Should Large Language Models Go to School for Phonology?

Ashima Suvarna, Harshita Khandelwal, Nanyun Peng
Work in Progress

Abstract

Phonology refers to the organization of sound in a language and plays a key role in language acquisition. Large Language Models (LLMs) that comprehend phonology can establish connections between orthographic representations and their corresponding phonetic forms, thus, facilitating many tasks that involve written and spoken languages such as text-based analysis of speech, and lyrics/poetry analysis and generation. Large Language Models (LLMs) such as GPT-4 have been trained on extensive orthographic data and have demonstrated astonishing results on many natural language processing (NLP) tasks. Therefore, we hypothesize that LLMs can learn imperfect regularities between orthographic and phonological representations of language thus performing seemingly well on phonological tasks without deep understanding of phonology. To quantify the performance of LLMs on phonologically grounded tasks, we design tasks that test the LLMs performance on varying levels of complexity. Specifically, we prompt models to (a) generate phonetic transcriptions, (b) classify rhyming and non-rhyming pairs, and (c) generate rhyming words. We find that GPT-4 generates the correct phonetic transcription for 50% of the common words and 29% of the rare words suggesting that GPT-4 struggles with basic phonetics. Our findings highlight that LLMs perform suboptimally on phonological tasks for English and encourage innovative solutions from the research community to train language models with improved phonological understanding.

1 Introduction

Phonology is the study of sound structure in a language and plays a vital role in language acquisition. Many real-world applications such as human-like speech generation and lyrics/poetry writing require deep phonological understanding. Recent advances in large language models (LLMs) (OpenAI, 2023; Taori et al., 2023; Touvron et al., 2023; Chowdhery et al., 2022; Hoffmann et al., 2022) have shown remarkable capabilities in generating fluent text by comprehending user intents (Bang et al., 2023; Liu et al., 2023b,a). Their impressive abilities to write coherent text have spurred interest in utilizing LLMs as writing assistants in domains that rely on phonology such as poetry and songwriting (Chakrabarty et al., 2022; Tian and Peng, 2022; Jammaround, 2023). As shown in Figure 1, sonnets rely on phonetic rules to adhere to strict metre-and-rhyme constraints.

So all their praises are but **prophecies**	**A**
Of this our time, all you **prefiguring**;	**B**
And for they looked but with divining **eyes,**	**A**
They had not skill enough your worth to **sing**:	**B**
For we, which now behold these present **days**,	**C**
Have eyes to wonder, but lack tongues to **praise**.	**C**

Figure 1: A structured poem like a sonnet often relies on phonological rules to satisfy strict metre-and-rhyme constraints. E.g., we know the words days and praise rhyme because we understand how they are pronounced.

While several large-scale analyses (Beeching et al., 2023; Liang et al., 2022; Gao et al., 2021) have evaluated LLMs across multiple NLP benchmarks, and other studies have specifically examined and assessed their reasoning (Liu et al., 2023a), creative (Borji, 2023; Liang et al., 2022), linguistic (Basmov et al., 2023; Beguš et al., 2023), and factual abilities (Tam et al., 2022) through qualitative and quantitative analyses, there has been a lack of systematic studies that assess the phonological understanding of LLMs. Such analysis is crucial to further understand the linguistic behavior of LLMs before deploying them in real-world applications.

In this work, we hypothesize that LLMs can learn imperfect regularities between orthographic and phonological representations of language without deep understanding of phonology and aim to quantify this performance through our analysis.

