This is *not* a Disimprovement: Improving Negation Reasoning in Large Language Models via Prompt Engineering

Joshua Dias Barreto
IIT Bhubaneswar, India
21cs01075@iitbbs.ac.in

Abhik Jana
IIT Bhubaneswar, India
abhikjana@iitbbs.ac.in

Abstract

Negation reasoning remains a challenge for large language models (LLMs), often causing incorrect interpretations of negated statements. In this study, we analyze various LLMs for their handling of negation and propose two genres of prompts (*Warning-based* and *Persona-based*), which improve overall absolute accuracy by up to 3.17% and distractor negation accuracy by up to 25.14% over most competitive baselines. Next, we assess the robustness of LLMs by reordering prompts while preserving meaning, observing instability linked to positional encoding schemes. Further, we introduce a negative token attention score (NTAS) to quantify attention to negation words. From the comprehensive analysis, we point out that within a specific LLM family, the performance of a model (measured using accuracy) correlates more with NTAS than with model size. The code is publicly available: https://github.com/Joshua-Dias-Barreto/This-is-not-a-Disimprovement

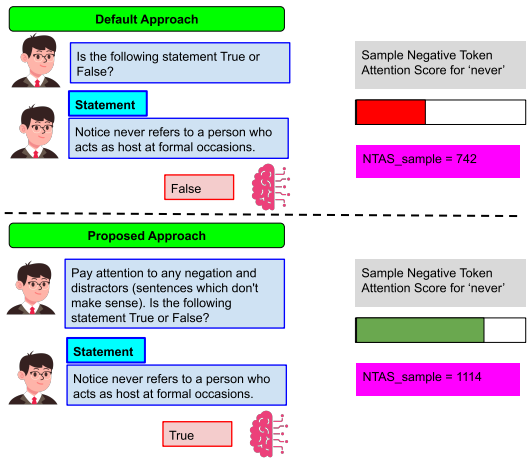

Figure 1: Improvement of Negation Reasoning using *Warning-based* prompt for Qwen2.5-14B, 'Default Approach' represents the *Baseline* Prompt and the 'Proposed Approach' represents the concatenation of *Warning-based* prompt and *Baseline* Prompt.

1 Introduction

Large language models (LLMs) have achieved state-of-the-art performance across diverse natural language processing (NLP) tasks, indicating their grasp of complex syntactic and semantic patterns. However, LLMs often struggle to correctly interpret sentences where negation alters the truth conditions of a proposition (Kassner and Schütze, 2020; Hossain et al., 2020; Truong et al., 2022; Ettinger, 2020; Shivagunde et al., 2023; Lialin et al., 2022). Various methods have been proposed to improve neural models' robustness to negated inputs. Hosseini et al. (2021) improve the BERT understanding of negation by combining unlikelihood training with syntactic data augmentation using negated LAMA (Kassner and Schütze, 2020). In another study, Singh et al. (2023) develop a pretraining strategy specifically designed to address negation, while Rezaei and Blanco (2024) explore paraphrasing negations into affirmative terms. Studies have also been attempted to investigate the robustness of LLMs with variations of prompts, size of models, etc. For example, LLMs have been found to be sensitive to variations in prompt format (Sclar et al., 2024) and content (Min et al., 2023). On the other hand, Shu et al. (2024) highlight that even minor prompt perturbations can significantly degrade performance, especially for negation. In one of the recent works, Yu et al. (2025) observe that LLMs exhibit a negative bias and propose a negative attention score (NAS) to measure it. Even though there have been studies dealing with negation reasoning, very few attempts have been made to improve the performance of LLMs for the same.

Hence, in this paper, we propose a prompt engineering-based framework to improve the performance of LLMs for negation reasoning. Towards that objective, we pose negation reasoning as a binary classification task where an LLM is supposed to classify a given statement as 'True' or

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 14149–14156
November 4-9, 2025

'False' (Figure 1). In our study, such statements are taken from a gold standard dataset, namely *This-is-not-a-Dataset* proposed by García-Ferrero et al. (2023), and as the baseline prompt, we consider a simple question - 'Is the following statement True or False?'(García-Ferrero et al., 2023). We propose two different genres of prompts- *Warning-based* and *Persona-based* to improve the negation reasoning capabilities and experiment with five different LLM families (FLAN-T5 (Chung et al., 2024), GPT-2 (Radford et al., 2019), Qwen1.5 (Bai et al., 2023), Qwen2.5 (Yang et al., 2024; Team, 2024) and IBM Granite 3.0 (Mishra et al., 2024; Granite Team, 2024)). Note that the baseline prompt is appended with the proposed prompts to make the entire prompt. We observe a distractor negation accuracy improvement of 25.14% and 10.9% for IBM-Granite-3.0-2B using *Warning-based* prompt and *Persona-based* prompts, respectively (Table 2).

