# Emergence of Context Characteristics Sensitivity in Large Language Models

Anonymous ACL submission


## Abstract

During instruction fine-tuning (IFT), large language models (LLMs) learn to follow instructions by using the provided context to answer a query. While prior work has studied how context characteristics correlate with context usage by the LLM, this analysis has been limited to inference time, leaving open *how* these relationships are acquired in the first place. Here, we measure how models' sensitivity on such characteristics shifts across successive IFT stages: supervised fine-tuning (SFT), direct preference optimization (DPO) and reinforcement learning with verifiable rewards (RLVR). Experiments across four models and three datasets show that SFT makes models more likely to use contexts that are easy to understand, such as containing high length, context-query similarity, and fluency. Post-SFT dynamics may either reinforce or resolve these preferences depending on the training dataset. Our findings reveal that context usage is actively reshaped at each IFT stage, and designing a balanced IFT dataset is important in ensuring robust context utilization of instruction-tuned models.


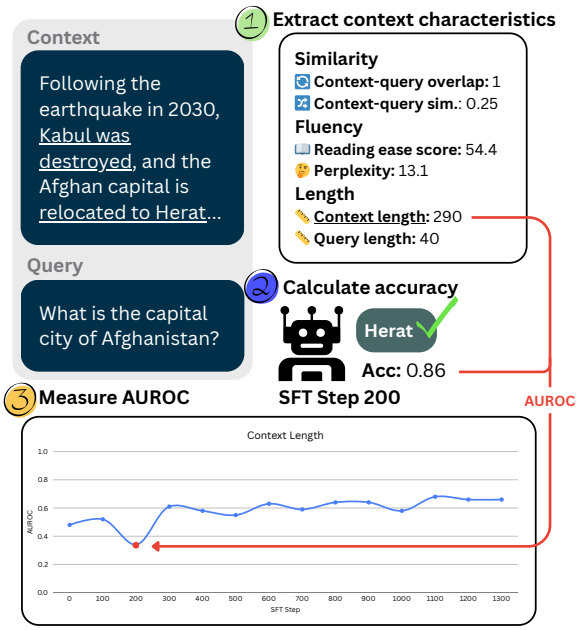


Figure 1: **Experimental setup.** From each dataset, three families of context characteristics (similarity, fluency, and overlap) are extracted. Accuracy is then calculated based on the model's generated output. Sensitivity is measured by calculating AUROC score (Section 4) between each characteristic and accuracy.

## 1 Introduction

LLMs can use the given context to answer queries; a valuable trait as it allows models to access up-to-date knowledge without re-training (Mosbach et al., 2023; Ovadia et al., 2024). This enables various applications, from retrieval-augmented generation (Lewis et al., 2020; Gao et al., 2024) to tool use and automated judges (Zheng et al., 2023; Chiang et al., 2024). However, models' context usage is often not robust, they sometimes ignore the provided context in favor of their parametric knowledge (Xie et al., 2024), or are easily distracted by irrelevant context (Shi et al., 2023). The consequences can be significant: for example, inappropriate context usage may make models susceptible to injected misinformation (Pan et al., 2023). Understanding *what* makes a model uses its context is important because it ensures reliable model behavior.

An intuitive approach to understanding context usage is to investigate the characteristics of the context itself. Recent work study lexical properties such as length, fluency, word overlap, and source reliability, finding that these characteristics correlate with context usage to varying degrees (Hagström et al., 2025; Hagström et al., 2026). Yet, little is known about how the models learn to use context in the first place and in particular, how sensitivities for certain context characteristics are acquired. In this work, *context characteristics sensitivity* refers to the degree to which a characteristic's value influences whether the model uses the provided context to answer the query. Prior work has established that

IFT plays a dominant role in teaching models to use context (Goyal et al., 2025) and in shaping LLM behavior more broadly (Bhatia et al., 2025). By understanding how IFT training dynamics shape context characteristic sensitivity, we can curate training datasets that elicit more robust context utilization in instruction-tuned LLMs.

Our analysis spans different IFT stages: SFT, DPO, and RLVR. Further, we train an LLM through SFT and DPO, and measure how this relationship evolves at each step; see Figure 1. We find that context usage is dynamic, and is actively reshaped across IFT stages in distinct ways. During SFT, LLMs develop a sensitivity towards easy-to-understand characteristics such as short and fluent contexts, and higher context-query similarity. The effect of DPO, in turn, is determined by the preference dataset, specifically the difference in lexical characteristics of the chosen and rejected responses. To produce robust context usage and counteract the context characteristic sensitivities developed during SFT, the preference dataset must be carefully curated such that the chosen and rejected responses share the same context characteristics.

## 2 Method

### 2.1 Context Characteristics

We examine three families of context characteristics motivated by prior work on the influence of context characteristics on context usage: similarity, fluency, and length (Hagström et al., 2025). We drop the unreliability family as it is specific to fact-checking tasks and yields neutral results (see Appendix C). Implementation of the extraction process is detailed at Appendix D.

