PhonologyBench: Evaluating Phonological Skills of Large Language Models

Ashima Suvarna, Harshita Khandelwal, Nanyun Peng
Department of Computer Science, University of California, Los Angeles
{asuvarna31, nanyunpeng}@cs.ucla.edu
harshitaskh@g.ucla.edu

Abstract

Phonology, the study of speech's structure and pronunciation rules, is a critical yet often overlooked component in Large Language Model (LLM) research. LLMs are widely used in various downstream applications that leverage phonology such as educational tools and poetry generation. Moreover, LLMs can potentially learn imperfect associations between orthographic and phonological forms from the training data. Thus, it is imperative to benchmark the phonological skills of LLMs. To this end, we present PhonologyBench, a novel benchmark consisting of three diagnostic tasks designed to explicitly test the phonological skills of LLMs in English: grapheme-to-phoneme conversion, syllable counting, and rhyme word generation. Despite having no access to speech data, LLMs showcased notable performance on the PhonologyBench tasks. However, we observe a significant gap of 17% and 45% on Rhyme Word Generation and Syllable counting, respectively, when compared to humans. Our findings underscore the importance of studying LLM performance on phonological tasks that inadvertently impact real-world applications. Furthermore, we encourage researchers to choose LLMs that perform well on the phonological task that is closely related to the downstream application since we find that no single model consistently outperforms the others on all the tasks. [1]

1 Introduction

Large language models (LLMs) are a class of deep generative models (Ruthotto and Haber, 2021; Bond-Taylor et al., 2021) that are trained on an internet-scale text corpora (e.g., two trillion tokens). To assess their usefulness in the real-world scenarios, prior work has focused on holistic evaluation of LLMs (Liang et al., 2022). For instance, they are evaluated on tasks that require syntactic and semantic understanding of the language such as summarization, rationale generation, story generation and question answering (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; Sun et al., 2023). However, there are various text-based tasks of practical importance that require joint understanding of the written and spoken language such as poetry generation (Ormazabal et al., 2022; Henderson, 1965; Suzuki, 1985). In particular, these tasks require the model to have phonological skills i.e., understanding the patterns of speech units and rules governing pronunciation in language. While LLMs have been adopted to perform text-based tasks that require phonological skills (Ding et al., 2024; Kwon, 2023), it is unclear to what extent they acquire phonological skills by training on large-scale text data, without access to speech data.

Despite being trained solely on textual data, LLMs have been applied to many tasks that benefit from a deep understanding of phonology. These applications include poetry generation, song writing, machine translation, and language learning (Ding et al., 2024; Kwon, 2023; Yu et al., 2024). For example, the generation of poetry and lyrics leverages the models' knowledge of rhyme and meter to produce rhythmically engaging content. Similarly, phonetic transcriptions from related Dravidian languages can enhance the accuracy of multilingual machine translations (Chakravarthi et al., 2019). We argue that due to their extensive training and further alignment, LLMs may leverage the learnt associations between written and spoken forms to accomplish these tasks. However, due to the lack of deep phonological understanding, they fail to perform phonological reasoning tasks in open-world scenarios (Peng et al., 2023). For example, although LLMs are popular for composing poetry, machine-generated poetry lacks diverse rhyming

---

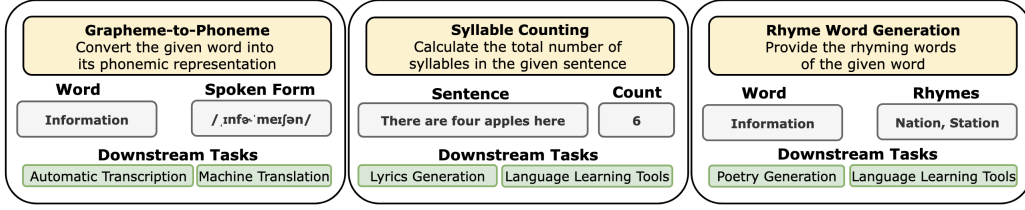[1] We will release the dataset and code in the camera-ready version.

Figure 1: Overview of tasks comprising PhonologyBench. a) Grapheme-to-Phoneme conversion translates written language into phonetic script b) Syllable Counting enumerates the syllables in a sentence c) Rhyme Word Generation identifies rhyming counterparts for a given word. Each task is presented with an example and various downstream tasks.

schemes and is highly criticised by experts (Vice, 2023).

While earlier studies (Jiao et al., 2023; Sawicki et al., 2023; Peng et al., 2023), have evaluated LLM performance on poetry generation, machine translation and language learning. These studies were designed to analyse LLM performance on the downstream task rather than probe the model's inherent understanding of phonology. To this end, we devise three diagnostic tasks that benchmark various LLMs on fine-grained tasks that explicitly apply phonological concepts in English. Our benchmark, **PhonologyBench**, comprises of three tasks - grapheme-to-phoneme (G2P) conversion, syllable counting and rhyme word generation, is shown in Figure 1. G2P is a widely adopted task for speech recognition applications that leverages pronunciation information to produce phonemes. Additionally, syllable counting and rhyme word generation are educational exercises used to enhance early language skills in children (Bruck and Genesee, 1995). Thus, PhonologyBench provides a systematic framework to analyse the phonological skills of LLMs that inadvertently affects their performance in real-world applications (writing assistants, translators) Figure 1.

