

Peer Review

Review of: "Fine-Tuning PHI-3 for Multiple-Choice Question Answering: Methodology, Results, and Challenges"

Evgenii Arsentev¹

1. Independent researcher

1. What the work claims

The paper takes a small open model from Microsoft's Phi-3 family, fine-tunes it with supervised fine-tuning on a preprocessed version of TruthfulQA, reworks the prompt format, and reports that multiple-choice answering improves sharply. The abstract gives two headline figures: perplexity falls from 4.68 to 2.27, and accuracy rises from 62% to 90.8%. Table 1 ("Performance vs. Model Size Comparison") places three rows side by side: GPT-3 (175B; perplexity 3.12, accuracy 85.7, F1 0.86, recall 0.83), "PHI-3 (Baseline)" (1.3B; 4.68, 78.3, 0.75, 0.73), and "PHI-3.5 (Fine-Tuned)" (1.3B; 2.27, 90.8, 0.90, 0.91). From this, the author concludes that a compact model, adapted with modest hardware (an NVIDIA GTX 1650, \$14.99), can outperform GPT-3 on MCQ answering and is a viable component for tutoring and assessment systems.

Three contributions are listed in §I: the fine-tuning study itself, "a novel approach to prompt design" that reduces hallucinations and irrelevant answers, and an evaluation with perplexity, accuracy, F1, and recall. I want to give credit where it is due. The paper reports its own failure mode openly: in §IV.C, the author describes how the first prompt format made the model "consistently choosing the last option," and says so again in §V.A. Many papers would have buried that. The hyperparameters in §IV.B are listed in unusual detail for a short paper (batch size, accumulation steps, learning rate, scheduler, warmup, optimizer, step count, seed, weight decay), and the Notes section promises both the preprocessed dataset and the code. The motivation, cheap models for educational platforms with limited compute, is a reasonable one. My concerns are about whether the numbers in Table 1 measure what the text says they measure.

2. Whether the conclusion holds on the data

2.1 The headline numbers disagree with the paper's own table. The abstract and the first paragraph of §V both say accuracy rose "from 62% to 90.8%." Table 1, row "PHI-3 (Baseline)," column Accuracy, gives 78.3, not 62. The same paragraph of §V says "F1 score increased from 66 to 90.6," whereas Table 1 gives F1 0.75 for the baseline row and 0.90 for the fine-tuned row. So there are two baselines (62 and 78.3) and two F1 pairs (66 → 90.6 and 0.75 → 0.90) in the same results section. I could not tell from the text which is correct, or whether 62% refers to a different configuration (for example, the untuned model under the original completion prompt). The gain the paper advertises is 28.8 points (90.8 – 62); the gain the table supports is 12.5 points (90.8 – 78.3). That is less than half the size.

2.2 The baseline and the fine-tuned model are not the same model. The baseline row is labelled "PHI-3," and the fine-tuned row "PHI-3.5." The abstract speaks of "PHI-3.5's MCQ handling post-fine-tuning," and §§III and IV speak of PHI-3 throughout. If the baseline was Phi-3 and the fine-tuned model was Phi-3.5, then the 12.5-point gap mixes two effects: a change of base model and the fine-tuning. The paper needs an untuned Phi-3.5 row under the same prompt; otherwise, no part of the gain can be attributed to fine-tuning.

2.3 The parameter count looks wrong. Both Phi rows in Table 1 give "Size (Billion Parameters)" = 1.3. The Phi-3 technical report that the paper cites as [1] describes phi-3-mini, the smallest member of the family, as a 3.8B-parameter model, and I am not aware of a 1.3B Phi-3 or Phi-3.5 checkpoint. Perhaps the author used a different model or copied the figure from an earlier Phi release. Either way, the exact Hugging Face checkpoint has to be named because the size column is the whole point of a table titled "Performance vs. Model Size Comparison."

2.4 I could not find a held-out test set. This is the problem I consider most serious. §I, contribution 1, says TruthfulQA is used "for training and evaluation". §III.A.1 and §IV.A describe preprocessing but never a split. No train/validation/test sizes are given anywhere. If the 90.8% was measured on questions the model was trained on, the number reflects memorization, and the drop in perplexity from 4.68 to 2.27 (Table 1, Perplexity column) is exactly what one would expect after training on those same strings. Even with a random split, TruthfulQA questions within a category are close paraphrases of one another, so a split by category would be the more honest choice. Until the split is documented, I do not think Table 1 can support the conclusion in §VI.

2.5 Positional bias may still be driving accuracy. §IV.C reports that the first prompt produced a model that always chose the last option. That is a very informative finding, and it raises an obvious question the paper does not answer: after preprocessing, where does the correct answer sit? TruthfulQA's multiple-choice targets list the correct answer first in each question's option set, and §III.A.1 says the author "retained the best correct answer for each question" and trimmed the wrong ones, but says nothing about shuffling. If options were not shuffled, or were shuffled once and never re-drawn, a model can reach high accuracy by learning a position. The revised prompt is said to be "detailed in Section 4.2" (end of §IV.C), but there is no Section 4.2 in the Qeios version; §IV has subsections A, B, and C only. So the one fix that matters for the positional problem is not shown. I would want to see (a) the distribution of correct-answer positions in train and test, (b) accuracy after random permutation of options at test time, and (c) the full final prompt.

