Compos Mentis at SemEval2024 Task6: A Multi-Faceted Role-based Large Language Model Ensemble to Detect Hallucination

Souvik Das , Rohini K. Srihari
{souvikda, rohini}@buffalo.edu
Department of Computer Science and Engineering, University at Buffalo, NY.

Abstract

Hallucinations in large language models (LLMs), where they generate fluent but factually incorrect outputs, pose challenges for applications requiring strict truthfulness. This work proposes a multi-faceted approach to detect such hallucinations across various language tasks. We leverage automatic data annotation using a proprietary LLM, fine-tuning of the Mistral-7B-instruct-v0.2 model on annotated and benchmark data, role-based and rationale-based prompting strategies, and an ensemble method combining different model outputs through majority voting. This comprehensive framework aims to improve the robustness and reliability of hallucination detection for LLM generations. Code and data[1]

1 Introduction

The modern natural language generation (NLG) (OpenAI et al., 2023; Touvron et al., 2023) landscape faces two interconnected challenges: firstly, current neural models have a tendency to produce fluent yet inaccurate outputs, and secondly, our evaluation metrics are better suited for assessing fluency rather than correctness(Bang et al., 2023; Guerreiro et al., 2023). This phenomenon, known as "hallucination," (Ji et al., 2023) where neural networks generate plausible-sounding but factually incorrect outputs, is a significant hurdle, especially for NLG applications that require strict adherence to correctness. For instance, in machine translation(Lee et al., 2019), producing a fluent translation that deviates from the source text's meaning renders the entire translation pipeline unreliable. This issue may arise as LLMs are trained on vast amounts of data from the internet, which can contain inaccuracies, biases, and false information. Also, it may arise due improper representations learned during training even if good quality data is used. As a result, LLMs can sometimes hallucinate or fabricate details, especially when prompted to discuss topics outside their training data or make inferences beyond their capabilities.

Hallucination detection (Liu et al., 2022), also known as factual verification or truthfulness evaluation, identifies and mitigates these hallucinations in the outputs of LLMs. This is an active area of research and development, as it is crucial for ensuring the reliability and trustworthiness of LLM-generated content, particularly in high-stakes domains such as healthcare, finance, and legal applications. In this task, the primary focus will be to classify whether a generation is hallucinated.

This work proposes a multi-faceted approach to detecting hallucinations in large language models' outputs. We employ automatic data annotation using a proprietary LLM (Claude 2.1[2]) to label examples from the provided training set as hallucinated or not. Then we fine-tune the Mistral-7B-instruct-v0.2[3] model on this annotated data as well as the HaluEval benchmark (Li et al., 2023) to create two fine-tuned models. To improve performance, we use role-based prompting that casts the task in specific contexts like fact-checking. We also leverage rationale-based prompting, asking the LLM to justify its hallucination label. Finally, an ensemble method combines outputs from the fine-tuned Mistral models, Claude 2.1, and different prompting strategies via majority voting. This comprehensive approach aims to enhance the robustness and reliability of hallucination detection across various language tasks.

2 Task Details

This shared task (Mickus et al., 2024) aims to foster the growing interest within the community in ad-

[1] https://github.com/souvikdgp16/shroom_compos_mentis

[2] https://www.anthropic.com/news/claude-2

[3] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 1449–1454
June 20-21, 2024 ©2024 Association for Computational Linguistics

dressing this issue. Participants are tasked with performing binary classification to identify instances of fluent overgeneration hallucinations in two different setups: a model-aware track and a model-agnostic track. Essentially, participants must detect grammatically sound outputs that contain incorrect or unsupported semantic information, inconsistent with the source input, with or without having access to the model that produced the output.

To facilitate this task, participants are provided with a collection of checkpoints, inputs, references, and outputs from systems covering three different NLG tasks: definition modeling (DM), machine translation (MT), and paraphrase generation (PG). These systems will be trained with varying degrees of accuracy. The validation and test sets will include binary annotations from at least five annotators, with a majority vote determining the gold label.

2.1 Data

The data split is shown in Table 1.

Task	Validation	Test
model agnostic	500	1500
model aware	500	1500

Table 1: Data-split statistics.

