UMUTeam at SemEval-2024 Task 6: Leveraging Zero-Shot Learning for Detecting Hallucinations and Related Observable Overgeneration Mistakes

Ronghao Pan[1], **José Antonio García-Díaz**[1], **Tomás Bernal-Beltrán**[1], **Rafael Valencia-García**[1]
[1] Facultad de Informática, Universidad de Murcia, Campus de Espinardo, 30100 Murcia, Spain
{ronghao.pan, joseantonio.garcia8, tomas.bernalb, valencia}@um.es

Abstract

In these working notes we describe the UMUTeam's participation in SemEval-2024 shared task 6, which aims at detecting grammatically correct output of Natural Language Generation with incorrect semantic information in two different setups: model-aware and model-agnostic tracks. The task is consists of three subtasks with different model setups. Our approach is based on exploiting the zero-shot classification capability of the Large Language Models LLaMa-2, Tulu and Mistral, through prompt engineering. Our system ranked eighteenth in the model-aware setup with an accuracy of 78.4% and 29th in the model-agnostic setup with an accuracy of 76.9333%.

1 Introduction

Recently, the emergence of Large Language Models (LLMs) has brought about a paradigm shift in Natural Language Processing (NLP), leading to unprecedented advances in Natural Language Understanding (NLU) (Huang et al., 2023) and Reasoning (Zhang et al., 2023). In general, LLMs refer to a set of general-purpose models based on the Transformer architecture and pre-trained on large text corpora, such as GPT-3 (Brown et al., 2020), LLaMa (Touvron et al., 2023), PaLM (Chowdhery et al., 2023) and GPT-4 (Achiam et al., 2023). By scaling the amount of data and model capacity, LLMs demonstrate incredible emergent capabilities, typically including In-Context Learning (ICL) (Brown et al., 2020), chain-of-thought prompting (Wei et al., 2022), and instruction following (Peng et al., 2023).

Natural Language Generation (NLG) faces two related challenges. First, current models often produce output that is fluent but inaccurate. Second, the metrics used to evaluate the LLMs performance prioritize fluency over correctness, exacerbating the problem of "hallucination", in which LLMs produce fluent but incorrect output. Consequently, significant research in underway to automatically detect such errors. In many NLG applications, output correctness is paramount, as in cases such as machine translation, where producing a plausible but inconsistent translation compromises the utility of the system.

Thus, the SHROOM shared-task focuses on identifying grammatically correct outputs that contain incorrect semantic information, regardless of whether the model producing the output is accessible or not (Mickus et al., 2024). To this end, the organizers have adapted a post-hoc environment in which the models have already been trained, and the outputs have already been produced. The participants' task is a binary classification problem to identify cases of hallucinations, i.e. to detect grammatically correct outputs that contain incorrect or unsupported semantic content, in two different setups: model-aware and model-agnostic tracks.

To address the SHROOM challenge, our team used a zero-shot learning (ZSL) approach with LLaMa-2 (Touvron et al., 2023), Mistral (Jiang et al., 2023), and Tulu (Ivison et al., 2023) LLMs to detect grammatically correct output that contains incorrect semantic information through the prompt. The ZSL technique refers to the ability of LLMs to perform tasks without being explicitly trained on them, meaning that the model can generate responses or make predictions on topics or domains that were not part of its explicit training. This is achieved by exploiting the general knowledge that the models have acquired during their massive pretraining on large text corpora.

During our experiments, we observed that these LLMs were able to identify hallucinations. In particular, Tulu is the one best suited for this task.

The rest of this paper is organized as follows. Section 2 provides a summary of important details about the task setup. Section 2 gives an overview of our system for two subtasks. Section 4 presents the specific details of our systems. Section 5 dis-

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 675–681
June 20-21, 2024 ©2024 Association for Computational Linguistics

cusses the results of the experiments, and finally, the conclusions are presented in Section 6.

