From Babble to Words 💬 :
Pre-Training Language Models on Continuous Streams of Phonemes

Zébulon Goriely 🍊 **Richard Diehl Martinez** 🍊
Andrew Caines 🍊🍋 **Lisa Beinborn** 🍏 **Paula Buttery** 🍊🍋
🍊 Department of Computer Science & Technology, University of Cambridge, U.K.
🍋 ALTA Institute, University of Cambridge, U.K.
🍏 University of Göttingen, Germany
🍊 firstname.secondname@cl.cam.ac.uk 🍏 lisa.beinborn@uni-goettingen.de

Abstract

Language models are typically trained on large corpora of text in their default orthographic form. However, this is not the only option; representing data as streams of phonemes can offer unique advantages, from deeper insights into phonological language acquisition to improved performance on sound-based tasks. The challenge lies in evaluating the impact of phoneme-based training, as most benchmarks are also orthographic. To address this, we develop a pipeline to convert text datasets into a continuous stream of phonemes. We apply this pipeline to the 100-million-word pre-training dataset from the BabyLM challenge, as well as to standard language and grammatical benchmarks, enabling us to pre-train and evaluate a model using phonemic input representations. Our results show that while phoneme-based training slightly reduces performance on traditional language understanding tasks, it offers valuable analytical and practical benefits.

🤗 phonemetransformers/FromBabbleToWords (CC BY 4.0)
codebyzeb/PhonemeTransformers (CC BY 4.0)
codebyzeb/CorpusPhonemizer (CC BY 4.0)

1 Introduction

The use of orthographic text to train neural networks is so commonplace that it is considered the default. This has not always been the case.

When neural networks were first applied to language, models were primarily trained on continuous streams of phonemes or graphemes, rather than orthographic text with its written artefacts. These early neural models demonstrated a striking ability to acquire phonology, syntax and semantics (Elman, 1990; Seidenberg and McClelland, 1989; Prince and Smolensky, 1997). As technology scaled, subword representations became the dominant representation, offering key advantages such as reducing computation costs and better capturing out-of-vocabulary items (Sennrich et al., 2016). Written text became favored over speech transcriptions due to matching the domain of downstream tasks and due to the abundance of diverse texts available through web-scraping (Bansal et al., 2022). Today, "large language models" (LLMs) all use subword-based text inputs and perform impressively on a variety of language understanding tasks (Zellers et al., 2019; Hendrycks et al., 2020; Suzgun et al., 2023).

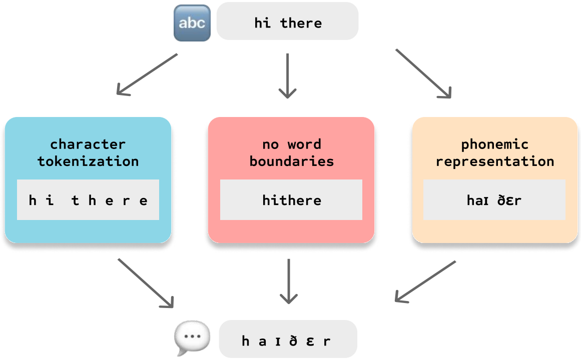

Figure 1: An illustration of all three adjustments that we make to convert text input to continuous streams of phonemes.

The success of these models on downstream tasks has motivated researchers to examine the internal representations of LLMs and analyze their ability to learn grammatical generalizations (Hewitt and Manning, 2019; Hu et al., 2020; Manning et al., 2020). However, their phonological capabilities remain understudied due to the orthographic nature of training data.

An alternative input representation for text-based language models is to use phonemes rather than graphemes, corresponding to how words are pronounced, rather than how they are written. The use of phonemes, such as those described by the International Phonetic Alphabet (IPA), as an underlying input representation, presents the following analytical and practical benefits over an orthographic representation that is the modern-day default.

Analytical: A phoneme-based representation is useful when using language models to study the distributional properties of phonemes (Mayer, 2020) and phonological systems of languages more broadly (Eden, 2018). Many language acquisition studies prefer using phonemes as a representation that more closely represents the human learning environment, which facilitates statistical learning experiments ranging from word segmentation (Çöltekin, 2017), to past-tense formation (Kirov and Cotterell, 2018), and broader lexico-syntactic knowledge (Lavechin et al., 2023).

Practical: IPA-encoded text has been found to be beneficial for a variety of NLP tasks including lyric generation (Ding et al., 2024), text-to-speech (Sundararaman et al., 2021; Li et al., 2023) and low-resource language modeling (Leong and Whitenack, 2022). Phonemes also benefit multi-lingual language modeling by establishing a universal representation shared between languages (Feng et al., 2023; Zhu et al., 2024).

Despite the analytical and practical advantages of training language models with phonemes, a key question remains: *Can modern language model architectures encode grammatical knowledge and succeed at language understanding tasks when trained with phoneme-based representations?*

Answering this question is challenging for two reasons. First, training and evaluation data need to be provided to a model in both a phonemic and graphemic representation. Second, it is nontrivial to select the transformations to convert orthographic text into phonemic representations and to evaluate how these individually affect a model's performance across a wide variety of benchmarks.

In this work, we address these challenges as follows. We first present a method for converting training data and evaluation benchmarks into a unified IPA representation. This enables language models to be trained and evaluated on graphemic and phonemic representations of the same data. We then identify three key transformations which enable us to map from the written representation typically used to train language models to the phonemic representation often used in analytical studies (see fig. 1). Finally, we conduct a careful ablation of the three transformations: we train a language model on the same corpus of 100 million words with all combinations of the three transformations (2^3 configurations), evaluating the model's grammatical capabilities and its resulting performance on downstream language understanding tasks.

We find that large language models are powerful statistical learners capable of learning grammar from a phonemic input representation. Although we observe a decrease in performance on some tasks, the degradation is not as substantial as has been anecdotally suggested by previous studies. Our ablation studies indicate that the impact of each transformation that we use to convert orthographic text to continuous phoneme streams depends on the downstream task; tasks in the BLiMP Supplement set are particularly sensitive to the use of phonemes, while those in GLUE are sensitive to character tokenization. A deeper analysis into these ablations reveals that many evaluation instances rely on information only present in written text (such as punctuation). Finally, we take advantage of the fact that we train models using phonemic streams and evaluate our models for phonological knowledge using the BabySLM benchmark. Our models achieve the best scores on this benchmark to date.

2 Related Work

The standard input representation for training large language models consists of written text split into subword units. By contrast, studies that train models using a phonemic input representation tend to split words into individual phonemes, without word boundaries (as spoken utterances are produced continuously, without clear pauses between words).

We identify three key transformations that bring us from the standard input representation used by language models to this alternative **phoneme stream** representation:

- Character tokenization Treating each phoneme or grapheme as a token, rather than using subwords.
- Word boundary removal Removing whitespace or other word boundary cues from the input.
- Phonemic transcription Converting words to a phonemic representation.

Each transformation can be made independently or in combination, as illustrated in fig. 1.

Previous literature has extensively explored these three transformations but they have typically been studied independently and been used for different downstream purposes.