We conduct a comprehensive study across 6 widely used LLMs, including three closed-source models - GPT-4 (OpenAI, 2023), Claude-3-Sonnet (Anthropic, 2024), GPT-3.5-Turbo (OpenAI, 2022), and three open-source models - , Mistral-7B (Jiang et al., 2023) and Mixtral-8X7B (Jiang et al., 2024). Furthermore, we study the impact of word frequency, tokenization and sentence complexity across all three tasks. We find that overall LLMs exhibit surprisingly good performance on the three evaluated tasks while being trained solely on textual data. However, we observe a gap of 45% between human performance and LLM performance on syllable counting whereas the gap is nearly 17% for rhyme word generation. We also note that, based on our evaluation, no single model consistently outperforms the others across all evaluated tasks. For example, GPT-4 is the best model in Rhyme Generation while Claude-3-Sonnet outperforms GPT-4 by a large margin in Syllable Counting. This highlights that models are good at different phonological tasks and researchers should choose LLMs based on their performance on the core task related to the downstream applications.

Our main contributions are:

1. We introduce PhonologyBench, a benchmark to evaluate the phonological awareness of LLMs on three diagnostic tasks in English - grapheme-to-phoneme conversion, syllable counting and rhyme word generation. PhonologyBench offers 4k datapoints along various axes to holistically evaluate LLMs.

2. We benchmark six LLMs and study the impact of word frequency, tokenization and sentence complexity on LLM performance.

3. Our findings suggest that LLMs perform suboptimally in comparison to human. In addtion, we show that no single model is consistently good at all the tasks and researchers should choose LLMs that perform well on the tasks that affect the related downstream application.

2 Related Work

Evaluation of LLM Linguistics. 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. Contrary to our work, (Beguš et al., 2023) presents qualitative case studies on phonology, syntax and semantics of GPT-4 by generating theoretical analyses

of linguistic phenomenon on toy languages. Additionally, (Basmov et al., 2023) evaluates the performance of LLMs on linguistic inferences such as grammatically-specified entailments and monotonicity entailments using natural language inputs. More recently, (Peng et al., 2023) proposes a multi-choice question answer dataset to evaluate the spoken language knowledge of LLMs. The proposed dataset comrpises of spoken language questions from an academic source (linguistics 101) whereas PhonologyBench comprises of tasks that closely align with real-world applications.

Grapheme-to-Phoneme. Many studies have been conducted on G2P conversion. Early works have proposed joint n-gram models (Galescu and Allen, 2002) and joint-sequence models (Bisani and Ney, 2008) for G2P conversion. Recent developments in G2P studies have shown LSTM (Toshniwal and Livescu, 2016) and Transformer (Yolchuyeva et al., 2019) to be powerful G2P models. These models have access to various pronunciation dictionaries and are trained to explicitly accomplish the grapheme-phoneme conversion for speech applications like Automatic Speech Recognition(ASR) and Text-To-Speech (TTS) (Masumura et al., 2020). Finally, (Park and Kim, 2019) combines the CMUdict [2] corpus with a neural-network model to convert graphemes to phonemes. However, previous works have not analysed the ability of LLMs as G2P models.

Syllable Counting. Syllable counting is used for testing the phonological awareness of children at an early age (Bruck and Genesee, 1995; Ukrainetz et al., 2011). It is also useful in second language acquisition and commonly used as an educational tool. Additionally, counting of syllables is crucial for composing songs, poems and haikus (Tian and Peng, 2022; Henderson, 1965; Suzuki, 1985). Several works have studied the syllable structure of haikus and poetry in English, haikus are particularly popular for they 5-7-5 syllable structure (Henderson, 1965; Suzuki, 1985). Recently, (Tian and Peng, 2022), (Tian et al., 2023) and (Ormazabal et al., 2022) have utilized syllable counts as decoding contraints or metre descriptors to generate formatted sonnets, lyrics and poems. (Sun et al., 2023) evaluates the ability of LLMs to generate syllable-controlled texts (e.g. - Complete a sentence in 5 syllables). In this work, we focus

---

on evaluating the ability of LLMs to recognise and count the number of syllables in a sentence.

Rhyming and Rhymes. Rhyming words are pivotal in early age evaluation of language development in children and popularly used as an education tool to teach languages (Bruck and Genesee, 1995). Additionally, rhyming words are critical components of creative writing tasks such as poetry and song writing (Caplan, 2014). Prior works in automatic poetry and sonnet generation rely on external rhyming dictionaries sch as CMUdict to induce rhyme during generation (Tian and Peng, 2022; Tian et al., 2023). Recently, several works have studied LLM-generated poetry and creative artifacts with a focus on rhyme schemes, style and creativity (Sawicki et al., 2023). However, previous works have not evaluated the ability of LLMs to generate corresponding rhyming words as a phonological skill assessment.

3 PhonologyBench

Here, we present our benchmark, PhonologyBench, that evaluates the English phonological skills of LLMs across three tasks: Grapheme-to-Phoneme Conversion (§3.1), Syllable Counting (§3.2), and Rhyme Word Generation (§3.3). An overview of these tasks, along with their significance in a variety of downstream applications, is provided in Figure 1. Starting with Grapheme-to-Phoneme Conversion, this task is pivotal for enhancing speech-related applications, including Automatic Speech Recognition and Text-to-Speech systems, by ensuring accurate phonetic interpretations of text. Furthermore, the incorporation of phonemic and phonetic transcriptions into machine translation models introduces additional layers of understanding through phonetic embeddings, thereby improving translations (Liu et al., 2019; Chakravarthi et al., 2019). In addition, Syllable Counting and Rhyme Word Generation are traditionally popular as educational tools. Beyond facilitating language learning, they are essential in the creative processes involved in poetry and song composition, illustrating their versatility. Overall, we find that the tasks curated in PhonologyBench shed light on a wide range of real-world applications for LLMs, and benchmarking LLM performance on these tasks can advace research in LLM development for improved linguistic and creative generations. We provide the dataset statistics in Table 1 and outline the experimental methodology in § 3.1- 3.3.