Each data split file is formatted as a JSON list. Each element in this list corresponds to a data point as shown:


```
{
        "hyp": "(uncountable) The study of trees.",
        "ref": "tgt",
        "src": "It is now generally supposed that the
forbidden fruit was a kind of citrus , but certain facts
connected with  arborolatry  seem to me to disprove this
opinion .",
        "tgt": "The worship of trees.",
        "model": "",
        "task": "DM",
        "labels": [
            "Hallucination",
            "Hallucination",
            "Hallucination"
        ],
        "label": "Hallucination",
        "p(Hallucination)": 1.0
}
```


Each data instance contains the following key elements: a task (task) indicating the language model's objective; a source (src) input; a target reference (tgt); a hypothesis (hyp) which is the model's actual output; a set of per annotator hallucination labels (labels); a majority-based gold hallucination label (label); and a probability score (p(Hallucination)) representing the proportion of annotators who labeled the instance as hallucinated.

2.2 Evaluation Protocol

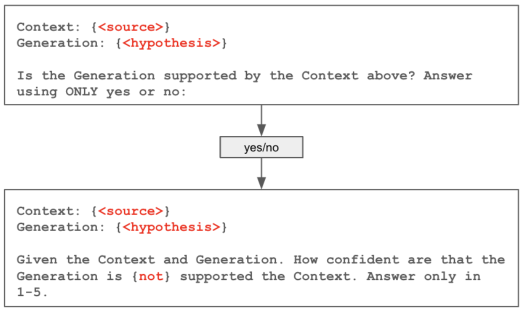
Submissions are evaluated using two criteria:

1. **Accuracy**: the system accuracy reached on the binary classification.
2. ρ: the Spearman correlation of the systems' output probabilities with the proportion of the annotators marking the item as overgenerating.

3 System Description

3.1 Automatic Data Annotation

We automatically annotate the unlabeled training data provided by the organizers. We use a strong proprietary Large Language Model(LLM) Claude 2.1 to annotate the data automatically. Since, annotations from Claude 2.1 might not be fully reliable we use a confidence-based measure to select only those training examples where the LLM is confident enough. We use the following prompts:



First, we prompt the LLM to get the hallucination label. Then we again prompt the LLM to do a retrospect on the decision it has made by asking it how confident it is with the decision. We filter out all the examples with a score less than 5.

3.2 Fine-tuning Mistral-7B-instruct-v0.2

We train two fine-tuned versions of Mistral-7B-instruct-v0.2 for this task:

Fine-tuned on our data: We split our automatically annotated dataset in 8:1:1 split for training, validation and testing. We adopt a generative approach for classification where the instruction was fed in this fashion: `[INST]`*prompt*`[/INST]`, where the *prompt* is the same as it is used during annotation phase. The goal is to generate the hallucination label. The test F1-score was 82.03%. We name this model as Mistral-7B-instruct-v0.2-halu-internal.

Fine-tuned on HaluEval dataset: Hallucination Evaluation benchmark for Large Language Models (HaluEval), a large collection of generated and

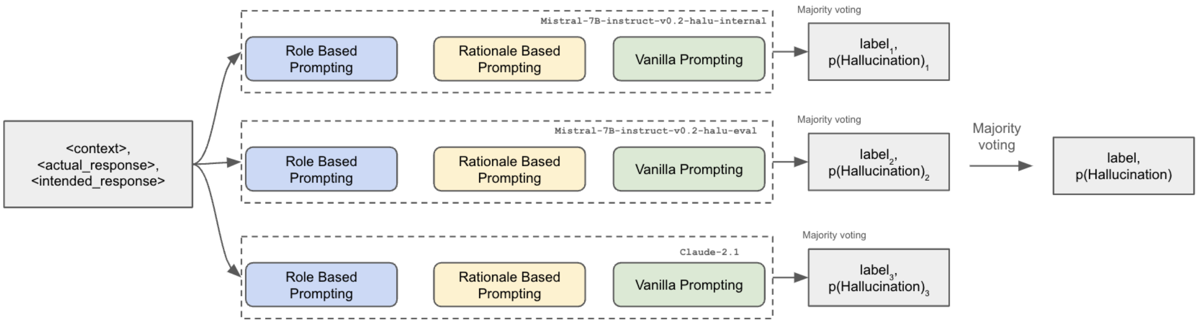


Figure 1: Our overall ensemble-based inference pipeline. We use majority voting at the model level and overall pipeline level to determine the final hallucination label.

human-annotated hallucinated samples for evaluating the performance of LLMs in recognizing hallucination HaluEval dataset contains $30,000$ hallucinated samples with $10,000$ examples for each task of QA, dialogue, and summarization. Here also, we adopt a generative approach for classification with the same instruction sequence as used during fine-tuning using our data. The test F1-score was 77.95%. We name this model as Mistral-7B-instruct-v0.2-halu-eval.