2.1 Training with Phonemes

Several language models have been trained with phonemic input (Sundararaman et al., 2021; Gale et al., 2023) but it remains a challenge to do so due to the lack of large phonemic corpora. While a number of well-known speech-based datasets include phonemic transcriptions, such as Switchboard (Godfrey et al., 1992) and TIMIT (Garofolo et al., 1993), these datasets are small compared to the trillions of tokens contained in standard language model pre-training corpora (Elazar et al., 2024). The majority of works that use phonemic representations typically rely on grapheme to phoneme conversion tools (Bisani and Ney, 2008; Hasegawa-Johnson et al., 2020) to generate coarse phonemic transliterations of text data.

It is also a challenge to evaluate the broad capabilities of language models trained with phonemes, as most benchmarks assume a graphemic representation, even some that assess phonological knowledge (Suvarna et al., 2024). One benchmark that assesses both the syntactic and phonological capabilities of language models is BabySLM (Lavechin et al., 2023). We discuss this benchmark further in section 5.1.

2.2 Character-based Language Models

The use of characters as the input representation, rather than words or subwords, has been extensively explored. Character-level language models offer a simplified input stream compared to the standard approach of training on learned subword tokens. Many studies have developed specialized architectures to train language models on characters (Jozefowicz et al., 2016; Kim et al., 2016; Ma et al., 2020; Al-Rfou et al., 2019) while other approaches seek to establish 'token-free' training regimes to eliminate the need for subwords entirely (Clark et al., 2022; Xue et al., 2022).

Another alternative input representation is to split words into morphemes, which provide theoretical benefits over subwords and have their own analytical and practical benefits particularly for morphologically rich languages (Üstün et al., 2018; Nzeyimana and Niyongabo Rubungo, 2022; Fan and Sun, 2023). Mapping orthographic text to morphemes continues to be a challenging task, requiring dedicated systems trained on labeled corpora (Batsuren et al., 2022) and we do not consider morphemes in this work.

2.3 Removal of Word Boundaries

When using a phonemic input representation to model speech, word boundaries are not typically included, as word boundaries are not explicitly marked in the speech stream. The phoneme stream representation (i.e., the combination of all three transformations) is the typical representation for word segmentation studies, where the task is to learn word boundaries without supervision (Brent, 1999). A wide variety of statistical, dynamic programming and neural approaches have been applied to the task, with consequences for acquisition research and low-resource language modeling (Blanchard et al., 2010; Çöltekin, 2017; Algayres et al., 2022; Goriely et al., 2023).

2.4 Input Representation Comparisons

To the best of our knowledge, a full systematic comparison of the three input transformations has not yet been conducted. Hahn and Baroni (2019) investigated the effect of removing word boundaries and using a word-level or character-level tokenization, evaluating on several psycholinguistic benchmarks. However, they only used graphemic text from Wikipedia and did not ablate the two transformations, only comparing a word-level model (with word boundaries) to a character-level model (without word boundaries). Nguyen et al. (2022) extend this work, comparing character-level graphemic input (with and without word boundaries) to character-level phonemic input (with and without word boundaries) by training on the Librispeech corpus (Panayotov et al., 2015). They also compare larger units of tokenization (BPE and word-level) for both graphemic and phonemic text, but only with word boundaries included, missing out on several key combinations.

In our work, we provide a complete comparison of these three input representation transformations by considering all combinations, leading to new input representations that have not been studied before (such as subword tokenization trained without word boundaries). We also use a larger model than previous work, a 12-layer transformer rather than a 3-layer LSTM.

3 Phoneme Stream Pipeline

To convert the data to a phonemic representation, we developed the **Corpus Phonemizer** tool:[1] a li-

[1] https://github.com/codebyzeb/Corpus-Phonemizer

brary to convert various corpora across many different languages to a unified phonemic representation in IPA, prepare them as Huggingface datasets and subsequently train Huggingface tokenizers.

3.1 Dataset Phonemization

Our toolkit leverages the `phonemizer` package (Bernard and Titeux, 2021) with the `espeak-ng` backend[2] which uses a combination of a pronunciation dictionary and pronunciation rules to convert orthographic transcriptions to IPA. We select the American English accent (`en-US`) for a consistent pronunciation.

The tool outputs phonemes separated by spaces.[3] For instance, the phonemic representation of "what a conundrum!" is:

w ʌ t ␣ ʌ ␣ k ə n ʌ n d ɹ ə m ␣

One limitation of our phonemization tool is that 'a' is not reduced to the shwah, 'ə' as it would be in continuous speech. We discuss the limitations of this phonemization process in section 6.2. Crucially, we lose punctuation marks, as they are an artefact of orthographic text and equivalent information in speech would be conveyed through prosody, stress, or non-linguistic signals such as gestures, none of which are included in this simple phonemic format. This has potential consequences for downstream tasks that rely on such markers, as discussed in section 5.3.

3.2 Tokenizer Preparation

Using the phonemic data transcribed by the Corpus Phonemizer tool, our pipeline then implements the three input transformations by preparing different tokenizers:

- Character tokenization We either train the tokenizer using the Byte-Pair Encoding (BPE) algorithm (Sennrich et al., 2016) (✗) or create a character-based tokenizer by extracting a vocabulary from the data (✓).
- Word boundary removal We either train the tokenizer with whitespace included (✗) or use the tokenizer's normalizer to strip whitespace (✓).
- Phonemic transcription The tokenizer is either trained on the original orthographic dataset (✗), or the phonemized version described above (✓).

These transformations can be made independently, allowing for all eight combinations of the transformations to be implemented as individual tokenizers. For the combination of BPE and no word boundaries, the whitespace is removed before training, so the model may learn 'subwords' that cross word boundaries.

Each tokenizer also adds a dedicated "utterance boundary" token `UTT_BOUNDARY` to the start of each sentence, representing the pauses between spoken utterances and serving as a dedicated start-of-sentence token. When sentences are collated, it also implicitly acts as an end-of-sentence token, as discussed in appendix B.2.

4 Experimental Setup

We evaluate the effect of our proposed input adjustments by training a GPT-2 model (Radford et al., 2019) using the BabyLM challenge framework (Choshen et al., 2024). The model is trained eight times with each combination of the three input adjustments. Following the STRICT track of the BabyLM challenge, we train on a provided corpus of 100 million words and evaluate on a series of benchmarks assessing the grammatical knowledge and the downstream capabilities of each model. We additionally evaluate on BabySLM (Lavechin et al., 2023) which provides syntactic and lexical scores specifically for speech-based models. Our phonemized dataset, trained models and tokenizers are hosted on Huggingface.[4]

4.1 Dataset

The BabyLM 2024 pretraining data contains 100 million words sourced from nine different corpora (Warstadt et al., 2023). Over 50% of the data consists of transcribed or scripted speech and over 40% comes from child-directed sources (written or spoken). We apply minor cleaning operations to the dataset, removing extraneous spaces and formatting anomalies using regular expressions.

4.2 Tokenizers

For each of the eight combinations of the three transformations, we train a tokenizer on the 'train' portion of the BabyLM dataset. We compare the

[2] `https://github.com/espeak-ng/espeak-ng`

[3] It is common practice to separate phonemes by spaces to make tokenization simple, as some individual phonemes may consist of several symbols, e.g. tʃ or ɜɪ.

