CHARPEVAL: Benchmarking Large Language Models' Contextual Reasoning in Knowledge-Grounded Dialogue

Abbas Ghaddar[1♠] **David Alfonso-Hermelo**[1] **Philippe Langlais**[2]
Boxing Chen[1] **Prasanna Parthasarathi**[1]
[1] Huawei Noah's Ark Lab, Montreal Research Center, Canada
[2] RALI/DIRO, Université de Montréal, Canada
abbas.ghaddar@huawei.com

Abstract

This paper presents CHARPEVAL, a challenging benchmark specifically designed to evaluate the ability of Large Language Models (LLMs) to perform contextualized reasoning in knowledge-grounded dialogue scenarios. The task involves selecting the correct response from 6 options, including 5 manually crafted distractors, given the conversation history. Extensive benchmarking experiments with a diverse set of state-of-the-art open-weight LLMs show poor performance on CHARPEVAL due to their inability to effectively reason over discontinuous chunks of text across the input. Our analysis reveals systematic error patterns across models with different properties, highlighting the need to improve LLMs beyond simply scaling-up data and compute. CHARPEVAL is publicly available at https://huggingface.co/datasets/huawei-noah/CHARP.[1]

1 Introduction

There have been ongoing efforts to develop new benchmarks that challenge the rapid advancements of LLMs (Hurst et al., 2024; Liu et al., 2024a; Dubey et al., 2024), by introducing more complex (Fan et al., 2023), reasoning-intensive (Wang et al., 2024), and domain-specific (Jain et al., 2024; Chen et al., 2023b) tasks. Knowledge-grounded dialogue (Ghazvininejad et al., 2018; Lewis et al., 2020; Dziri et al., 2022b) is a task that requires responding to user queries while staying faithful to the provided knowledge.

In this paper, we propose CHARPEVAL, a challenging benchmark that evaluates LLMs ability to identify discontinuous spans of relevant information and perform complex reasoning on them. CHARPEVAL is build on top of CHARP (Ghaddar et al., 2024), which is a testbed designed for probing the conversation history awareness of knowledge-grounded dialog systems. Specifically, we expand CHARP annotations with 5 hand-crafted challenging distractor responses, each designed to test a specific potential limitation in LLMs contextual reasoning. As shown in Figure 1, CHARPEVAL not only presents a serious challenge to modern LLMs, but also enables a self-contained and reproducible evaluation process by eliminating the need for external evaluation tools.

Therefore, CHARPEVAL is more tailored to LLM evaluation compared to existing dialog benchmarks —including CHARP itself— by addressing issues such as misalignment with true human performance (Sinha et al., 2020), reproducibility challenges with API deprecated (Chen et al., 2023a), outdated Judge LLMs (Szymanski et al., 2024), and costly human expertise. These issues are common in approaches relying on lexical overlap/semantic similarity scorers (Lin, 2004; Zhang et al., 2019), closed-source LLM APIs (Ahmed et al., 2024), LLM-as-Judge (Zhu et al., 2023), or human expert as evaluation methods, respectively. In contrast to our CHARPEVAL, which is a human-generated benchmark emphasizing contextual reasoning in dialog, the recently proposed DialogBench (Ou et al., 2024) is automatically generated by GPT-4 (OpenAI, 2023) and focuses on evaluating LLM instruction-following in dialog tasks.

A comprehensive evaluation using a diverse set of recent open-weight LLMs (Yang et al., 2024; Dubey et al., 2024) reveals that these models perform poorly on CHARPEVAL, with the best models scoring around 50%. Our extensive analysis reveals systematic error patterns, regardless of the model size, tuning process, or organization behind the model. It also highlights an inherent bias that favors irrelevant responses, with highly frequent text segments encountered during pretraining being favored over task-relevant ones.

♠Corresponding author.

[1]Instructions on how to conduct the evaluation using the `lm-evaluation-harness` library (Gao et al., 2024) are provided in the dataset's README file.

Findings of the Association for Computational Linguistics: ACL 2025, pages 16764–16775

