An analysis of Answer Inconsistency in Language Models under Input Format Changes: An Approach via Hidden State Analysis

Junhyeok Park
Pusan National University
Busan, South Korea
eppi001004@gmail.com

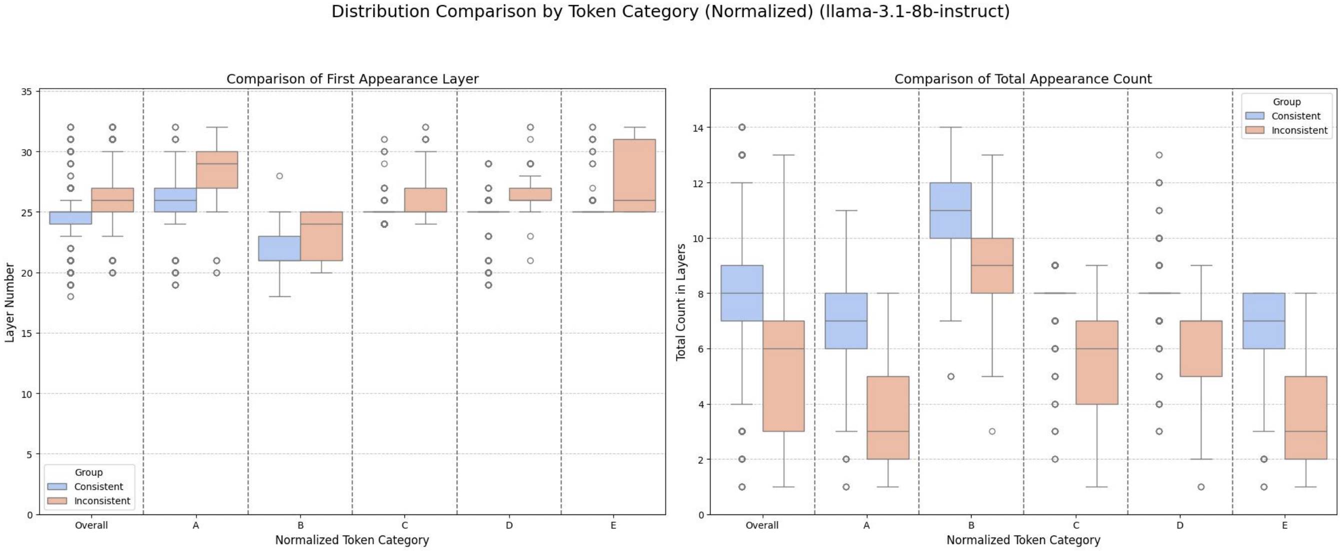

Figure 1: Comparison of answer emergence in the Llama-3.1-8B-Instruct model between Consistent and Inconsistent cases. The figure shows the distribution of the first layer where the final answer emerges (left) and the total number of layers it persists as the top prediction (right).

Abstract

The fundamental reliability of Large Language Models (LLMs) is challenged by inconsistency, where minor input perturbations lead to different outputs. To understand this phenomenon, we employ a Multiple-Choice Question Answering (MCQA) dataset to conduct a mechanistic analysis of model failures. By applying vocabulary projection to trace the internal decision-making process, we uncover the internal dynamics that drive unreliable model behavior. Our analysis reveals that inconsistent predictions are characterized by a prolonged state of uncertainty, with the final decision emerging in the later layers of the model. This contrasts with the early and stable prediction observed in consistent cases, suggesting that inconsistency is a failure to stabilize a prediction during the early processing stages.

1 Introduction

Ensuring the fundamental reliability of Large Language Models (LLMs) is a critical challenge for their widespread and safe application. A major challenge to this reliability is the phenomenon of inconsistency, where minor, semantically irrelevant perturbations in an input prompt can cause the model to produce different outputs. This instability is particularly evident and problematic in the Multiple-Choice Question Answering (MCQA) format (Zheng et al. [5]). The dominance of MCQA in major evaluation benchmarks implies that such failures have broad implications for model capabilities. More importantly, the shift from one option to another provides a clear, unambiguous signal of failure, making MCQA the ideal testbed for moving beyond observing inconsistencies to investigating their underlying mechanisms.