Next, we introduce a negative token attention score (NTAS) to quantify attention to negative words (e.g., *"no"*, *"never"*, and *"not"*) and examine its link to improved negation reasoning performance. Further, we assess LLM robustness by swapping prompt order (Proposed+Baseline vs. Baseline+Proposed) and grouping models by positional encoding: *Relative*, *Absolute*, and *Rotary*. Our analysis shows that models with *Relative* encoding (e.g., FLAN-T5) show stable performance, while others vary when prompts are reordered. We also find that larger models don't always outperform smaller ones in negation tasks. Experiments show that within a specific LLM family, the performance of a model (measured using accuracy) correlates more with NTAS than with model size.

2 Task and Dataset

Task Formulation: We follow the task formulation proposed by García-Ferrero et al. (2023), which focuses on the binary classification ('True' or 'False') of sentences in the dataset. Given a statement *st*, the model must generate *True* or *False* tokens. Following the work of García-Ferrero et al. (2023) and Scheurer et al. (2023), the answer *A* is computed as follows:

$$A = \begin{cases} \text{True} & \text{if } \frac{p(\text{True}|st)}{p(\text{True}|st)+p(\text{False}|st)} > 0.5 \\ \text{False} & \text{otherwise} \end{cases}$$

Dataset: The dataset introduced by García-Ferrero et al. (2023) comprises approximately 400,000 semi-automatically generated English sentences encapsulating commonsense knowledge, with about two-thirds containing various forms of negation. The dataset features sentences structured into 11 distinct patterns, each designed to test different aspects of negation understanding. These patterns are based on WordNet (Fellbaum, 1998) relations, including synonymy, hypernymy, antonymy, and meronymy, as well as the semantic roles like agent, instrument, etc., which are defined by Morphosemantic Links (Fellbaum et al., 2009). We use its test set, which has approximately 90,000 sentences.

3 Proposed Framework

LLMs: We use a total of five LLM families, which are again categorized into three classes based on types of positional encodings. Note that positional encodings indicate how LLMs handle sequential information. We only use models available through open-access APIs.The details of the LLM families, along with the classes, are described below.
Relative positional encoding (Shaw et al., 2018) captures the distance between tokens to focus on relative positions, which is used by **FLAN-T5**;
Absolute positional encoding (Vaswani et al., 2017) assigns fixed positional values based on token order, which is used by the **GPT-2** family;
Rotary positional encoding (Su et al., 2024) applies a rotational transformation for flexible relative positioning, which is used by the **Qwen1.5**, **Qwen2.5**, and **IBM Granite 3.0** families[1] .

Next, we describe the two genres of prompts to improve negation reasoning in LLMs and negative token attention score (NTAS) to hypothesize why certain prompts or models have higher accuracies.

3.1 Prompt Engineering Set-up

We propose a set of compact prompts (one or two sentences) to improve negation reasoning and promote efficient inference. The two proposed genres of prompts (*Warning-based* and *Persona-based*) and the *Baseline* prompt are described below.
Baseline Prompt: Default prompt from (García-Ferrero et al., 2023) `-> Is the following statement True or False?`
***Warning-based* Prompts (WP):** The choice of Warning-based prompts was motivated by findings

[1]In this paper, Qwen1.5, Qwen2.5, Granite-3.0 refer to the Qwen1.5-Chat, Qwen2.5-Instruct, and Granite-3.0-Instruct models, respectively.

that LLMs are sensitive to instructional cues (Wei et al., 2022).