Grapheme-to-Phoneme	Number
Number of Words	3K
High Frequency/Low Frequency	1K/2k
Whole-Word/Split-Word (High Frequency)	700/200
Syllable Counting	
Number of Sentences	1K
Simple/Complex	740/260
Rhyme Word Generation	
Number of Words	300
Common/Rare	200/100

Table 1: Dataset Statistics of the proposed Phonology-Bench.

3.1 Grapheme-to-Phoneme Generation

Task Description. The grapheme-to-phoneme (G2P) task involves converting the orthographic representation of a word into its phonemic representation. Writing systems often do not have a one-to-one mapping with spoken forms in English. For example, the element *o* in 'olive(//)' differs phonetically from the *o* in 'rose(/o/)'. Therefore, inferring phonetic transcriptions solely from orthography is challenging. **In this task, we prompt LLMs with a word to predict the correct phonemic form in International Phonetic Alphabet (IPA).**

Dataset and Evaluation Metric. We sample 3,000 words and their corresponding pronunciations from the SIGMORPHON 2021 G2P task (Ashby et al., 2021) for American English to curate our dataset. Then, we probe the LLMs with the prompt (zero-shot) shown in Table 6 and report the accuracy for each model. Accuracy is the percentage of words whose predicted phoneme sequences were identical to the gold references.

Baseline. We report the performance of the G2P (Park and Kim, 2019) library on our dataset as the baseline. This library is a combination of a dictionary look-up and a neural network and serves as a reasonable baseline with access to pronunciation information.

3.2 Syllable Counting

Task Description. In syllable counting, individuals must identify the vowel peaks and the consonants that may precede or follow these peaks to determine the number of syllables in a word (Bruck and Genesee, 1995). Syllable counting is an educational tool used to improve a child's phonemic awareness. Therefore, assessing the performance of LLMs can provide insights into the model's phonemic awareness. Here, **we prompt LLMs with a sentence to count the number of syllables** in the sentence.

Dataset and Evaluation Metric. Following (Sun et al., 2023), we curate 1,000 sentences from the datasets of Romance Books and Reddit Short Stories [3]. After that, we employ a grapheme-to-phoneme library to count the syllables in the sentences (Park and Kim, 2019). Finally, we probe LLMs with the prompt shown in Table 6 and report the accuracy for each model. Accuracy is the percentage of sentences whose predicted syllable counts are correct.

Baseline. We implement a naive approach by counting the number of vowels (a,e,i,o,u) in the sentence (text) as a baseline. Additionally, we ask an annotator with graduate-level engineering education and working proficiency in English (US) to perform the task and treat it as the human performance baseline.

3.3 Rhyme Generation

Task Description. In this task, we aim to analyse the LLMs' capability to generate correct rhyming words for a given word. Rhyme words are a crucial aspect of creative writing and rely on pronunciation knowledge. Since, LLMs are used popularly as creative writing assitants, LLM performance on this task provides insights about creative generation.

Dataset and Evaluation Metric. We collect 300 words from the Spelling Bee Study List (Maguire, 2006) and the Google One Trillion corpus [4] to curate our dataset. We retrieve all the rhyming words (slant and strict rhymes) for a given word from an online rhyming dictionary, WordHippo[5] and treat these as the gold reference. We obtain an average of 1200 rhyming words for a word to ensure a good coverage in our gold references. Similar to subsection 3.1, we categorized the dataset into two segments - high-frequency words and low-frequency words by using WIMBD (Elazar et al., 2024) index. Finally, we probe LLMs with the prompt shown in Table 6 and report the Success Rate (SR). We compute the word-specific success rate as the number of generated rhyming candidates that belong to the ground-truth set of rhyming words. SR is the average success rate for all the words.

---

[3]https://www.kaggle.com/datasets/trevordu/reddit-short-stories

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

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

Baseline. We present human performance as a baseline. Two human annotators with college-level education and native proficiency in English (US) performed the task. All human annotators were paid an $18 per hour and we spent $100 to acquire all annotations. We finetune a LLaMa-2-13B-Chat model (2 epochs, 2e-6 learning rate) on the common words and evaluate its performance on the rare words as a task-specific baseline.

4 Experimental Results

Here, we aim to benchmark the phonological skills of LLMs across the three tasks - G2P (§ 4.1), Syllable Counting (§ 4.2) and Rhyme Word Generation (§ 4.3). We further study the impact of word frequency in the pretraining dataset, tokenization strategy, and sentence complexity on LLM performance across all the tasks. Figure 7 shows human as well as model responses on the three tasks along with the gold references. We prompt all models in a zero-shot setting.

4.1 Grapheme-to-Phoneme

Overall Model Performance. Our results in Table 2 show that LLMs are worse than the phonologically trained baseline, G2P by ∼10% with Claude-3-Sonnet and GPT-4 achieving the highest performance. We observe that the best performing open-source model (with <15B params), is ∼34% behind GPT-4. We provide qualitative examples in Table 5.