**Similarity** LLMs often exploit high query-context similarity as a shortcut for whether to use context (Modarressi et al., 2025; Du et al., 2023), which we score with: *Jaccard similarity* (Jaccard, 1901) after lowercasing and punctuation removal; and the fraction of named entities that appear both in the query and context (*query-entity overlap*).

**Fluency** LLMs tend to follow factually incorrect evidence when it is coherent and fluent (Xie et al., 2024; Stureborg et al., 2024). We measure fluency through *Flesch reading ease score* (Flesch, 1948), a readability metric based on sentence length and syllable count, with higher scores indicating text that is easier to read, and *Perplexity*, where lower values indicate the context is considered fluent under the model's learned distribution.

**Length** Previous work documents two opposing forces affecting LLMs' context usage: they use long contexts less effectively (Xu et al., 2024), but also treat longer passages as more credible (Dubois et al., 2024). We investigate this by measuring *Context length* and *Query length*, both calculated as the raw character counts of each string.

### 2.2 Instruction Fine-Tuning

We conduct (1) a stage-wise comparison of sensitivities across IFT stages, and (2) a step-wise analysis tracking how they evolve within each stage.

For the stage-wise analysis, we evaluate four models: LLaMA-3.1-8B, LLaMA-3.2-1B (Grattafiori et al., 2024), OLMo-2-1B and OLMo-2-7B (OLMo et al., 2024). We use the final IFT model (SFT, DPO, RLVR) of LLaMA-3.1-8B and OLMo 2 (1B and 7B variants) released by the Open Instruct library[1] for our stage-wise analysis. Note that the LLaMA-3.1-8B model is trained using the Tulu3 dataset (Lambert et al., 2025), which is different from the datasets used to train the OLMo models detailed by OLMo et al. (2024).

Since the intermediate SFT and DPO checkpoints are not released, to conduct the step-wise analysis, we additionally fine-tune LLaMA-3.2-1B on the same SFT and DPO dataset as OLMo-2-1B. LLaMA-3.2-1B is chosen as it is small, while still able to use context well, see Appendix A. We adopt the hyperparameters of Lambert et al. (2025) unchanged. We exclude RLVR from step-wise analysis due to compute cost, but since it shares the same KL-regularized objective as DPO (Rafailov et al., 2023), DPO trends likely extend to RLVR. Detailed IFT configuration is in Appendix B.

## 3 Experimental Setup

**Datasets** We use three datasets chosen for diversity in task format and in the relationship between context and parametric knowledge. ConflictQA (Xie et al., 2024) and Context-Reliance (Goyal et al., 2025) are open-ended QA tasks with counterfactual contexts, ensuring that accuracy purely reflects context use. In contrast, DRUID (Hagström et al., 2025) is a claim verification task that reflects realistic context–knowledge interactions.

[1] https://allenai.github.io/open-instruct/

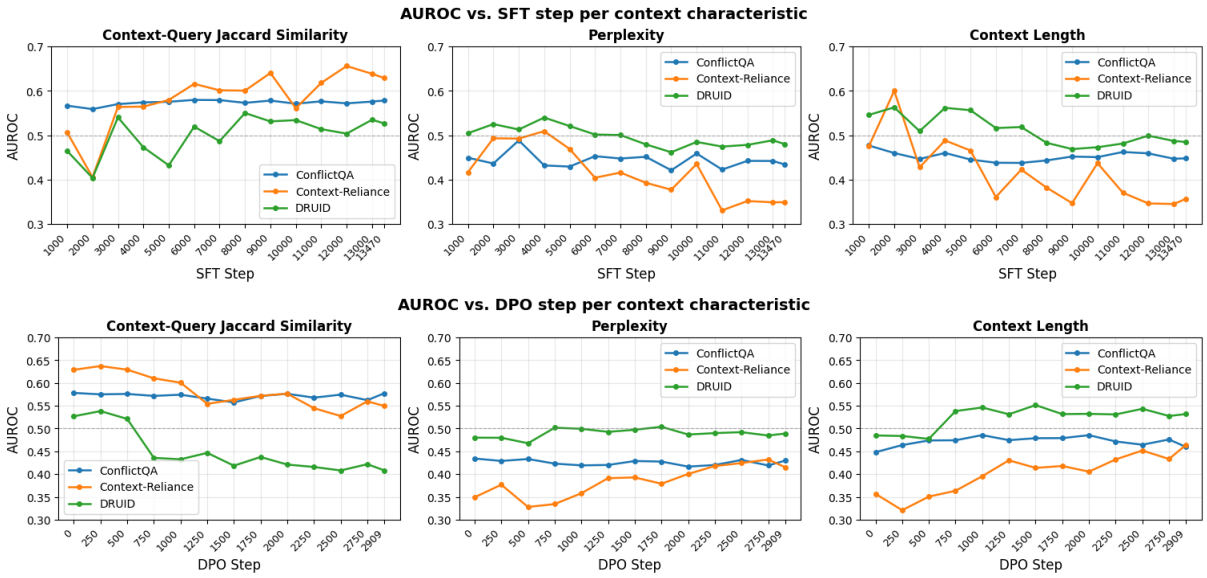


Figure 2: **Step-wise analysis across three datasets** of LLaMA-3.2-1B (Section 2.2). One metric of each characteristic family is chosen; other metrics are in Appendix C. During SFT (top), the model develops sensitivities towards easy-to-understand context, while DPO (bottom) resolves the learned sensitivities.