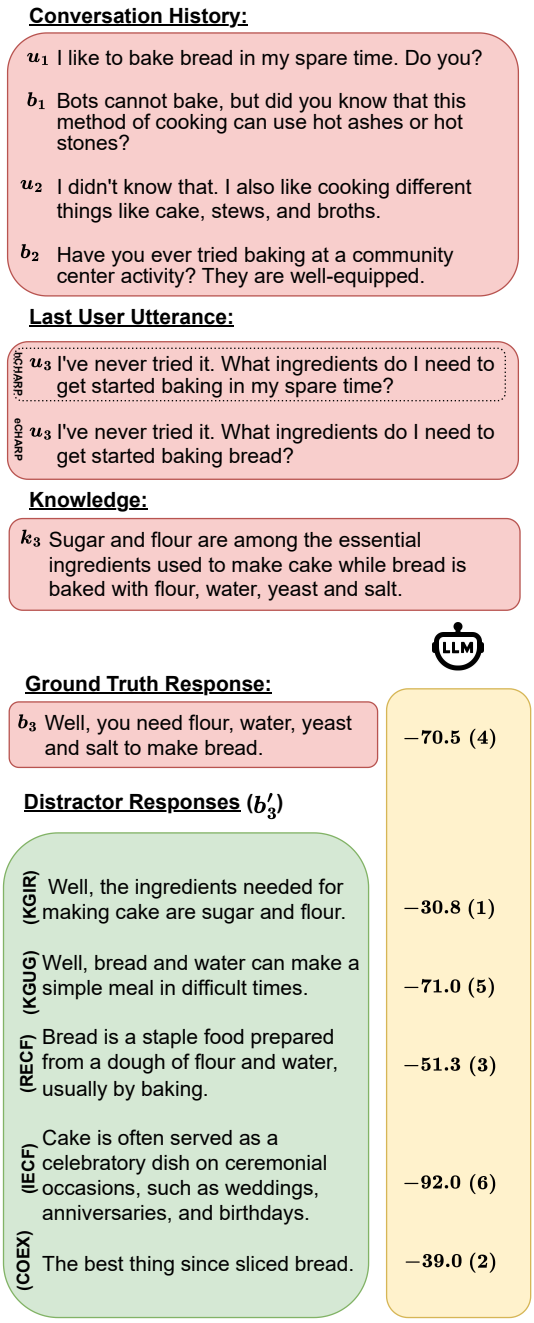

Figure 1: An illustrative example showing a sample from CHARPEVAL along with an LLM benchmark evaluation performed on it. Pink and green boxes show the original CHARP annotations and our CHARPEVAL augmented distractor responses, respectively. The yellow box contains the log-likelihood scores and the responses rank (in parentheses) of the *Qwen2.5-70B-Instruct* LLM when using hCHARP last user utterance (dotted box).

2 Methodology

2.1 Task Formulation

We adopt the knowledge-grounded dialogue task formulation as defined in previous works (Dinan et al., 2018; Dziri et al., 2022a,b). Formally, let $\mathcal{H} = \{(u_i, b_i)\}_{i=1}^{n}$ represent the user-bot conversational history, where u_i and b_i denote the user's utterance and the bot's response at the i-th turn, respectively. Given a new user utterance u_{n+1} and a piece of knowledge k_{n+1}, the task is to generate the bot's response b_{n+1}, such that it both answers the u_{n+1} and faithfully incorporates the provided knowledge. The latter is provided as given, and no retrieval step is performed.

2.2 CHARP

CHARP (Ghaddar et al., 2024) is designed to diagnose knowledge-grounded dialogue systems that generate responses solely based on the provided knowledge, while ignoring content from the conversation history. CHARP was built by post-editing annotations from the FaithDial (Dziri et al., 2022a) dataset, ensuring that the response requires reasoning over the conversation history, the last user utterance, and the provided knowledge. CHARP consists of two subsets, differing only in the last user utterance (u_{n+1}): eCHARP (easy), which requires reasoning only over the knowledge, and hCHARP (hard), which requires reasoning over the conversation history, the provided knowledge, and the last user utterance. Overall, CHARP contains $2,160$ samples, split equally between hCHARP and eCHARP.

2.3 CHARPEVAL

CHARPEVAL is an extended version of CHARP that aims to create a standard and reproducible evaluation benchmark to test the ability of modern LLMs to perform contextual reasoning in the context of knowledge-grounded dialogue. To this end, we hire professional annotators to augment samples in CHARP with 5 distractor responses (b'_{n+1}), each requiring discerning reasoning from an LLM to avoid being selected as the correct answer.

2.3.1 Knowledge Irrelevant (KGIR)

In this case, b'_{n+1} is a distracting response that contains irrelevant factual information that exists in the provided knowledge k_{n+1}.

2.3.2 Knowledge Ungrounded (KGUG)

The distractor response b'_{n+1} contains valid factual information, but it is not mentioned (ungrounded) in k_{n+1}, while still containing words from b'_{n+1}.

2.3.3 Relevant Entity Common Fact (RECF)

We asked our annotators to identify the main entity in the ground truth response b_{n+1} and create a distractor response b'_{n+1} that contains a common fact (e.g., an extract from the first passage of a Wikipedia article) about that entity, but not mentioned in k_{n+1}. Intuitively, a common factual segment will have a higher log-likelihood if the model does not consider the task and the provided context.

2.3.4 Irrelevant Entity Common Fact (IECF)

Similar to the distractor in § 2.3.3, but this time using a common fact about an entity that is mentioned in k_{n+1} but not in b_{n+1}.

2.3.5 Common Expression (COEX)

We prompt our annotators to use a common expression (e.g. idioms, catchphrases, proverbs, and clichés) as a distractor response, provided it aligns with the conversation flow. Since these phrases commonly occur in conversations and are highly frequent, a model that does not reason about the task and its underlying knowledge is likely to assign a high likelihood to this particular distractor.

2.4 Benchmark Evaluation

We perform benchmark evaluation of a LLM by calculating the accuracy of the model in selecting the ground truth b_{n+1} as the most likely response compared to the distractors b'_{n+1}. Specifically, each sample in CHARPEVAL is expanded into 6 sequences, where the input context (task prompt, conversation history, and knowledge segment) is concatenated with 6 continuation segments: the ground-truth response and 5 distractor responses. The best response is determined by the one with the highest probability (lowest perplexity), and a model receives full credit for a sample if the ground-truth response is ranked top. Evaluation can be conducted on both eCHARP and hCHARP subsets of CHARPEVAL (1080 each), and under both zero-shot and few-shot settings. Implementation details for the task prompt and few-shot examples can be found in Appendix A, while Appendix B describes the manual quality validation of CHARPEVAL.