To mechanistically investigate this problem, we adapt tools from a critical area of research focused on analyzing model internals. A powerful line of work has employed tools like Vocabulary Projection to analyze the internal mechanisms of *successful* predictions (Wiegreffe et al. [4]). Our primary contribution is to apply this diagnostic tool to the problem of *failure* within the controlled MCQA environment. The fixed set of candidate answers in MCQA enables the application of these methods, allowing us to mechanistically trace why a model's prediction shifts.

While the existence of inconsistency is established, the specific internal dynamics that differentiate a stable prediction from an unstable one remain underexplored. Our work aims to fill this critical gap by investigating the internal mechanisms of failure.

The research code for this paper can be found at https://github.com/JakeFRCSE/KAIRI_2025_Summer.

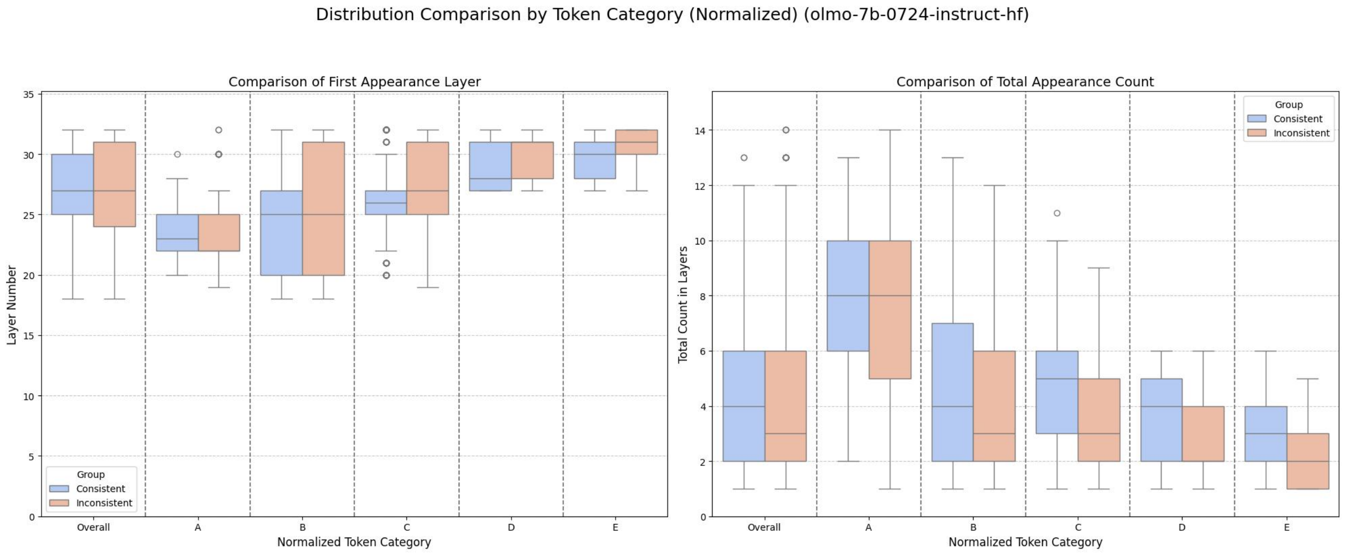

Figure 2: Comparison of answer emergence in the OLMo-7B-0724-Instruct model between Consistent and Inconsistent cases. The figure shows the distribution of the first layer where the final answer emerges (left) and the total number of layers it persists as the top prediction (right).

Therefore, to answer the following central research questions, we employ the Vocabulary Projection technique to trace the layer-by-layer evolution of the model's prediction for each candidate answer token:

(1) *Can we identify specific processing stages (i.e., layers) where the mechanism for a consistent answer significantly diverges from that of a inconsistent one?*
(2) *What qualitative changes occur in an LLM's internal decision-making process when its prediction becomes inconsistent due to a minor format change?*

2 Related Work

This study is situated within the recent line of work concerning the fundamental reliability of Large Language Models (LLMs). While traditional benchmarks have focused on aggregate performance, our work aligns with a growing body of research that scrutinizes the consistency and trustworthiness of model behavior. This section contextualizes our research by reviewing three key areas. First, we discuss the specific evaluation benchmark chosen to test the limits of a model's capabilities. Second, we frame our work within the paradigm shift from evaluating performance to probing reliability. Finally, we review the mechanistic analysis techniques that enable us to investigate the internal dynamics that lead to unreliable model behavior.