[4] `https://huggingface.co/collections/phonemetransformers/from-babble-to-words-66e068b54765a48ff30273c9`

Model	Character tokenization	Word boundary removal	Phonemic transcription	Vocabulary Size	Example Tokenization	BLiMP Filtered	BLiMP Supplement	GLUE	BabySLM (Syntactic)	BabySLM (Lexical)
Baby Llama	✗	✗	✗	16,000	␣what ␣a ␣con und rum ␣!	73.1	60.6	69.0	94.0	-
LTG-BERT	✗	✗	✗	16,000	␣what ␣a ␣con und r um ␣!	69.3	66.5	68.4	75.8	-
GPT-2	✗	✗	✗	16,000	␣what ␣a ␣con und rum ␣!	**77.8**	**69.4**	**71.6**	92.8	-
	✗	✓	✗	16,000	what acon un drum !	73.9	64.3	68.6	73.9	-
	✗	✗	✓	16,000	␣wʌt ␣ʌ ␣kən ʌnd ɹəm	74.7	59.6	68.6	85.8	67.3
	✗	✓	✓	16,000	wʌt ʌkən ʌnd ɹəm	71.7	56.7	65.5	74.7	71.2
	✓	✗	✗	115	w h a t ␣ a ␣ c o n u n d r u m ␣ !	77.4	63.6	64.4	**94.9**	-
	✓	✓	✗	114	w h a t a c o n u n d r u m !	75.1	64.8	64.8	88.3	-
	✓	✗	✓	51	w ʌ t ␣ ʌ ␣ k ə n ʌ n d ɹ ə m	74.7	58.5	65.6	90.5	**89.6**
	✓	✓	✓	50	w ʌ t ʌ k ə n ʌ n d ɹ ə m	72.5	57.6	65.4	83.9	87.8

Table 1: Results for the two BabyLM baseline models and the GPT-2 model trained under all eight conditions. On the left, we compare the effects of each of the three transformations across all eight possible combinations, by tokenizing the example phrase "what a conundrum!". The '␣' character denotes word boundaries. On the right, we report BLiMP, GLUE and BabySLM scores achieved by each model, with the best scores in each column in **bold**.

output of the eight tokenizers in table 1. We used a vocabulary size of 16,000 for the BPE tokenizers to match the vocabulary size used by the two baseline models provided by the BabyLM challenge (described below).

Note that the vocabulary size for the character-level tokenizers operating on phonemes is less than half the vocabulary size of their orthographic equivalents. This is because the phonemic data only consists of the 47 phonemes produced by the American English accent, but the orthographic data includes numbers, punctuation and other symbols.

4.3 Model

Our experiments use the GPT-2 architecture. We train the model using all eight tokenizers (using the phonemized dataset for the phoneme-based tokenizers) for 400k steps, selecting the checkpoint with the lowest perplexity.[5] See appendix A for a full description of the chosen model parameters and training procedure.

We also report the results from two baseline models which achieved the highest scores at the 2023 BabyLM challenge. These are Baby Llama, an auto-regressive model, which was trained using knowledge distillation from an ensemble of teachers (Timiryasov and Tastet, 2023) and LTG-BERT, an architectural variation of the standard auto-encoding BERT architecture optimized for small, speech-based corpora (Samuel et al., 2023; Charpentier and Samuel, 2023). Both models use a BPE tokenizer with a vocabulary size of 16,000 and have a similar number of parameters to our model.[6]

4.4 Evaluation

We follow the BabyLM Challenge's framework and evaluate on BLiMP (Warstadt et al., 2020), BLiMP Supplement (Choshen et al., 2024) and a subset of the (Super)GLUE tasks (Wang et al., 2018, 2019). BLiMP assesses a model's ability to distinguish grammatical sentences from ungrammatical sentences across 67 subtasks covering a range of linguistic phenomena. BLiMP Supplement consists of 5 BLiMP-style tasks covering additional linguistic phenomena not tested by BLiMP. The GLUE suite assesses a language model's language understanding abilities on typical downstream tasks using fine-tuning.

We also evaluate our models on BabySLM (Lavechin et al., 2023), a benchmark specifically designed for probing speech-based LMs at a *syntactic* level and a *lexical* level. The benchmark was

[5] The best checkpoint for five of the eight models was the final checkpoint but a visual inspection of the curve revealed that differences between the final checkpoints were minimal.

[6] Our GPT-2 model has 85M non-embedding parameters. Baby Llama has 41M and LTG-Bert has 110M.

also designed to compare text-based models (those considered here, including both orthographic text and phonemic transcriptions) to speech-based models (which learn directly from audio) by providing parallel text and audio test instances. Finally, the vocabulary items were chosen to be compatible with children's language experiences, aiming to better reflect the input that children are exposed to as they begin to acquire language.

The BabySLM syntactic metric is similar to BLiMP, using pairs of grammatical and ungrammatical sentences, but consists of shorter sentences across just six simple syntactic phenomena. By comparison, BLiMP complicated many grammatical phenomena which may be rarely used even in adult–adult spontaneous conversation.

The lexical metric consists of minimal pairs of words and pseudo-words in a phonemic representation, representing a 'real-word recognition' task to assess a model's lexicon and phonemic capabilities. For instance, the model should assign a higher likelihood to the real-word t ɛ m p ɹ ə tʃ ə ɹ (temperature) compared to the pseudo-word t ɛ m p f ə tʃ ə ɹ (tempfature). This metric is related to the pronunciation of words, rather than the spelling of words and so cannot be used to evaluate models trained on orthographic text (which have no concept of pronunciation).

To evaluate our phoneme-based models, we run our phonemizer tool on all test instances across these benchmarks (except for the BabySLM lexical examples, which are already in IPA).

5 Results

In table 1, we report a summary of the results obtained by the two BabyLM baseline models and our GPT-2 model trained in all eight conditions. Due to limited computational resources we only train a single run per condition, limiting our ability to critique them individually. Exact results may be subject to variance across random seeds but we can still observe trends over the whole set.

The base GPT-2 model with no input adjustments outperforms the two baselines for BLiMP, BLiMP Supplement and GLUE, validating our selection of hyper-parameters and choice of architecture as described in appendix A.

Comparing the GPT-2 model with no input transformations (top row) to the same model with all three transformations applied (bottom row), we notice a decrease in performance across all benchmarks. Although this indicates that the GPT-2 architecture is best suited for the standard orthographic input representation (word boundaries, graphemes and subword tokenization), the decrease in performance when the three transformations are applied is not substantial and scores remaining competitive with the baseline models (all combinations still outperform LTG-BERT on BLiMP). It is clear that the model is still capable of learning grammatical rules and excelling at downstream tasks when the input consists of individual phonemes with no word boundaries.

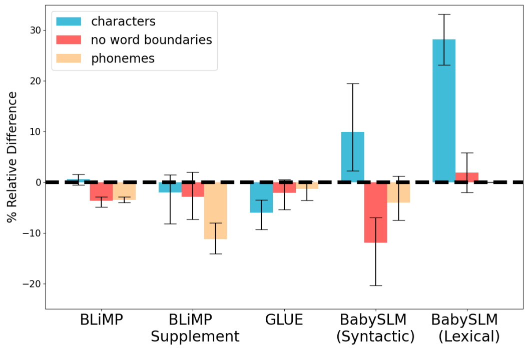

Figure 2: Mean (with Min and Max range) percentage difference achieved on each benchmark's macro score as a result of the three adjustments.

In section 5.1 we investigate this result further through an ablation of the three transformations, noting the effect of punctuation and context size. In section 5.2 we focus on the BabySLM metrics, which demonstrate a different pattern to the other benchmarks. Finally, in section 5.3 we investigate the consequences of removing punctuation in our phonemic transcriptions.