2 Background

NLG is a branch of Artificial Intelligence (AI) and computational linguistics that deals with the automated generation of text in human language. NLG covers a wide range of tasks, such as text generation for chatbots, automatic summarization, machine translation, story generation, and others. NLG relies on models and algorithms that enable machines to understand and generate text that is coherent and intelligible to humans. However, current models can produce inaccurate but fluent output, while the metrics tend to describe fluency rather than correctness. This leads to models producing "hallucinations", i.e. generated content that appears nonsensical or unfaithful to the given source content.

In general, hallucinations in NLG tasks can be divided into two main types (Ji et al., 2023): intrinsic hallucinations and extrinsic hallucinations. On the one hand, intrinsic hallucinations refer to the output of LLMs that conflict with the source content. On the other hand, extrinsic hallucinations refer to LLM generations that cannot be verified from the source content.

The SHROOM task aims to automatically detect hallucinations and related observable overgeneration errors. To achieve this, the organizers have provided a collection of checkpoints, inputs, references and outputs from systems covering three different NLG tasks: (1) definition modeling (DM), (2) machine translation (MT), and (3) paraphrase generation (PG), trained with different levels of accuracy. The development set includes binary annotations from at least five different annotators and a majority vote gold label.

The generalizability of LLMs is very attractive because it allows us to adapt state-of-the-art methods to specific goals. For example, an LLM trained on multilingual texts can perform translations without being explicitly trained to do so (known as zero-shot capability, ZSL). Another possibility is to guide models by providing them with examples of the input and the desired output (known as few-shot learning, FSL). For example, in (García-Díaz et al., 2023), LLMs have shown good performance in a ZSL scenario for identifying hate speech in Spanish and English. In this sense, it is possible to ask for a sentence and its translation before asking it to translate another sentence. This additional information helps to improve the quality of the output. For text classification tasks, the ability to make such predictions with little or no training makes these models particularly promising for empirical research, as they have the potential to perform accurately without the need for costly and time-consuming annotation procedures.

Therefore, we took advantage of this ZSL classification capability of LLMs to detect hallucinations and related observable overgeneration errors.

The following models are evaluated:

- **Mistral** (Jiang et al., 2023). Higher model performance often requires an escalation in model size. However, this scalability tends to increase computational cost and inference latency, raising the barriers to implementation in practical real-world scenarios. Mistral 7B is a high-performance LLM that maintains efficient inference. Mistral 7B outperforms the 13 billion parameter LLaMa-2 model on all benchmarks. In addition, Mistral 7B approaches the coding performance of Code-Llama 7B without sacrificing performance on non-code benchmarks.

- **LLaMa-2** (Touvron et al., 2023). Llama 2 and Llama 2-Chat are pre-trained and fine-tuned LLMs, both at scales of up to 70B parameters. In several benchmarks tested, Llama 2-Chat models generally outperformed existing open-source models. For our system, we used an instructively fine-tuned version of LLaMa-2 with 7B parameters from the Orca (Mukherjee et al., 2023) set called "stabilityai/StableBeluga-7B[1]".

- **Tulu** (Ivison et al., 2023). TuLu is a family of pre-trained and fine-tuned LLMs. Unlike other existing LLMs, distilled data mixtures from TuLu have been shown to significantly improve downstream performance over instruction and datasets available, with a new mixture outperforming its predecessor by an average of 8%. In addition, TuLu models use a fine-tuned version of Direct Preference Optimization (DPO) that scales to 70 billion parameter models and significantly improves open-response generation metrics without compromising model performance, im-