WP_1 (Negation-Distraction) warns the model of negation using distractors -> `Pay attention to any negation and distractors (sentences that don't make sense).`

WP_2 (Negation Spoonfeeding) informs the model about negation words -> `Pay attention to any negation (words like 'not', 'no', 'never', etc.).`

***Persona-based* Prompts (PP):** These prompts incorporate persona-based language while retaining the core task. The choice of Persona-based prompts was motivated by prior prompt engineering work, which suggests that establishing a role or persona can significantly impact model output (Tseng et al., 2024).

PP_1 (Linguistic Expert) assumes the role of a linguistic expert with this persona -> `You are a linguistic expert who understands the effect of negation words.`

PP_2 (Negation Expert) is designed to give confidence to the model beforehand, and that it can analyse negations in sentences -> `You are an expert in analysing negations in sentences.`

3.2 Prompt Reordering

3.2.1 Reordering Procedure

For this experiment, we reordered the baseline and the *Negation-Distraction* prompts to test the model's consistency in understanding the same content presented in different sequences. The two prompts were ordered in the following configurations:

1. Baseline, *Negation-Distraction* – The baseline prompt followed by the *Negation-Distraction* prompt.

2. *Negation-Distraction*, Baseline – The *Negation-Distraction* prompt followed by the baseline prompt.

The goal of this experiment was to observe whether changing the order of the prompts affected the model's ability to understand negation reasoning, as models using positional encodings may rely on the order of tokens for processing the input.

3.2.2 Example of Reordering

After Reordering, the two configurations would appear as:

```
Is the following statement True or False? Pay attention to any negation and distractors (sentences which don't make sense).
```

and

```
Pay attention to any negation and distractors (sentences which don't make sense). Is the following statement True or False?
```

In this experiment, we did not modify the content of the prompts, only their order. This allowed us to test if positional encodings influenced performance based on the sequence of prompts.

3.3 Negative Token Attention Score (NTAS)

The *Negative Token Attention Score* (NTAS) is inspired by *Negative Attention Score* (NAS) (Yu et al., 2025). NTAS is designed to quantify the attention that a model pays to negative answer tokens such as *"no"*, *"never"*, and *"not"* during inference. A sample x refers to a negation input sentence from a dataset X consisting of words such as *"no"*, *"never"*, and *"not"*. Let $A^{(l,h)} \in R^{L \times L}$ denote the attention matrix of head h in layer l, where L is the length of the tokenized input sequence.

Let t_{neg} be the position of the last occurring negation token in the input sequence, selected from the set {"no", "never", "not"}. The **sample NTAS** for a given input x is defined as the total attention directed toward the negation token position, summed over all heads and layers:

$$\text{NTAS}_{\text{sample}}(x) = \sum_{l \in L} \sum_{h \in H} \sum_{i=1}^{L} A_{i,t_{\text{neg}}}^{(l,h)} \quad (1)$$

Here, H and L are the sets of all attention heads and layers, respectively, and $A_{i,t_{\text{neg}}}^{(l,h)}$ represents the attention weight from token position i to the negation token t_{neg} at head h and layer l.

Model NTAS is defined as the average of the sample NTAS across all samples in a dataset X:

$$\text{NTAS}_{\text{model}}(X) = \frac{1}{|X|} \sum_{x \in X} \text{NTAS}_{\text{sample}}(x) \quad (2)$$

Model	Baseline	WP_1	WP_2	PP_1	PP_2
FLAN-T5 Base	50.75	50.25	50.83	**51.19**	50.52
GPT-2 Large	50.04	49.70	50.00	49.80	**50.20**
Qwen1.5-1.8B	**53.61**	53.01	50.01	51.94	50.66
Qwen2.5-14B	66.87	**68.55**	66.27	64.38	64.68
IBM-Granite-3.0-2B	57.77	**60.94**	60.29	60.15	59.21

Table 1: Accuracy(%) of different models across various prompts. WP_1 and WP_2 are the *Negation-Distraction* and the *Negation Spoonfeeding* prompts respectively. PP_1 and PP_2 are the *Linguistic Expert* and the *Negation Expert* persona-styled prompts. One of these outperforms the baseline for all models except Qwen1.5.