To conduct out analysis, we design a set of three phonological tasks to probe state-of-the-art LLMs

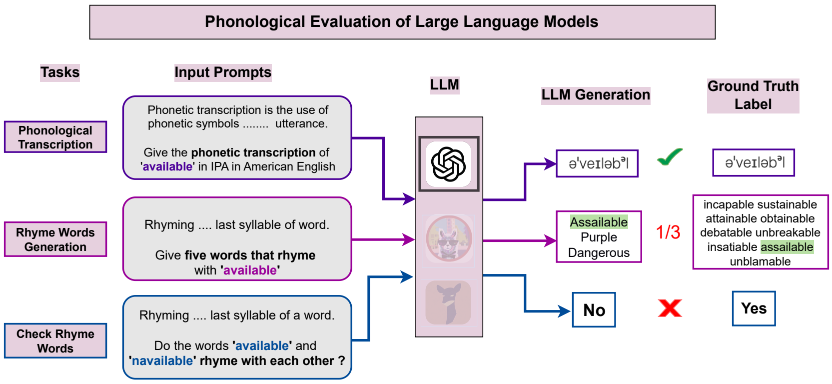

Figure 2: We analyse the phonological understanding of various LLMs on three tasks that require deep understanding of phonology by comparing model outputs with ground truth labels.

Words	Phonetic Transcription	Rhyming Words
Frequent Words		
Everything	/ˈɛvrəi,θɪŋ/	Anything, Betting, Getting, Coloring
Sponsored	/ˈspan.səɹd/	Forward, Offered, Awkward, Ordered
Rare Words		
Cenote	/sɪˈnoʊti/	Pretty, Coffee, Dirty, Guarantee
Zazen	/zɑːˈzɛn/	Happen, Passion, Imagine, Fashion

Table 1: We show the ground truth labels for **frequent words** and **rare words** that are used as inputs for our tasks.

for their ability to understand phonology at various levels of abstraction. Specifically, in §3.2.1, we prompt models to generate phonetic transcriptions, which requires accurately representing the pronunciation of words and find that they struggle on to accomplish this task. §3.2.2 prompts the LLMs to classify rhyming and non-rhyming pairs, testing their discriminative skills in identifying phonetic similarity between words and find that models have some understanding of rhyme words. Finally, in §3.2.3, we probe LLMs to generate rhyming words, showcasing their understanding of sound patterns and their ability to produce words that share similar phonetic endings.

Figure 2 gives an overview of our study. Our main contributions are: (a) We present the first systematic study to evaluate the phonological understanding of LLMs at various levels of complexity. (b) Our results quantify that performance of LLMs on tasks that require deep phonological understanding. (c) Our findings suggest that LLMs learn only superficial regularities from their pre-training data that may not be grounded in phonology to accomplish these tasks thus their sub-optimal performance and highlights the importance of incorporating phonological information in LLMs.

2 Related Work

Recently, various works have focused on evaluating the capabilities of LLMs on various NLP tasks (Liang et al., 2022; Zheng et al., 2023; Valmeekam et al., 2023; Bang et al., 2023; Beeching et al., 2023; Qin et al., 2023; Kocoń et al., 2023). Additionally, (Liang et al., 2022; Beeching et al., 2023) provide multi-metric evaluations of LLMs on a broach range of scenarios such as question answering, summarization and sentiment analysis. However, these works do not specifically study the linguistic capabilities of LLMs. Hu and Levy (2023) assesses the efficacy of prompting as a way of probing the model's metalinguistic ability, i.e., the ability to perform linguistic analyses given a natural language input while Beguš et al. (2023) presents qualitative case studies on phonology, syntax and semantics of GPT-4. Even though Beguš et al. (2023) has also demonstrated the behavior of LLMs on multiple phonology-related tasks, they differ from our work. They focus on generating theoretical analyses of phonology given linguistic structures as input while we focus on the probing phonological understanding of LLMs given natural language input. Basmov et al. (2023) evaluates the performance of LLMs on linguistic inferences such as grammatically-specified entailments and monotonicity entailments using natural language inputs. While Basmov et al. (2023) focus on linguistic inferences of ChatGPT, in this work, we analyse the phonological capabilities of several LLMs.

3 Method

To evaluate LLMs on their phonological understanding We do not distinguish between American and British English and reward the model for phonetic transcriptions from either dialect., we assess the performance of these models on the following three tasks. We prompt all the models in a zero-shot setting with the same prompt. We provide the prompts for each task in the Appendix §C.