[1] https://huggingface.co/stabilityai/StableBeluga-7B

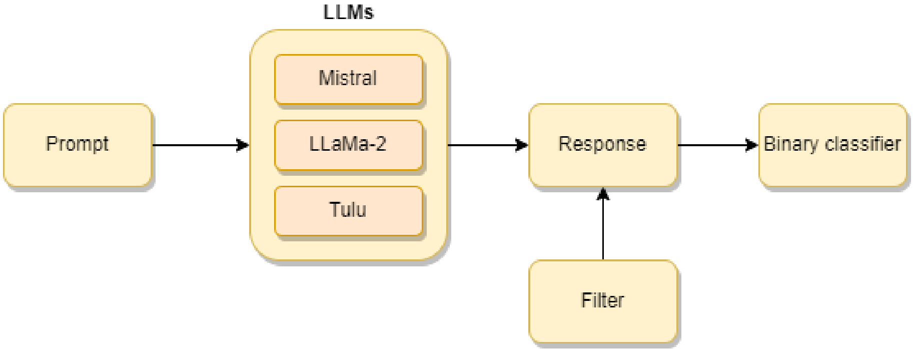


Figure 1: System architecture approach

proving AlpacaEval performance by an average of 13% across all model scales. For this task, we have evaluated the 7 billion parameter DPO version of Tulu with called "tulu-2-dpo-7b".[2]

3 System overview

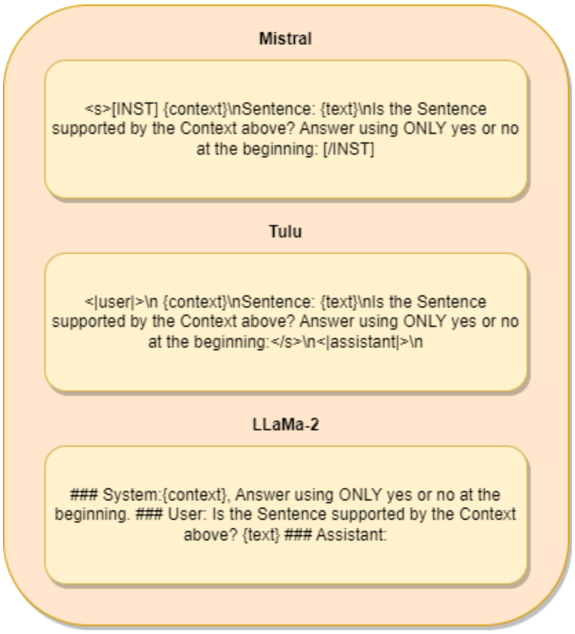
Figure 1 shows the architecture of our system. We can see that we have introduced a specific prompt for each LLM to generate a response with the desired structure. Then we have a module called "filter" that extracts a binary response based on the response and the correlation value.

3.1 Prompt

The prompt in the context of LLMs refers to a specific input provided to the model to elicit a desired response or to guide the text generation. This prompt can be a sentence, a question, or even a fragment of text that sets the context or direction for the model's text generation. In our proposal, we use prompt engineering, which involves the design and careful wording of these prompts to elicit specific model responses and optimally influence the model's response.

Figure 2 shows the prompts used for each LLM, in which we can see that each LLM has its own control tokens to indicate which parts are system control sequences and which parts are user questions. For example, in LLaMa-2 "`### System`" is used to indicate the control sequence, and "`### User`" is used to indicate the user question. However, Mistral and Tulu do not have tokens to indicate system control sequences, but we can append the control sequence to the user question.

In our system, we have used the same prompt structure for all LLMs: (1) **System control sequence**. It is used to specify the context and instruct the model to respond only with "yes" or "no" at the beginning of the response; and (2) **User question**. It is used to introduce the text and specify the question "the Sentence supported by the Context above". Once the response generated by the LLMs is obtained, it is passed through the filter module, which identifies the first word of the response and classifies it as "Hallucination" or "Not Hallucination". To obtain the correlation value, we have used the same approach as the baseline provided by the organizers, which consists of extracting the log probability value of the first token of the response generated by the LLM.



Mistral

```
<s>[INST] {context}\nSentence: {text}\nIs the Sentence supported by the Context above? Answer using ONLY yes or no at the beginning: [/INST]
```

Tulu

```
<|user|>\n {context}\nSentence: {text}\nIs the Sentence supported by the Context above? Answer using ONLY yes or no at the beginning:</s>\n<|assistant|>\n
```

LLaMa-2

```
### System:{context}, Answer using ONLY yes or no at the beginning. ### User: Is the Sentence supported by the Context above? {text} ### Assistant:
```

Figure 2: Examples of prompts for each LLMs

4 Experimental setup

In this section, we explain the dataset used, the hyperparameters used in the LLMs to generate re-