3 Experiments

3.1 Models

We conduct a performance comparison across a diverse set of open-weight LLMs, encompassing models with varying providers, parameter counts, release periods, training paradigms, and architectural designs. Specifically, we experiment with base and instruction-tuning models ranging in size from 3B to 72B from Qwen2.5 (Yang et al., 2024), Llama (Touvron et al., 2023; Dubey et al., 2024), and Mistral (Jiang et al., 2024) families.

3.2 Main Results

Table 1 shows the benchmark evaluation performance in both zero-shot (ZS) and few-shot (FS) settings for LLMs with diverse properties on the hCHARP and eCHARP subsets of CHARPEVAL.

Model	#P	IT	hCHARP		eCHARP	
			ZS	FS	ZS	FS
Qwen2.5	72B	✓	48.7	43.3	49.3	44.0
Qwen2.5	32B	✓	47.0	49.8	47.2	50.5
Qwen2.5	7B	✓	46.9	46.6	46.1	48.7
Qwen2.5	7B	✗	18.8	25.6	20.7	26.8
Qwen2.5	3B	✓	42.5	41.6	42.9	44.6
Llama3.3	70B	✓	48.3	48.9	47.1	49.4
Llama3.1	70B	✓	44.0	45.7	42.5	46.0
Llama3.1	8B	✓	41.9	39.2	41.3	38.3
Llama2	13B	✓	27.7	28.0	29.7	29.9
Llama2	7B	✓	24.1	25.7	24.9	25.4
Llama2	7B	✗	10.4	20.1	11.4	23.4
Mistralv0.1	52B	✓	46.1	49.7	46.7	51.7
Mistralv0.2	7B	✓	43.8	42.8	43.6	43.3

Table 1: CHARPEVAL benchmark accuracy on hCHARP and eCHARP under both zero-shot (**ZS**) and few-shot (**FS**) settings. We evaluate LLMs with varying numbers of parameters (**#P**), **IT** indicates if it is an instruction-tuned (✓) or base model (✗).

Overall, we observe that while some models perform better than others, their performance remains poor across both CHARPEVAL subsets and evaluation settings, with the highest scores barely reaching 50%. This observation indicates that CHARPEVAL poses challenges and exposes a potential limitation for current state-of-the-art open-weight small and medium-sized LLMs.

Expectedly, we observe that adding few-shot examples leads to significant improvements (6%-12%) in performance only for the base models, as including a few demonstration examples in the input prompt is crucial for performance when using a pretrained LLM (Brown et al., 2020). In contrast, instruction-tuned models show either a moderate gain (except on Mixtral-52B) or a slight drop in performance (except on Qwen2.5-72B) of less 2% in both cases. These small variations in scores indicate that the overall poor performance is not due to the models' inability to follow task instructions.

Interestingly, we observe a small gap of 3-5% between few-shot and instruction-tuned Llama2-7B, compared to a significantly larger gap (>20%) between the base and instruction-tuned versions of Qwen2.5-7B. This contrasting observation suggests that the supervised fine-tuning and preference alignment phases are the main drivers of performance improvements in recent LLM iterations (e.g. Qwen2.5), while the performance of earlier iterations (e.g. Llama2) relies more heavily on the pretraining phase.

We notice that model version upgrades, which include scaling up data and compute power, along with improvements in modeling and training techniques, are much more effective for improving performance than simply scaling up model size within the same model family. For instance, performance improves by roughly 6% on both subsets when scaling Qwen2.5 from 3B to 72B, and by 3% when scaling Mistral from 7B to 52B. In contrast, we observe a much larger gain of +12% when comparing Llama3.1-8B with Llama2-13B, and +4% when comparing the same-sized 70B Llama3.1 and Llama3.3. Still, models from different providers seem to converge to a performance range of around 50% on CHARPEVAL.

In addition, we observe that models tend to perform slightly better, with only a small margin of 1%-3%, on the less reasoning-intensive eCHARP subset compared to the hCHARP subset, despite a few exceptional cases (e.g., LLaMA3.1-70B-IT and Qwen2.5-7B-IT). These findings suggest that a significant number of challenging dialog-oriented examples need to be employed in the fine-tuning and preference alignment phases during future model upgrades to significantly improve the contextual reasoning of LLMs in scenarios like those exhibited by CHARPEVAL.

3.3 Analysis and Discussion

To better understand the LLMs limitations on CHARPEVAL, we analyze the distribution of response rankings on the hCHARP subset of the CHARPEVAL for a few selected models in Table 2. Other models, as well as on eCHARP subset, mostly exhibit similar result patterns. Detailed performance data are presented in Table 4 and Table 5 in Appendix C. We observe that, regardless of model family, size, or whether the model has been instruction-tuned, the distribution of errors is not uniformly random across the five distractor types. More precisely, we observe that selecting the wrong fact from k_{n+1} (**KGIR**) is the most dominant error type across all models, while selecting an ungrounded response (**KGUG**) is the least common error (near 0%). This suggests that the LLMs followed the prompt's instructions and attempted to reason to some extent in order to solve the task.