5.1 Teasing Apart the Three Transformations

By running our GPT-2 model with all eight combinations of the three input adjustments, we can tease apart the effect of each transformation.

For each transformation, we can create four pairs of runs that only differ with respect to that transformation (e.g. the four runs with a phonemic transcription and the four runs with orthographic text). For each pair, we calculate the percentage increase in each metric caused by the transformation. In fig. 2 we plot the average of these four percentage differences, allowing us to identify the overall effect of each transformation. We can also use the averaged scores for each subtask within a benchmark (such as the 67 BLiMP subtasks) to assess whether differences are significant for BLiMP, BLiMP Supplement, GLUE and BabySLM (Syntactic) using a paired t-test (see appendix B.1 for details and p-values for each test conducted).

Character Tokenization We find that character tokenization does not significantly decrease performance on BLiMP or BLiMP Supplement compared to subword tokenization. This validates previous work which found that despite the higher computation costs, character-based language models are just as capable of learning language (Al-Rfou et al., 2019; Hahn and Baroni, 2019). We do find a significant decrease for GLUE but this may be due to the fact that many of the finetuning examples for GLUE are very long and our model's context size is only 128 tokens, leading to severe truncation. As character-based tokenizers output more tokens for the same sentence than BPE tokenizers, this means that for many GLUE tasks, necessary information is lost.

Word boundary removal We find that removing word boundaries significantly decreases the BLiMP score, but the decreases for BLiMP Supplement and GLUE are not significant.[7] In their investigation, Nguyen et al. (2022) found a decrease of 7-8% on their own phonemic version of BLiMP when word boundaries were removed, but here we observe only an average decrease of 3.7%. As they only trained 3-layer LSTMs, it is possible that larger models like ours are required to overcome the loss of word boundaries.

Phonemic Transcription Finally, we find that using a phonemic transcription instead of the original written text significantly decreases performance on BLiMP and GLUE, although the percentage decreases are small (3.5% and 1.5% respectively). It also leads to the largest decrease of 11.3% for BLiMP Supplement. We discuss a possible explanation for this particular decrease in section 5.3.

5.2 BabySLM

Unlike the other benchmarks, our best BabySLM score is not achieved by the model trained with the standard orthographic input representation. Instead, the best syntactic score of 94.9 is achieved by the model that uses character-based tokenization (on written text, with word boundaries) and the best lexical score of 89.6 is achieved by the model that uses character-based tokenization for phonemes. It is also worth noting that, to the best of our knowledge, these are the best BabySLM scores to date (see appendix B.3 for a detailed comparison).

Examining the effect of each condition, we find that using a phonemic transcription on average reduces the syntactic score by 4.0%, which is in line with the other benchmarks discussed above. Unlike the other benchmarks, the character tokenization condition **always leads to an improvement** for both BabySLM scores: an average increase of 9.9% for the syntactic score and 23.9% for the lexical score. The sentences used for the syntactic test are all very short compared to the BLiMP sentences (4 words long on average) so a more fine-grained representation may be more useful. For the lexical test, where single words are compared that often only differ by a single phoneme, it seems more appropriate to use a character-based tokenization as the model needs to learn the distributional properties of individual phonemes, which may be lost in subword units.

The removal of word boundaries has a contrasting effect on the two scores. It reduces the syntactic score by 11.9% but increases the lexical score by 1.9%, the only benchmark where removing word boundaries is a positive change. However, the best individual lexical score was achieved by the model that did include word boundaries, suggesting that word boundaries are a helpful signal for a model learning to distinguish words from non-words, possibly because they help separate short sequences of phonemes that appear across word boundaries but not within words.

For the syntactic score, the worst scores are achieved by the models that learn subwords without word boundaries. For these models, the BPE algorithm is essentially acting as an unsupervised word segmentation algorithm learning to split entire sentences into useful units. With a vocabulary size of 16,000, it seems we learn units smaller than words (morpheme-sized units such as "un" in table 1) but also units that cross word boundaries (such as "acon" in table 1). The resulting implicit subword boundaries seem to have particular consequences when evaluating the shorter BabySLM sentences. Using the BPE algorithm in this way could be of interest for word segmentation studies.

5.3 The Effect of Punctuation

Punctuation is a feature of written text that is rarely included in phonemic transcriptions, as it does not typically change the way that words are pro-

[7] Since there are only 5 tasks for BLiMP Supplement it is difficult to get a p-value below 0.05.

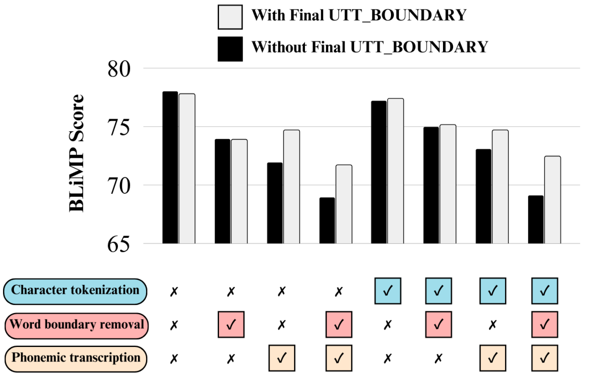

Figure 3: The overall BLiMP scores achieved by GPT-2 in our eight conditions with and without the `UTT_BOUNDARY` token (used to separate sentences) included at the end of evaluation instances.

nounced. However, punctuation in written text does carry important meaning about the structure and tone of sentences. In speech, this information is typically conveyed through intonation, stress and rhythm. By simply stripping punctuation in our phonemic transcriptions, we may be removing information that is important for a model's ability to learn and process language.

In some instances, naïvely stripping punctuation can even lead to nonsense sentences. This may explain the large dip in performance for BLiMP Supplement, as three of the five subtasks rely on punctuation to simulate question-answer pairs or dialogue, such as:

```
A: What did you break?\nB: I broke a bowl.
```

In the example above, the line break, colon and question mark are used to indicate speaker turns and convey the question-answer nature of the prompt. Removing the punctuation leads to a nonsense sentence, especially when read aloud with no pauses or change in tone to indicate the structure:

ʌ ␣ w ʌ t ␣ d ɪ d ␣ j uː ␣ b ɹ eɪ k ␣ b iː ␣ aɪ ␣
b ɹ o ʊ k ␣ ʌ ␣ b oʊ l ␣

Without punctuation, the names "A" and "B" seem out of place. A model trained on written text can use punctuation to possibly understand that these are names, but a spoken model without punctuation would struggle to process this sentence.

This reliance on punctuation seems to be the leading cause of the drop in performance on BLiMP Supplement. If we remove the three subtasks where an understanding of punctuation is required to process the sentence, the effect of switching to a phonemic representation reduces the drop in performance considerably from 11.3% to 0.9%.

There is another subtle yet crucial consequence of removing punctuation: stripping punctuation at the end of sentences, if not handled correctly, can lead to significant decreases in performance on these benchmarks. This is because without an end-of-sentence marker, certain evaluation examples are no longer valid. In order to mark the end of the sentences without puncutation, we needed to ensure that our dedicated sentence-separation token was added to the end of each evaluation instance. The effect of this adjustment is highlighted in fig. 3. The increase in BLiMP score for our phonemic models confirms that this change was necessary and highlights the importance of carefully investigating the role of tokenization in the evaluation of large language models. We discuss this effect further in appendix B.2.