[2] https://huggingface.co/allenai/tulu-2-dpo-7b

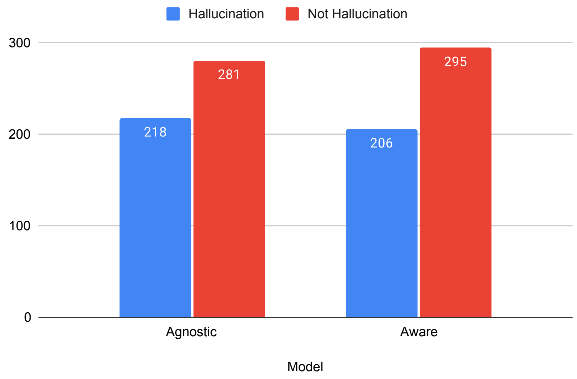


Figure 3: Validation set distribution

sponses, and details of the metrics used by the organizers for evaluation.

In this task, the organizers provided participants with unlabeled train data, trial data, and validation data for both the model-aware and model-agnostic setups. We have only used the validation data to evaluate the performance of different models using a ZSL approach. Figure 3 displays the distribution of the validation set.

The hyperparameters used in the LLMs to generate the response are 0.95 for top_p, 0 for top_k, 256 for max_new_tokens and the default temperature for each LLM.

The evaluation metrics used are accuracy for binary classification and rho to evaluate correlation. The rho metric, commonly known as the rho correlation coefficient (ρ), is a statistical measure that evaluates the relationship between two ordinal variables. It is particularly useful when the variables are not continuous but are divided into ordered categories.

5 Results

Table 1 shows the results of different LLMs in the validation set with two different configurations: (1) model-aware and (2) model-agnostic tracks. Thus, the system has to identify when a text is grammatically correct but contains incorrect information inconsistent with the source input, either with or without access to the model that produced the text. We can see that the Tulu performed best in both the model-aware and model-agnostic configurations. It obtained an accuracy of 76.6467% and a rho of 0.521104 in the model-aware configuration and an accuracy of 73.7475% and a rho of 0.553962 in the model-agnostic metric.

According to the results with development, we used Tulu in the task. Table 2 shows the official ranking for the task. We achieved the eighteenth best result out of a total of 46 teams in the model-aware setup, with a precision of 78.4% and a rho of 0.506895. Compared to result to the best result, our model is 2.866% worse in precision and 19.25% worse in rho. Regarding the model independent setup, our system achieved the nineteenth best result out of 49 participants, with a precision of 76.9333%, which is 7.8% worse than the best team (ahoblitz), and a rho of 0.560945, which is 20.86% worse than the best team.

Table 1: Results obtained with different LLMs in the validation set.

LLM	Accuracy	Rho
	Aware	
Mistral	52.2954	0.345239
Tulu	**76.6467**	**0.521104**
LLaMa-2	66.8663	0.487483
	Agnostic	
Mistral	50.3006	0.229504
Tulu	**73.7475**	**0.553962**
LLaMa-2	65.5310	0.521414

Table 2: Official raking table

LLM	Rank	Accuracy	Rho
		Aware	
HaRMoNEE	1	81.2666	0.699316
ahoblitz	2	80.6000	0.714712
TU Wien	3	80.6000	0.707192
	...		
UMUTeam	18	**78.4000**	**0.506895**
		Agnostic	
ahoblitz	1	84.7333	0.769512
OPDAI	2	83.6000	0.732195
HIT_WL	3	83.0666	0.767700
	...		
UMUTeam	29	**76.9333**	**0.560945**

5.1 Error analysis

We perform an error analysis of our system. For this, we extracted the confusion matrix from Tulu on the test set of the two configurations (model-aware and model-agnostic).