Model	GTRS	KGIR	KGUG	RECF	IECF	COEX
Zero-Shot						
Q-72B-IT	48%	33%	1%	8%	6%	4%
L3.1-70B-IT	47%	33%	0%	11%	7%	2%
L3.1-8B-IT	41%	48%	0%	6%	3%	2%
Q-7B-IT	46%	36%	1%	6%	5%	6%
Q-7B	18%	55%	0%	7%	4%	16%
L2-7B	10%	61%	0%	9%	7%	13%
Few-Shot						
L3.1-8B-IT	39%	56%	0%	3%	1%	1%
Q-7B-IT	46%	39%	0%	7%	3%	5%
Q-7B	25%	58%	0%	7%	4%	5%
L2-7B	20%	59%	0%	8%	9%	4%

Table 2: Models responses ranking distribution on the hCHARP subset of the CHARPEVAL benchmark under zero-shot and few-shot (3) settings. **GTRS** shows accuracy on the ground truth response, while the other columns indicate the distractor types described in § 2.3.

Although less significant ($\approx$ 10% across models) than **KGUG**, the errors on **RECF** and **IECF** highlight that, while the model can identify that the last user utterance relates to entities mentioned in the knowledge, it often gets distracted by common facts about these entities rather than focusing on the one mentioned in k_{n+1}. This pattern is consistent across instruction-tuned models, regardless of their size (e.g., Q-72B-IT and Q-7B-IT) or family (e.g., Q-72B-IT and L-70B-IT). This is further supported by the analysis conducted in Appendix C.1 on the distribution of models' attention scores.

Surprisingly, we found that the second most common error for base models in the zero-shot setting is their tendency to identify common expressions (**COEX**), such as idioms and chat phrases, as the most likely valid responses. **COEX** responses intuitively receive higher probabilities due to their high frequency in the pretraining corpus compared to other distractor types. Interestingly, we notice that few-shot examples in the prompt lead both the Qwen and Llama2 7B base models to reduce the error on **COEX** from 16% and 13% to 5% and 4%, respectively. These findings suggest that, despite the progress made in LLMs' reasoning, these models (especially pretrained ones) can still get partially distracted from the task and exhibit a bias toward favoring high-frequency textual segments.

4 Conclusion

We introduce CHARPEVAL, a benchmark that is both challenging and has a reproducible evaluation method to test modern LLMs' contextual reasoning abilities in knowledge-grounded dialogue tasks. We hope that our CHARPEVAL can guide the development of future LLM upgrades to address the limitations identified in this paper.

Limitations

Potential limitations of this work include not experimenting with other families of open-weight LLMs, such as the Gemma (Team et al., 2024), Phi (Abdin et al., 2024), or DeepSeek (Liu et al., 2024a; Bi et al., 2024), as well as larger models above 100B scale (e.g. Llama3.1-405B). The choice of models was primarily based on state-of-the-art models (within their size category) for knowledge and reasoning benchmarks (Hendrycks et al., 2021; Wang et al., 2024) during the submission period. Additionally, the study did not consider closed-source models such as GPT-4 (OpenAI, 2023) or Claude (AnthropicAI, 2023), as these powerful models have shown to perform quite well on CHARP response generation in (Ghaddar et al., 2024). Since CHARPEVAL is not a privately held dataset (e.g., in a leaderboard), there is a risk that it could become invalid for evaluation if it is incorporated into the training data of future LLM iterations. Finally, the CHARPEVAL annotation style could be extended to support a wider range of dialogue tasks, such as conversational search (Mo et al., 2024) or citation-based dialogue (Dehghan et al., 2024), in addition to other task types (Alfonso-Hermelo et al., 2021; Lu et al., 2021; Zhou et al., 2024) or languages (Ghaddar and Langlais, 2020; Ghaddar et al., 2021; Alghamdi et al., 2023).

Ethics Statement

The human annotators involved in this project are two NLP data labeling experts, each with over two years of experience in this field. They are hired as local contractors, working 40 hours per week for 2 months on this project, and are paid 60% above the local minimum hourly wage. The annotators received prior training and were provided with guidelines that included instructions and examples of both standard and extreme cases they could encounter during the annotation process. Furthermore, domain experts reviewed the annotations daily and held video meetings with the annotators as needed.

Acknowledgements

We would like to thank Ella Cho and Abdulmuizz Yusuf, the professional annotators without whom this work would have not been possible. We thank the anonymous reviewers for their insightful comments.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Toufique Ahmed, Christian Bird, Premkumar Devanbu, and Saikat Chakraborty. 2024. Studying llm performance on closed-and open-source data. *arXiv preprint arXiv:2402.15100*.

David Alfonso-Hermelo, Ahmad Rashid, Abbas Ghaddar, Philippe Langlais, and Mehdi Rezagholizadeh. 2021. Nature: Natural auxiliary text utterances for realistic spoken language evaluation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Asaad Alghamdi, Xinyu Duan, Wei Jiang, Zhenhai Wang, Yimeng Wu, Qingrong Xia, Zhefeng Wang, Yi Zheng, Mehdi Rezagholizadeh, Baoxing Huai, et al. 2023. Aramus: Pushing the limits of data and model scale for arabic natural language processing. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2883–2894.