Frequently Used Words. Usually, words that are more commonly used in real-world scenarios are more prevalent in the training dataset. Highly frequent words may also afford LLMs greater opportunities to learn their correct pronunciations during training as shown in numerical reasoning tasks (Razeghi et al., 2022). To explore LLM performance on different frequency words, we categorized the dataset into two segments - high-frequency words and low-frequency words. We count the frequency of all the words in our dataset using the index in WIMBD (Elazar et al., 2024) [6]. Our dataset comprises 1,000 high-frequency words with more than 10M occurrences and 2,000 low-frequency words that occur less than 1M in pretraining corpora. Results in Table 2 indicate that models (including the baseline) generally have higher

---

[6]WIMBD returns the number of documents where a word occurs in the C4 (Raffel et al., 2023) dataset. We use this as a proxy for occurrence frequency.

performance on high-frequency words than on low-frequency words. This could be attributed to wider availability of pronunciation information for high-frequency words in the training dataset as well as online dictionaries. However, our baseline model still outperforms the best LLM (Claude-3-Sonnet) on both high-frequency and low-frequency words highlighting the need for curating better phonology-rich datasets.

Tokenization in LLMs. Tokenization, refers to the division of input sequence of bytes into discrete tokens. LLMs primarily use Byte Pair Encoding (BPE) (Shibata et al., 1999) or its variants (Kudo and Richardson, 2018) for tokenization, leading to a generative process predominantly based on sub-word generation. We argue that this tokenization process can result in the loss of a word's phonological structure, which in turn, may impede the phoneme generation process. To investigate this further, we divided high-frequency words into two categories: whole-word tokens and split-word tokens. We utilize the tiktoken library, which provides open access to the tokenizer used in OpenAI's models, for word tokenization and subsequent data segmentation. Our analysis revealed that 30% of the high frequency words were tokenized as whole words by the OpenAI tokenizer. Results in Table 3 show that LLMs achieve higher accuracy with whole-word tokens compared to split-word tokens. We also observe that GPT-4 outperforms Claude-3-Sonnet on split-words. This highlights that subword tokenization may lead to loss of phonological information that eventually affects the phonological skills of LLMs.

4.2 Syllable Counting

Overall Model Performance. We find that Claude-3-Sonnet achieves the best performance of 55% which is far behind the human baseline at 90%. All the closed-source models beat the vowel baseline. Open-source models like and Mistral-7B perform worse than the vowel baseline indicating that these models do not have an innate understanding of syllable structure and their relation with vowels. Surprisingly, GPT-4 falls behind Claude-3-Sonnet by ∼22% while being at-par in G2P task. We provide qualitative examples in Table 4.

Sentence Complexity and Sentence Length. A syntactically complex sentence is hard to comprehend due to the increased cognitive load required to grasp the syntax and semantics of the sentence

Model	High Frequency	Low Frequency
Open-Sourced Models		
	18.0	12.6
Mistral-7B-Instruct	5.3	2.4
Mixtral-8X7B-Instruct	22.0	18.1
Closed-Sourced Models		
GPT-3.5-Turbo	47.6	34.4
GPT-4	51.9	38.1
Claude-3-Sonnet	`52.7`	`40.2`
Baseline		
G2P (Park and Kim, 2019)	62.4	52.8

Table 2: Results for grapheme-to-phoneme conversion by LLMs. We report the *Accuracy* as the percentage of correct phonemes generated by each LLM. *High Frequency* words are words that occur 100 times more than *Low Frequency* words in pretraining corpora.

Model	Whole-Word	Split-Word
Open-Sourced Models		
	22.7	15.9
Mistral-7B-Instruct	8.0	4.0
Mixtral-8X7B-Instruct	27.9	19.5
Closed-Sourced Models		
GPT-3.5-Turbo	53.1	45.1
GPT-4	58.0	`49.2`
Claude-3-Sonnet	`64.8`	47.3

Table 3: Results for grapheme-to-phoneme conversion by LLMs. We report the *accuracy* as the percentage of correct phonemes generated by each LLM. *Whole-Words* are preserved during tokenization by OpenAI tokenizer while Split-Words are split into sub-word tokens. We report LLM performance on the high-frequency words.

(Mikk, 2008). We posit that LLMs face similar difficulties in performing reasoning tasks like syllable counting over syntactically complex sentences. Therefore, we group our dataset into two categories : simple and complex sentences [7]. We utilize the spacy library [8] to identify the number of clauses in a sentence. Our dataset comprises 74% simple sentences and 26% complex sentences. Our results are shown in Table 4. We observe that with increasing sentence complexity model performance drops significantly. Usually, syntactically complex sentences are positively correlated with sentence length, hence, we show the performance of LLMs on differing sentence lengths in Appendix A. This highlights that LLMs can leverage the statistical correlations they learned from orthography to accomplish simple tasks. However, they tend to degrade in performance when faced with complex task and increased ambiguity. This shows that

LLMs are accomplishing phonology-rich tasks by utilizing side evidence from the training data instead of phonological concepts and reasoning as humans do. To our surprise, for Claude-3-Sonnet the performance is higher for complex sentences than simple sentences. This could be due to training data bias where models have seen many complex and longer sentences found in literature, academic papers, and professional communications (Elazar et al., 2024). Moreover, Claude-3 models have shown near perfect in long context evaluations indicating that they are good at processing longer inputs (Anthropic, 2024).