3.1 Models

We analyse four state-of-the-art LLMs : ChatGPT (OpenAI, 2022) , GPT-4 (OpenAI, 2023), Alpaca-7B (Taori et al., 2023) and Vicuna-13B (Chiang et al., 2023). We focus on ChatGPT and GPT-4 since they have been pretrained on web-data and can comprehend user queries well (Liu et al., 2023b). Additionally, we also evaluate Alpaca-7B and Vicuna-13B as they are popular open-source LLMs that are instruction-tuned on human or machine-generated instructions. We provide details of model decoding in Appendix §B.

3.2 Tasks

Input Words We curate three types of words for our tasks: Frequent Words, Rare Words and Nonsense Words as shown in Table 1. We provide further details in Appendix §A.

3.2.1 Phonetic Transcription Generation

Task In this task, our objective is to evaluate the model's proficiency in generating phonetic transcriptions of provided words using the International Phonetic Alphabet (IPA) [1]. Phonetic transcription serves as a means to distinguish between different sounds that may share the same spelling (homographs) , such as the verb 'read' (present tense) and the verb 'read' (past tense). Analyzing LLM performance here reveals their utilization of phonological knowledge instead of relying on orthography.

Setup Since, nonsensical words will not have a corresponding entry in the dictionary, we perform this task only for **Frequent Words** and **Rare Words**. We retrieve the phonetic transcriptions of the words in our collection from the Oxford Dictionaries API (Oxford, March 2023) and treat them as the ground truth. Finally, we prompt the models to generate the phonetic transcription in IPA given the word in natural language and compute the accuracy of the model generations.

3.2.2 Rhyming Word Classification

Task In this task, our goal is to test the model's ability to discern rhyming vs. non-rhyming words. A rhyme is a repetition of similar phonemes in the final stressed syllables and any following syllables of two or more words. Thus, detecting whether two given words rhyme or not requires knowledge of the phonemes composing each word. Unlike the previous task, this task evaluates whether LLMs can apply the phonological knowledge as intermediary representations (phonemes) to accomplish a high level task (detect rhymes).

Setup We perform this task for **Frequent Words** and **Nonsense Words**. The models are required to classify whether the given pair of words rhyme with each other or not. Example prompts are shown in Appendix Table 9. We prompt the models to generate binary responses 'yes' or 'no' and compute the accuracy of the classification.

3.2.3 Rhyming Words Generation

Task In this task, we aim to analyse the LLMs' capability in *generating* correct rhyming words for a given word. Specifically, we prompt the models to generate rhyming words for a word **Frequent Words** and **Rare Words**. Since generating rhymes requires one to decompose a word into phonemes, LLMs that cannot comprehend phonology well will struggle and perform poorly, often generating rhymes based solely on the spelling of a word.

Setup We retrieve all the rhyming words (slant and strict rhymes) for a given word from an online rhyming dictionary, WordHippo[2] and treat these as the ground truth. We could not find accurate rhyming words for Nonsense Words and therefore, do not analyse this task on them. In this task, we prompt the models to generate five rhyming words for **Frequent Words** and **Rare Words**. We then compute the word-specific success rate as the number of generated rhyming candidates that belong to the ground-truth set of rhyming words. The final success rate is the average success rate for all the words for a particular type.

[1] www.internationalphoneticassociation.org/

[2] https://www.wordhippo.com/

Task	Prompt
Phonetic Transcription	Phonetic transcription is the use of phonetic symbols to represent speech sounds. Ideally, each sound in a spoken utterance is represented by a written phonetic symbol, so as to furnish a record sufficient to render possible the accurate reconstruction of the utterance. The International Phonetic Alphabet (IPA) is a set of about a hundred alphabetic symbols (e.g. 1), a) together with a handful of non-alphabet symbols (e.g. the length mark :) and about thirty diacritics (e.g. those exemplified in S, d). Give the phonetic transcription of '<input>' in IPA.
Rhyme Word Generation	Rhyming words are words that have the same ending sound. In simpler terms, it can be defined as the repetition of similar sounds. Give 5 words that rhyme with '<input>'.
Rhyme Check	Rhyming words are words that have the same ending sound. In simpler terms, it can be defined as the repetition of similar sounds. Does '<input>' rhyme with '<input>'? Please answer 'Yes' or 'No'.