2.1 A Testbed for Probing Decision-Making Failures

For the sensitivity analysis in this study, we selected CommonsenseQA (Talmor et al. [3]) as the experimental dataset. The goal of CommonsenseQA is to evaluate commonsense inference capabilities, which requires understanding the relationships between concepts, rather than simple factual recall. A key feature of this dataset is its inclusion of plausible distractors—incorrect options that are designed to be semantically similar to the correct answer. This structure demands that the model not only identify the correct answer but also precisely distinguish between subtle semantic differences. The characteristics of this dataset provide an ideal environment for observing the fragility of a model's decision-making process to external perturbations like format changes. Therefore, we utilize this dataset to analyze how such minor variations can disrupt the model's decision-making process and lead to prediction instability.

2.2 From Performance to Reliability in MCQA

Traditionally, LLM evaluation, including the aforementioned CommonsenseQA benchmark, has focused on measuring performance. This paradigm primarily assesses a model's capabilities by testing its accuracy across a vast and diverse set of questions, effectively measuring the breadth of its knowledge and skills. However, a recent and critical line of inquiry has shifted the focus from broad performance to a more fundamental question: reliability. Instead of asking how many different questions a model can answer correctly, this new paradigm examines the robustness of a model's answer to a single question by testing its stability against minor perturbations.

For example, Zheng et al. [5] experimentally demonstrated that LLMs are not robust selectors in MCQA, as their choices are heavily influenced by factors unrelated to content, such as token bias, where the model's choice is affected by the specific tokens used as option identifiers (e.g., 'A', 'B', 'C'). This finding has significant implications, suggesting that current MCQA-based evaluation methods may not accurately measure a model's true language understanding capabilities. Our research shares this critical perspective and builds

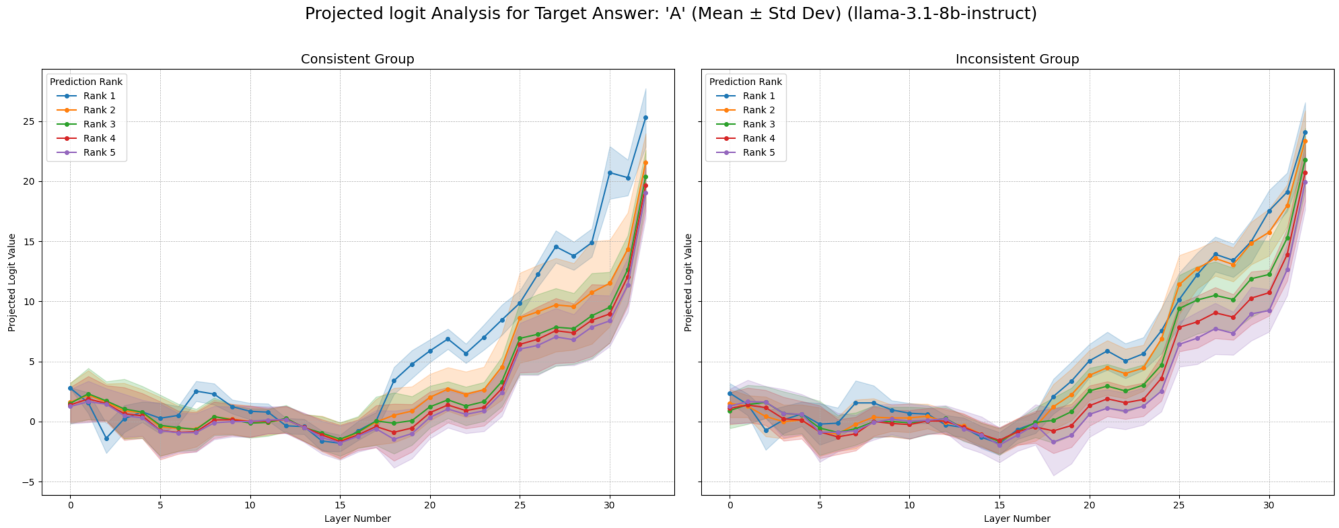

Figure 3: Aggregated logit trajectories for the Llama-3.1-8B-Instruct model, comparing a Consistent (left) and Inconsistent cases (right) for answer 'A'.

upon it. Moving beyond implicit biases, we analyze the phenomenon of how subtle changes in the question format can destabilize a model's predictions.