Model	Baseline	WP_1	WP_2	PP_1	PP_2
FLAN-T5 Base	90.47	**91.48**	77.36	49.91	74.84
GPT-2 Large	86.29	69.03	**89.23**	88.41	85.74
Qwen1.5-1.8B	4.37	5.96	**16.55**	10.93	10.82
Qwen2.5-14B	19.25	25.98	**29.98**	23.65	22.17
IBM-Granite-3.0-2B	7.22	24.29	**32.36**	16.74	18.12

Table 2: Distractor Negation Accuracy(%) of different models across various prompts. It is the accuracy for only those sentences that are negated using distractors (Appendix A). WP_2 improves this accuracy by upto 25.14% over the baseline.

This single score reflects the average amount of attention a model devotes to negative words across an entire dataset.

4 Experimental Results and Analysis

Experimental set-up: For all experiments, we used a single NVIDIA Quadro RTX 6000 Ada Generation GPU. To ensure uniform inference behaviour, Flash Attention (Dao et al., 2022) is disabled for all models. We use 'bfloat16' precision where supported, which helps optimize memory usage (Kalamkar et al., 2019; Burgess et al., 2019). To facilitate efficient inference on a single GPU, we employ 4-bit quantization (Liu et al., 2023). We mainly focus on *accuracy* as the evaluation metric.

Model	Baseline	WP_1	WP_2	PP_1	PP_2
FLAN T5 Base	**3.03**	1.99	2.48	2.25	2.54
GPT2-Large	652	534	**773**	545	544
Qwen1.5-1.8B	140	125	**169**	130	130
Qwen2.5-14B	621	593	**830**	645	636
Granite 3.0-2B	492	434	**523**	473	484

Table 3: Model NTAS (Negative Token Attention Score) of LLMs across various prompts. WP_2 or the *Negation Spoonfeeding* prompt causes models to pay more attention to the last negative token in the input sentence.

Model	Accuracy (Prompt/ Reordered)	Positional Encoding
GPT-2 Small	50.00 / 50.41	Absolute
GPT-2 Large	49.70 / 49.52	Absolute
FLAN-T5 Small	**49.67 / 49.59**	**Relative**
FLAN-T5 Base	**50.25 / 50.23**	**Relative**
Qwen1.5-1.8B	53.01 / 50.64	Rotary
Qwen1.5-7B	60.41 / 62.62	Rotary

Table 4: Accuracy (%) comparison of models with different positional encodings on standard vs. reordered prompts. Models with relative encodings show stable performance under *Negation-Distraction* prompt reordering (see subsection 3.2 for details).

Model Family	# Parameters	Accuracy	NTAS
FLAN-T5	0.077B	49.85	1.04
	0.783B	52.11	4.81
	2.85B	**52.3**	**7.07**
GPT-2	0.355B	49.92	318
	0.774B	50.04	652
	1.5B	**50.38**	**1221**
Qwen1.5	0.5B	51.38	170
	4B	55.93	206
	14B	**65.4**	**480**
Qwen2.5	0.5B	47.86	128
	3B	61.53	217
	14B	**66.87**	**621**
IBM-Granite 3.0	2B	57.77	492
	3B	50.26	222
	8B	**59.96**	**536**

Table 5: Accuracy(%) of model sizes for the baseline prompt. Larger models may not always perform better, but models with larger NTAS have higher accuracies, see IBM-Granite 3.0 2B and 3B models.

Results:

Prompt Variations vs Model Performance: Our prompts likely outperform the baseline on distractor negation accuracy because they explicitly highlight negation and distractors (Table 2). Distractors can make sentences misleading, as in "A use may be a document in certain contexts," where the word "use" misleads the model. The Negation-Distraction prompt helps Qwen1.5-1.8B correctly classify it as False, while the baseline does not. See Appendix A for more on distractors. Table 1 reveals that our prompts and personas outperform the baseline for all models except Qwen1.5-1.8B. The *Negation Spoonfeeding* prompt likely causes models to pay more attention to negative words as it explicitly warns the model of such words (Table 3).

