha, Sehoon Kim, Ryan Tabrizi, Suhong Moon, Coleman Hooper, Gopala Anumanchipalli, Kurt Keutzer, and Amir Gholami. 2024. TinyAgent: Function calling at the edge. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 80–88, Miami, Florida, USA. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *Preprint*, arXiv:1503.02531.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Chen, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8003–8017, Toronto, Canada. Association for Computational Linguistics.

Aref Jafari, Mehdi Rezagholizadeh, Pranav Sharma, and Ali Ghodsi. 2021. Annealing knowledge distillation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2493–2504, Online. Association for Computational Linguistics.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, Austin, Texas. Association for Computational Linguistics.

Zuxin Liu, Thai Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, and 1 others. 2024. APIGen: Automated pipeline for generating verifiable and diverse function-calling datasets. *arXiv preprint arXiv:2406.18518*. NeurIPS 2024 Datasets and Benchmarks.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781, Toronto, Canada. Association for Computational Linguistics.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning BERT: Misconceptions, explanations, and strong baselines. In *International Conference on Learning Representations*.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. 2023. Gorilla: Large language model connected with massive apis. *Preprint*, arXiv:2305.15334.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, dahai li, Zhiyuan Liu, and Maosong Sun. 2024. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *The Twelfth International Conference on Learning Representations*.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, Copenhagen, Denmark. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A. Alemi, and Andrew Gordon Wilson. 2021. Does knowledge distillation really work? In *Advances in Neural Information Processing Systems*, volume 34, pages 6906–6919.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

Fusheng Wang, Jianhao Yan, Fandong Meng, and Jie Zhou. 2021. Selective knowledge distillation for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6456–6466, Online. Association for Computational Linguistics.

Fanjia Yan, Huanzhi Huang, Sida Wang, Qianli Mao, Tianjun Zhang, , and 1 others. 2024. Berkeley function calling leaderboard. *arXiv preprint arXiv:2407.01536*.

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 23 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

A Per-seed accuracy values

Method	Per-seed test accuracy (%)
`ce_only`	87.4, 85.6, 75.7, 85.6, 83.8, 77.5
`progressive_kd`	85.6, 81.1, 83.8, 79.3
`rank_kd`	83.8, 77.5, 77.5, 84.7
`full`	85.6, 75.7, 70.3, 81.1
`ce_paraphrase`	77.5, 80.2, 52.3, 85.6, 82.9
`gated_kd`	74.8, 64.9, 67.6, 88.3
`ce_Knowledge Distillation`	81.1, 82.0, 48.7, 31.5, 62.2
`reasoning_kd`	86.5, 83.8, 0.9
Controlled prompt-format comparison (Section 4.5)	
V1, names-only	85.6, 75.7, 87.4
V1, descriptions	86.5, 67.6, 86.5

Table 5: Per-seed accuracies for all multi-seed conditions. Seed assignments use the base set 42, 123, 7; additional seeds (456, 789, 1729, 2024, 2025) were added incrementally.