Fine-Tune on the Format First: Improving Multiple-Choice Evaluation for Intermediate LLM Checkpoints

Alec Bunn[♣] **Sarah Wiegreffe**[*†] **Ben Bogin**[*]
[♣]University of Washington [†]Allen Institute for AI (Ai2)
abunn2@uw.edu

Abstract

Evaluation of intermediate language model checkpoints during training is critical for effective model development and selection. However, reliable evaluation using the popular multiple-choice question (MCQ) format is challenging, as small and non instruction-tuned models often lack the symbolic reasoning required for the task. This is despite the fact that MCQ evaluation is often used and needed to distinguish between the performance of different training runs. In particular, when prompted with a question and a set of labeled answer choices (e.g., "A. . . . , B. . . . , C. . . ."), many models struggle to emit the correct label (e.g., "C"), even when they can select the correct *string* answer choice. We propose an alternative evaluation method: fine-tuning the model on an auxiliary MCQ dataset prior to outputting labels. We validate this approach empirically by showing that training on auxiliary data improves MCQ ability on all our test datasets except 1. This approach provides a more accurate signal of model capability at intermediate checkpoints, as it disentangles the evaluation of core knowledge from the model's emerging ability to follow formatting instructions.

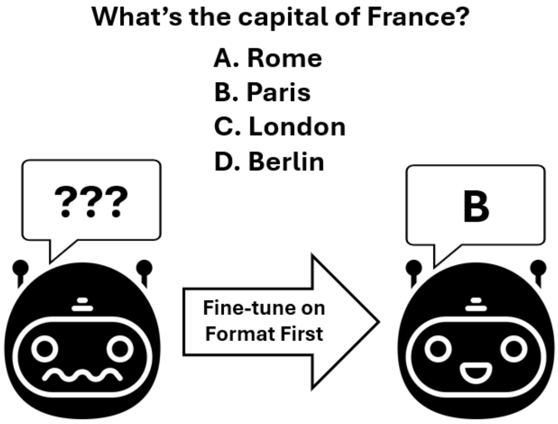

Figure 1: Many intermediate models do not understand MCQ format and may fail to provide a valid answer to this question (left). We propose fine-tuning on the MCQ format prior to evaluation so that the fine-tuned model (right) learns to output the correct label ('B'). This allows for a more robust test of its underlying skills. We demonstrate this improves MCQ evaluation reliably.

1 Introduction

Robust and accurate evaluation of Large Language Models (LLMs) is crucial for their development, guiding the design decisions model developers make when selecting from different model candidates. More specifically, it is common practice to evaluate intermediate model checkpoints over the course of a training run to estimate the final model's abilities before training is completed (Biderman et al., 2023; Liu et al., 2023; OLMo et al., 2024; Snell et al., 2024, *i.a.*). Therefore, it is important to have robust ways to evaluate these intermediate checkpoints. However, intermediate model checkpoints are significantly harder to evaluate consistently than the final model, given that they often do not possess prerequisite skills, for example, in-context learning, instruction following, or chain-of-thought reasoning. This makes it difficult to distinguish between the performance of different training runs or assess true model capability.

One common LM evaluation format is that of Multiple Choice Questions (MCQs; Rogers et al., 2023), which are easy to automatically score due to the existence of one pre-specified correct answer. In this format, the model is asked a question, given different answer choices, and must select the correct answer from the provided choices (Figure 1). This format thus avoids the pitfalls of evaluating the correctness of open-ended LLM-generated text.

However, it often proves challenging in practice to evaluate intermediate model checkpoints on MCQs, because learning to answer MCQs is a skill

[*] Co-authors. Work done at Ai2.

Proceedings of the Fourth Workshop on Generation, Evaluation and Metrics (GEM² 2025), pages 511–521
July 31 - August 1, 2025 ©2025 Association for Computational Linguistics

in-and-of-itself that must also be learned during pretraining. More specifically, the ability to map a predicted answer choice string (e.g., "Paris") to its respective symbol (e.g., "B") and then generate that symbol, known as "symbol binding" (Robinson and Wingate, 2023), is learned only after some number of pretraining steps (Wiegreffe et al., 2025). In light of this issue, how can we best standardize model evaluation across checkpoints of varying instruction-following abilities? Prior work has proposed to evaluate each checkpoint with multiple formats and take the maximal score (Gu et al., 2025), but this approach both requires double the number of evaluations and adds complexity to results by introducing a format confounder.