Positional Encoding vs Model Performance: Models with relative positional encoding (e.g., FLAN-T5) show little performance change after syntactic reordering (see subsection 3.2 for details), while those with absolute or rotary encodings are

Model	FLAN-T5	GPT-2	Qwen1.5	Qwen2.5	IBM-Granite 3.0
Pearson	0.9521	0.9929	0.9761	0.8249	0.9962

Table 6: Correlation of model NTAS and accuracy for model families in Table 5. With such a limited sample size, we do not claim statistical significance, but instead present it as an exploratory observation.

less consistent (Table 4). This may stem from how each encoding handles token positions: relative encoding focuses on token relationships, making it more robust to reordering. In contrast, absolute encodings rely on fixed positions, leading to more sensitivity. For rotary positional encodings, despite their ability to model token relationships continuously, they still retain some dependence on absolute positions, which could explain their instability under syntactic reordering.

Model Size vs Model Performance: Most LLMs in our experiment show higher accuracy and model NTAS as model size increases, indicating better attention to key input elements (Table 5). However, IBM-Granite 2B outperforms the 3B model with a higher model NTAS and accuracy, suggesting that smaller models can outperform larger ones if they focus more on negative tokens. Correlations in Table 6 highlight that accuracy gains may depend not just on size but also on attention to negation.

5 Conclusion

This study aimed to enhance LLMs' negation reasoning through two genres of prompts: *Warning-based* and *Persona-based*. Experiments showed that small prompt variations can significantly boost performance up to 3.17% over baseline accuracy and 25.14% for distractor negation accuracy. We proposed a negative token attention score (NTAS) to quantify attention to negative words. Our analysis showed that inconsistencies with prompt reordering are likely tied to positional encodings. Though larger models usually perform better, this trend is inconsistent; accuracy correlated better with model NTAS than size within the same family of LLMs. Immediate future work could be to do a thorough pattern-wise analysis of the eleven linguistic patterns present in the target dataset to identify which patterns LLMs handle well and which they struggle with. Using other binary labels such as *Yes/ No* instead of *True/ False* is something that can be explored. Examining multilingual negation reasoning to evaluate generalization across languages and investigating whether effective prompts in one language transfer to others could be another future direction to explore.

6 Limitations

While our study provides valuable insights into the negation reasoning abilities of LLMs, it has several limitations. First, our experiments focus primarily on first-token probabilities rather than full-text responses. While this approach reduces inference time, it may not capture cases where models exhibit reasoning errors later in their responses. A more detailed evaluation of multi-token completions could offer additional insights into model behaviour.

Second, our study considers only a limited number of prompting strategies. Although we observe significant performance variations, there may be other prompt designs that further reveal model weaknesses or strengths. Future work should explore a wider range of prompt formulations, including those optimized through reinforcement learning or adversarial testing.

Finally, while we analyze the impact of model scaling on negation reasoning, our results are constrained to specific architectures and families of models. Different pretraining objectives, training data compositions, or fine-tuning techniques may influence performance in ways not captured by our study. Extending our analysis to a broader set of models could help generalize our findings.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Neil Burgess, Jelena Milanovic, Nigel Stephens, Konstantinos Monachopoulos, and David Mansell. 2019. Bfloat16 processing for neural networks. In *2019 IEEE 26th Symposium on Computer Arithmetic (ARITH)*, pages 88–91.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2024. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 25:70:1–70:53.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: fast and memory-efficient exact attention with io-awareness. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Language, Speech and Communication. Mit Press.

Christiane Fellbaum, Anne Osherson, and Peter E. Clark. 2009. Putting semantics into wordnet's "morphosemantic" links. pages 350–358. Springer Nature.

Iker García-Ferrero, Begoña Altuna, Javier Alvez, Itziar Gonzalez-Dios, and German Rigau. 2023. This is not a dataset: A large negation benchmark to challenge large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, page 8596–8615. Association for Computational Linguistics.

IBM Granite Team. 2024. Granite 3.0 language models.

Md Mosharaf Hossain, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, and Eduardo Blanco. 2020. An analysis of natural language inference benchmarks through the lens of negation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9106–9118, Online. Association for Computational Linguistics.

Arian Hosseini, Siva Reddy, Dzmitry Bahdanau, R Devon Hjelm, Alessandro Sordoni, and Aaron Courville. 2021. Understanding by understanding not: Modeling negation in language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1301–1312, Online. Association for Computational Linguistics.