**Evaluation** For each combination of model, dataset, and characteristic, we compute the AUROC between the characteristic value and accuracy. The model's generation is considered accurate if it matches the context-based ground truth. We use AUROC because accuracy is binary, making other correlation measures, such as Pearson and Spearman, less appropriate. See Appendix E for details.

## 4 Results

Figures 2 and 3 show how context characteristics sensitivity evolves step-wise and stage-wise respectively. To gauge sensitivity, we use the AUROC score, which measures how well the context characteristic ranks accurate examples above inaccurate ones, ranging from 0.5 (chance) to 1 (perfect separation). Values above 0.5 indicates high characteristic values correspond to accurate examples, while values below 0.5 suggest the inverse.

### 4.1 Models learn sensitivities towards easy-to-understand contexts during SFT

For each characteristic, Figures 2 (top) and 3 show consistent AUROC trend during SFT, across all models and all datasets. This indicates that models learn the same context characteristics sensitivities when deciding whether to follow the provided context.

Specifically, the AUROC score of *Context-query Jaccard similarity* increases during SFT. Models increasingly use contexts that are lexically similar to the query. This is intuitive: higher token overlap signals relevance, making the context appear more directly applicable to the question. On the other hand, AUROC score of *Perplexity* decreases, meaning models increasingly follow low-perplexity contexts. Lower perplexity indicates more fluent, natural text, so the model learns to treat well-formed contexts as more trustworthy. Similarly, AUROC score of *Context length* decreases, suggesting models increasingly follow shorter contexts, possibly as they are easier to integrate compared to longer ones and therefore more reliable.

Our results support prior findings, where models are shown to use contexts with high similarity (Modarressi et al., 2025; Du et al., 2023) and high fluency (Xie et al., 2024; Stureborg et al., 2024) more frequently. Our investigation reveals that this sensitivity mainly forms during SFT. Notably, while context characteristic sensitivities increase across SFT steps, they do not translate into improved accuracy (Appendix A). The model exploits context characteristics as a shortcut; it uses context only when the context are easy-to-understand, rather than genuinely assessing its relevance.

### 4.2 DPO sensitivities are shaped by characteristic differences in training data

From Figure 2 (bottom), DPO resolves the sensitivities learned during SFT by shifting sensitivities to pre-SFT level. However, Figure 3 shows that LLaMA-3.1-8B does not follow the same trend dur-