AnthropicAI. 2023. Introducing claude.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023a. How is chatgpt's behavior changing over time? *arXiv preprint arXiv:2307.09009*.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023b. Theoremqa: A theorem-driven question answering dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7889–7901.

Mohammad Dehghan, Mohammad Alomrani, Sunyam Bagga, David Alfonso-Hermelo, Khalil Bibi, Abbas Ghaddar, Yingxue Zhang, Xiaoguang Li, Jianye Hao, Qun Liu, et al. 2024. Ewek-qa: Enhanced web and efficient knowledge graph retrieval for citation-based question answering systems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14169–14187.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*.

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 preprint arXiv:2407.21783*.

Nouha Dziri, Ehsan Kamalloo, Sivan Milton, Osmar Zaiane, Mo Yu, Edoardo M Ponti, and Siva Reddy. 2022a. Faithdial: A faithful benchmark for information-seeking dialogue. *Transactions of the Association for Computational Linguistics*, 10:1473–1490.

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. 2022b. Evaluating attribution in dialogue systems: The begin benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083.

Lizhou Fan, Wenyue Hua, Lingyao Li, Haoyang Ling, and Yongfeng Zhang. 2023. Nphardeval: Dynamic benchmark on reasoning ability of large language models via complexity classes. *arXiv preprint arXiv:2312.14890*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.

Abbas Ghaddar, David Alfonso-Hermelo, Philippe Langlais, Mehdi Rezagholizadeh, Boxing Chen, and Prasanna Parthasarathi. 2024. CHARP: Conversation history AwaReness probing for knowledge-grounded dialogue systems. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1534–1551, Bangkok, Thailand. Association for Computational Linguistics.

Abbas Ghaddar and Philippe Langlais. 2020. Sedar: a large scale french-english financial domain parallel corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3595–3602.

Abbas Ghaddar, Yimeng Wu, Ahmad Rashid, Khalil Bibi, Mehdi Rezagholizadeh, Chao Xing, Yasheng Wang, Duan Xinyu, Zhefeng Wang, Baoxing Huai, et al. 2021. Jaber and saber: Junior and senior arabic bert. *arXiv preprint arXiv:2112.04329*.

Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2018. A knowledge-grounded neural conversation model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024b. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Peng Lu, Abbas Ghaddar, Ahmad Rashid, Mehdi Rezagholizadeh, Ali Ghodsi, and Philippe Langlais. 2021. Rw-kd: Sample-wise loss terms re-weighting for knowledge distillation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3145–3152.

Fengran Mo, Abbas Ghaddar, Kelong Mao, Mehdi Rezagholizadeh, Boxing Chen, Qun Liu, and Jian-Yun Nie. 2024. Chiq: Contextual history enhancement for improving query rewriting in conversational search. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2268.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

Jiao Ou, Junda Lu, Che Liu, Yihong Tang, Fuzheng Zhang, Di Zhang, and Kun Gai. 2024. Dialogbench: Evaluating llms as human-like dialogue systems. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6137–6170.

Koustuv Sinha, Prasanna Parthasarathi, Jasmine Wang, Ryan Lowe, William L Hamilton, and Joelle Pineau. 2020. Learning an unreferenced metric for online dialogue evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2430–2441.

Annalisa Szymanski, Noah Ziems, Heather A Eicher-Miller, Toby Jia-Jun Li, Meng Jiang, and Ronald A Metoyer. 2024. Limitations of the llm-as-a-judge approach for evaluating llm outputs in expert knowledge tasks. *arXiv preprint arXiv:2410.20266*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

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*.

Jiaming Zhou, Abbas Ghaddar, Ge Zhang, Liheng Ma, Yaochen Hu, Soumyasundar Pal, Mark Coates, Bin Wang, Yingxue Zhang, and Jianye Hao. 2024. Enhancing logical reasoning in large language models through graph-based synthetic data. *arXiv preprint arXiv:2409.12437*.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2023. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*.

A Prompt Design and Implementation Details

We designed the instruction component through trial and error, examining the outputs of two base models—Qwen2.5 and Llama2 7B—along with instruction-tuned versions, until we confirmed that all models could follow the instructions and generate outputs in the required format. It is important to note that, as mentioned in the main paper, models are evaluated in perplexity-based response selection mode, while generative mode is used only for prompt engineering purposes. Here is the prompt related to the illustration example in Figure 1:

```
You are given a conversation history between a "User" and a "Bot", along with a piece of "Knowledge" containing factual information. Your goal is to produce a response to the User's last message, relying only on the provided Knowledge. Do not introduce any new information that is not present in the Knowledge. If the User asks about something that is not covered by the Knowledge, you may express uncertainty, but do not invent details.

User: I like to bake bread in my spare time. Do you?
Bot: Bots cannot bake, but did you know that this method of cooking can use hot ashes or hot stones?
User: I didn't know that. I also like cooking different things like cakes, stews, and broths.
Bot: Have you ever tried baking at a community center activity? They are well-equipped.
User: I've never tried it. What ingredients do I need to get started baking in my spare time?

Knowledge: Sugar and flour are among the essential ingredients used to make cake while bread is baked with flour, water, yeast and salt

Bot:
```

Model evaluation is performed as follows: both the ground truth response and the five distractors are independently scored by concatenating each with the above prompt, resulting in a total of 6 sequences. Perplexity is then calculated for each sequence, and the response from the sequence with the lowest perplexity is selected as the predicted response. It is worth noting that when a response is concatenated at the end of the prompt, perplexity is influenced by the fluency of the entire sequence, not just the response part. Since we have a controlled setting (each response option has a logic behind it), we draw conclusions about the model's behavior and its utilization of knowledge and contextual history based on which response leads to the lowest sequence perplexity.