2.6 Chance level is unknown, so 78.3 and 90.8 have no floor. §III.A.1 says the number of wrong answers was "limited" but not to how many. With two options, chance is 50%; with four, 25%. A reader cannot judge whether 78.3% is strong or weak without this, and the comparison to GPT-3 (85.7) is equally uninterpretable. The number of options per question after preprocessing should appear in §IV.A.

2.7 The GPT-3 row has no provenance. Table 1, row GPT-3, reports perplexity 3.12, accuracy 85.7, F1 0.86, and recall 0.83. The paper does not say whether these numbers were measured by the author (which API model, which prompt, which options format, when) or taken from somewhere else, and no reference is attached to the row. The original TruthfulQA paper, cited here as [12], reports results for GPT-3 on its multiple-choice task, but under its own MC1/MC2 protocol with the full option sets, not under this paper's reduced option format, so those results cannot simply be placed in this table. There is also a basic measurement problem: perplexity depends on the tokenizer and on which tokens are scored. GPT-3 and Phi models use different tokenizers, so 3.12 and 2.27 are not on a common scale, and the paper does not say whether perplexity was computed over the whole prompt, the answer span only, or the answer letter. I would drop the GPT-3 perplexity cell entirely unless it can be computed on the same token set, and either measure GPT-3 accuracy under the paper's protocol or remove the row.

2.8 F1 and recall are undefined. For single-answer multiple choice, F1 and recall need an averaging scheme (per option letter? macro? micro? per correct/incorrect label?). With micro-averaging over classes, F1 equals accuracy, which would make the separate columns redundant; with another scheme, the numbers mean something different. Table 1 gives F1 0.90 and recall 0.91 against accuracy 90.8 for the fine-tuned row, and F1 0.75, recall 0.73 against accuracy 78.3 for the baseline row. Those gaps are

informative only once the definition is stated. Precision is not reported at all, though it would be the natural companion to recall.

2.9 The training budget does not cover the dataset once. From §IV.B: per-device batch size 2, gradient accumulation 4, so an effective batch of 8, and a maximum of 60 steps. That is $60 \times 8 = 480$ training examples in total. §III.A.1 says the dataset has 1,000 questions (I note below that the TruthfulQA paper describes 817). Either way, the model sees fewer than 60% of the questions, less than one epoch, before training stops, and that is before any test split is removed. The text says 60 steps "was sufficient due to the dataset's small size". That may be true, but a loss curve or a validation point would show it; right now it is asserted.

2.10 There are no uncertainty estimates and only one run. Every number in Table 1 comes from a single training run with seed 3407. No confidence intervals, no repeated seeds, no test-set size. To show why this matters, suppose the test set were 200 items (my hypothetical, not a figure from the paper). A Wilson 95% interval around 90.8% would then span roughly 86% to 94%, which comes within half a point of GPT-3's 85.7 at its lower edge. With the true test size, the interval could be wider or narrower. I cannot compute it because the size is not given.

2.11 Hallucination reduction is claimed but not measured. §I, contribution 2, and §III.A.2 say the revised prompt reduces "hallucinations and irrelevant responses". No metric in Table 1 captures this. If the author counted invalid outputs (answers that are not one of the option letters, or free text), that rate before and after the prompt change would be a useful and easy addition. Without it, the claim goes beyond the evidence.

Taken together: the direction of the result (a small model improves on a narrow MCQ task after task-specific tuning) is plausible, and I do not doubt it. But the magnitude, the comparison to GPT-3, and the implication that the model understands MCQs better are not established by what is on the page.

3. Reproducibility

The paper is ahead of many short LLM papers on training hyperparameters. §IV.B lists batch size 2, accumulation 4, learning rate 2×10^{-4} with a linear scheduler and 5 warmup steps, FP16/BF16 selected by `is_bfloat16_supported()`, `adamw_8bit`, 60 steps, weight decay 0.01, seed 3407, and the `SFTTrainer` from TRL together with the `unsloth` library. That is enough to rerun training, given the rest. The rest is where it breaks down.

Model identity. No checkpoint name or revision is given for either Phi row, and the 1.3B size in Table 1 conflicts with the cited technical report (see 2.3). For GPT-3, there is no model name, API endpoint, date, or decoding setting. For LLM papers, this omission is almost universal, and it is present here for all three rows.

Fine-tuning method. §III.A.3 says full SFT was used and that PEFT "did not yield significant improvements", and also that quantization was applied. The hyperparameter set in §IV.B (unsloth, adamw_8bit, learning rate 2×10^{-4} , 60 steps, seed 3407) matches what is commonly used for LoRA adapters on quantized models, and a GTX 1650 is a 4 GB card, on which full-parameter training of a multi-billion-parameter model is hard to picture. I may be wrong about what was run. The paper should state plainly whether the reported model is fully fine-tuned or a LoRA/QLoRA adapter, the quantization scheme (4-bit? 8-bit? which library?), LoRA rank and target modules if applicable, and the maximum sequence length as a number (§IV.B says only that it was set to the longest sequence).