2.3 Mechanistic Analysis of Model Behavior in MCQA

To understand the mechanistic origins of such unreliable behavior, we turn to the field of mechanistic interpretability. This line of research provides tools to move beyond observing model outputs and instead analyze the internal processes that produce them. A notable example is the work of Wiegreffe et al. [4], which used techniques like Vocabulary Projection to successfully analyze the internal mechanisms of how a model arrives at correct predictions in MCQA. While their work provided a powerful methodology for understanding successful reasoning, it did not address the mechanisms behind failures. Our study bridges this gap. By adapting the analytical tools from Wiegreffe et al. [4] to the reliability problems highlighted by Zheng et al. [5], we move beyond observing inconsistency to explore its mechanistic origins that drive such unreliable behavior.

3 Methods

This section details the experimental methodology used to investigate the internal dynamics of inconsistent model behavior in Multiple-Choice Question Answering (MCQA). We first describe the models and dataset used, then outline the procedure for eliciting and categorizing inconsistent predictions, and finally explain the analytical techniques employed to probe the model's internal mechanisms.

3.1 Experimental Setup

Language Models. We conducted experiments with two instruction-tuned models: Llama-3.1-8B-Instruct (Dubey et al. [1]) and OLMo-7B-0724-Instruct (Groeneveld et al. [2]). Both models were loaded and run using bf16 precision.

Dataset. Our experiments were performed on the CommonsenseQA (Talmor et al. [3]) dataset. We randomly sampled 80% of the CommonsenseQA validation set, using a random state of 0 for reproducibility. Our final analysis set consists of 977 question-answer pairs.

Prompt Templates. To systematically introduce minor format variations, we created six distinct prompt templates. These templates varied only in the capitalization of keywords and the spacing around the keywords and their subsequent colons. Each prompt was provided directly to the model as raw text, bypassing any model-specific chat or instruction-following templates. The model's task was to predict a single token representing the option label. The full set of six templates is provided in appendix A.

3.2 Experimental Procedure

Data Collection. For each of the 977 questions selected for our analysis, we generated six prompts using the templates described above. For each of the six prompt variations, we performed a single forward pass to obtain the model's one-token answer. This resulted in a total of six answers per question for each model.

Defining Consistency. We categorized each question into one of two groups based on the six collected answers:

- **Consistent Cases:** All six generated answers are identical.
- **Inconsistent Cases:** The six generated answers are not all identical.

For this categorization, we normalized the output tokens by stripping any leading and trailing whitespace (e.g., treating ' A' and