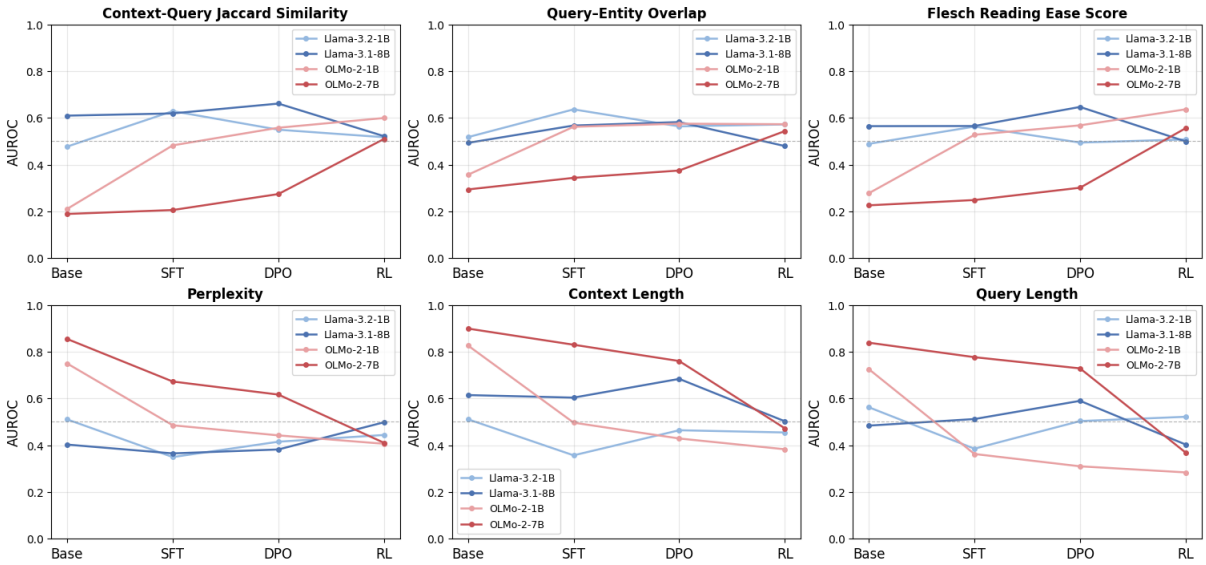


Figure 3: **Stage-wise analysis across four models**. Each model is evaluated on the Context-Reliance dataset. SFT induces the same sensitivity in all models, while DPO's effect depends on characteristic differences in its training data. Evaluation results on ConflictQA and DRUID dataset are in Appendix F.

| | **LLaMA-3.2-1B** | | | **LLaMA-3.1-8B** | | |
|---|---|---|---|---|---|---|
| Characteristic | chosen | rejected | $\Delta$ | chosen | rejected | $\Delta$ |
| Context-query Jaccard similarity | 0.158 | 0.150 | **0.008** | 0.192 | 0.161 | **0.031** |
| Query-entity overlap | 0.409 | 0.376 | **0.033** | 0.395 | 0.354 | **0.041** |
| Flesch reading ease score | 46.454 | 40.143 | **6.311** | 45.273 | -165.690 | **210.963** |
| Context length | 1844.392 | 1844.489 | **-0.097** | 1727.275 | 1705.265 | **22.010** |
| Query length | 946.300 | 946.300 | 0.000 | 910.274 | 910.274 | 0.000 |

Table 1: **Mean context characteristics for chosen and rejected responses for the DPO dataset** used to train the LLaMA models in Figure 3. $\Delta$ = chosen $-$ rejected. Bolded $\Delta$ indicates statistical significance ($p < 0.05$), more details in Appendix G.

ing DPO. For example, DPO amplifies, rather than resolves, sensitivity towards *Context-query Jaccard similarity* in the 8B variant. As the two models are trained on different DPO datasets (Section 2.2), we attribute this observation to the characteristics of the training dataset, specifically the difference between the chosen and rejected responses ($\Delta$) (Table 1). We find that whether the post-DPO AUROC lies above or below 0.5 broadly tracks the sign of $\Delta$. This highlights the importance of balancing context characteristics of responses in the DPO dataset. Any imbalances can reinforce SFT-stage sensitivities and context usage that is not robust. We note a caveat: this pattern holds for QA datasets (ConflictQA and Context-Reliance), but is less consistent for DRUID (Figure 2; bottom), a claim verification task. This suggests that DPO's effect on context characteristic sensitivities may be task-dependent. Finally, we observe that RLVR mostly preserves the direction of DPO's effects as expected.

## 5 Conclusion

We investigate how the context characteristic sensitivity evolves across IFT stages. We find that SFT consistently instills the same sensitivities, favoring lexically similar, fluent, and shorter contexts, across datasets and model families. This sensitivity formation is distinct from accuracy, suggesting that SFT teaches models to exploits these easy-to-read characteristics to signal context usage, rather than robustly assessing the context. For DPO, we attribute sensitivity shifts directly to characteristic differences between chosen and rejected contexts, underscoring the need to design a balanced preference dataset to neutralize context characteristics sensitivities inherited from SFT. Together, these findings suggest that context usage is not a static model property but is actively shaped at each IFT stage, and can be steered through careful data curation choices.

## Limitations

Our study is limited to two model families (LLaMA and OLMo) at two scales each. We consider this scope appropriate for our research questions which focus on how training dynamics shape context characteristic sensitivity. However, broader coverage across different architectures remains an interesting open line of work. Additionally, we examine three families of context characteristics: similarity, fluency, and length. These are prevalent and well-grounded in prior work, though there may well be other characteristics that influence context usage, and would be worth exploring in future work.

An AI writing assistant tool is used for a grammar check and stylistic improvement. None of the AI generations is used as is.

## References


Mehar Bhatia, Shravan Nayak, Gaurav Kamath, Marius Mosbach, Karolina Stańczak, Vered Shwartz, and Siva Reddy. 2025. Value drifts: Tracing value alignment during llm post-training. *Preprint*, arXiv:2510.26707.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios N. Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: an open platform for evaluating llms by human preference. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. 2023. Shortcut learning of large language models in natural language understanding. *Commun. ACM*, 67(1):110–120.

Yann Dubois, Percy Liang, and Tatsunori Hashimoto. 2024. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Sachin Goyal, Christina Baek, J Zico Kolter, and Aditi Raghunathan. 2025. Context-parametric inversion: Why instruction finetuning may not actually improve context reliance. In *The Thirteenth International Conference on Learning Representations*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Lovisa Hagström, Sara Vera Marjanovic, Haeun Yu, Arnav Arora, Christina Lioma, Maria Maistro, Pepa Atanasova, and Isabelle Augenstein. 2025. A reality check on context utilisation for retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19691–19730, Vienna, Austria. Association for Computational Linguistics.