For the few-shot setting, we progressively included in-context examples until the output of all models stabilized, showing little to no variation in the model responses. We set the number of in-context examples to 3, as adding more examples did not yield any further improvements. The few-shot samples were manually designed to ensure full alignment with the CHARP samples. The new prompt consists of placing the three few-shot samples (which are the same as the input example but with the ground truth response) between the instruction and the test sample. We carefully integrated CHARPEVAL as a new task into a local fork of the `lm-evaluation-harness` [2] library (Gao et al., 2024) and used it to perform standard evaluations of models imported from the Hugging Face Transformers library (Wolf et al., 2020).

A.1 Evaluation Design Choice

Another way to frame the evaluation is like a multiple-choice question, where all six options (the gold response and five distractors) are concatenated as response options with indexes (e.g., A, B, C, D, E, F), as in MMLU (Hendrycks et al., 2021) benchmark. The model can either be asked to generate the index of the most likely option as you propose, or the index of each option can be concatenated to the end of the prompt, and the perplexity of the lowest index can be measured. However, this task formulation has some remarks and limitations (based on our experiments) in the context of our work, which led us to decide against using it.

While LLMs are trained on multiple-choice question (MCQA) answering data during SFT, response selection for knowledge-grounded dialogue data is rare. It is unlikely that models have encountered such data during SFT. In contrast, multi-turn dialogue (including knowledge-grounded dialogue) chat datasets are common and abundant as training data for SFT, similar to MCQA. Therefore, the mismatch between what models have been tuned on is minimal (and a more natural occurring choice) with the dialogue task formulation compared to the response selection task formulation.

Using the option selection task formulation would require placing all response options in the same prompt, which may introduce biases for models under this specific setting. This is because, by design, our responses are not independent, which differs from common MCQA tasks. More precisely, the model may use the interaction between

[2] https://github.com/EleutherAI/lm-evaluation-harness

the response options themselves to eliminate certain choices, disregarding knowledge and context history. For instance, in Figure 1, the ground truth, **KGIR**, and **KGUG** response options all start with the same phrase (*Well, ...*), which the model may interpret as a hint to disregard the remaining three options. Alternatively, it may disregard the options containing *cake* simply because the other four options contain *bread*. In MCQA tasks, like coding and math, this is not an issue because the options are typically numbers (e.g., formula answers or program outputs), which do not have relationships with each other. Therefore, scoring each option independently using perplexity with a dialogue task formulation is less prone to bias compared to the option selection formulation.

Also, we observed that numerous LLMs are biased toward selecting a particular option based on its position (mainly option A), which was also pointed out in the literature (Zheng et al., 2024). Randomizing the options leads to results that are close across runs but not deterministic, which impacts the reproducibility of the results across models—our main motivation for creating CHARPEVAL.

It is worth noting that in common evaluation practices for LLMs benchmark evaluation on MCQA tasks, the models are not queried to generate the option index. Instead, the option indices are concatenated at the end of the prompt, and the one with the lowest perplexity is selected. This approach helps avoid cases where the model generates text that is either not in the option list or does not follow the expected format (or requires parsing the answer). In conclusion, both the dialogue task and response selection formulations lead to the same outcome (selecting the best answer). However, we choose the former as it is a more natural formulation of the task, avoids option bias, and is more deterministic (no option shuffling).

Finally, we would like to point out that human evaluation is not required in our scenario. This is because our evaluation is automatic and deterministic, similar to a multiple-choice question, where there is a single correct response and multiple (wrong) distractor responses. Human evaluation in the context of answer generation—is what CHARP does. Our contribution, CHARPEVAL, builds on top of CHARP to enable deterministic and reproducible evaluation while ensuring a challenging benchmark.

B Inter-annotator Agreement

First, it is important to clarify that in our case, inter-annotator agreement is tricky because we are not evaluating the correct or gold response directly (which is the usual approach in most annotation tasks). Instead, we annotate incorrect distractor responses, where the 'gold' correct answer is already provided to the annotators. In addition, many distractors (e.g., **IECF**, **COEX**) can have valid annotations, which limits the ability to measure similarities between the same distractors annotated twice by different annotators.

Since there is no ambiguity about the correct answer, we implemented measures during the annotation process to ensure consistency among different annotators in generating plausible wrong distractors that adhere to our rules. At the early stage of the annotation process, both annotators were asked to annotate a subset of 100 samples (roughly 10% of CHARPEval). We then measured the performance discrepancy of a model to ensure consistency in following our annotation rules.