Table 2: Example Prompts for Tasks. Each prompt is prefixed with general information about the task and concepts.

Model	Frequent Words (↑)	Rare Words (↑)
ChatGPT	32.0%	18.2%
GPT-4	**50.4%**	**29.1%**
Alpaca-7B	5.6%	2.3%
Vicuna-13B	8.7%	4.2%

Table 3: Performance accuracy of LLMs on Phonetic Transcription of Frequent and Rare Words. We reward the model for phonetic transcriptions from both American and British English.

Model	Frequent Words (↑)	Nonsense Words (↑)
ChatGPT	**82.2%**	**44.4%**
GPT-4	79.1%	35.9%
Alpaca-7B	42.7%	28.6%
Vicuna-13B	52.3%	41.3%

Table 4: Performance accuracy of LLMs on Rhyming Word Classification task.

4 Results

In this section, we discuss the performance of LLMs on the proposed phonological tasks.

We present the accuracy of LLMs on the phonetic transcription task (§3.2.1) in Table 3. Interestingly, all models exhibit poor performance, with GPT-4 achieving the highest accuracy of 50.4% for frequent words. This suggests that the models struggle with phonetic transcriptions due to the limited availability of comprehensive phonetic data during training. Consequently, LLMs may not effectively utilize linguistic principles for phonological tasks and rely more on statistical correlations from pretraining data. This is evident in GPT-4's 21% drop in performance for rare words compared to frequent words. These models perform worse on rare words than frequent words so there is evidence of a performance gap. A specialized model that has truly learned phonological information will perform similarly on both types of words.

We present the performance accuracy of different LLMs on Rhyme Classification (§3.2.2) in Table 4. We observe models perform well on Frequent Words, with ChatGPT accurately classifying two words as rhyming or non-rhyming in 82% of the cases. However, model performance drops signficantly for rare words. This suggests that high performance on frequent words could also be attributed to the presence of these words in the pretraining data unlike the rare words. We also observed that the models were biased to classifying pairs of words as non-rhyming in most cases, thus, indicating that the LLMs are not relying on phonological theories of rhyme to accomplish this task.

We investigate the models' ability to generate rhyming words for a given word (§3.2.3) in Table 5. GPT-series LLMs show greater than 70% success rate. However, we observe a significant drop of 30% for Rare Words, indicating that they fail to generalize their phonological knowledge on words that are unlikely to be in the pretraining data. We also observe that for most rare words the model generated rhyming words that were spelled similarly to the given word but not pronunced the same,

Model	Frequent Words (↑)	Rare Words (↑)
ChatGPT	71.1%	38.5%
GPT-4	**77.8%**	**40.4%**
Alpaca-7B	6.5%	2.6%
Vicuna-13B	27.8%	3.8%

Table 5: Overall Success Score of LLMs on Rhyming Word Generation Task.

for example, GPT-4 generated 'riot', 'pilot' and 'violet' as rhyming candidates for 'coyote'. This suggests that these models are establishing their generations for this task on the orthography of the word rather than the phonology. We present more examples in Appendix Tables 7 and 8.

Overall, both Alpaca-7B and Vicuna-13B performed very poorly across all the tasks. Interestingly, Vicuna-13B outperformed GPT-4 on Rhyme Classification of Nonsense Words. We analyze the amount of phonology-inspired tasks in the 52K Alpaca dataset and find that 0.55% of the total instructions focused on generating rhyming words, count syllables in words, and phonetic transcriptions generation. Appendix §D presents the setup of our analysis. Hence, this poor performance could be attributed to the lack of phonetic data in their pretraining and finetuning data.