Lovisa Hagström, Youna Kim, Haeun Yu, Sang goo Lee, Richard Johansson, Hyunsoo Cho, and Isabelle Augenstein. 2026. Cub: Benchmarking context utilisation techniques for language models. *Preprint*, arXiv:2505.16518.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Jumayel Islam, Lu Xiao, and Robert E. Mercer. 2020. A lexicon-based approach for detecting hedges in informal text. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3109–3113, Marseille, France. European Language Resources Association.

Paul Jaccard. 1901. Etude comparative de la distribution florale dans une portion des alpes et des jura. In *Bulletin de la Société Vaudoise des Sciences Naturelles*.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James Validad Miranda, Alisa Liu, Nouha Dziri, Xinxi Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Christopher Wilhelm, Luca Soldaini, and 4 others. 2025. Tulu 3: Pushing frontiers in open language model post-training. In *Second Conference on Language Modeling*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. *Preprint*, arXiv:cs/0205028.

Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Trung Bui, Ryan A. Rossi, Seunghyun Yoon, and Hinrich Schuetze. 2025. NoLiMa: Long-context

evaluation beyond literal matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 44554–44570. PMLR.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12284–12314, Toronto, Canada. Association for Computational Linguistics.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, and 21 others. 2024. 2 olmo 2 furious. *Preprint*, arXiv:2501.00656.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2024. Fine-tuning or retrieval? comparing knowledge injection in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 237–250, Miami, Florida, USA. Association for Computational Linguistics.

Yikang Pan, Liangming Pan, Wenhu Chen, Preslav Nakov, Min-Yen Kan, and William Wang. 2023. On the risk of misinformation pollution with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1389–1403, Singapore. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Qwen Team. 2026. Qwen3.6-Plus: Towards real world agents.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *Preprint*, arXiv:2405.01724.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2024. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Retrieval meets long context large language models. In *The Twelfth International Conference on Learning Representations*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

## A LLaMA-3.2-1B Accuracy

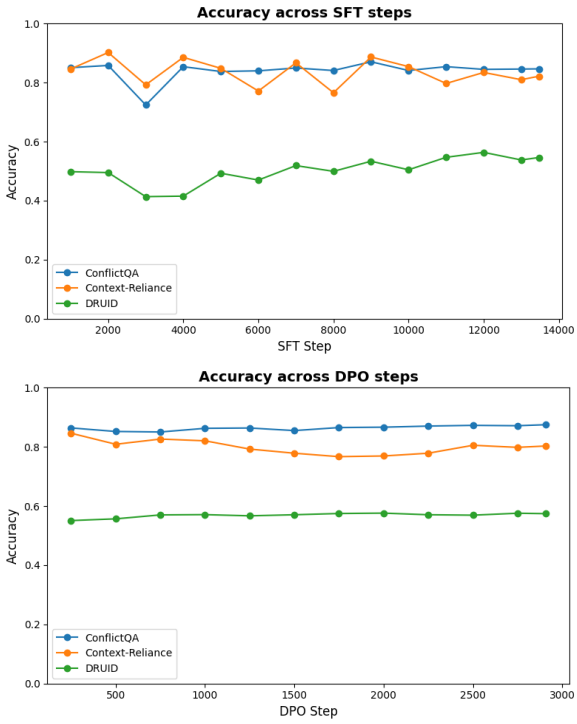


Figure 4: **Context-based accuracy of LLaMA-3.2-1B per step**, throughout SFT (top) and DPO (bottom).

Figure 4 shows that small LLaMA-3.2-1B model are able to use context properly. For ConflictQA and Context-Reliance, the accuracy throughout SFT and DPO training hovers around 0.8 to 0.9. For DRUID, accuracy is lower at around 0.5, but it is still higher than random chance (0.33; three-way classification). This shows that we can use LLaMA-3.2.1B as a proxy to study changes in context usage during SFT and DPO.

## B Configuration for Instruction Fine-Tuning

Table 2 shows the config for IFT used to train LLaMA-3.2-1B, which checkpoints are used to create Figure 2.

| Stage | Parameter | Value |
|---|---|---|
| SFT | Learning rate | 3e-5 |
| | Batch size/device | 2 |
| | Gradient accumulation steps | 16 |
| | Epochs | 2 |
| | Max sequence length | 4096 |
| | LR scheduler | Linear |
| DPO | Learning rate | 2.5e-6 |
| | Batch size/device | 2 |
| | Gradient accumulation steps | 16 |
| | Epochs | 1 |
| | Max sequence length | 2048 |
| | LR scheduler | Linear |
| | $\beta$ | 5 |

Table 2: Detailed configuration for SFT and DPO fine-tuning.