4.3 Rhyme Word Generation

Overall Model Performance. We find that GPT-4 achieves the best overall performance of 57.6%. In comparison, open-source models such as LLaMA and Mixtral-8X7B lag behind their closed-source counterparts, with Mixtral-8X7B securing the highest success rate among them at 27.9%. Overall, humans beat all the evaluated models by

---

[7]We overload the term 'complex' sentence to refer to sentences with more than one clause to simplify our analysis
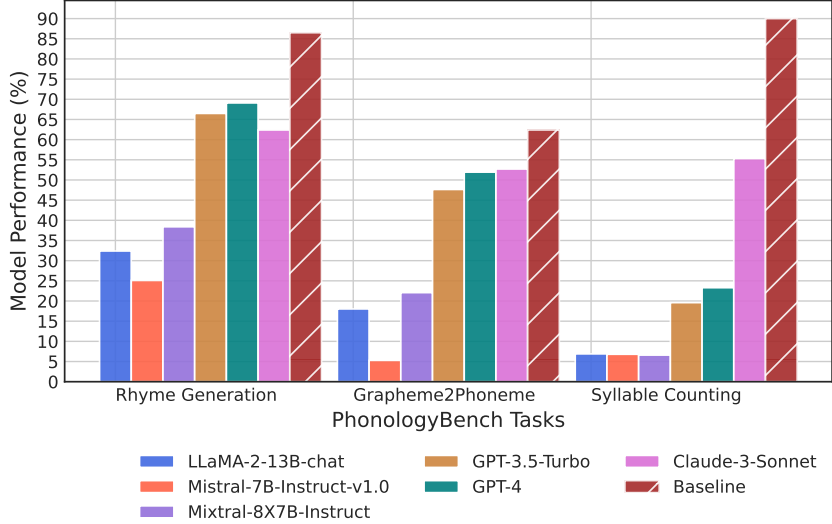
[8]https://spacy.io/

Figure 2: Performance of the 6 evaluated models on PhonologyBench. We consider human performance on Rhyme Word Generation and Syllable Counting as the baseline. For G2P, we consider a dictionary look-up based approach as the baseline.

Model	Overall	Simple	Complex
Open-Sourced Models			
	6.9	7.5	5.1
Mistral-7B-Instruct	6.8	7.0	6.2
Mixtral-8X7B-Instruct	6.6	7.0	5.5
Closed-Sourced Models			
GPT-3.5-Turbo	19.6	20.2	14.0
GPT-4	23.3	24.2	15.3
Claude-3-Sonnet	55.3	53.4	65.0
Baseline			
Vowel Baseline	12.1	12.4	12.1
Human	90.0	93.3	86.7

Table 4: Results for syllable counting by LLMs. Here, we report the *Accuracy* as the percentage of correct syllable counts predicted by LLMs. *Simple* sentences contain only one clause while *Complex* sentences contain more than one clause.

17%. Intriguingly, supervised fine-tuning of a -Chat model does not yield any performance enhancements. This suggests that additional training focused on orthography does not contribute to improving the model's capabilities in generating rhyme. We provide qualitative examples in Table 6.

Impact of Word Frequency Similar to § 3.1, we study LLM performance on words with varying frequency of usage. Our dataset comprises 200 common words with more than 10M appearances and 100 rare words that occur less than 10K in in the proxy pretraining dataset (C4). Rare words are less commonly used by English speakers and their pronunciations may not be widely known though available in large datasets. Results in Table 5 indi-

cate that LLMs exhibit higher accuracy in rhyme generation for common words compared to their rare counterparts. This could be due to less exposure to rare words and their pronunciations during training and highlights the importance of phonological information for improved LLM performance.

4.4 Discussion

In summary, our findings suggest that despite being trained on orthographic form **LLMs perform surprisingly well on English phonological tasks**. This could be attributed to the imperfect associations between text and speech learned by LLMs during to their training. Additionally, in Figure 2 we show that despite notable performance on different tasks, **LLMs fall behind relevant base-**

Model	Common Words	Rare Words
Open-Sourced Models		
	32.4	15.6
Mistral-7B-Instruct-v0.1	25.1	8.3
Mixtral-8X7B-Instruct	38.4	17.5
Closed-Sourced Models		
GPT-3.5-Turbo	66.5	42.7
GPT-4	69.1	46.1
Claude-3-Sonnet	62.4	39.6
LLaMA-2-13B (SFT)	15.8	15.8
Baseline		
Human	86.4	60.4

Table 5: Results for rhyme word generation by LLMs. Here, we report the *Success Rate* as the correct rhyming words out the five generated per word by LLM. *Common Words* are words that occur 1000 times more than *Rare Words* in pretraining corpora.

lines across all tasks. We find that the evaluated open-source models are consistently worse than the closed-source models. However, we also observe that **no single model consistently** outperforms the others in these tasks. For example, while outperforms Mistral-Instruct-7B on rhyme generation and G2P, it has the same performance on Syllable Counting. This indicates that higher performance in one task does not signify higher performance across all tasks, thus researchers should choose LLMs based on their performance on the core task for related downstream applications.

5 Conclusion

In this work, we present, PhonologyBench that consists of 4k data samples to evaluate the phonological skills of LLMs across G2P conversion, syllable counting and rhyme word generation in English. We show that LLMs fail to outperform human performance in syllable counting and rhyme word generation. Our findings thus highlight that there is scope of improving LLM performance on these core phonological task and encourages future research in training phonologically aware models. A straightforward approach to improving 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, we show that no single model consistently performs well on all the tasks. Thus, we encourage researchers to develop downstream applications

that rely on these phonological tasks by carefully selecting LLMs that perform well on the particular task.