5 Conclusion

In this work, we investigate the hypothesis that LLMs rely on superficial regularities learned from pre-training data without a deep understanding of phonology to accomplish phonological tasks such as generating rhymes or counting lexical stresses. We further quantify the performance of LLMs on various phonologically-motivated tasks in English. Our findings reveal that LLMs generally perform sub-optimally on these phonological tasks with GPT-4 performing the best across the various tasks. A straightforward approach to improve model performance across these phonological tasks is to add more phonological data during pre-training. Prior work by Liu et al. (2019) have shown the efficacy of joint textual and phonetic embedding in neural machine translation, thus, future work can focus on augmenting LLMs with phonetic representations. Overall, our results highlight that LLMs do not have a deep understanding of phonology for English and encourages research in training models for improved phonological understanding.

Limitations

Our experiments investigate the phonological tasks only in English. We tested the LLMs in a zero-shot setting and did not try sophisticated prompting techniques like chain-of-thought prompting. In practice, zero-shot prompts were sufficient to highlight the gaps in LLM capabilities from a phonological perspective. We acknowledge that due to the models' sensitivity to prompt design, we can achieve better task performance with more sophisticated techniques, however, we believe that the main findings will remain consistent. Future research can focus on the impact of different prompting techniques on the phonological tasks.

Another limitation of our study is that we analysed a small class of models. Future works could replicate our work on models of different sizes and objectives (such as OPT (Zhang et al., 2022), Flan-T5 (Chung et al., 2022)). We also note that the results from the OpenAI models are not necessarily reproducible due to the models being closed behind an API. Furthermore, since ChatGPT and GPT-4 undergoes continuous updates, the experimental results presented here are likely to change over time.

Ethics Statement

This work does not involve the creation of new models. Instead, its objective is to offer valuable insights into the methodology used to evaluate the phonological knowledge of state-of-the-art LLMs , ultimately contributing to the interpretability of these models. However, it is crucial to acknowledge that the broader ethical concerns associated with LLMs remain relevant to our work. LLMs have demonstrated the ability to generate outputs that are factually incorrect, offensive, or discriminatory. As a result, their use should be approached with utmost caution, particularly in commercial applications or user-facing contexts. Any demonstrations of the phonological capabilities and general functionality of LLMs should be interpreted within the context of these ethical considerations.

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.

Victoria Basmov, Yoav Goldberg, and Reut Tsarfaty. 2023. Chatgpt and simple linguistic inferences: Blind spots and blinds.

Edward Beeching, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open llm leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.

Gašper Beguš, Maksymilian Dąbkowski, and Ryan Rhodes. 2023. Large linguistic models: Analyzing theoretical linguistic abilities of llms.

Ali Borji. 2023. A categorical archive of chatgpt failures.

Tuhin Chakrabarty, Vishakh Padmakumar, and He He. 2022. Help me write a poem: Instruction tuning as a vehicle for collaborative poetry writing.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

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 Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2021. A framework for few-shot language model evaluation.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. 2022. Training compute-optimal large language models.

Jennifer Hu and Roger Levy. 2023. Prompt-based methods may underestimate large language models' linguistic generalizations.

Jammaround. 2023. Unleashing the potential of llms and audio models on music.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, Anna Kocoń, Bartłomiej Koptyra, Wiktoria Mieleszczenko-Kowszewicz, Piotr Miłkowski, Marcin Oleksy, Maciej Piasecki, Łukasz Radliński, Konrad Wojtasik, Stanisław Woźniak, and Przemysław Kazienko. 2023. ChatGPT: Jack of all trades, master of none. *Information Fusion*, page 101861.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2022. Holistic evaluation of language models.