We save intermediate checkpoints every 1,000 steps during SFT and every 250 steps during DPO. All fine-tuning is done on 4XA100 GPU. SFT took 24 hours and DPO took 6 hours. Conducting RLVR would have taken approximately 344 hours (14X more compute than SFT).

## C Full Context-Characteristic Results

The step-wise relationship for *Query-entity overlap*, *Flesch reading ease score*, and *Query length* is shown in Figure 5. We include an additional context characteristic family that measures *Unreliability* of the context. This is omitted from the main paper as the results are neutral, and the characteristic is only relevant to claim verification tasks.

## D Context Characteristic Extraction

**Similarity** *Context-query similarity* is computed as the Jaccard index between the sets of lower-cased tokens of the context and the query, after tokenization with NLTK (Loper and Bird, 2002) and removal of punctuation and typographic quote characters. *Query-entity overlap* is computed by extracting named entities from both the context and the query using `spaCy`'s transformer-based pipeline (Honnibal et al., 2020) and reporting the fraction of query entities that also appear in the context; queries with no detected entities are excluded.

**Fluency** *Flesch reading ease score* is calculated via the `textstat` library[2]. Meanwhile, *Perplexity* is calculated according to a HuggingFace guide[3].

**Length** *Context length* and *Query length* are measured as the raw number of characters in the corresponding strings.

**Uncertainty** As mentioned in Appendix C, in Figure 5, we include an additional context characteristic family, *Unreliability*. We follow procedure detailed by Islam et al. (2020). Context is parsed sentence by sentence with `Stanza` library (Qi et al., 2020), and two boolean signals are extracted per sentence: (i) whether any $n$-gram of the lemmatized sentence ($n \in \{1, \ldots, 7\}$) reaches a Jaccard similarity of at least $0.8$ with a *discourse marker* from the word list; and (ii) whether any *hedge terms* from the word list occurs and survives dependency-based filtering rules that suppress non-hedging usages (e.g., *think* followed by a preposition).

## E Detailed Set-up

For DRUID (5,490 rows), predictions are mapped to one of the three labels (*supports*, *refutes*, *insufficient*) via string matching against the generated output; mismatches and abstentions count as incorrect. For ConflictQA (8,067 rows) and Context-Reliance (1,064 rows), we extract the context-based and parametric-based ground truths and string-match the model's output against them. When the result is ambiguous (both or neither match), we fall back to an LLM-as-a-judge using Qwen-3.6 Plus (Qwen Team, 2026), as it performed best in held-out evaluation set (See Appendix H). Prompt used for inference are shown in Appendix I.

All inference uses greedy decoding with `max_new_tokens=1024` and a fixed random seed (1000) across PyTorch, CUDA, NumPy, and Python; cuDNN is set to deterministic mode to ensure reproducibility. Inference is run on a single A100 GPU.

## F Extended Model-Dataset Results

Figure 6 shows the stage-wise relationship using ConflictQA and DRUID datasets.