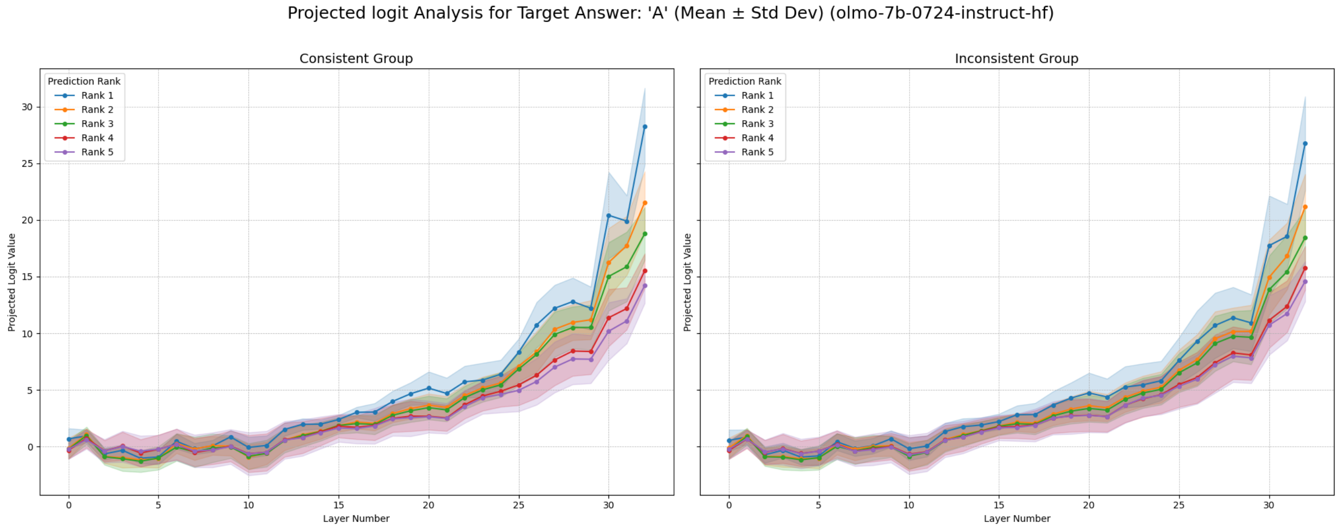

Figure 4: Aggregated logit trajectories for the OLMo-7B-0724-Instruct model, presenting a more nuanced comparison between Consistent (left) and Inconsistent (right) cases for answer 'A'.

'A' as the same). It is important to note that this study focuses on inconsistency in the *chosen option* (e.g., 'A' vs. 'B').

3.3 Analysis Method

To analyze the internal dynamics of the models' decision-making process, we employed Vocabulary Projection, following the methodology of Wiegreffe et al. [4]. This technique involves extracting the hidden state of the last token of the input prompt from each model layer. As the hidden state of the last token serves as an aggregate representation of the entire sequence for predicting the next token, we project it into the vocabulary space using the model's unembedding matrix after applying layer normalization. This allows us to inspect the resulting logit values for any candidate answer token (e.g., 'A', 'B') at each layer of the model's computation.

We applied this technique to compare the internal processes of consistent and inconsistent cases. For both of the following analyses, we first segmented our data by both consistency status and the model's final answer, creating groups such as 'Consistent-A' and 'Inconsistent-A'. This allowed for a targeted comparison of the model's internal state leading to a specific outcome. Our analysis then focused on two aspects:

(1) **First Answer Emergence and Total Count.** For each forward pass within our segmented groups, we measured two aspects of the prediction process. First, we identified the First Answer Emergence layer, defined as the first layer at which the token corresponding to the final output achieves the highest logit value among all vocabulary tokens. Second, we calculated the Total Count by counting the total number of layers where that same final output token held the highest logit value. Together, these metrics allow us to indirectly measure how robustly a decision, once formed, is sustained throughout the model's subsequent layers.

(2) **Layer-wise Logit Trajectory Visualization.** To compare the internal dynamics of stable versus unstable predictions, we visualized aggregated layer-by-layer logit trajectories for each group. For each group, we calculated the mean and standard deviation of logit trajectories for a set of key tokens. These key tokens were defined by identifying the top five highest-logit tokens from the final layer of each individual prediction. We then aggregated the trajectories by rank (e.g., the mean trajectory for all rank 2 tokens), while also tracking the specific trajectory for the group's target answer. The results were plotted as a line graph showing the mean trajectory, with shading to represent one standard deviation.

4 Results

This section presents the findings from our experiments. We begin by reporting the overall inconsistency rates for both models, followed by the results from our two primary analyses: the emergence layer of the final answer and the layer-wise logit trajectories.

4.1 Overall Inconsistency Rates

We first quantified the overall sensitivity of each model to the six format variations. Table 1 summarizes the number of consistent and inconsistent cases found among the 977 questions in our dataset.

Table 1: Inconsistency rates for each model.