We investigate an alternative approach to evaluating intermediate model checkpoints on MCQ datasets: fine-tune each checkpoint on an auxiliary MCQ dataset (potentially from a different task) to teach the evaluation format, and then evaluate on the target dataset. This method gives the model explicit exposure to the multiple-choice format prior to evaluation, improving its ability to follow the format. This approach can thus give an arguably better estimate of a model's true capability on a given skill or domain, mitigating issues such as all answer choices being assigned low probability (Holtzman et al., 2021), answers differing based on evaluation format (Wiegreffe et al., 2023; Lyu et al., 2024), or preambling (Wang et al., 2024b).

In this work, we address the following research questions: (1) Can an intermediate model effectively acquire the ability to follow the MCQ format through fine-tuning? (2) Does this format learning on an auxiliary dataset transfer to improved performance on other, unseen MCQ datasets? (3) How does the model's final evaluation accuracy scale with the number of auxiliary training examples?

Our empirical studies reveal three key findings. First, we demonstrate that intermediate models can effectively acquire the MCQ format through auxiliary fine-tuning, and that this capability transfers across datasets. Second, using a more diverse auxiliary dataset leads to stronger performance on the target task. Finally, we find that model accuracy on the target dataset increases with the number of auxiliary training examples. Taken together, these findings provide a practical methodology for more reliably evaluating and comparing intermediate language models on MCQ tasks.

2 Current Evaluation Methodology for Multiple Choice Questions

There are two primary methodologies for evaluating models on MCQ datasets: label-based and sequence-based formatting, with examples of each shown in Figure 2. In this context, the word "format" refers to both the prompt structure and the model's expected answer. **Label-based** formatting assigns a symbol, such as A, B, C, or D, to each choice. The symbol with the highest probability is selected as the model's prediction. **Sequence-based** formatting, by contrast, calculates which answer string the model is most likely to generate. The answer string with the highest probability is then selected as the model's prediction.

2.1 Label-Based Formatting

Formally, let a question x be presented with a set of n choice-symbol pairs, $\{(s_1, c_1), \ldots, (s_n, c_n)\}$, where choices c_i are from a set C and are uniquely paired with symbols s_i from a set S. The correct answer choice, $y \in C$, corresponds to a target symbol $s^* \in S$. The goal in this format is to correctly predict the symbol s^*.

Let M be a model parameterized by θ that defines a probability distribution over a vocabulary of tokens T, where $S \subset T$. The model's prediction, $\hat{s}$, is found by selecting the symbol in S with the highest conditional probability:

$$\hat{s} = \arg\max_{s \in S} P_\theta(s|x, \{(s_1, c_1), \ldots, (s_n, c_n)\}) \tag{1}$$

The model's prediction for a given question is considered correct if the predicted symbol $\hat{s}$ matches the target symbol s^*.

2.2 Sequence-based Formatting

In sequence-based formatting, given a question x and a set of choices C, the goal is to identify the correct choice, $y \in C$. This is done by calculating the model's likelihood of generating the full text of each choice.

Using a model M parameterized by θ, the prediction is the choice $c \in C$ that the model assigns the highest conditional probability to:

$$\hat{y} = \arg\max_{c \in C} P_\theta(c|x) \tag{2}$$

A prediction is correct if $\hat{y} = y$. We do not normalize these probabilities by length, because it does not consistently improve performance (Liang et al., 2022; Biderman et al., 2024; Gu et al., 2025).

Label-based format:

Question: What home entertainment equipment requires cable?

A. radio shack   B. substation

C. cabinet   D. television

E. desk

Answer:

A, B, C, **D**, E

Sequence-based format:

Question: What home entertainment equipment requires cable?

Answer:

radio shack   substation   cabinet   **television**   desk

Figure 2: Comparison of Label-based and Sequence-based MCQ Formats.

3 Difficulties with MCQ Evaluation

Evaluating language models on multiple-choice questions presents several challenges, with distinct problems arising from both of the primary evaluation formats.