Subset	**GTRS**	**KGIR**	**KGUG**	**RECF**	**IECF**	**COEX**
hCHARP						
all	24%	63%	0%	6%	4%	3%
100 (A#1)	23%	62%	1%	6%	3%	3%
100 (A#2)	24%	64%	0%	5%	4%	3%
eCHARP						
all	25%	61%	0%	6%	5%	3%
100 (A#1)	24%	60%	0%	7%	5%	4%
100 (A#2)	25%	62%	0%	6%	5%	3%

Table 3: Zero-shot performance of the LLaMA2-7B Instruct (L2-7B-IT) model on all samples ('all'), as well as on two subsets of 100 samples each, annotated by 2 different annotators (A#1 and A#2), on both hCHARP and eCHARP splits of the CHARPEVAL benchmark.

Table 3 shows the zero-shot performance of LLaMA2-7B-Instruct (selected as a representative example) on the full CHARPEVAL, as well as on the 100-sample subsets jointly annotated by both annotators for the hCHARP and eCHARP. The results indicate consistent annotations between Annotator #1 and Annotator #2, with only minor differences of at most 1%. This close alignment reflects our thorough preparatory training and the measures implemented during the annotation process to ensure both annotators adhered closely to the same guidelines throughout the project.

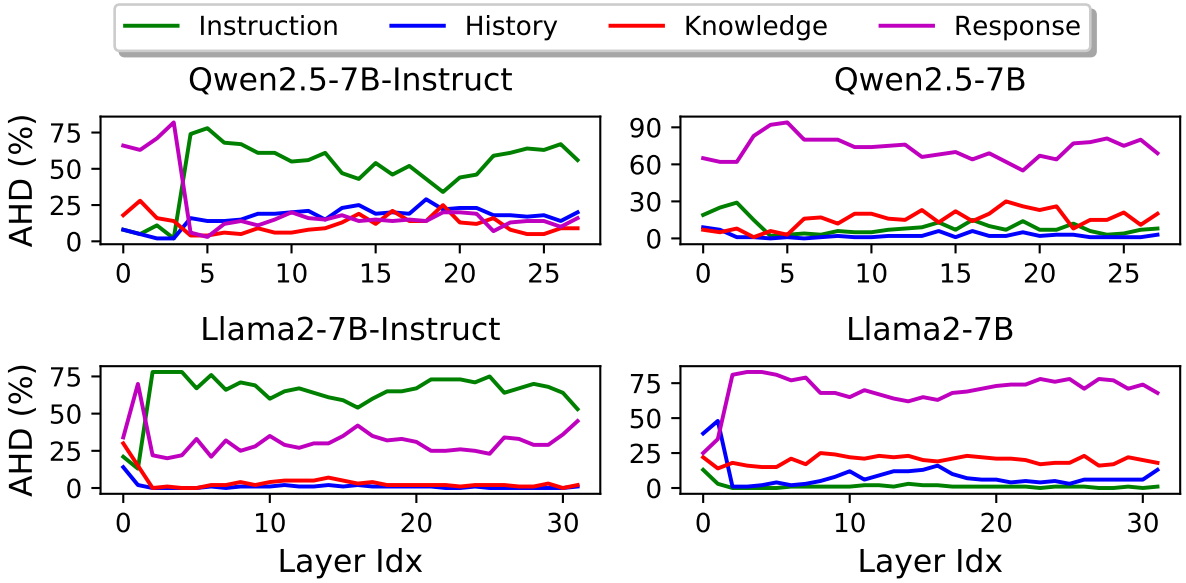

Figure 2: For each LLM layer (x-axis), the attention head distribution (y-axis) of response tokens is plotted, showing how they point to the four chunks of the input sequence: *Instruction*, *History*, *Knowledge*, and *Response*. Experiments are conducted on four LLMs using samples from the hCHARP subset of CHARPEVAL.

C Results

C.1 Attention Analysis

We leverage the attention map of LLMs to better understand which parts of the input are influencing the predicted response. To this end, we calculate the distribution of attention heads for each token in the response across the entire input sequence. More precisely, we split the sample into 4 chunks: the instruction, the history (including the last user turn), the given knowledge segment, and the response itself. Then, for each token in the response, we collect the chunk that each attention head points to, and compute the distribution across the four categories. Finally, we aggregate the distributions for all response tokens and normalize them (in the range of [0-100%]) into a distribution across the input sequence for each example.

This procedure is applied to each model layer, and the distribution is averaged across all samples in a given testbed. As shown in Figure 2, we conduct this experiment on hCHARP subset of CHARPEVAL[3] with namely 4 models: Qwen2.5-7B-Instruct, Qwen2.5-7B, Llama2-7B-Instruct, and Llama2-7B. These models are selected to focus the comparison between base and instruction-tuned models, as well as between two families of models of the same size but with a large performance gap. Qwen2.5-7B and Llama2-7B have 28 heads and 28 layers, and 32 heads and 32 layers, respectively.

On one hand, we notice that the response token attention is influenced by the instruction segment, with significantly less attention given to the other three segments. For instance, attention to the instruction segment is less than 40% for Qwen2.5-7B-Instruct and less than 50% for Llama2-7B-Instruct, respectively. Intuitively, this aligns with the role of supervised fine-tuning and preference alignment, which are primarily aimed at adapting the model to follow the task instructions.