6 Limitations

Our work analyses LLM performance on various phonological tasks. The limitations of our work are two-fold, including concept coverage and data quality. While we present a comprehensive study on the various facets of LLM performance on G2P, syllable counting and rhyme word generation, our work only covers a limited portion of the variety of phonological phenomenon such as blending, homonyms and homographs. Additionally, our work focuses on American English and does not extend to various dialects. Future studies can focus on multilingual phonological phenomenon as well as LLM performance on different dialects. We encourage research that addresses these various languages, dialects and phenomenon to further the understanding of LLM language skills. Finally, our benchmark though extensive is limited by the quality of the curated gold references. For example, despite a large coverage of rhyming words we cannot be certain that our gold references are exhaustive. Similarly, LLMs can process syllable counts differently for clinical data or scientific communications that is not covered in PhonologyBench. Therefore, we encourage further contributions from the research community to help develop high-quality evaluation sets.

7 Acknowledgements

References

Anthropic. 2024. Introducing the next generation of claude. https://www.anthropic.com/news/claude-3-family.

Lucas FE Ashby, Travis M Bartley, Simon Clematide, Luca Del Signore, Cameron Gibson, Kyle Gorman, Yeonju Lee-Sikka, Peter Makarov, Aidan Malanoski, Sean Miller, et al. 2021. Results of the second sigmorphon shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 115–125.

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. *Preprint*, arXiv:2302.04023.

Victoria Basmov, Yoav Goldberg, and Reut Tsarfaty. 2023. Chatgpt and simple linguistic inferences: Blind spots and blinds. *Preprint*, arXiv:2305.14785.

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. *Preprint*, arXiv:2305.00948.

Maximilian Bisani and Hermann Ney. 2008. Joint-sequence models for grapheme-to-phoneme conversion. *Speech communication*, 50(5):434–451.

Sam Bond-Taylor, Adam Leach, Yang Long, and Chris G Willcocks. 2021. Deep generative modelling: A comparative review of vaes, gans, normalizing flows, energy-based and autoregressive models. *IEEE transactions on pattern analysis and machine intelligence*, 44(11):7327–7347.

Maggie Bruck and Fred Genesee. 1995. Phonological awareness in young second language learners. *Journal of child Language*, 22(2):307–324.

David Caplan. 2014. *Rhyme's challenge: hip hop, poetry, and contemporary rhyming culture*. Oxford University Press, USA.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Bernardo Stearns, Arun Kumar Jayapal, S Sridevy, Mihael Arcan, Manel Zarrouk, and John P McCrae. 2019. Multilingual multimodal machine translation for dravidian languages utilizing phonetic transcription. In *Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages*, pages 56–63.

Shuangrui Ding, Zihan Liu, Xiaoyi Dong, Pan Zhang, Rui Qian, Conghui He, Dahua Lin, and Jiaqi Wang. 2024. Songcomposer: A large language model for lyric and melody composition in song generation. *Preprint*, arXiv:2402.17645.

Yanai Elazar, Akshita Bhagia, Ian Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, Hanna Hajishirzi, Noah A. Smith, and Jesse Dodge. 2024. What's in my big data? *Preprint*, arXiv:2310.20707.

Lucian Galescu and James F Allen. 2002. Pronunciation of proper names with a joint n-gram model for bi-directional grapheme-to-phoneme conversion. In *Seventh International Conference on Spoken Language Processing*.

Harold G Henderson. 1965. Haiku in english.

Jennifer Hu and Roger Levy. 2023. Prompt-based methods may underestimate large language models' linguistic generalizations. *Preprint*, arXiv:2305.13264.

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*.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Xing Wang, and Zhaopeng Tu. 2023. Is chatgpt a good translator? yes with gpt-4 as the engine. *Preprint*, arXiv:2301.08745.

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.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Taeahn Kwon. 2023. *Interfaces for Personalized Language Learning with Generative Language Models*. Ph.D. thesis, Columbia University.

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. *Preprint*, arXiv:2211.09110.

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.

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

Ryo Masumura, Naoki Makishima, Mana Ihori, Akihiko Takashima, Tomohiro Tanaka, and Shota Orihashi. 2020. Phoneme-to-grapheme conversion based large-scale pre-training for end-to-end automatic speech recognition. In *INTERSPEECH*, pages 2822–2826.

Jaan Mikk. 2008. Sentence length for revealing the cognitive load reversal effect in text comprehension. *Educational Studies*, 34(2):119–127.

OpenAI. 2022. Introducing chatgpt.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Aitor Ormazabal, Mikel Artetxe, Manex Agirrezabal, Aitor Soroa, and Eneko Agirre. 2022. Poelm: A meter-and rhyme-controllable language model for unsupervised poetry generation. *arXiv preprint arXiv:2205.12206*.

Kyubyong Park and Jongseok Kim. 2019. g2pe. https://github.com/Kyubyong/g2p.

Linkai Peng, Baorian Nuchged, and Yingming Gao. 2023. Spoken language intelligence of large language models for language learning. *Preprint*, arXiv:2308.14536.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *Preprint*, arXiv:2302.06476.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot reasoning. *arXiv preprint arXiv:2202.07206*.

Lars Ruthotto and Eldad Haber. 2021. An introduction to deep generative modeling. *GAMM-Mitteilungen*, 44(2):e202100008.

Piotr Sawicki, Marek Grzes, Fabricio Goes, Dan Brown, Max Peeperkorn, and Aisha Khatun. 2023. Bits of grass: Does gpt already know how to write like whitman? *arXiv preprint arXiv:2305.11064*.

Yusuxke Shibata, Takuya Kida, Shuichi Fukamachi, Masayuki Takeda, Ayumi Shinohara, Takeshi Shinohara, and Setsuo Arikawa. 1999. Byte pair encoding: A text compression scheme that accelerates pattern matching.