3.1 Problems with Label-Based Formatting

A primary challenge with label-based formatting is label bias, where models exhibit a strong preference for certain labels (e.g., "A") regardless of the question's content (Zheng et al., 2024; Pezeshkpour and Hruschka, 2024; Alzahrani et al., 2024; Wang et al., 2024b). This bias can stem from the higher base frequency of certain tokens in the pretraining corpus or from primacy effects related to the ordering of the choices. Another challenge is the tendency of models, particularly instruction-tuned ones, to generate conversational preambles (e.g., "Yes, I can answer that question, my answer is...") before their answer (Wang et al., 2024b). Forcing a model to produce an immediate single-token response can alter its prediction compared to when it is allowed to generate a preamble first.

Beyond these general issues, label-based evaluation is especially problematic for intermediate model checkpoints. These models often lack the fundamental ability to follow the MCQ format, causing them to fail even on simple questions. This difficulty arises because symbol binding—the process of mapping a semantic choice to an arbitrary symbol—is a non-trivial skill that models must acquire through training. Due to this limitation, researchers evaluating intermediate checkpoints often resort to using sequence-based formatting instead. However, as the next section details, this alternative has its own significant drawbacks.

3.2 Problems with Sequence-Based Formatting

While avoiding the symbol-binding problem, sequence-based formatting introduces its own significant challenges, the most prominent of which is Surface Form Competition (Holtzman et al., 2021). This phenomenon occurs when a model's probability mass is split across many synonymous or similarly phrased expressions, effectively "stealing" probability from the correct answer choice. For instance, consider a model tasked with completing the sentence, "After his model overfit the data, Adam was ___." If the correct choice is "disheartened," the model may still assign a higher probability to a more common synonym like "disappointed," even if that word is not among the provided choices. This can cause the model to select a common but incorrect option (e.g., "bored") over the correct but less frequent one ("disheartened").

This issue becomes more pronounced for multi-token answers where minor variations in phrasing can dilute the probability of the correct sequence. Furthermore, the method is susceptible to length bias, where models may inherently favor shorter or longer answer choices, though this can be partially mitigated through normalization techniques (Holtzman et al., 2021). The format is also ill-suited for questions that use referential answers, such as "all of the above," as each choice is evaluated in isolation.

Finally, sequence-based formatting is computationally expensive. It requires a separate forward pass of the model for each answer choice to calculate its probability, whereas label-based methods require only a single pass per question. Due to these collective drawbacks, label-based formatting is often the preferred and more robust method for evaluating final, well-tuned models.

4 Auxiliary Format Fine-Tuning

To address the challenges of standard MCQ evaluation, we propose and investigate a two-stage methodology. First, an intermediate model checkpoint is briefly fine-tuned on an auxiliary MCQ dataset. During this stage, the model is trained exclusively on the label-based format: given a question and choices mapped to symbols, it learns to output the single token for the correct answer. Second, this newly fine-tuned model is evaluated on the target MCQ dataset using the same label-based format.

This approach is designed to disentangle a model's underlying knowledge from its ability to follow a specific format, thereby mitigating issues from both standard evaluation techniques. The fine-tuning stage explicitly teaches the skill of symbol binding, addressing the primary failure point for intermediate models in standard label-based evaluation. This process also targets format-specific artifacts; because the correct symbol's identity and position are varied across training examples, inherent label bias is reduced. Similarly, training the model to maximize the first-token probability of the correct symbol inherently penalizes the generation of any preamble.

Crucially, our method retains the primary strengths of the original formats. After the one-time fine-tuning, evaluation remains computationally efficient, requiring only a single forward pass per question. By using label-based prediction, it also completely avoids the problem of Surface Form Competition inherent to sequence-based evaluation.

5 Experimental Setup

5.1 Data

To assess the generalization of format understanding across diverse domains, we use a variety of natural and synthetic MCQ datasets. The number of answer choices per question is denoted by N.

Auxiliary Fine-Tuning Sets To test cross-domain generalization, we use two distinct datasets for auxiliary fine-tuning: SciQ (N=4; Welbl et al., 2017), a science question-answering dataset with supporting passages which has 11,679 questions in the trainset, and SWAG (N=4; Zellers et al., 2018), which focuses on commonsense reasoning and has 73,546 questions in the trainset. We experiment with fine-tuning on each individually and on a 50/50 mixture.