On the other hand, we observe that attention in both base models is highly concentrated (with more than 60% in most layers) on the response tokens, followed by the knowledge to some extent (20%-30%), with only minimal attention given to the instruction and history segments (less than 10%). This is expected behavior for pre-trained models, where attention is typically more concentrated on local context, while long-distance dependencies may play a minor role in predicting the next token during unsupervised learning (Liu et al., 2024b). This contrastive observation of attention distribution between the base and instruction-tuned models suggests that instruction tuning alters LLM behavior not only by exposing models to task-relevant data but also by shifting their attention allocation across the input sequence.

Interestingly, we observe that Qwen2.5-7B-Instruct exhibits a similar attention distribution

[3]Similar trends are observed on eCHARP

range for both historical knowledge and responses, showing a distinct pattern compared to its base model. In contrast, Llama2-7B-Instruct allocates near zero attention to knowledge and history, but significantly higher attention to the response, showing a similar pattern (except on instruction) of its base model. This difference may help explain the 20% and 22% performance gaps between the two models on hCHARP and eCHARP, respectively.

Model	#P	IT	hCHARP						eCHARP					
			GTRS	KGIR	KGUG	RECF	IECF	COEX	GTRS	KGIR	KGUG	RECF	IECF	COEX
Qwen2.5	72B	✓	48%	33%	1%	8%	6%	4%	48%	32%	1%	9%	6%	4%
Qwen2.5	32B	✓	47%	38%	0%	8%	3%	4%	47%	36%	0%	9%	3%	5%
Qwen2.5	7B	✓	46%	36%	1%	6%	5%	6%	46%	37%	0%	6%	5%	6%
Qwen2.5	7B	✗	18%	55%	0%	7%	4%	16%	20%	53%	0%	7%	4%	16%
Qwen2.5	3B	✓	42%	46%	0%	5%	3%	4%	42%	44%	0%	7%	3%	4%
Llama3.3	70B	✓	47%	33%	0%	11%	7%	2%	47%	33%	0%	11%	7%	2%
Llama3.1	70B	✓	44%	38%	1%	7%	5%	5%	42%	40%	1%	7%	5%	5%
Llama3.1	8B	✓	41%	48%	0%	6%	3%	2%	41%	48%	0%	6%	3%	2%
Llama2	13B	✓	28%	66%	0%	3%	2%	1%	30%	64%	0%	3%	2%	1%
Llama2	7B	✓	24%	63%	0%	6%	4%	3%	25%	61%	0%	6%	5%	3%
Llama2	7B	✗	10%	61%	0%	9%	7%	13%	11%	57%	0%	10%	7%	15%
Mistralv0.1	52B	✓	46%	33%	0%	11%	4%	6%	46%	32%	0%	11%	5%	6%
Mistralv0.2	7B	✓	43%	33%	1%	12%	6%	5%	43%	33%	1%	13%	6%	4%

Table 4: Models responses ranking distribution on the hCHARP and CHARPEVAL subsets of the CHARPEVAL benchmark under zero-shot setting. **GTRS** shows accuracy on the ground truth response, while the other columns indicate the distractor types described in § 2.3.

Model	#P	IT	hCHARP						eCHARP					
			GTRS	KGIR	KGUG	RECF	IECF	COEX	GTRS	KGIR	KGUG	RECF	IECF	COEX
Qwen2.5	72B	✓	43%	43%	0%	7%	4%	3%	44%	41%	0%	8%	4%	3%
Qwen2.5	32B	✓	50%	38%	0%	6%	2%	4%	50%	37%	0%	6%	3%	4%
Qwen2.5	7B	✓	46%	39%	0%	7%	3%	5%	48%	35%	1%	7%	4%	5%
Qwen2.5	7B	✗	25%	58%	0%	7%	4%	5%	27%	60%	0%	7%	5%	1%
Qwen2.5	3B	✓	41%	51%	0%	3%	2%	3%	44%	47%	0%	4%	2%	3%
Llama3.3	70B	✓	48%	39%	0%	9%	4%	2%	49%	36%	0%	9%	4%	2%
Llama3.1	70B	✓	45%	42%	1%	6%	3%	3%	46%	41%	1%	6%	3%	3%
Llama3.1	8B	✓	39%	56%	0%	3%	1%	1%	37%	57%	0%	3%	2%	1%
Llama2	13B	✓	28%	66%	0%	3%	2%	1%	30%	64%	0%	3%	2%	1%
Llama2	7B	✓	26%	62%	0%	5%	5%	2%	25%	63%	0%	5%	5%	2%
Llama2	7B	✗	20%	59%	0%	8%	9%	4%	20%	57%	0%	9%	9%	5%
Mistralv0.1	52B	✓	49%	33%	0%	9%	4%	5%	52%	27%	1%	10%	5%	5%
Mistralv0.2	7B	✓	43%	34%	1%	12%	6%	4%	43%	36%	1%	12%	5%	3%

Table 5: Few Models responses ranking distribution on the hCHARP and CHARPEVAL subsets of the CHARPEVAL benchmark under zero-shot setting. **GTRS** shows accuracy on the ground truth response, while the other columns indicate the distractor types described in § 2.3.