[2] https://github.com/textstat/textstat

[3] https://huggingface.co/docs/transformers/perplexity

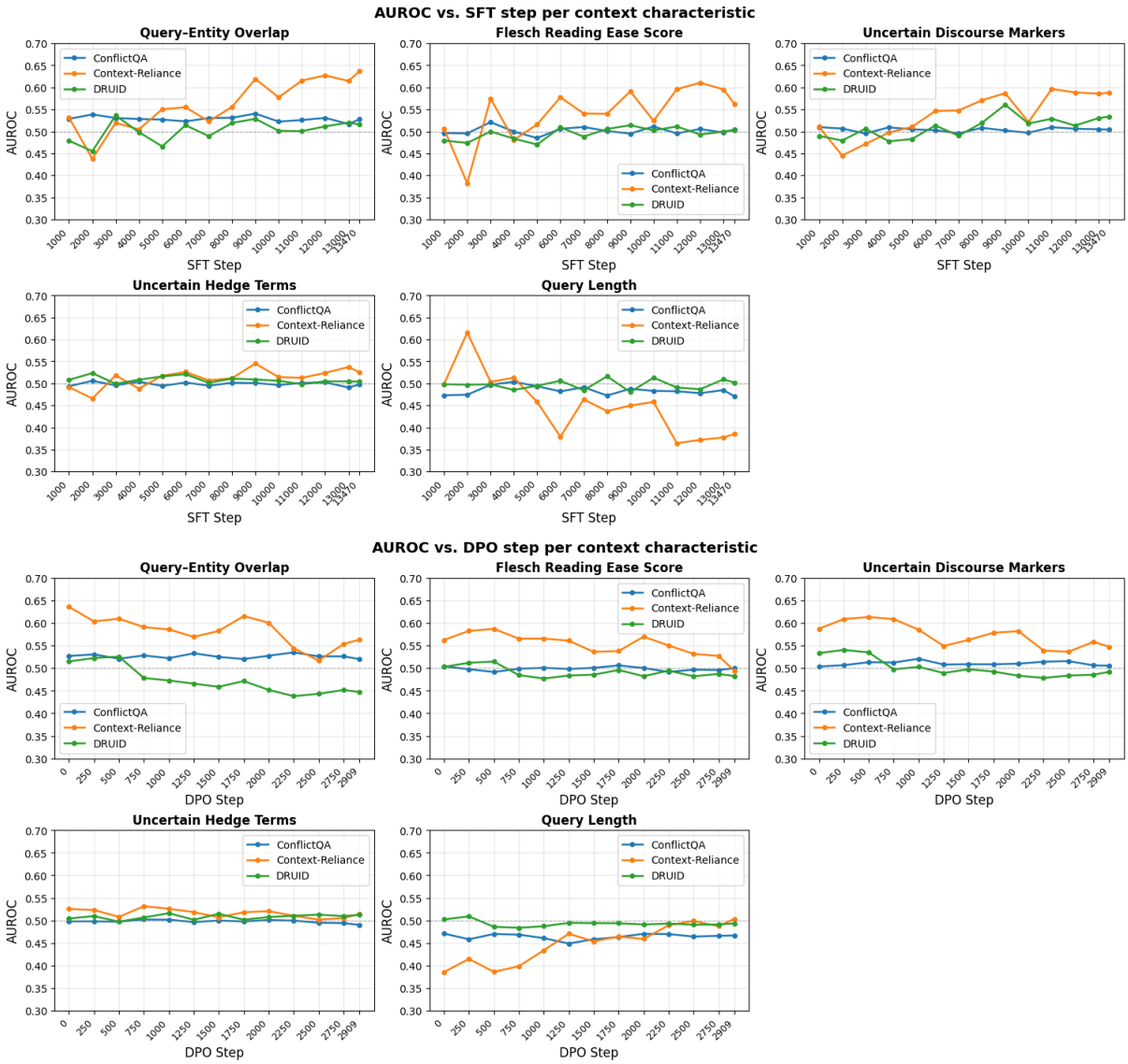


Figure 5: **Step-wise analysis between the remaining other context characteristic with accuracy** including the unreliable context characteristic family. LLaMA-3.2-1B is used (Section 2.2).

## G Statistical Tests

We use Wilcoxon signed rank test to determine significance. It tests the null hypothesis that two related paired samples (the chosen and rejected answers from the DPO training dataset) come from the same distribution. Since $p < 0.05$ for all of $\Delta$ in Table 1, the null hypothesis is rejected. The difference ($\Delta$) is significant.

Student t-test is not used as we are unable to assume the difference in mean of each context characteristic is normally distributed.

## H LLM Judge

For ConflictQA and Context-Reliance dataset, if the model's answer contains both (or neither of) context-based and parametric-based ground truths, we fall back to LLM judge to determine accuracy. Figure 7 shows the prompt for the LLM judge.

To find the best judge model to use, we manually annotate 50 samples of model outputs (using the same prompt in Figure 7) and validate them with 10 different judges. Agreement between each judge and the manual annotation is shown in Table 3. For our analysis, we use Qwen-3.6 Plus as it is the best-performing and most cost-efficient judge. All judge inferences use the OpenRouter [4] API. Only input pricing is considered, as the model's output is short and therefore negligible.