Evaluation Sets Our evaluation suite includes the test sets of SciQ and SWAG, as well as several other benchmarks: ARC-Easy (N=3–5; Clark et al., 2018), HellaSwag (N=4; Zellers et al., 2019), OpenBookQA (N=4; Mihaylov et al., 2018), PIQA (N=2; Bisk et al., 2019), and SocialIQA (N=3; Sap et al., 2019). To isolate format-following ability, we also include the synthetic dataset CopyColors (N=2, 4, 10; Wiegreffe et al., 2025).

For all datasets, evaluations are run on a randomly sampled subset of 1,000 test examples due to compute constraints.

5.2 Model

We use the OLMo-1B model (Groeneveld et al., 2024), trained for 1T tokens (400,000 steps) on the Dolma 1.6 dataset (Soldaini et al., 2024). For our analysis, we select 10 evenly spaced checkpoints from this pretraining run, corresponding to every 40,000 steps.

5.3 Baselines

We compare our method against several baselines that do not require fine-tuning. We report performance using both the standard label-based and sequence-based formats. We also include a 3-shot label-based baseline, where each prompt is conditioned on three in-context examples to provide format exposure without updating model weights; this serves as a conceptual parallel to our fine-tuning method. Finally, to establish a performance lower-bound, we report a random chance baseline, calculated as the average reciprocal of the number of choices per question.

5.4 Fine-tuning

To apply our proposed evaluation procedure, we fine-tune each model checkpoint on an auxiliary dataset (SciQ, SWAG, or a 50/50 mixture). For these main experiments, we use a fixed training run of 1,000 steps with a batch size of 32 so 32,000 training instances total and a learning rate of 10^{-6} on a linear decay schedule. Afterward, each fine-tuned checkpoint is evaluated on the target datasets using label-based formatting.

In a separate experiment to analyze the effect of data scale, we fine-tune the model on 10 subsets of

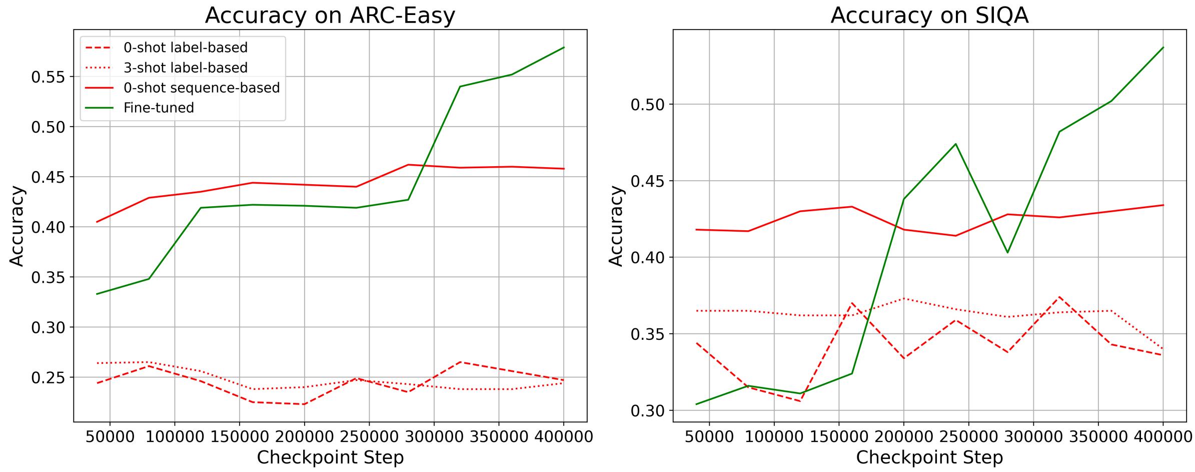

Figure 3: Accuracy using various evaluation methods across varying checkpoints. In this case the "Fine-tuned" metric used the mixture of both SciQ and SWAG.

SWAG, with sizes ranging from 10 to 50,000 examples (spaced log-linearly). For these runs, we use a fixed training budget of 1,562 steps to ensure a fair comparison across the different data sizes. We bump up the number of steps since when training on all 50,000 examples with a batch size of 32 we can do one full epoch (i.e. $32 \times 1562 \approx 50,000$). For the runs with less data, we keep iterating through them for 1,562 steps.