6 Discussion

In this work, we set out to establish whether modern language model architectures can encode grammatical knowledge and succeed at language understanding tasks when trained with phonemic input representations. By identifying three key transformations, carefully ablating them and evaluating our models on a wide variety of benchmarks, we found that these transformations do lead to decreased performance on standard benchmarks, but that this decrease is not substantial, and the effect of each transformation varies according to the evaluation. Generally, we conclude that language models are capable learners and training with these input representations is completely viable.

In this section, we consider explanations for the difference in performance across the benchmarks and discuss the limitations of phonemic transcriptions and our monolingual approach. Our work also has implications for human acquisition investigations and studies that train models directly from raw audio, which we discuss in appendix C.

6.1 The Effect of Input Transformations

There are many possible explanations for the decrease in performance for BLiMP, BLiMP Supplement and GLUE. In section 4.4 and section 5.3 we discuss two possibilities; the fact that character tokenization causes more substantial truncation (affecting GLUE) and the fact that phonemic transcriptions do not include punctuation (which particularly affects BLiMP Supplement). Another factor to consider is that although we do not change the

GPT-2 architecture or training parameters, the vocabulary size does change, which affects the size of the embedding layer. Character tokenization also leads to reduced exposure to each sentence during training (fewer epochs) because each sentence is represented with more tokens, increasing the number of steps required for each epoch. Furthermore, our initial choice of model parameters may have implicitly favored the standard orthographic input representation given that the language modeling community has been collectively optimizing these architectures to learn representations for written text, not phonemic streams. Just as the BabyLM challenge seeks to find solutions for low-resource language modeling, we may require an equivalent challenge to identify new methods and architectures for a phonemic input representation.

We also found a different pattern for the BabySLM benchmark, that certain transformations increased performance. In some cases, the transformations were even necessary (the lexical measure requiring a model to be trained on phonemic input). Given that the BabySLM benchmark more closely relates to child-language acquisition with its shorter sentences and vocabulary taken from child-directed speech, this result will be of interest to studies using language models to study acquisition.

6.2 Limitations and advantages of phonemic transcriptions

One difficulty in training models from ecological long-form child-centered audio is the lack of corpora available. Papers reporting research on daylong recordings tend not to release the raw data due to privacy concerns (e.g. Bergelson et al. (2023); Léon and Cristia (2024)). Our method allows us to convert text (which is much more readily available) into a speech representation (phoneme streams), meaning that we could quickly prepare a corpus of 100 million words.

There are also limitations in our transcription generation process. The fact that phonemes are an abstraction of speech means that we lose key information contained in speech such as prosody, stress and allophonic variation. Using a single accent to generate our phonemes, we also lose inter-speaker variability. Children also learn from non-linguistic cues, multi-modal input and interaction. If anything, it is a striking result that a model trained only on a set of 51 discrete symbols is able to demonstrate grammatical knowledge and perform competitively at downstream linguistic tasks.

6.3 Multi-lingual evaluation

A final important remark is that our experiments are conducted only in English. It is possible that language models trained on phonemic data in other languages would exhibit different trends in downstream performance. Although a multilingual analysis is outside the scope of our paper, we have applied our data processing pipeline to prepare phonemized datasets for 26 of the languages contained in the CHILDES database and hope to release this dataset in the near future.

7 Conclusion

Our study explores the effect of training language models using phonemic input representations, which offer both analytical and practical advantages. We develop a pipeline to convert orthographic datasets into a continuous stream of phonemes and leverage this pipeline to train a language model on phoneme streams and evaluate its grammatical and language understanding abilities. Our findings suggest that while phoneme-based input representations result in a slight decrease in model performance on traditional language understanding tasks, it is nonetheless a feasible training paradigm, facilitating future language modeling work, improving phonological interpretability and enhancing speech-based applications.

Acknowledgements

Our experiments were performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service, provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council. Zébulon Goriely's work is supported by The Cambridge Trust. Richard Diehl Martinez is supported by the Gates Cambridge Trust (grant OPP1144 from the Bill & Melinda Gates Foundation). Andrew Caines and Paula Buttery are supported by Cambridge University Press & Assessment. Lisa Beinborn's work is partially supported by the Dutch National Science Organisation (NWO) through the VENI program (Vl.Veni.211C.039).

References

Rami Al-Rfou, Dokook Choe, Noah Constant, Mandy Guo, and Llion Jones. 2019. Character-level language modeling with deeper self-attention. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3159–3166.

Robin Algayres, Tristan Ricoul, Julien Karadayi, Hugo Laurençon, Salah Zaiem, Abdelrahman Mohamed, Benoît Sagot, and Emmanuel Dupoux. 2022. DP-Parse: Finding Word Boundaries from Raw Speech with an Instance Lexicon. *Transactions of the Association for Computational Linguistics*, 10:1051–1065.

Yamini Bansal, Behrooz Ghorbani, Ankush Garg, Biao Zhang, Colin Cherry, Behnam Neyshabur, and Orhan Firat. 2022. Data scaling laws in NMT: The effect of noise and architecture. In *International Conference on Machine Learning*, pages 1466–1482. PMLR.

Marco Baroni. 2022. On the proper role of linguistically oriented deep net analysis in linguistic theorising. In *Algebraic structures in natural language*, pages 1–16. CRC Press.

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova. 2022. The SIGMORPHON 2022 shared task on morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 103–116, Seattle, Washington. Association for Computational Linguistics.

Lisa Beinborn and Nora Hollenstein. 2024. *Cognitive plausibility in natural language processing*. Springer.

Elika Bergelson, Melanie Soderstrom, Iris-Corinna Schwarz, Caroline F. Rowland, Nairán Ramírez-Esparza, Lisa R. Hamrick, Ellen Marklund, Marina Kalashnikova, Ava Guez, Marisa Casillas, Lucia Benetti, Petra van Alphen, and Alejandrina Cristia. 2023. Everyday language input and production in 1,001 children from six continents. *Proceedings of the National Academy of Sciences*, 120(52):e2300671120.

Mathieu Bernard and Hadrien Titeux. 2021. Phonemizer: Text to phones transcription for multiple languages in python. *Journal of Open Source Software*, 6(68):3958.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430.

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

Daniel Blanchard, Jeffrey Heinz, and Roberta Golinkoff. 2010. Modeling the contribution of phonotactic cues to the problem of word segmentation. *Journal of Child Language*, 37(3):487–511.

Benjamin Börschinger, Mark Johnson, and Katherine Demuth. 2013. A joint model of word segmentation and phonological variation for English word-final /t/-deletion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1508–1516, Sofia, Bulgaria. Association for Computational Linguistics.

Michael R. Brent. 1999. Efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, 34(1):71–105.

Lucas Georges Gabriel Charpentier and David Samuel. 2023. Not all layers are equally as important: Every layer counts BERT. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 238–252, Singapore. Association for Computational Linguistics.

Leshem Choshen, Ryan Cotterell, Michael Y Hu, Tal Linzen, Aaron Mueller, Candace Ross, Alex Warstadt, Ethan Wilcox, Adina Williams, and Chengxu Zhuang. 2024. Call for papers – The BabyLM Challenge: Sample-efficient pretraining on a developmentally plausible corpus. *arXiv preprint arXiv:2404.06214*.

Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2022. Canine: Pre-training an efficient tokenization-free encoder for language representation. *Transactions of the Association for Computational Linguistics*, 10:73–91.

Çağrı Çöltekin. 2017. Using Predictability for Lexical Segmentation. *Cognitive Science*, 41(7):1988–2021.

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. *arXiv preprint arXiv:2402.17645*.

Ewan Dunbar, Nicolas Hamilakis, and Emmanuel Dupoux. 2022. Self-Supervised Language Learning From Raw Audio: Lessons From the Zero Resource Speech Challenge. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1211–1226. Conference Name: IEEE Journal of Selected Topics in Signal Processing.

Emmanuel Dupoux. 2018. Cognitive science in the era of artificial intelligence: A roadmap for reverse-engineering the infant language-learner. *Cognition*, 173:43–59.

S Elizabeth Eden. 2018. *Measuring phonological distance between languages*. Ph.D. thesis, UCL (University College London).

Yanai Elazar, Akshita Bhagia, Ian Helgi Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Evan Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, et al. 2024. What's in my big data? In *The Twelfth International Conference on Learning Representations*.

Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science*, 14(2):179–211.

Linnea Evanson, Yair Lakretz, and Jean Rémi King. 2023. Language acquisition: do children and language models follow similar learning stages? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12205–12218, Toronto, Canada. Association for Computational Linguistics.

Allison Fan and Weiwei Sun. 2023. Constructivist tokenization for English. In *Proceedings of the First International Workshop on Construction Grammars and NLP (CxGs+NLP, GURT/SyntaxFest 2023)*, pages 36–40, Washington, D.C. Association for Computational Linguistics.

Pablo Picasso Feliciano de Faria. 2019. The role of utterance boundaries and word frequencies for part-of-speech learning in Brazilian Portuguese through distributional analysis. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 152–159, Minneapolis, Minnesota. Association for Computational Linguistics.

Siyuan Feng, Ming Tu, Rui Xia, Chuanzeng Huang, and Yuxuan Wang. 2023. Language-universal phonetic representation in multilingual speech pretraining for low-resource speech recognition. In *INTERSPEECH 2023*, Dublin, Ireland. ISCA.

Robert Gale, Alexandra Salem, Gerasimos Fergadiotis, and Steven Bedrick. 2023. Mixed orthographic/phonemic language modeling: Beyond orthographically restricted transformers (BORT). In *Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023)*, pages 212–225, Toronto, Canada. Association for Computational Linguistics.

John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett. 1993. Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1. *NASA STI/Recon technical report n*, 93:27403.

John J Godfrey, Edward C Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *Acoustics, speech, and signal processing, ieee international conference on*, volume 1, pages 517–520. IEEE Computer Society.

Zébulon Goriely, Andrew Caines, and Paula Buttery. 2023. Word segmentation from transcriptions of child-directed speech using lexical and sub-lexical cues. *Journal of Child Language*, pages 1–41.

Michael Hahn and Marco Baroni. 2019. Tabula nearly rasa: Probing the linguistic knowledge of character-level neural language models trained on unsegmented text. *Transactions of the Association for Computational Linguistics*, 7:467–484.

Mark Hasegawa-Johnson, Leanne Rolston, Camille Goudeseune, Gina-Anne Levow, and Katrin Kirchhoff. 2020. Grapheme-to-phoneme transduction for cross-language asr. In *International Conference on Statistical Language and Speech Processing*, pages 3–19. Springer.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Nora Hollenstein, Federico Pirovano, Ce Zhang, Lena Jäger, and Lisa Beinborn. 2021. Multilingual language models predict human reading behavior. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 106–123, Online. Association for Computational Linguistics.

Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. 2020. A systematic assessment of syntactic generalization in neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1725–1744, Online. Association for Computational Linguistics.

Philip A. Huebner, Elior Sulem, Fisher Cynthia, and Dan Roth. 2021. BabyBERTa: Learning more grammar with small-scale child-directed language. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 624–646, Online. Association for Computational Linguistics.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *Preprint*, arXiv:1602.02410.

Jacob Kahn, Morgane Riviere, Weiyi Zheng, Evgeny Kharitonov, Qiantong Xu, Pierre-Emmanuel Mazaré, Julien Karadayi, Vitaliy Liptchinsky, Ronan Collobert, Christian Fuegen, et al. 2020. Libri-light: A benchmark for asr with limited or no supervision. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673.

Nina Kazanina, Jeffrey S Bowers, and William Idsardi. 2018. Phonemes: Lexical access and beyond. *Psychonomic bulletin & review*, 25(2):560–585.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander Rush. 2016. Character-aware neural language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Christo Kirov and Ryan Cotterell. 2018. Recurrent Neural Networks in Linguistic Theory: Revisiting Pinker and Prince (1988) and the Past Tense Debate. *Transactions of the Association for Computational Linguistics*, 6:651–665.

Marvin Lavechin, Maureen De Seyssel, Hadrien Titeux, Hervé Bredin, Guillaume Wisniewski, Alejandrina Cristia, and Emmanuel Dupoux. 2022. Can statistical learning bootstrap early language acquisition? a modeling investigation.

Marvin Lavechin, Yaya Sy, Hadrien Titeux, María Andrea Cruz Blandón, Okko Räsänen, Hervé Bredin, Emmanuel Dupoux, and Alejandrina Cristia. 2023. BabySLM: language-acquisition-friendly benchmark of self-supervised spoken language models. In *INTERSPEECH 2023*, pages 4588–4592, Dublin, Ireland. ISCA.

Colin Leong and Daniel Whitenack. 2022. Phone-ing it in: Towards flexible multi-modal language model training by phonetic representations of data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5306–5315, Dublin, Ireland. Association for Computational Linguistics.

Yinghao Aaron Li, Cong Han, Xilin Jiang, and Nima Mesgarani. 2023. Phoneme-level BERT for enhanced prosody of text-to-speech with grapheme predictions. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Mathilde Léon and Alejandrina Cristia. 2024. *Data Protection Handbook for Long-Form Recording Research: Navigating Data Protection Laws across the Globe*.

Wentao Ma, Yiming Cui, Chenglei Si, Ting Liu, Shijin Wang, and Guoping Hu. 2020. CharBERT: Character-aware pre-trained language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 39–50, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Brian MacWhinney and Catherine Snow. 1985. The Child Language Data Exchange System. *Journal of Child Language*, 12(2):271–295.

Christopher D Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Yevgen Matusevych, Thomas Schatz, Herman Kamper, Naomi H Feldman, and Sharon Goldwater. 2023. Infant phonetic learning as perceptual space learning: A crosslinguistic evaluation of computational models. *Cognitive Science*, 47(7):e13314.

Connor Mayer. 2020. An algorithm for learning phonological classes from distributional similarity. *Phonology*, 37(1):91–131.

Tu Anh Nguyen, Maureen De Seyssel, Robin Algayres, Patricia Roze, Ewan Dunbar, and Emmanuel Dupoux. 2022. Are word boundaries useful for unsupervised language learning? *arXiv preprint arXiv:2210.02956*.

Antoine Nzeyimana and Andre Niyongabo Rubungo. 2022. KinyaBERT: a morphology-aware Kinyarwanda language model. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5347–5363, Dublin, Ireland. Association for Computational Linguistics.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an ASR corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32.