Data. The preprocessing in §III.A.1 is described in one sentence. Missing: how many wrong answers were kept, how the "best correct answer" was chosen, whether options were shuffled, the split, and its sizes. The count of 1,000 questions differs from the 817 described in the TruthfulQA paper; if extra items were added or the count is a rounding, the paper should say so.

Prompts. Only the first, abandoned prompt appears (§IV.C). The Alpaca-style prompt and the final "combined" prompt described in §III.A.2 are not shown. Since prompt design is listed as a contribution, this is a real gap.

Inference. Nothing is said about decoding at evaluation time: greedy or sampled, temperature, maximum new tokens, and how a generated string was mapped to an option (exact letter match, first token, log-likelihood over options). These choices can move MCQ accuracy by several points.

Code and data links. The Notes section says the dataset and code are available "in here," but as rendered on Qeios, the hyperlinks did not come through in the text I reviewed, so I could not check what the repository contains. Figures 1 and 2 are likewise not legible in the version I read, so I have not commented on the pipeline diagram.

Compute. The GPU is stated. Training time, peak memory, and the CUDA workaround mentioned in §III.A.3 ("which we overcame by modifying the training code") are not. The title promises "Challenges," and this workaround is the one engineering challenge mentioned; readers with the same hardware would value the details.

4. What specifically to fix

1. **Abstract and §V, first paragraph.** Reconcile the baseline accuracy (62% in the text vs 78.3 in Table 1, row "PHI-3 (Baseline)") and the F1 figures ("66 to 90.6" in the text vs 0.75 and 0.90 in Table 1). If 62% belongs to a different configuration, add that configuration as its own row.
2. **§IV.A.** Add the train/validation/test split, with sizes, and say how it was drawn (random, stratified, by category). Report every Table 1 number on the held-out portion only.
3. **Table 1, "Size" column and model names.** Give the exact checkpoint for each row and correct the parameter count.
4. Add an untuned Phi-3.5 row, evaluated with the final prompt, so that base-model change and fine-tuning can be separated (see 2.2).
5. **§IV.C, reference to "Section 4.2".** This section does not exist. Either add it, with the full Alpaca-style prompt and the final combined prompt verbatim, or fix the cross-reference.
6. **§III.A.1.** State the number of options per question after preprocessing, the rule for choosing the "best correct answer," and whether and how options were shuffled. Report the chance-level accuracy.
7. Run a permutation test for positional bias: re-evaluate the fine-tuned model with options randomly reordered (several draws) and report accuracy for each, plus the distribution of correct-answer positions in train and test. Given the finding in §IV.C, this is the single most persuasive experiment the paper could add.
8. **Table 1, row GPT-3.** Either document how these four numbers were obtained (model, date, prompt, option format) or remove the row. Remove the perplexity cell in any case unless it is computed on an identical token set; cross-tokenizer perplexity is not comparable.
9. **§V and Table 1 header.** Define F1 and recall (averaging scheme, what the positive class is), define perplexity (over which tokens), and add precision.
10. **§III.A.3 and §IV.B.** Say whether the final model is full SFT or LoRA/QLoRA, give the quantization scheme, LoRA settings if used, and the numeric max sequence length. Explain what the CUDA fix was.
11. Show the arithmetic of the training budget ($60 \times 2 \times 4 = 480$ examples) next to the training-set size and add a training/validation loss curve to back the claim that 60 steps was enough.
12. Report results over at least three seeds, with mean and spread, and a 95% interval for each accuracy figure.

13. **SV, evaluation protocol (new paragraph).** Describe decoding at test time and the rule mapping generated text to an option. Report the rate of invalid or off-option outputs before and after the prompt change; that is the measurable version of the hallucination claim in SI, contribution 2.
14. **SIID and the abstract.** TruthfulQA is cited as [1], which in the reference list is the Phi-3 technical report; the TruthfulQA paper is [12]. The abstract also attaches [1] to PHI-3, which is correct there, so only the SIID citation needs fixing. SVI ends with a stray citation [12].
15. **SIIL.A.1 vs TruthfulQA.** Explain the count of 1,000 MCQs, since the source paper describes 817 questions.
16. Soften SIIE ("remains largely unexplored") and SVI. Fine-tuning small models for multiple-choice QA has been studied before; the paper's contribution is a concrete low-resource recipe, and it is stronger when framed that way.
17. Restore the links in the Notes section and pin a commit or release, so that readers can match code to results.

5. Recommendation

Major revisions. The paper is honest about the positional bias it ran into, lists its training hyperparameters in useful detail, and addresses a practical question for low-resource education settings. I would like to see it succeed. As it stands, though, the abstract and Table 1 give different baselines, the baseline and tuned rows appear to use different base models, the parameter count conflicts with the cited source, and I could not find a held-out test set. Any one of these could change the headline result. None of them is hard to fix, and a revised version with a documented split, an untuned Phi-3.5 row, and a shuffled-option test would be a small but genuinely useful study.

Declarations

Potential competing interests: No potential competing interests to declare.