Hyperparameters: We use the original weights of Mistral-7B-instruct-v0.2 released by Mistral AI. We use QLoRA(Dettmers et al., 2023) for parameter-efficient fine-tuning. We set the maximum length of the input sequence to 512 and the rank k and α in QLoRA to 16 and 8, respectively. We use the `bitsandbytes` library to initialize the QLoRA parameters. We use an 8-bit Paged Adam optimizer to update QLoRA parameters with a batch size of 64 and learning rates of 1e-7. The trainable QLoRA parameters ($\sim$ 19.5M) are fine-tuned on 2 NVIDIA A5000-24GB GPUs. All the hyperparameter are tuned using the provided trial data, k and α were varied in the range of [4,16] with a step of 4, batch size was varied in the range of [32,72] with a step of 16, and the learning rate was varied from 1e-8 to 1e-7, the best performing hyperparameters are reported.

3.3 Role Based Prompting

Since we are dealing with multiple tasks, the same prompt might not be suitable for all the tasks during inference. We create task-specific role-based prompt for each task using the following prompt template:

```
Given the <intended_response>, <context> and
<actual_response> :

<intended_response>: {tgt}.
<context>: {src}.
<actual_response>: {hyp}.

{ROLE}

State whether the <actual_response> supports <context> and
<intended_response>. Answer using ONLY yes or no:
```

<intended_response> is the golden response, <actual_response> is the actually generated response. Here the inference-time roles will be based on the following Table:

	Role
Definition Modelling	Imagine yourself as a fact-checker; your job is to check whether <actual_response>is the definition of <context>.
Paraphrase Generation	Imagine yourself as a paraphrase-checker; your job is to check whether <actual_response> is an actual paraphrase of <context>. That means the meaning of <actual_response> should be the same as <context>, however <actual_response> will contain lesser words than <context>.
Machine Translation	Imagine yourself as a translation-checker; your job is to check whether <actual_response> is an actual translation of <context>.

Table 2: Role Definitions.

3.4 Rationale Based Prompting

We notice that when LLMs are prompted to produce rationale for its decision it often elicits more truthful response. Due to this observation we prompt the LLM to generate the explanation classifying the generation is hallucinated or not. The prompt is as follows:

```
Given the <intended_response>, <context> and
<actual_response> :

<intended_response>: {tgt}.
<context>: {src}.
<actual_response>: {hyp}.

{ROLE}

Come up with an explanation of whether the <actual_response>
supports <context> and <intended_response>.
Using the explanation, choose a final answer between yes or
no.
The final answer should be in this format:
{"explanation":<explanation>, "final_answer":<yes or no>}
```

3.5 Inference Ensemble

We combine all our prompting strategies to simulate an annotator for each sample. Also, we create an ensemble of three models: (1) Mistral-7B-instruct-v0.2-halu-internal (2) Mistral-7B-instruct-v0.2-halu-eval (3) Claude 2.1. Along with the role-based and rationale-based prompting we also incorporate a vanilla prompting where we just ask the LLM to come up with the hallucination label without assuming any role or generating a rationale, like this:

```
Given the <intended_response>, <context> and
<actual_response> :

<intended_response>: {tgt}.
<context>: {src}.
<actual_response>: {hyp}.

State whether the <actual_response> supports <context> and
<intended_response>. Answer using ONLY yes or no:
```

For each model pipeline, we get 3 hallucination labels; the pipeline label is the most common label out of 3. The hallucination probability score is determined by this equation: $p(\text{Hallucination}) = \frac{\#\text{halluciation\_labels}}{3}$. We get the hallucination label and $p(\text{Hallucination})$ for the three pipelines, and again we do a majority voting to get the final hallucination label. The final $p(\text{Hallucination})$ is set to the maximum probability of the selected hallucination label across the pipeline. We use greedy decoding for the Mistral-based models with a temperature of 0.8. Average cost of running Claude APIs for each is about 7$ for validation set and 16$ for test set. For Claude inference we use a temperature of 0.9.