Hairong Liu, Mingbo Ma, Liang Huang, Hao Xiong, and Zhongjun He. 2019. Robust neural machine translation with joint textual and phonetic embedding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3044–3049, Florence, Italy. Association for Computational Linguistics.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023a. Evaluating the logical reasoning ability of chatgpt and gpt-4.

Yiheng Liu, Tianle Han, Siyuan Ma, Jiayue Zhang, Yuanyuan Yang, Jiaming Tian, Hao He, Antong Li, Mengshen He, Zhengliang Liu, Zihao Wu, Dajiang Zhu, Xiang Li, Ning Qiang, Dingang Shen, Tianming Liu, and Bao Ge. 2023b. Summary of chatgpt/gpt-4 research and perspective towards the future of large language models.

James Maguire. 2006. *American bee: The national spelling bee and the culture of word nerds*. Rodale Books.

OpenAI. 2022. Introducing chatgpt.

OpenAI. 2023. Gpt-4 technical report.

University Press Oxford. March 2023. Oed online.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver?

Jonathan A Simon. 2018. Entendrepreneur: Generating humorous portmanteaus using wordembeddings. In *Second Workshop on Machine Learning for Creativity and Design (NeurIPS 2018)*.

Derek Tam, Anisha Mascarenhas, Shiyue Zhang, Sarah Kwan, Mohit Bansal, and Colin Raffel. 2022. Evaluating the factual consistency of large language models through summarization.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Yufei Tian and Nanyun Peng. 2022. Zero-shot sonnet generation with discourse-level planning and aesthetics features.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Karthik Valmeekam, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. 2023. Large language models still can't plan (a benchmark for llms on planning and reasoning about change).

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pretrained transformer language models.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Appendix

A Input Words

Throughout our experiments, we use three types of words - **Frequent words**, **Rare Words**, **Nonsense Words**. Specifically, we collect **Frequent Words** from Google Trillion Word Corpus [3]. Since, it is a popular web-based dataset, these words are likely to be in the pretraining data for models and we expect models to perform well for these words. Therefore, we do not the select frequent words from the Google Corpus as rare words. We compile 150 words from the Spelling Bee Study Guide [4] as **Rare Words**. Maguire (2006) have demonstrated that the Spelling Bee Study Guide is frequently updated to filter out common words, thus, LLMs are less likely to have seen **Rare Words** in training. Finally, we create **Nonsense Words** to evaluate LLMs performance on gibberish that may not appear in the online resources and hence, their pretraining data. We generate Nonsense words by naively combining a set of commonly used **Affixes** in English & **Root words** derived from Greek and Latin sources. The use of known root words and affixes helps us control the rhyming possibility of the word while creating truly nonsensical words that the models may not have seen during pretraining. Future work can also analyse the phonological understanding of models on portmanteaus (Simon, 2018).

B Decoding settings for the Models

ChatGPT and GPT-4 were accessed through the OpenAI API while Alpaca-7b and Vicuna-13b were setup locally.

- ChatGPT & GPT-4 responses were decoded using the openai.ChatCompletion.create() function with default parameters.
- Vicuna was decoded using a greedy approach with parameters - temperature: 0.7, repetiton penalty: 1.0
- Alpaca was decoded using a greedy approach with parameters - temperature: 1.1.

C Prompt Template for each task

We present the prompts used for each task in Table 6. Each prompt is prefixed with general information about the task and concepts followed by the

[3] `https://github.com/first20hours/google-10000-english`

[4]

task instruction. We evaluate the models with the same prompt for each task. Results for the same can be seen in Table 8. Additionally, we show example prompts for Rhyme Classification to better exemplify how the pairs of words were selected in Table 9.