6 Results

6.1 Can Intermediate Model Checkpoints Learn the Label-Based Format?

Our primary result demonstrates that auxiliary fine-tuning provides a clear signal of model improvement over the course of pretraining. As shown for ARC-Easy and SIQA in Figure 3, our proposed method of fine-tuning in this case on both SciQ and SWAG (solid green line) is the only metric that reveals a consistent, monotonic increase in performance across the 10 model checkpoints. In contrast, the baseline metrics—zero-shot label-based, few-shot label-based, and zero-shot sequence-based—remain largely flat or noisy, showing little correlation with training progress. This indicates that standard evaluation methods fail to reliably distinguish between weaker and stronger intermediate checkpoints, whereas our approach effectively captures model improvement. Accuracy graphs for all evaluation datasets are in Appendix A.

This performance advantage generalizes across a wide range of domains, as shown by the results from the final model checkpoint in Table 1. Our fine-tuning approach consistently yields higher accuracy scores than all baselines, even on datasets topically dissimilar to the SciQ and SWAG auxiliary sets. This suggests that standard methods underestimate a model's latent knowledge when the model has not been explicitly exposed to the evaluation format.

The synthetic CopyColors dataset isolates this format-following ability in a controlled setting. On CopyColors-4 (four choices), the fine-tuned model achieves near-perfect accuracy, confirming it has learned the symbol-binding task. However, performance drops substantially on CopyColors-10 (ten choices), indicating that the generalization is limited when the number of choices deviates significantly from the training condition (N=4).

6.2 Effect of Auxiliary Data Diversity

To assess the importance of diversity in the auxiliary set, we compare fine-tuning the final OLMo checkpoint on a single dataset (either SciQ or SWAG) versus a 50/50 mixture of both. The results in Table 1 show that fine-tuning on the mixed dataset yields more robust and consistent performance. While the mixed-data approach is not always the top scorer on every individual dataset, it avoids the significant performance degradation sometimes observed when using a single, more specialized auxiliary set. This highlights the importance of a diverse auxiliary dataset for achieving broad generalization.

We also observe strong in-domain generalization effects. For instance, fine-tuning on SciQ leads to strong performance on ARC-Easy, likely

Method	SciQ	SWG	ARC	CSQA	HSWG	OBQA	PIQA	SIQA	CC2	CC4	CC10
Random	25.0	25.0	25.0	20.0	25.0	25.0	50.0	33.3	50.0	25.0	10.0
0-shot	24.6	23.8	24.7	20.9	24.6	27.5	51.6	33.6	48.0	31.0	9.0
3-shot	26.6	26.7	24.4	21.8	23.8	28.8	46.9	34.0	52.0	22.0	12.0
Seq	58.6	37.9	46.1	33.9	39.8	22.5	**73.4**	41.0	97.0	97.0	97.0
Both	95.1	77.0	57.9	49.0	51.0	37.6	62.4	**53.7**	**100.0**	**100.0**	85.0
SciQ	**95.5**	44.3	**58.3**	**49.1**	33.5	**40.7**	52.2	52.6	**100.0**	**100.0**	**99.0**
SWG	50.5	**81.2**	35.3	35.2	**52.8**	29.6	57.3	45.6	60.0	67.0	15.0

Table 1: Performance of final checkpoint across test datasets. Methods include baselines (top four rows) and models fine-tuned on training data from SciQ, SWAG (SWG), or both (bottom three rows). CC=CopyColors with 2, 4, or 10 answer choices.

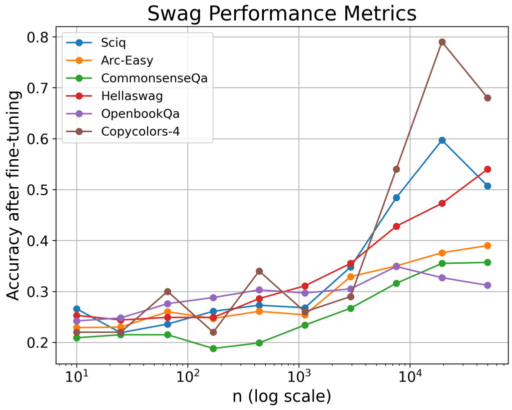

Figure 4: Performance of the final model checkpoint on test sets when trained on differing amounts of SWAG training examples.

due to their shared focus on scientific question-answering. The structural similarity of providing contextual passages also appears to aid transfer to SocialIQA and CopyColors. In contrast, the context-free, short-form reasoning of SWAG transfers most effectively to similarly structured datasets like HellaSwag and PIQA.