4 Results

Table 3 and 4 show the results for model-aware and model-agnostic hallucination detection tasks for validation split. For both cases, we notice increased performance with rationale-based prompts for all the models. Subsequently, our ensemble-based pipeline boosts the performance even more. On the other hand, the performance of the HaluEval fine-tuned dataset is superior to our annotated dataset because there is a large possibility of noise getting introduced during our annotation process. Our annotation process uses verbalized model confidence as a proxy for data filtration; if the model is not calibrated correctly, this might lead to a faulty filtration process.

Configuration	Prompting Technique	Accuracy	Rho
Mistral-7B-instruct-v0.2-halu-internal	role-based	0.711	0.562
Mistral-7B-instruct-v0.2-halu-eval	role-based	0.724	0.588
Claude2.1	role-based	0.723	0.563
Mistral-7B-instruct-v0.2-halu-internal	rationale-based	0.724	0.566
Mistral-7B-instruct-v0.2-halu-eval	rationale-based	0.73	0.564
Claude2.1	rationale-based	0.728	0.566
Mistral-7B-instruct-v0.2-halu-internal	vanilla	0.712	0.562
Mistral-7B-instruct-v0.2-halu-eval	vanilla	0.72	0.553
Claude2.1	vanilla	0.712	0.565
3-model-ensemble	all	0.738	0.568

Table 3: Validation results for model-agnostic task.

Configuration	Prompting Technique	Accuracy	Rho
Mistral-7B-instruct-v0.2-halu-internal	role-based	0.713	0.568
Mistral-7B-instruct-v0.2-halu-eval	role-based	0.733	0.576
Claude2.1	role-based	0.723	0.556
Mistral-7B-instruct-v0.2-halu-internal	rationale-based	0.726	0.567
Mistral-7B-instruct-v0.2-halu-eval	rationale-based	0.723	0.569
Claude2.1	rationale-based	0.731	0.572
Mistral-7B-instruct-v0.2-halu-internal	vanilla	0.708	0.562
Mistral-7B-instruct-v0.2-halu-eval	vanilla	0.723	0.533
Claude2.1	vanilla	0.726	0.566
3-model-ensemble	all	0.736	0.579

Table 4: Validation results for model-aware task.

Task	Configuration	Accuracy	Rho
model agnostic	3-model-ensemble	0.738	0.595
model aware	3-model-ensemble	0.756	0.566

Table 5: Evaluation results.

During evaluation, we ran our best-performing pipeline i.e., the ensemble of 3 models. A performance similar to the validation set is observed here. Our team ranked 33 out of 48 for model agnostic sub-task and 29 out of 45 for model aware sub-task.

5 Conclusion

This work proposes a multi-faceted approach for detecting hallucinations in large language model outputs across various natural language tasks. It employs automatic data annotation, fine-tuning state-of-the-art models on annotated data and benchmarks, role-based and rationale-based prompting strategies, and an ensemble method combining multiple model outputs. The ensemble pipeline achieves promising results on model-agnostic and model-aware evaluation settings for hallucination detection. While challenges remain, this comprehensive framework highlights the potential of carefully designed prompting, model fine-tuning, and ensembling techniques to enhance the robustness and reliability of factual verification in language model generations, paving the way for developing more trustworthy natural language generation systems.

Acknowledgements

We thank the anonymous reviewers for providing valuable feedback on our manuscript. This work is partly supported by NSF grant number IIS-2214070. The content in this paper is solely the responsibility of the authors and does not necessarily represent the official views of the funding entity.

References

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Nuno M. Guerreiro, Duarte Alves, Jonas Waldendorf, Barry Haddow, Alexandra Birch, Pierre Colombo, and André F. T. Martins. 2023. Hallucinations in large multilingual translation models.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2019. Hallucinations in neural machine translation.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. Halueval: A large-scale hallucination evaluation benchmark for large language models.

Tianyu Liu, Yizhe Zhang, Chris Brockett, Yi Mao, Zhifang Sui, Weizhu Chen, and Bill Dolan. 2022. A token-level reference-free hallucination detection benchmark for free-form text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6723–6737, Dublin, Ireland. Association for Computational Linguistics.

Timothee Mickus, Elaine Zosa, Raúl Vázquez, Teemu Vahtola, Jörg Tiedemann, Vincent Segonne, Alessandro Raganato, and Marianna Apidianaki. 2024. Semeval-2024 shared task 6: Shroom, a shared-task on hallucinations and related observable overgeneration mistakes. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1980–1994, Mexico City, Mexico. Association for Computational Linguistics.

OpenAI, :, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan

Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2023. Gpt-4 technical report.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.