Jiao Sun, Yufei Tian, Wangchunshu Zhou, Nan Xu, Qian Hu, Rahul Gupta, John Frederick Wieting, Nanyun Peng, and Xuezhe Ma. 2023. Evaluating large language models on controlled generation tasks. *Preprint*, arXiv:2310.14542.

Seiichi Suzuki. 1985. The role of syllable structure in old english poetry. *Lingua*, 67(2-3):97–119.

Yufei Tian, Anjali Narayan-Chen, Shereen Oraby, Alessandra Cervone, Gunnar Sigurdsson, Chenyang Tao, Wenbo Zhao, Yiwen Chen, Tagyoung Chung, Jing Huang, and Nanyun Peng. 2023. Unsupervised melody-to-lyric generation. *Preprint*, arXiv:2305.19228.

Yufei Tian and Nanyun Peng. 2022. Zero-shot sonnet generation with discourse-level planning and aesthetics features. *Preprint*, arXiv:2205.01821.

Shubham Toshniwal and Karen Livescu. 2016. Jointly learning to align and convert graphemes to phonemes with neural attention models. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 76–82. IEEE.

Teresa A Ukrainetz, Janae J Nuspl, Kimberly Wilkerson, and Sarah Rose Beddes. 2011. The effects of syllable instruction on phonemic awareness in preschoolers. *Early childhood research quarterly*, 26(1):50–60.
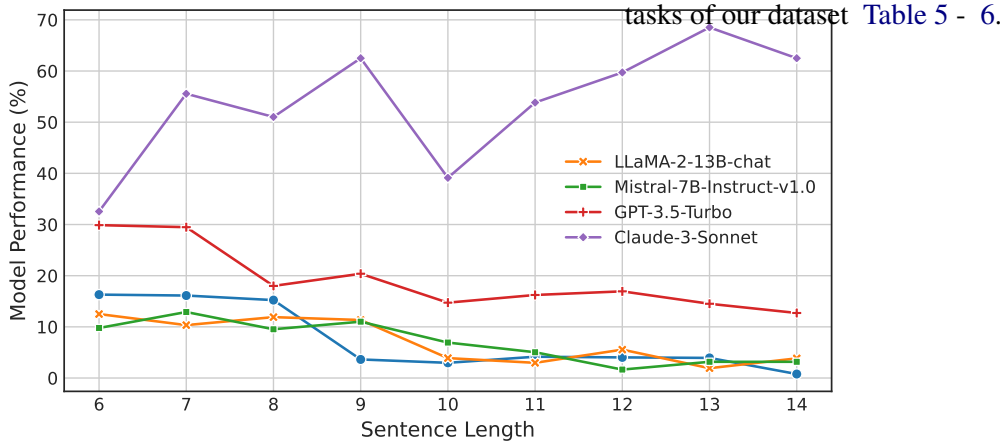
tasks of our dataset Table 5 - 6.

Figure 3: Performance of the 4 models on syllable counting at varying sentence lengths.

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). *Preprint*, arXiv:2206.10498.

Vice. 2023. Chatgpt is pretty bad at poetry, according to poets. https://www.vice.com/en/article/7kx9d9/chatgpt-is-pretty-bad-at-poetry-according-to-poets.

Sevinj Yolchuyeva, Géza Németh, and Bálint Gyires-Tóth. 2019. Transformer based grapheme-to-phoneme conversion. In *Interspeech 2019*, interspeech2019. ISCA.

Chengyue Yu, Lei Zang, Jiaotuan Wang, Chenyi Zhuang, and Jinjie Gu. 2024. Token-free llms can generate chinese classical poetry with more accurate format. *arXiv preprint arXiv:2401.03512*.

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. *Preprint*, arXiv:2306.05685.

A   Syllable Counting at Varying Sentence Lengths

In this section, we study the impact of sentence length on LLMs on the syllable counting task. Figure 3 shows that overall models have higher accuracy for shorter sentence (length=6) than longer sentences(length=13). This highlights the relationship between sentence complexity and sentence length with similar trends as § 4.2.

B   Qualitative Examples

In this section, we present the qualitative examples to show the varied model responses on different

Task	Prompt
Grapheme-to-Phoneme	You are an expert in American English phonology, phonetics and morphology. In this task, you are required to map a sequence of graphemes -characters representing a word to a transcription of that word's pronunciation (phonemes). If you cannot complete the task, respond with - NONE.
	Generate the phoneme for the grapheme :'<input>'
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 ending sounds.
	Give 5 words that rhyme with '<input>'.
Syllable Counting	Syllable is a unit of pronunciation having one vowel sound, with or without surrounding consonants, forming the whole or a part of a word.
	Count the number of syllables in the given sentence :'<input>'

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

Figure 7: Samples from all three tasks in PhonologyBench

Instruction: Count syllables in this sentence: "She meets him randomly in the woods at his family's cabin."

Ground Truth Count: 16   **GPT-4:** Counting the syllables in the phrase "She meets him randomly in the woods at his family's cabin," we have:

She: 1 syllable
meets: 1 syllable
him: 1 syllable
randomly: 3 syllables
in: 1 syllable
the: 1 syllable
woods: 1 syllable
at: 1 syllable
his: 1 syllable
family's: 3 syllables
cabin: 2 syllables
So, the total is 16 syllables in this phrase.