6.3 Effect of Auxiliary Data Size

To assess the impact of auxiliary data size on evaluation performance, we conducted additional experiments using subsets of SWAG. We varied the training set size from 10 to approximately 50,000 examples. As shown in Figure 4, performance improves consistently with more data, up to the maximum tested size. These results suggest that larger auxiliary datasets are beneficial, although further work is needed to determine where performance plateaus.

7 Related Work

While MCQs are commonly used to evaluate LLMs due to their simplicity and efficiency (Robinson and Wingate, 2023; Wang et al., 2024a), the reliability of these evaluation methods is disputed. Prior work has identified many issues with MCQ evaluation. For instance, there seem to be inconsistent results when comparing probability-based scoring (which encompasses both sequence-based and label-based formatting) and generation-based scoring (Tsvilodub et al., 2024; Lyu et al., 2024). Additionally, Holtzman et al. (2021) demonstrates that surface form competition can cause sequence-based formatting to underrepresent model ability significantly. Many authors have also pointed out that option order has a large effect in label-based formatting (Zheng et al., 2024; Pezeshkpour and Hruschka, 2024; Alzahrani et al., 2024; Wang et al., 2024b).

Efforts to improve MCQ robustness have focused on mitigating biases in scoring methods. For example, Zheng et al. (2024) proposed addressing position bias by finding the prior probabilities that the LLM would place on each position, while Holtzman et al. (2021) addresses surface form competition by reweighting answer likelihoods. However, the efficacy of such methods remains inconsistent: Wiegreffe et al. (2023) demonstrates that increasing probability mass on answer choices can paradoxically harm accuracy for certain LLMs. While some studies advocate for task-specific calibration (Pezeshkpour and Hruschka, 2024; Wang et al., 2024a), others caution against these methods of correcting for biases since they may not generalize across models or datasets (Li et al., 2024; Tsvilodub et al., 2024).

Our method of fine-tuning a model to follow a specific format is conceptually related to instruction tuning (Weller et al., 2020; Mishra et al., 2022), where a pretrained model is further trained on a collection of instructions and desired responses. However, a key distinction lies in the goal and application. Instruction tuning is typically a large-scale, final training stage meant to create a general-purpose, obedient model. In contrast, our method is a lightweight, targeted fine-tuning step designed specifically as a pre-evaluation probe to assess the knowledge of *intermediate* checkpoints. It is therefore a tool for evaluation rather than a final step in model creation.

Most similar to our work is Snell et al. (2024), who also finetune intermediate model checkpoints and evaluate performance as a means to predict when and whether certain "emergent" skills will be learned, some of which are instantiated as MCQA datasets. However, their goal is not to predict the success of any particular training run or standardize evaluation format, but rather to predict scaling laws for emergent behaviors.

8 Conclusion

In this paper, we address issues with evaluating intermediate LLM checkpoints on MCQ-style datasets. Standard evaluation methods such as sequence-based and label-based formatting have significant issues that make them ill-suited candidates for evaluation. Scoring with label-based formatting is impossible when the model does not have the capability to symbol bind, and sequence-based formatting suffers from Surface Form Competition as well as numerous other issues. To mitigate these problems, we propose fine-tuning on an auxiliary MCQ dataset followed by scoring with label-based formatting on the target datasets. This allows models to explicitly learn the MCQ format while reducing bias and improving robustness.

The empirical results we present in this paper demonstrate that this fine-tuning approach shows significant promise to improve evaluation consistency for intermediate model checkpoints. Furthermore, we show that not much data is actually required to make significant improvements to label-based formatted evaluation. We also demonstrate that this method provides a better metric to distinguish model ability in intermediate model checkpoints. We believe that this is a promising direction that requires further study.

References

Norah Alzahrani, Hisham Alyahya, Yazeed Alnumay, Sultan AlRashed, Shaykhah Alsubaie, Yousef Almushayqih, Faisal Mirza, Nouf Alotaibi, Nora Al-Twairesh, Areeb Alowisheq, M Saiful Bari, and Haidar Khan. 2024. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13787–13805, Bangkok, Thailand. Association for Computational Linguistics.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 2397–2430. PMLR.

Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, and 1 others. 2024. Lessons from the trenches on reproducible evaluation of language models. ArXiv preprint arXiv:2405.14782.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. Piqa: Reasoning about physical commonsense in natural language. In *AAAI Conference on Artificial Intelligence*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, and 1 others. 2024. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809, Bangkok, Thailand. Association for Computational Linguistics.

Yuling Gu, Oyvind Tafjord, Bailey Kuehl, Dany Haddad, Jesse Dodge, and Hannaneh Hajishirzi. 2025. OLMES: A standard for language model evaluations. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5005–5033, Albuquerque, New Mexico. Association for Computational Linguistics.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference* *on Empirical Methods in Natural Language Processing*, pages 7038–7051, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wangyue Li, Liangzhi Li, Tong Xiang, Xiao Liu, Wei Deng, and Noa Garcia. 2024. Can multiple-choice questions really be useful in detecting the abilities of LLMs? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2819–2834, Torino, Italia. ELRA and ICCL.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. ArXiv:2211.09110.

Zhengzhong Liu, Aurick Qiao, Willie Neiswanger, Hongyi Wang, Bowen Tan, Tianhua Tao, Junbo Li, Yuqi Wang, Suqi Sun, Omkar Pangarkar, and 1 others. 2023. Llm360: Towards fully transparent open-source llms. ArXiv preprint arXiv:2312.06550.

Chenyang Lyu, Minghao Wu, and Alham Aji. 2024. Beyond probabilities: Unveiling the misalignment in evaluating large language models. In *Proceedings of the 1st Workshop on Towards Knowledgeable Language Models (KnowLLM 2024)*, pages 109–131, Bangkok, Thailand. Association for Computational Linguistics.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, and 1 others. 2024. 2 olmo 2 furious. ArXiv preprint arXiv:2501.00656.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017, Mexico City, Mexico. Association for Computational Linguistics.

Joshua Robinson and David Wingate. 2023. Leveraging large language models for multiple choice question answering. In *The Eleventh International Conference on Learning Representations*.

Anna Rogers, Matt Gardner, and Isabelle Augenstein. 2023. Qa dataset explosion: A taxonomy of nlp resources for question answering and reading comprehension. *ACM Computing Surveys*, 55(10):1–45.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

Charlie Victor Snell, Eric Wallace, Dan Klein, and Sergey Levine. 2024. Predicting emergent capabilities by finetuning. In *First Conference on Language Modeling*.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, and 17 others. 2024. Dolma: an open corpus of three trillion tokens for language model pretraining research. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15725–15788, Bangkok, Thailand. Association for Computational Linguistics.

Polina Tsvilodub, Hening Wang, Sharon Grosch, and Michael Franke. 2024. Predictions from language models for multiple-choice tasks are not robust under variation of scoring methods. *Preprint*, arXiv:2403.00998.

Xinpeng Wang, Chengzhi Hu, Bolei Ma, Paul Rottger, and Barbara Plank. 2024a. Look at the text: Instruction-tuned language models are more robust multiple choice selectors than you think. In *First Conference on Language Modeling*.

Xinpeng Wang, Bolei Ma, Chengzhi Hu, Leon Weber-Genzel, Paul Röttger, Frauke Kreuter, Dirk Hovy, and Barbara Plank. 2024b. "my answer is C": First-token probabilities do not match text answers in instruction-tuned language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7407–7416, Bangkok, Thailand. Association for Computational Linguistics.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

Orion Weller, Nicholas Lourie, Matt Gardner, and Matthew E. Peters. 2020. Learning from task descriptions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1361–1375, Online. Association for Computational Linguistics.

Sarah Wiegreffe, Matthew Finlayson, Oyvind Tafjord, Peter Clark, and Ashish Sabharwal. 2023. Increasing probability mass on answer choices does not always improve accuracy. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8392–8417, Singapore. Association for Computational Linguistics.

Sarah Wiegreffe, Oyvind Tafjord, Yonatan Belinkov, Hannaneh Hajishirzi, and Ashish Sabharwal. 2025. Answer, assemble, ace: Understanding how LMs answer multiple choice questions. In *The Thirteenth International Conference on Learning Representations*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*.