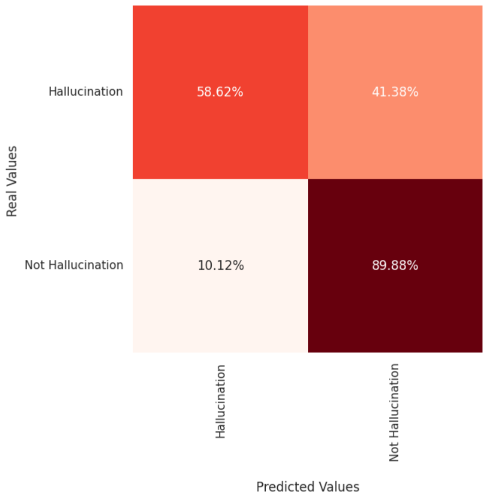


Figure 4: Confusion matrix of Tulu with test dataset in model-aware setup.

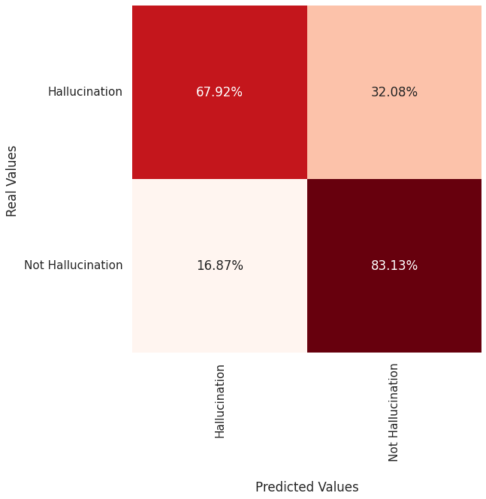


Figure 5: Confusion matrix of Tulu with test dataset in model-agnostic setup.

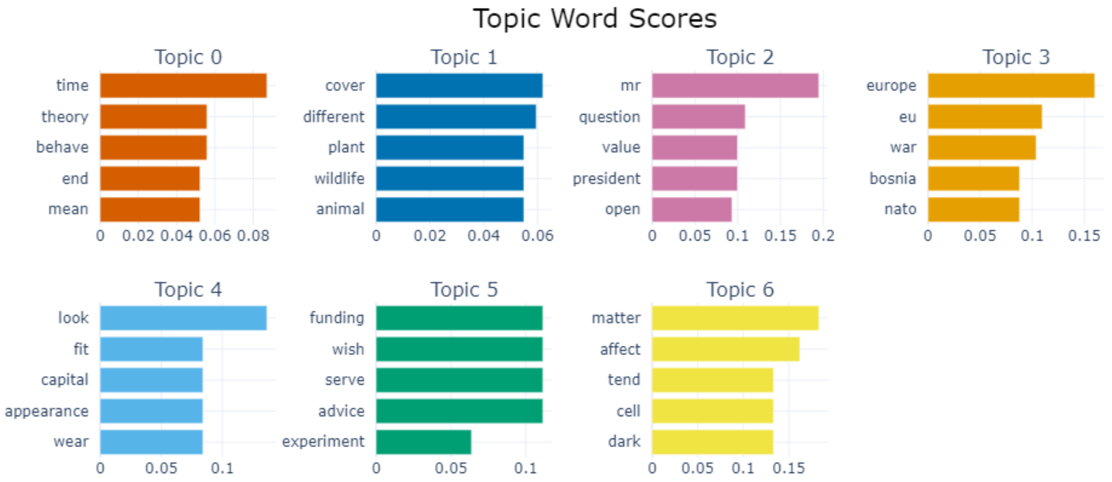


Figure 6: The most frequent topics associated with misclassification in model-aware setup.

Figure 4 shows the confusion matrix of TuLu from the test set in the model-aware setup. This approach tends to confuse hallucinations with non-hallucinations with a probability of 41.38%. However, it performs very well at detecting “not hallucination” with a probability of 89.88%. Regarding the model-agnostic setup, our model tends to confuse hallucinations with "not hallucination" with a probability of 32.08%, but is able to identify “not hallucination” with an accuracy of 83.13%.

The Tulu model from the test set in the model-aware setup has obtained a total of 324 misclassifications, of which 165 are of the definition modeling type, 100 are of the machine translation type, and 59 are of the paraphrase generation type. Therefore, we have a total of 165 misclassifications with the Flan-T5[3] model, 100 with the NLLB[4] model, and 59 with the Pegasus Paraphrase[5] model. In order to know the most common topic that the model comments on the classification error, we used the BERTopic model to identify and group topics in the context of the failed cases. In Figure 6 we can see the 7 topics in which the TuLu model usually misidentifies.