Model	Consist.	Inconsist.	Rate (%)
Llama-3.1-8B-Instruct	790	187	19.1%
OLMo-7B-0724-Instruct	697	280	28.7%

As shown in Table 1, both models exhibited sensitivity to minor prompt variations. OLMo-7B-0724-Instruct was less robust, with an inconsistency rate of 28.7%, compared to 19.1% for Llama-3.1-8B-Instruct.

4.2 Analysis of Answer Emergence Layer

To understand at which processing stage the models' decisions were being formed, we measured the layer at which the final answer emerged and its persistence across subsequent layers. Figures 1 and 2 present the distribution of these emergence layers for Llama-3.1-8B-Instruct and OLMo-7B-0724-Instruct, respectively, comparing consistent and inconsistent cases.

For both models, a clear trend was observed where the median emergence layer for Consistent Cases was significantly earlier than for Inconsistent Cases. In addition, the total count of the answer token was substantially higher for consistent predictions.

This indicates that when the model's prediction was stable, the decision was typically formed early and persisted. Conversely, unstable predictions tended to be finalized only in the later stages of computation, suggesting a more prolonged and less certain process.

4.3 Layer-wise Logit Trajectory Analysis

To investigate the qualitative differences in the decision-making process, we visualized the aggregated logit trajectories.

Llama-3.1-8B-Instruct. Figure 3 compares the aggregated logit trajectories for the Llama-3.1-8B-Instruct model, illustrating the distinct internal dynamics between the Consistent and Inconsistent groups. In Consistent cases, the mean logit for the rank 1 answer token showed a clear divergence from competing tokens, indicating a stable and confident decision-making process. In contrast, for Inconsistent cases, the logits of the top-ranked tokens remained closely contested throughout most of the model's depth, with a decisive preference emerging only in the final few layers.

OLMo-7B-0724-Instruct. In contrast, Figure 4 reveals different characteristics for the OLMo-7B-0724-Instruct model. For this model, a strong divergence in the mean logit trajectories was not the primary distinguishing feature. Instead, a key difference between the groups emerged in the variance of the predictions in the final layers, as indicated by the shaded standard deviation areas. In Consistent cases, the standard deviation areas for the top-ranked token and its main competitor became clearly separated at the final layer. For Inconsistent cases, however, these shaded areas remained overlapped at the final layer, pointing to high variance and persistent uncertainty.

5 Discussion

The results of our exploratory analysis provide initial insights into the internal dynamics of LLM inconsistency. This section discusses the interpretation of these findings, their relationship to prior work, their broader implications for LLM reliability, and the limitations of this study.

5.1 Interpretation of Our Findings

Our first research question sought to identify the processing stages where inconsistency arises. The answer emergence layer analysis showed that while stable decisions are often formed in the mid-to-late layers, unstable decisions are finalized at the final layer of the models. This strongly suggests that the inconsistency observed in this study is about a fragility in the final decision-making process.

Our second research question asked what qualitative changes occur internally during an inconsistency failure. The logit trajectory analysis elucidates this internal process. For a stable, consistent prediction, a model appears to form a confident opinion relatively early. This was particularly evident in the Llama-3.1-8B-Instruct model. In contrast, an inconsistent prediction is characterized by a prolonged state of uncertainty where multiple candidate answers remain viable until the final computational stage, often resulting in a last-minute tie-breaker rather than a confident conclusion.

5.2 Contextualizing with Prior Work

These findings both support and extend prior works. Our results empirically validate the claims of Zheng et al. [5] that LLMs are not robust selectors in MCQA. While their work highlighted the existence of token bias, our analysis shows how such biases can manifest as internal uncertainty and last-minute decisions. Furthermore, we build directly on the methodology of Wiegreffe et al. [4]. They used Vocabulary Projection to map the mechanisms of successful predictions; our study demonstrates the utility of this technique for the equally critical task of diagnosing failures, thereby extending its application to the domain of model reliability.

5.3 Broader Implications

The primary implication of this study is that high performance on benchmarks can mask underlying fragility. A model can arrive at the correct answer, yet its internal decision-making process may be characterized by high uncertainty. This was evident in the comparison between Llama-3.1-8B-Instruct and OLMo-7B-0724-Instruct; although both models showed instabilities, their internal dynamics differed, with OLMo exhibiting higher uncertainty. This suggests that future evaluations should not only consider final accuracy but also incorporate metrics of internal stability. Understanding these failure modes is a critical step towards building genuinely reliable and trustworthy AI systems.