Alan Prince and Paul Smolensky. 1997. Optimality: From neural networks to universal grammar. *Science*, 275(5306):1604–1610.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

David Samuel, Andrey Kutuzov, Lilja Øvrelid, and Erik Velldal. 2023. Trained on 100 million words and still in shape: BERT meets British National Corpus. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1954–1974, Dubrovnik, Croatia. Association for Computational Linguistics.

Thomas Schatz, Naomi H Feldman, Sharon Goldwater, Xuan-Nga Cao, and Emmanuel Dupoux. 2021. Early phonetic learning without phonetic categories: Insights from large-scale simulations on realistic input. *Proceedings of the National Academy of Sciences*, 118(7):e2001844118.

Mark S. Seidenberg and James L. McClelland. 1989. A distributed, developmental model of word recognition and naming. *Psychological Review*, 96:523–568.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Mukuntha Narayanan Sundararaman, Ayush Kumar, and Jithendra Vepa. 2021. PhonemeBERT: Joint Language Modelling of Phoneme Sequence and ASR Transcript. In *Proc. Interspeech 2021*, pages 3236–3240.

Ashima Suvarna, Harshita Khandelwal, and Nanyun Peng. 2024. PhonologyBench: Evaluating phonological skills of large language models. In *Proceedings of the 1st Workshop on Towards Knowledgeable Language Models (KnowLLM 2024)*, pages 1–14, Bangkok, Thailand. Association for Computational Linguistics.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Inar Timiryasov and Jean-Loup Tastet. 2023. Baby llama: knowledge distillation from an ensemble of teachers trained on a small dataset with no performance penalty. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 279–289, Singapore. Association for Computational Linguistics.

Ahmet Üstün, Murathan Kurfalı, and Burcu Can. 2018. Characters or morphemes: How to represent words? In *Proceedings of the Third Workshop on Representation Learning for NLP*, pages 144–153, Melbourne, Australia. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt and Samuel R Bowman. 2022. What artificial neural networks can tell us about human language acquisition. In *Algebraic structures in natural language*, pages 17–60. CRC Press.

Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell. 2023. Findings of the BabyLM challenge: Sample-efficient pretraining on developmentally plausible corpora. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 1–34, Singapore. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Jian Zhu, Changbing Yang, Farhan Samir, and Jahurul Islam. 2024. The taste of IPA: Towards open-vocabulary keyword spotting and forced alignment in any language. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 750–772, Mexico City, Mexico. Association for Computational Linguistics.

A Implementation Details

We implement all experiments using the `PyTorch` framework (Paszke et al., 2019) and the `Transformers` library (Wolf et al., 2020).

A.1 Hardware Details

We use a server with one NVIDIA A100 80GB PCIe GPU, 32 CPUs, and 32 GB of RAM for all experiments. Below, we report a subset of the output of the *lscpu* command:

```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Address sizes:         46 bits physical,
                       48 bits virtual
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Vendor ID:             GenuineIntel
Model name:            Intel(R) Xeon(R)
                       Silver 4210R CPU
                       @ 2.40GHz
CPU family:            6
Model:                 85
Thread(s) per core:    1
Core(s) per socket:    1
Socket(s):             8
Stepping:              7
BogoMIPS:              4800.11
```

A.2 Model Parameters and Training Procedure

Parameter	Value
Layers	12
Heads	12
Dropout	0.1
Embedding Size	768
Inner Size	3072
Max Example Length	128
Learning Rate	0.001
Optimizer	AdamW
Scheduler Type	Linear
Max Steps	400,000
Warm-up Steps	90,000
Per Device Batch Size	32

Table 2: Hyperparameter settings for training the GPT-2 architecture. Vocabulary size varies according to the tokenizer used, but all other parameters are constant across experiments. Where values are not reported, they may be assumed to be default values.

We describe the model and training parameters in table 2. The model parameters were chosen to match those of the Pythia-170M model from the Pythia suite (Biderman et al., 2023). The model has 85M non-embedding parameters and is also equivalent in size to GPT-Neo 125M and OPT-125M. The Pythia models use the GPTNeoX architecture which is slightly different to GPT-2. In initial experiments, we found that GPT-2 performed better on the benchmarks across all eight of our conditions.

Data is prepared into batches by first tokenizing the entire dataset, combining all tokens into one long vector, and then splitting the vector into chunks of 128 tokens. Only the very last example is padded, if required. At each step during training, random chunks are selected and combined into batches.

Checkpoints are taken every 50,000 steps during training. At each checkpoint, the perplexity is evaluated on the held-back evaluation set, and at the end of training the checkpoint with the lowest perplexity is returned as the best model.

B Evaluation Details

B.1 Significance Tests

It is difficult to determine whether the results for a given benchmark are significant given that we only train a single run for each of the eight conditions. Instead, we calculate the significance of a particular transformation by comparing the scores for each subtask of a benchmark. We average the scores achieved by the four models with a transformation applied and average the scores achieved by the four models without the transformation applied, giving us paired results for each subtask. We then use a paired student t-test to assess the significance of the transformation. We give the p-values for our significance tests in table 3.

Note that there are 67 subtasks for BLiMP, 5 for BLiMP Supplement, 9 for GLUE and 9 for BabySLM (Syntactic). With only 5 pairs for BLiMP Supplement, the test is under-powered and low p-values are unlikely. There are no subtasks for BabySLM (Lexical) so significance cannot be computed in the same way.

B.2 The Effect of End-of-Sentence Tokens

By default, our tokenizers add a special start-of-sentence token `UTT_BOUNDARY` to all sentences. This corresponds to the <s> token often used by tokenizers to help transformers with sentence-level processing, and also represents utterance boundaries, which unlike word boundaries are a clear cue present in speech and often included in word segmentation studies (Feliciano de Faria, 2019).

Since sentences are collated together during training, this means that these tokens also appear at

	BLiMP	BLiMP Supplement	GLUE	BabySLM (Syntactic)
orthographic vs. phonemic	**0.0001**	0.0780	**0.0149**	0.1884
word boundaries vs. no word boundaries	**0.0000**	0.1831	0.0813	**0.0118**
character tokenization vs. subword tokenization	0.5069	0.4832	**0.0010**	0.1500

Table 3: p-values from the paired student t-tests for each experiment. Significant results are given in **bold** using an alpha level of 0.05.

the end of every sentence, implicitly acting as end-of-sentence tokens. As a result, the model may use them to represent sentence-level information (especially given that these models are auto-regressive). However, in most evaluation tasks, sentences are presented individually (with padding) and so by default the tokenizer does not add this token to the end of sentences.

This has consequences for zero-shot evaluation tasks where the grammaticality of the sentence depends on the sentence being marked as complete, which is the case for several of the BLiMP subtasks. For instance, one subtask evaluates a model's understanding of filler-gap dependencies by presenting grammatical "wh"-phrases with "that"-phrases that are ungrammatical due to a missing dependency. An example is given in table 4 along with the tokens produced by two of our tokenizers. Crucially, our phonemic transcriptions do not include punctuation (see section 5.3) and for this task, without an end-of-sentence marker, the "ungrammatical" sentence is no longer ungrammatical, as it could just be incomplete.