Dhiraj D. Kalamkar, Dheevatsa Mudigere, Naveen Mellempudi, Dipankar Das, Kunal Banerjee, Sasikanth Avancha, Dharma Teja Vooturi, Nataraj Jammalamadaka, Jianyu Huang, Hector Yuen, Jiyan Yang, Jongsoo Park, Alexander Heinecke, Evangelos Georganas, Sudarshan Srinivasan, Abhisek Kundu, Misha Smelyanskiy, Bharat Kaul, and Pradeep Dubey. 2019. A study of BFLOAT16 for deep learning training. *CoRR*, abs/1905.12322.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Vladislav Lialin, Kevin Zhao, Namrata Shivagunde, and Anna Rumshisky. 2022. Life after BERT: What do other muppets understand about language? In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3180–3193, Dublin, Ireland. Association for Computational Linguistics.

Shih-yang Liu, Zechun Liu, Xijie Huang, Pingcheng Dong, and Kwang-Ting Cheng. 2023. LLM-FP4: 4-bit floating-point quantized transformers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 592–605, Singapore. Association for Computational Linguistics.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Comput. Surv.*, 56(2).

Mayank Mishra, Matt Stallone, Gaoyuan Zhang, Yikang Shen, Aditya Prasad, Adriana Meza Soria, Michele Merler, Parameswaran Selvam, Saptha Surendran, Shivdeep Singh, and et al. 2024. Granite Code Models: A Family of Open Foundation Models for Code Intelligence. *arXiv e-prints*, arXiv:2405.04324.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*. Accessed: 2024-11-15.

MohammadHossein Rezaei and Eduardo Blanco. 2024. Paraphrasing in affirmative terms improves negation understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 602–615, Bangkok, Thailand. Association for Computational Linguistics.

J'er'emy Scheurer, Jon Ander Campos, Tomasz Korbak, Jun Shern Chan, Angelica Chen, Kyunghyun Cho, and Ethan Perez. 2023. Training language models with language feedback at scale. *ArXiv*, abs/2303.16755.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting.

In *The Twelfth International Conference on Learning Representations*.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, New Orleans, Louisiana. Association for Computational Linguistics.

Namrata Shivagunde, Vladislav Lialin, and Anna Rumshisky. 2023. Larger probes tell a different story: Extending psycholinguistic datasets via in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2094–2107, Singapore. Association for Computational Linguistics.

Bangzhao Shu, Lechen Zhang, Minje Choi, Lavinia Dunagan, Lajanugen Logeswaran, Moontae Lee, Dallas Card, and David Jurgens. 2024. You don't need a personality test to know these models are unreliable: Assessing the reliability of large language models on psychometric instruments. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, page 5263–5281. Association for Computational Linguistics.

Rituraj Singh, Rahul Kumar, and Vivek Sridhar. 2023. NLMs: Augmenting negation in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13104–13116, Singapore. Association for Computational Linguistics.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Thinh Hung Truong, Yulia Otmakhova, Timothy Baldwin, Trevor Cohn, Jey Han Lau, and Karin Verspoor. 2022. Not another negation benchmark: The NaN-NLI test suite for sub-clausal negation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 883–894, Online only. Association for Computational Linguistics.

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. 2024. Two tales of persona in LLMs: A survey of role-playing and personalization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16612–16631, Miami, Florida, USA. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Sangwon Yu, Jongyoon Song, Bongkyu Hwang, Hoyoung Kang, Sooah Cho, Junhwa Choi, Seongho Joe, Taehee Lee, Youngjune Gwon, and Sungroh Yoon. 2025. Correcting negative bias in large language models through negative attention score alignment. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 9979–10001, Albuquerque, New Mexico. Association for Computational Linguistics.

A Example of Distractor Sentences

Distractor sentences are designed to introduce misleading or subtly incorrect information by replacing a key term in a statement with another term that alters its truth value. These distractors can confuse language models, leading to incorrect responses if the model fails to detect the alteration.

Consider the following original sentence:

> **Original Sentence:** "A study may be a document in certain contexts." **Correct Answer:** True

In this case, "study" correctly refers to something that can be a document in some contexts. However, introducing a distractor leads to a false statement:

Distractor Sentence: "A use may be a document in certain contexts." **Correct Answer:** False

Here, "use" replaces "study," creating false knowledge. While the original sentence is true, the distractor sentence is not.