5.4 Limitations

We acknowledge several limitations in this study. First, our analysis was conducted on two 8B-scale models and a single commonsense reasoning dataset; these qualitative trends may not generalize to significantly larger models or different task domains. Second, our analysis is qualitative and exploratory. While we identify clear patterns, we do not make quantitative claims. Third, our aggregated analyses are subject to data imbalance, as inconsistent cases were not uniformly distributed across all target answers, the full details of which are provided in Appendix B. Trends observed for subgroups with fewer instances should be interpreted with caution. Future work should aim to quantify these dynamics and test the resulting hypotheses on a wider and more balanced range of models and benchmarks.

Acknowledgments

The author would like to thank SAILAB at KAIST for the valuable internship opportunity.

References

[1] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv e-prints* (2024), arXiv–2407.

[2] Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838* (2024).

[3] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937* (2018).

[4] Sarah Wiegreffe, Oyvind Tafjord, Yonatan Belinkov, Hannaneh Hajishirzi, and Ashish Sabharwal. 2024. Answer, assemble, ace: Understanding how LMs answer multiple choice questions. *arXiv preprint arXiv:2407.15018* (2024).

[5] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2023. Large language models are not robust multiple choice selectors. *arXiv preprint arXiv:2309.03882* (2023).

A Prompt Templates

This appendix details the six prompt templates used in our experiments to assess model inconsistency against minor format variations. These templates were systematically generated based on two factors: (1) the capitalization of the keywords (`question`, `options`, `answer`), and (2) the presence of a leading space before the keyword and a space after the colon.

The placeholders `{question}` and `{options}` were populated with content from the CommonsenseQA dataset. The `{options}` placeholder was replaced with a string where each choice was presented on a new line, preceded by its label (e.g., `"\nA. choice1\nB. choice2"`). Note that in all templates, the prompt ends immediately after the colon of the `answer` keyword, with no subsequent space.

Below are the six exact templates used for the experiments.

Template Set 1: No Extra Spacing

```
question:{question}
options:{options}
answer:
```

```
Question:{question}
Options:{options}
Answer:
```

```
QUESTION:{question}
OPTIONS:{options}
ANSWER:
```

Template Set 2: With Leading and Trailing Spaces

```
 question: {question}
 options: {options}
 answer:
```

```
 Question: {question}
 Options: {options}
 Answer:
```

```
 QUESTION: {question}
 OPTIONS: {options}
 ANSWER:
```

B Detailed Data Counts per Target Answer

This section provides a detailed breakdown of the number of consistent and inconsistent cases for each model, segmented by the target answer choice. This data forms the basis for the aggregated trajectory plots presented in the main text and the following appendix section.

Table 2: Number of consistent and inconsistent cases for each model, broken down by the target answer choice.

Model	Target Answer	Consistent	Inconsistent
Llama-3.1	‘A’	141	27
	‘B’	175	26
	‘C’	164	31
	‘D’	150	43
	‘E’	160	25
OLMo	‘A’	102	70
	‘B’	126	57
	‘C’	151	31
	‘D’	183	31
	‘E’	135	45

C Additional Logit Trajectory Visualizations

C.1 Llama-3.1-8B-Instruct

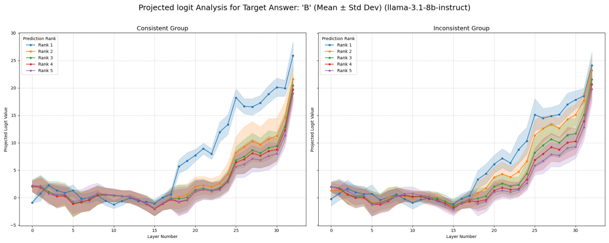

Figure 5: Consistent (left) and Inconsistent (right) Trajectory for answer 'B'.