A Additional Results

In this section, we show the results of the different evaluation methodologies for all datasets across the checkpoints. These are shown in Figure 5, which broadly line up with the rest of the results discussed throughout this paper.

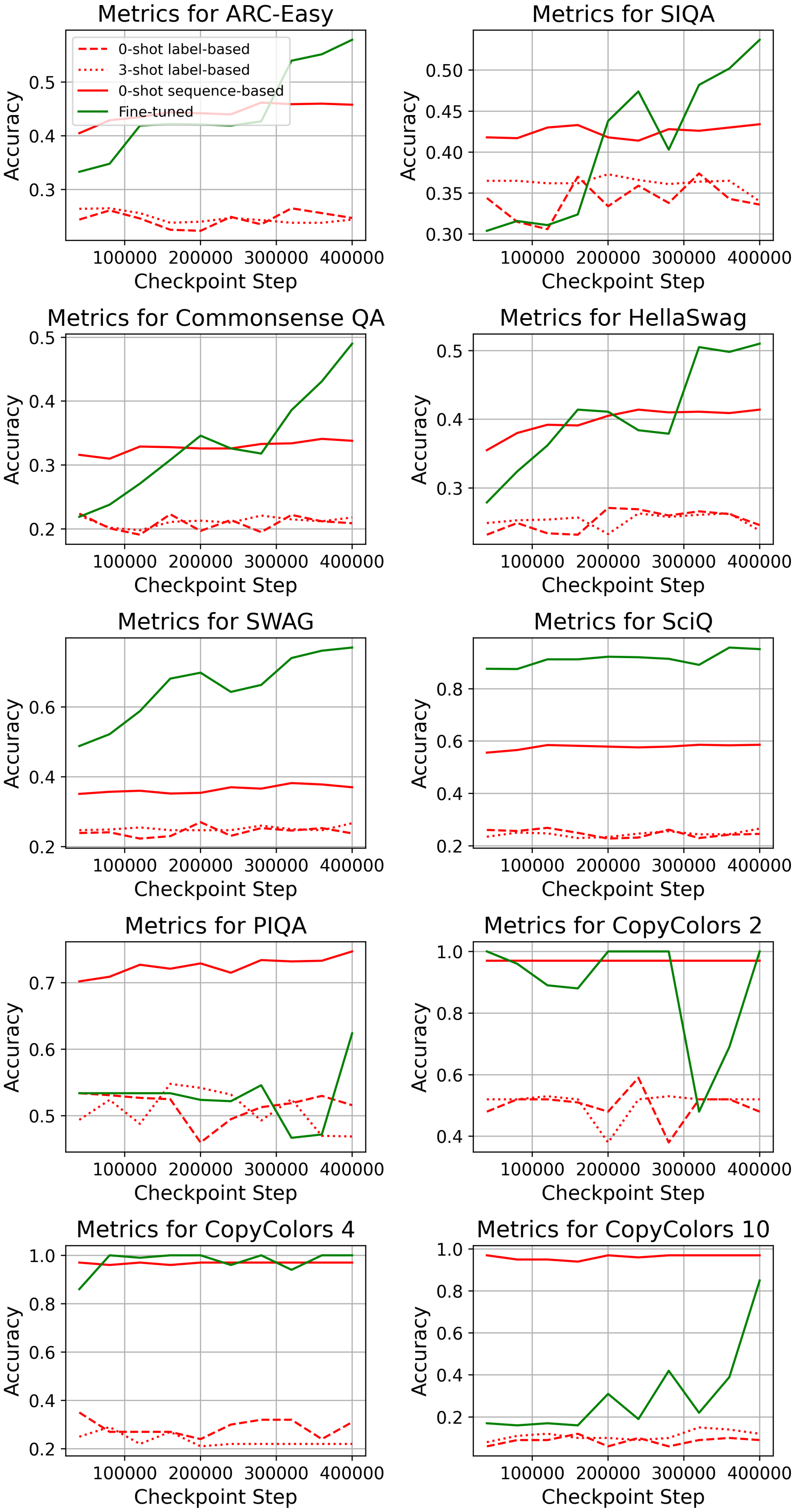

Figure 5: Results across all datasets and checkpoints for different evaluation methods. These were again fine-tuned on both SciQ and SWAG