[4] https://openrouter.ai/

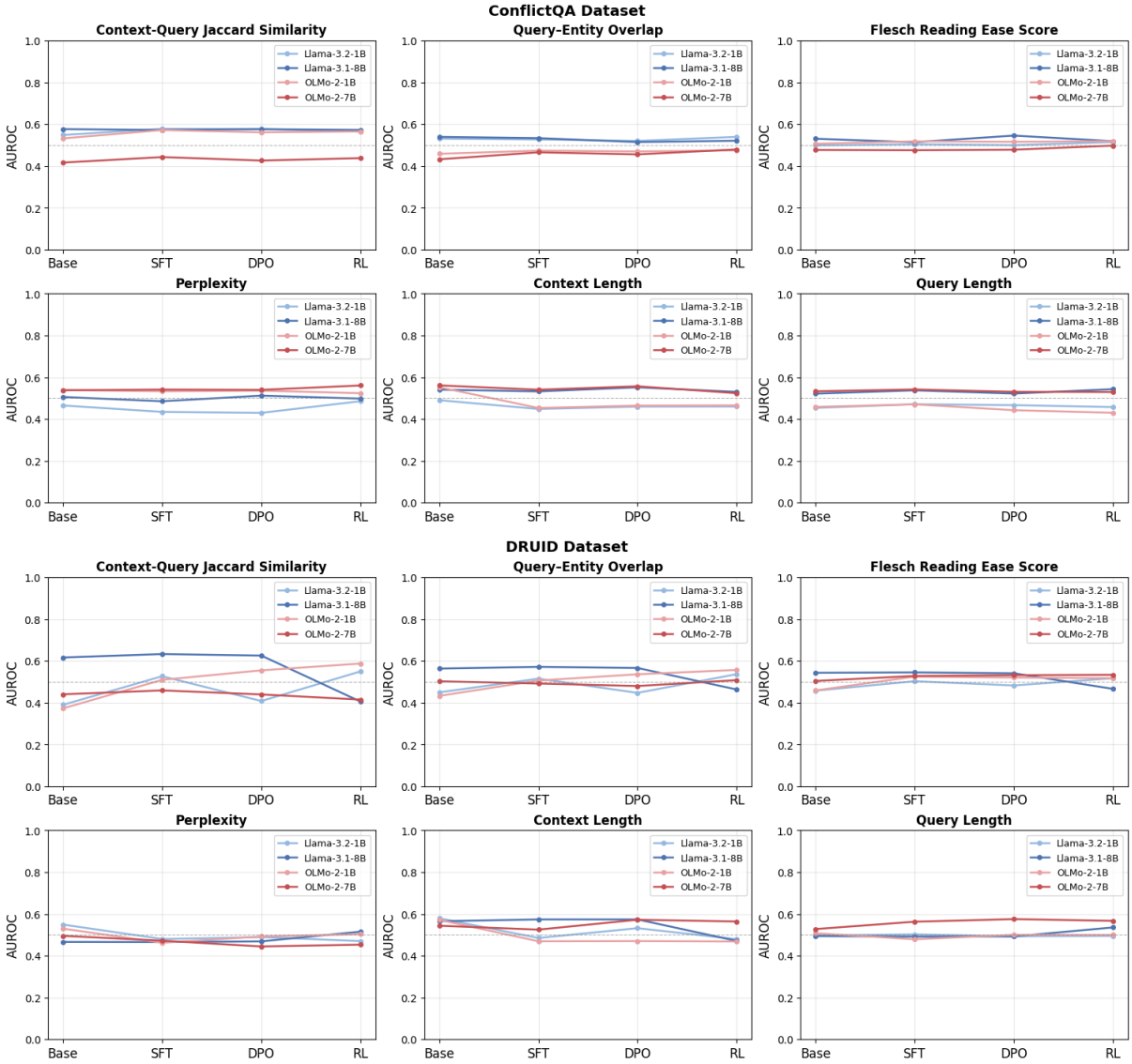


Figure 6: **Stage-wise analysis across four models on two extra datasets**: ConflictQA (top) and DRUID (bottom).

| LLM Judge | $\kappa$ | Input Price |
|---|---|---|
| Gemini 2.5 Flash | 0.716 | 0.30 |
| Qwen3 235B | 0.676 | 0.07 |
| Qwen 3.6 Plus | **0.920** | 0.17 |
| Llama 4 Maverick | 0.735 | 0.15 |
| Llama 3.3 70B | 0.401 | 0.10 |
| MiniMax M2.7 | 0.671 | 0.28 |
| GLM-5.1 | 0.880 | 0.98 |
| GLM-4.7 | **0.920** | 0.45 |
| GPT 5.4-mini (high reasoning) | **0.920** | 0.75 |
| GPT 5.4-mini (low reasoning) | 0.880 | 0.75 |

Table 3: LLM Judge agreement (Cohen's $\kappa$) with manual annotation and API input pricing (US$/M tokens).

# I Prompts Used for Inference

Figure 8 shows the prompts used for inference, based on the task.

**Prompt for LLM Judge**

You are a judge.
Your task: determine whether a generated answer correctly matches a reference answer.

Rules:
- Output ONLY the digit 1 or 0. Nothing else.
- Output 1 if the generated answer is correct, i.e., it expresses, implies, or contains the reference answer.
- Partial matches are acceptable (e.g., "Trump" matches "Donald Trump"), but not wrong answers (e.g., "Hudson Williams" does not match "Hudson Lee").
- Paraphrases are acceptable (e.g., "What is the capital of France? It is moved from Paris to Versailles" matches "Versailles").
- Output 0 if the generated answer does not answer the question directly, or require strong inference to get the answer.
- Output 0 if the generated answer is wrong, irrelevant, or contradicts the reference answer.
- If there is an explicit final answer in the generated answer, only consider the final answer in your decision.

Question: "<question>"
Reference answer: "<context-based ground truth>"
Generated answer: "<model's output>"

Does the generated answer match the reference answer? Output 1 or 0.

Figure 7: Prompt used for LLM judge as a tie-breaker.

**Prompt for Claim Verification Task (DRUID)**

Based on the provided context, is the claim True or False? If you are not sure or cannot answer, say None.
<user>
Context: "<context>"
Claim: "<claim>"
<assistant>

**Prompt for Open-ended QA Task (ConflictQA, Context-Reliance)**

Answer the following reading comprehension question.
<user>
Context: "<context>"
Query: "<query>"
<assistant>

Figure 8: Prompts used for model output generation.