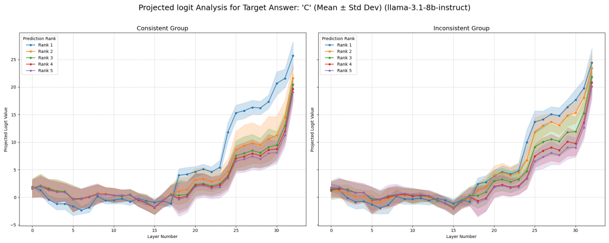

Figure 6: Consistent (left) and Inconsistent (right) Trajectory for answer 'C'.

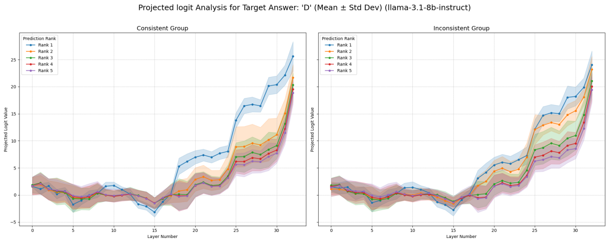

Figure 7: Consistent (left) and Inconsistent (right) Trajectory for answer 'D'.

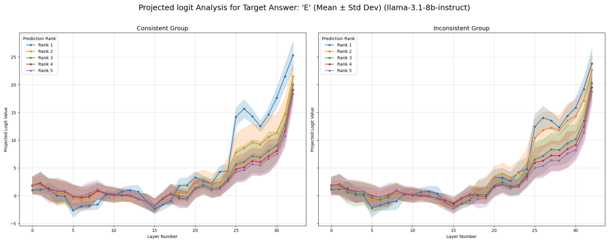

Figure 8: Consistent (left) and Inconsistent (right) Trajectory for answer 'E'.

C.2 OLMo-7B-0724-Instruct

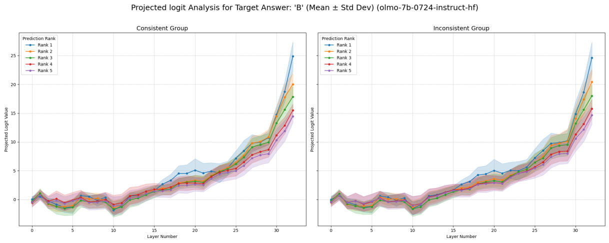

Figure 9: Consistent (left) and Inconsistent (right) Trajectory for answer 'B'.

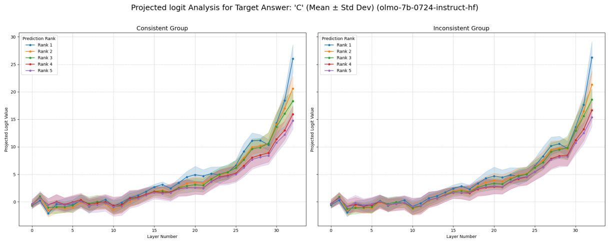

Figure 10: Consistent (left) and Inconsistent (right) Trajectory for answer 'C'.

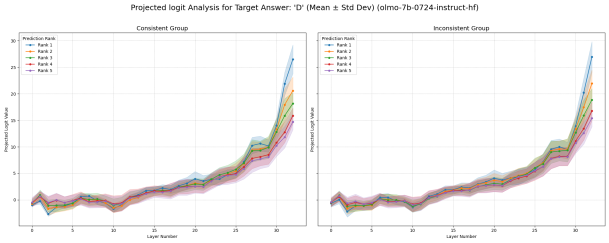

Figure 11: Consistent (left) and Inconsistent (right) Trajectory for answer 'D'.

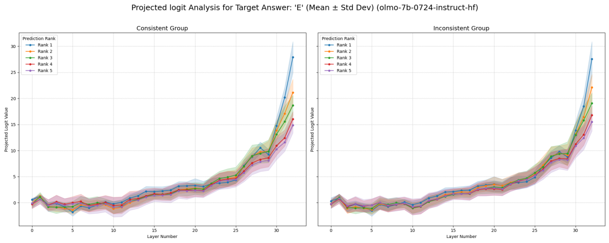

Figure 12: Consistent (left) and Inconsistent (right) Trajectory for answer 'E'.