This means that the subtask remained a valid test for our orthographic models (due to the inclusion of punctuation to mark the end of the sentence), but not the phonemic ones, since for the phonemic models both the "grammatical" and "ungrammatical" sentences could be considered grammatical. Since this task is not balanced, any preference for the word "that" over the "wh"-words would lead to the model consistently choosing the "that" sentences and achieving results below chance (which is 0.5 for all BLiMP tasks).

In our initial experiments we found that the models trained on phonemes achieved scores between 0.06 and 0.14 for this task whereas the orthographic models achieved scores between 0.35 and 0.53. We then added the `UTT_BOUNDARY` token to the end of every evaluation instance and found that the phonemic models could then achieve scores between 0.26 and 0.34 (with little change for the orthographic models). These results also held for several other BLiMP tasks with similar constructions.

We thus decided to ensure that the token was added to the end of every evaluation instance for all benchmarks reported in this paper for two reasons. First, it acts as a necessary end-of-sentence marker to ensure certain tests remain valid for the phonemic models, and second, because the token may encode useful sentence-level information for all models (particularly for GLUE tasks, as only the encoding of the final token is used for predictions).

We present the effect of this decision in fig. 3 which reports the overall BLiMP scores for our eight conditions with and without the inclusion of the `UTT_BOUNDARY` token at the end of each evaluation sentence. There is a very large increase for all four phonemic models with little change for the orthographic models, confirming how crucial this change was to make.

B.3 BabySLM Comparison

In table 1 we report the BabySLM scores achieved by our models and in section 5.2 we mention that these are the highest scores achieved on this benchmark to date. It is worth noting that this is only in comparison to the baseline scores released with the BabySLM benchmark (Lavechin et al., 2023), as at the time of writing no other scores have been published for this benchmark, given how recently it was introduced.

In their study, Lavechin et al. (2023) achieved their highest syntactic score of 70.4 using BabyBERTa (Huebner et al., 2021) trained on only 5 million words from CHILDES (MacWhinney and Snow, 1985). All of our models beat this score, with the highest achieving 94.9. BabyBERTa also uses a BPE tokenizer whereas we found that a character-based tokenizer consistently gave better performance (see section 5.2). There is also an architectural difference, BabyBERTa is an autoencoder trained using masked language modeling, whereas our model is autoregressive, using next-token prediction. The LTG-BERT baseline, which is a similarly sized model also trained on 100 million words, only achieves a score of 75.8. The Baby Llama baseline, by comparison, achieves 94.0. It is possible that the autoregressive architecture is much more suited to the syntactic task than the

	Grammatical	Ungrammatical
Original	Patrick revealed what a lot of men wore.	Patrick revealed that a lot of men wore.
BPE Text Tokenizer	<s> ␣patrick ␣revealed ␣what ␣a ␣lot ␣of ␣men ␣wore ␣.	<s> ␣patrick ␣revealed ␣that ␣a ␣lot ␣of ␣men ␣wore ␣.
BPE Phoneme Tokenizer	<s> ␣pætɹɪk ␣ɹɪviːld ␣wʌt ␣ʌ ␣lɑt ␣ʌv ␣mɛn ␣wɔɹ	<s> ␣pætɹɪk ␣ɹɪviːld ␣θæt ␣ʌ ␣lɑt ␣ʌv ␣mɛn ␣wɔɹ

Table 4: An example sentence pair from the wh_vs_that_with_gap subtask in BLiMP and the outputted tokens from our two tokenizers that use subwords but do not remove word boundaries. The '␣' character denotes word boundaries and the '<s>' token represents our UTT_BOUNDARY token which acts as an utterance boundary and a start-of-sentence token.

autoencoder architecture of BERT.

When it comes to the lexical test, the highest score achieved by Lavechin et al. (2023) was 75.4 using a 3-layer LSTM trained on 1.2 million words from the Providence corpus (Börschinger et al., 2013) which they converted to a stream of phonemes with no word boundaries using a similar tool to ours. Our highest-scoring model was also trained with character-based tokenization of phonemes, but did include word boundaries, achieving a score of 89.6. Our model without word boundaries got the second-highest score with 87.8.

In both cases, our model is larger (12 layers) and trained on much more data (100 million words) than the BabySLM baselines. Also, our pre-training dataset contains a wider variety of sentences than just the child-directed utterances in CHILDES. We are currently investigating the effect of model size and training size on the BabySLM scores. In initial experiments, we found that even a 6-layer model trained on only 7 million words from CHILDES was able to achieve a lexical score of 82, but this model also only achieved a syntactic score of 70. We hypothesize that lexical-level knowledge can be learned with less data and by smaller models when compared to learning syntactic knowledge, but this research is ongoing.

C Further Implications

C.1 Comparing Human Acquisition to Language Model Learning

The capacity of LMs to learn language from text alone has spurred interest in using such models for acquisition and psychology studies, such as comparing model learning trends to child learning behaviour (Evanson et al., 2023) and using model outputs to predict human reading times (Hollenstein et al., 2021).

To push this research further, recent efforts aim to make language modeling more cognitively plausible (Beinborn and Hollenstein, 2024) by reducing the advantages that typical language models have over humans during the learning process (Warstadt and Bowman, 2022). One approach is to limit and curate the dataset to that which a typical human may be exposed to, such as is done in the BabyLM challenge (Warstadt et al., 2023). Another approach is to use an input representation that more closely mimics speech rather than written text (Dupoux, 2018). Finally, we must consider whether the architectures themselves are suitable linguistic theories, given that they were developed for downstream tasks (Baroni, 2022).

In this work we contribute to all three approaches by training a language model with streams of phonemes and assess whether the language model architecture used is advantaged or disadvantaged by these changes according to a wide variety of benchmarks. We hope that this leads to further work studying acquisition using phoneme streams as an input representation. However, while streams of phonemes may seem more cognitively plausible than written text, many studies go further than we do and seek to train directly on raw audio.

C.2 Learning directly from audio

Our study focused on alternative input representations for text-based language models, but there is also a field of work dedicated to training models directly from raw audio. In recent years, the Zero Resource Speech Challenge has helped pioneer the development of models that learn unsupervised from raw audio (Dunbar et al., 2022). Models such as STELA (Schatz et al., 2021; Lavechin et al., 2022) use a two-stage approach, learning a discrete symbolic representation by clustering 10ms chunks of audio, then feeding these to a multi-layered LSTM language model.

These models are also used to study acquisition, regarding raw audio as an input representation that is more cognitively plausible than phonemes; a continuous signal full of noise and non-linguistic information that children must learn to filter. Whether adults even use phonemes as a core linguistic representation, and whether children learn phonemic categories before other stages of acquisition both continue to be a matter of debate (Kazanina et al., 2018; Matusevych et al., 2023) and the symbolic representations learned by models such as STELA have a duration four times shorter than phonemes, challenging the assumption that phonemic categories are precursors to later stages of acquisition.

The gap in linguistic performance between text-based models and audio-based models continues to be substantial. Lavechin et al. (2023) developed BabySLM to compare text-based models to speech-based models and highlighted this gap, but further noted that even speech-based models may not always train on plausible input, many often using audiobooks as their training data (Kahn et al., 2020). When training the STELA model on 1024 hours of ecological long-form child-centered audio compared to 1024 hours of audiobooks, Lavechin et al. (2023) found that the model trained on long-form audio achieved chance-level syntactic and lexical capabilities, highlighting how far we are from producing architectures that can learn from the same signals as human children.