Regarding the model-agnostic setup, our ap-

[3] ltg/flan-t5-definition-en-base
[4] facebook/nllb-200-distilled-600M
[5] tuner007/pegasus_paraphrase

proach has obtained a total of 346 misclassifications, of which 138 are of the definition modeling type, 107 are of the machine translation type, and 101 are of the paraphrase generation type. In contrast to the model-aware setup, there is an increase in the accuracy of the identification of definition modeling misclassifications, but a decrease in the identification of paraphrase generation misclassifications. Figure 7 shows the three most common topics associated with the classification errors.

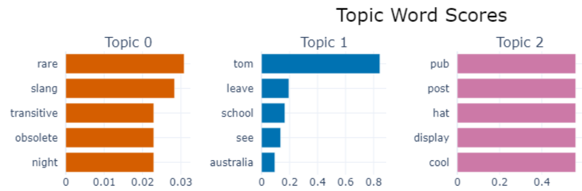


Figure 7: The most frequent topics associated with misclassification in model-agnostic setup.

6 Conclusion

Here we describe the UMUTeam's participation in SHROOM (SemEval 2024), concerning the development of models for detecting grammatically correct output from NLGs, but with incorrect semantic information in two different setups: model-aware and model-agnostic tracks. We have used the ZSL approach with LLaMa-2 (Touvron et al., 2023), Mistral (Jiang et al., 2023), and Tulu (Ivison et al., 2023) LLMs to detect output that contains incorrect semantic information through the prompt. Tulu performed best in the evaluation set. Using this model, we ranked eighteenth in the model-aware setup with an accuracy of 78.4% and nineteenth in the model-agnostic setup with an accuracy of 76.9333%.

As further work, we propose to investigate hallucination detection in the political domain. In politics, automated content generation can help politicians to generate text on a variety of political topics, which can help political campaigns, think tanks, and government agencies quickly produce tailored content. Hallucination detection can help to mitigate misleading or fabricated content. In this sense, we propose to generate political discourse that imitates politicians from different political wings (García-Díaz et al., 2022) and to identify the generated hallucinations by different LLMs.

Acknowledgements

This work is part of the research projects LaTe4PoliticES (PID2022-138099OB-I00) funded by MICIU/AEI/10.13039/501100011033 and the European Regional Development Fund (ERDF)-a way to make Europe and LT-SWM (TED2021-131167B-I00) funded by MICIU/AEI/10.13039/501100011033 and by the European Union NextGenerationEU/PRTR. In addition, this work was funded by the Spanish Government, the Spanish Ministry of Economy and Digital Transformation through the Digital Transformation through the "Recovery, Transformation and Resilience Plan" and also funded by the European Union NextGenerationEU/PRTR through the research project 2021/C005/0015007. Mr. Ronghao Pan is supported by the "Programa Investigo" grant, funded by the Region of Murcia, the Spanish Ministry of Labour and Social Economy and the European Union - NextGenerationEU under the "Plan de Recuperación, Transformación y Resiliencia (PRTR)".

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

José Antonio García-Díaz, Salud M Jiménez Zafra, María Teresa Martín Valdivia, Francisco García-Sánchez, Luis Alfonso Ureña López, and Rafael Valencia García. 2022. Overview of politices 2022: Spanish author profiling for political ideology. *Procesamiento del Lenguje Natural*.

José Antonio García-Díaz, Ronghao Pan, and Rafael Valencia-García. 2023. Leveraging zero and few-shot learning for enhanced model generality in hate speech detection in spanish and english. *Mathematics*, 11(24):5004.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, et al. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. *arXiv preprint arXiv:2305.08322*.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023. Camels in a changing climate: Enhancing lm adaptation with tulu 2.

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.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

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.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

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*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B Hashimoto. 2023. Benchmarking large language models for news summarization. *arXiv preprint arXiv:2301.13848*.