Claude-3-Sonnet: To count the number of syllables in the given sentence, we need to break it down into individual words and count the syllables in each word. Here's the breakdown:

"She" - 1 syllable
"meets" - 1 syllable
"him" - 1 syllable
"ran-dom-ly" - 3 syllables
"in" - 1 syllable
"the" - 1 syllable
"woods" - 1 syllable
"at" - 1 syllable
"his" - 1 syllable
"fam-i-ly's" - 3 syllables
"cab-in" - 2 syllables

Adding up the syllables for each word, we get: 16 syllables

Mistral-7b-instruct: The number of syllables in the given sentence is 12.

Llama-2-Chat: Sure! Here's the number of syllables in each word of the sentence:

1. She - 1 syllable
2. meets - 2 syllables
3. him - 1 syllable
4. randomly - 3 syllables
5. in - 1 syllable
6. the - 1 syllable
7. woods - 2 syllables
8. at - 1 syllable
9. his - 1 syllable
10. family's - 2 syllables
11. cabin - 1 syllable

Total number of syllables: 11

Instruction: Count syllables in this sentence: "To top it all off, I miss my Stunner"

Ground Truth Count: 10   **GPT-4:** Counting the syllables in the phrase "To top it all off, I miss my Stunner," we have:
To: 1 syllable
top: 1 syllable
it: 1 syllable
all: 1 syllable
off: 1 syllable
I: 1 syllable
miss: 1 syllable
my: 1 syllable
Stun-ner: 2 syllables
The total is 10 syllables in this phrase.

Claude-3-Sonnet: To count the number of syllables in the given sentence, we need to break it down into individual words and then count the syllables in each word.

The sentence is: "To top it all off, I miss my Stunner."

Here's the breakdown:

1. "To" - 1 syllable
2. "top" - 1 syllable
3. "it" - 1 syllable
4. "all" - 1 syllable
5. "off" - 1 syllable
6. "I" - 1 syllable
7. "miss" - 1 syllable
8. "my" - 1 syllable
9. "Stun-ner" - 2 syllables

By adding up the syllables in each word, we get:

1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 2 = 10 syllables
Therefore, the given sentence "To top it all off, I miss my Stunner." has 10 syllables.

Mistral-7b-instruct: The number of syllables in the given sentence is 10.

Llama-2-Chat: Sure! Here's the syllable count for the sentence "To top it all off, I miss my Stunner":
1. To
2. top
3. it
4. all
5. off
6. I
7. miss
8. my
9. Stunner
There are 9 syllables in the sentence.

Figure 4: We provide responses of GPT-4, Claude-3-Sonnet, and Mistral-Instruct-7B on two samples from the Syllable Counting task.

Instruction: Give the phonetic transcription of 'Brass' in IPA"
Ground Truth: /b r æ s/
GPT-4: Word : The phonetic transcription for "brass" in IPA in American English is /bræs/.

Claude-3-Sonnet: The phonemic transcription of the word "brass" in American English is: /bræs/

Mistral-7b-instruct: The phoneme transcription for the grapheme sequence "brass" is /bræs/

Llama-2-Chat: Sure, I can help you with that! The word "brass" has the following graphemes: b-r-a-s-s And its transcription in phonemes would be: /bræs/

Instruction: Give the phonetic transcription of 'Oath' in IPA"
Ground Truth: /o/
GPT-4: The phonetic transcription for "oath" in IPA in American English /o/

Claude-3-Sonnet: The phonemic transcription of the word "oath" in American English is: /o /

Mistral-7b-instruct: The phoneme transcription for the word "oath" is /o:/

Llama-2-Chat: Sure, I can help you with that! The word "oath" has a pronunciation of /O/. So, the phoneme for the grapheme "oath" is /O/.

Figure 5: We provide responses of GPT-4, Claude-3-Sonnet, and Mistral-Instruct-7B on two samples from the G2P task.

Instruction: Give 5 words that rhyme with submitted."
Ground Truth: Committed, Admitted, Acquitted, Outwitted, Transmitted, Permitted, Uplifted, Omitted.
GPT-4: Word : submitted
1. Admitted
2. Committed
3. Permitted
4. Omitted
5. Transmitted

Claude-3-Sonnet: Here are 5 words that rhyme with "submitted":

1. Committed
2. Omitted
3. Transmitted
4. Outwitted
5. Permitted

Mistral-7b-instruct: Some words that rhyme with "submitted" are:
1. Consumed
2. Exhumed
3. Jumped
4. Lumbered
5. Rumpus

Llama-2-Chat: Sure! Here are five words that rhyme with "submitted":

1. submitted
2. committed
3. submitted
4. neglected
5. invested

Instruction: Give 5 words that rhyme with trademarks."
Ground Truth: Remarks, Landmarks, Watermarks, Marks, Barks, Sharks, Hallmarks, Parks, Sparks, Benchmarks, Larks, Monarchs. **GPT-4:**Word : trademarks
1. Parks
2. Sparks
3. Sharks
4. Larks
5. Marks

Claude-3-Sonnet: Here are 5 words that rhyme with "trademarks":

1. Spacebarks
2. Masquerades
3. Charades
4. Crusades
5. Blockades

Mistral-7b-instruct: Sure are five words that rhyme with "trademarks":
1. Sparklines
2. Landmarks
3. Darkening
4. Outsmarting
5. Repairs

Llama-2-Chat: Sure, here are five words that rhyme with "trademarks":
1. brands
2. marks
3. sparks
4. sharks
5. arks

Figure 6: We provide responses of GPT-4, Claude-3-Sonnet, and Mistral-Instruct-7B on two samples from the Rhyme Word Generation task.