D Phonology Motivated Tasks in Alpaca Dataset

We filter instructions that mention ‘syllable’, ‘phonetics’, ‘phonology’ and ‘rhyming/rhyme’ in the 52k instruction dataset used to train Alpaca. We find 286 such instructions and provide some examples below :

- instruction": "Generate a list of words that rhyme with ‘cat’.", "input": "", "output": "Words that rhyme with cat include bat, hat, mat, fat, pat, rat, sat, chat, and scat."
- "instruction": "Spell out the sentence using the NATO phonetic alphabet.", "input": "A cat has nine lives.", "output": "Alpha Charlie Alpha Tango Hotel Alpha Sierra Hotel Echo Lima India Sierra Lima Echo Sierra."
- instruction": "Find the count of the number of syllables in the following word:Aardvark", "input": "", "output": "The word ’Aardvark’ has 3 syllables."
- "instruction": "Generate a rhyme for the word "falling.", "input": "", "output": "Calling"

Task	Prompt
Phonetic Transcription	Phonetic transcription is the use of phonetic symbols to represent speech sounds. Ideally, each sound in a spoken utterance is represented by a written phonetic symbol, so as to furnish a record sufficient to render possible the accurate reconstruction of the utterance. The International Phonetic Alphabet (IPA) is a set of about a hundred alphabetic symbols (e.g. 1), a) together with a handful of non-alphabet symbols (e.g. the length mark :) and about thirty diacritics (e.g. those exemplified in S, d). Give the phonetic transcription of ’<input>’ in IPA.
Rhyme Word Generation	Rhyming words are words that have the same ending sound. In simpler terms, it can be defined as the repetition of similar sounds. Give 5 words that rhyme with ’<input>’.
Rhyme Check	Rhyming words are words that have the same ending sound. In simpler terms, it can be defined as the repetition of similar sounds. Does ’<input>’ rhyme with ’<input>’? Please answer ‘Yes’ or ‘No’.

Table 6: Example Prompts for Tasks. Each prompt is prefixed with general information about the task and concepts.

Task	ChatGPT	GPT-4
Phonetic Transcription		
University Development	junəˈvɜsəti dɪˈvɛləpmənt	ˌjunɪˈvɜrsɪti ˌdE vəlˈoʊpmənt
Rhyme Word Generation		
demulcent	effulgent succulent incandescent convalescent luminescent	convalescent, flourescent, incandescent, adolescent, effervescent
cenote	remote, vote, tote, promote, note	remote, promote, anecdote, demote, afloat
Rhyme Word Classification		
their, chair pseudoism, skepticism	Yes. No.	Yes No

Table 7: ChatGPT and GPT-4 responses for each task in our analysis.

Task	Alpaca-7B	Vicuna-13B
Phonetic Transcription		
University Development	Give the phonetic transcription of ’university’ in IPA in American English ’ɪnklɪŋ	ˌənɪˈvɛrsti ˈdeveləpment
Rhyme Word Generation		
demulcent	What is the name of a demulcent expectorant? A demulcent expectorant & Ameliorate	‘Confluent’, ‘Deglutition’, ‘Elation’, ‘Exultation’
cenote	Give 5 words that rhyme with ’cenote’ and it won’t come up with much.	’note’, ’moist’, ’mist’, ’listed’, ’kist’
Rhyme Word Classification		
their, chair	Yes.	Yes
pseudoism, skepticism	No, "pseudoism" and "skepticism" do not rhyme with each other.	Yes, the word "pseudoism" does rhyme with "skepticism".

Table 8: Alpaca-7B and Vicuna-13B responses for each task in our analysis.

Rhyming Pair Type	**Example**	**Ground Truth Label**
Most Frequent + Most Frequent (Non-Rhymes)	Does ‘cloudy’ rhyme with ‘class’ ? Yes/No	No
Most Frequent + Most Frequent (Rhymes)	Does ‘ground’ rhyme with ‘crowned’ ? Yes/No	Yes
Most Frequent + Nonsense (Non-Rhymes)	Does ‘exist’ rhyme with ‘anthropoel’ ? Yes/No	No
Most Frequent + Nonsense (Rhymes)	Does ‘assurance’ rhyme with ‘vocance’ ? Yes/No	Yes

Table 9: Prompts generated by combining different types of rhyming pairs for Check Rhyme Task