ConcreteGPT: A Baby GPT-2 Based on Lexical Concreteness and Curriculum Learning

Luca Capone[1*†] and **Alessandro Bondielli**[1,2†] and **Alessandro Lenci**[1†]

[1]CoLing Lab, Department of Philology, Literature and Linguistics, University of Pisa
[2]Department of Computer Science, University of Pisa

luca.capone@fileli.unipi.it, {alessandro.bondielli, alessandro.lenci}@unipi.it

Abstract

We present a model for the Strict-Small track of the BabyLM Challenge 2024 (Choshen et al., 2024). We introduce a Curriculum Learning approach for training a specialized version of GPT-2 (Radford et al., 2019), that we name ConcreteGPT. We utilize the norms from Brysbaert et al. (2014), which provide concreteness ratings for 40,000 English lexical items based on human subjects. Using these norms, we assign a concreteness score to each sentence in the training dataset and develop two curriculum strategies that progressively introduce more complex and abstract language patterns in the training data. Compared to the baselines, our best model shows lower performance on zero-shot tasks but demonstrates superior performance in fine-tuning tasks. Notably, our curriculum-trained models exhibit significant improvements over a non-curriculum based training of the same model.

1 Introduction

Optimising language model training to enhance efficiency without compromising performance presents a significant challenge, especially in the era of Large Language Models (LLMs) which require trillions of input tokens and millions of PetaFLOPs for training (Villalobos et al., 2024). A promising approach lies in exploring training strategies that streamline the learning process and maximise resource utilisation. Initiatives like the BabyLM Challenge (Warstadt et al., 2023) aim to find strategies to train effective LLMs under specific data constraints, that naturally reflect also on model sizes constraints following known scaling laws.

One possible area of interest in this context is the use of training strategies related to Curriculum Learning, which refers to the idea of training machine learning models on meaningfully ordered data, for instance from easier to harder samples (Bengio et al., 2009). This approach has yielded beneficial results on many tasks (Soviany et al., 2022), but has not been widely adopted in the context of language modelling. Typically, LLMs are trained on data scraped from the Web, for which it is difficult to obtain a meaningful ordering. In the present work, we evaluate the hypothesis that a curriculum learning strategy informed by evidence from human language acquisition can enhance model performance in data- and/or compute-constrained settings. Specifically, we attempt to understand the impact of considering **word concreteness** for ordering training data. In this context, concreteness refers to how tangible or perceptible the referent of a word is, with more concrete words being those that refer to physical objects or sensory experiences, while abstract words relate to concepts and ideas (Brysbaert et al., 2014). Word concreteness is often considered a proxy for the natural order in which children acquire language, beginning with words that represent familiar objects and situations (Bergelson and Swingley, 2013; Schwanenflugel, 2013). As language development progresses, children gradually learn terms that describe more complex concepts or relationships, which typically rely on the prior acquisition of simpler linguistic elements. Understanding the impact of word concreteness on language model training could potentially lead to models that better grasp and generate language in a more nuanced manner, and more importantly, that learn faster and more efficiently. While some studies have explored different language complexity metrics for Curriculum Learning (Opper et al., 2023; Mi, 2023; Martinez et al., 2023), none of the methods proposed in the 2023 BabyLM Challange employed curriculum

[*]Corresponding author

[†]For the specific purposes of Italian Academy, Luca Capone is responsible for Sections 2 and 5, Alessandro Bondielli is responsible for sections 3 and 4, Alessandro Lenci is responsible for sections 1 and 6

November 15-16, 2024 ©2024 Association for Computational Linguistics

criteria related to lexical concreteness (Warstadt et al., 2023).

In this work, we introduce a Curriculum Learning approach to train a specialised version of GPT2 (Radford et al., 2019), that we call **ConcreteGPT**, which leverages word concreteness ratings. We exploit the concreteness norms from Brysbaert et al. (2014), which include concreteness ratings obtained from human subjects for 40,000 lexical items in English. Using the norms, we compute a concreteness score for each sentence in the training dataset and create a curriculum that progressively emphasises more complex and abstract language patterns. We evaluate our approach on the Strict-Small track for the 2024 BabyLM Challenge. For the track, participants are provided with a dataset of 10M tokens for pre-training their model. Then, the model is evaluated in two ways: first, on a set of tasks in zero-shot settings, using Perplexity (PPL) or Pseudo-Log-Likelihood (PLL) metrics as a proxy of model understanding; second, the model is fine-tuned using standard fine-tuning or LoRA on the GLUE benchmark tasks. We evaluate two different models that employ a slightly different approach to building the curricula for the training, and compare them with a baseline model trained with the same amount of FLOPs without curriculum learning. We show that the curriculum-based models tend to outperform the non-curriculum model, while generally matching or slightly underperforming compared to the strong baselines provided by the task organizers (i.e., the winning models from the 2023 edition), despite a possibly lower computational cost.

The paper is organised as follows. First, we outline the motivations behind our curriculum learning approach in Section 2. Section 3 details the methodology used to create the datasets and describe the curriculum design. Sections 4 and 5 provide an in-depth discussion of the model, covering training specifics and results, and discuss the impact of two variations of the curriculum learning strategy. Finally, Section 6 draws some conclusions and highlights possible future directions.

2 Motivation Behind Curriculum Design

The motivation behind this approach stems from the hypothesis that a curriculum guided by word concreteness can enhance the model's learning trajectory by starting with more concrete, easily grasped examples and gradually advancing to more sophisticated verbal items. This method aims to improve the model's ability to handle a broader range of linguistic phenomena, potentially leading to more robust and contextually aware text generation. Given that the model is not multimodal, one might initially question the value of using a Curriculum Learning approach based on lexical concreteness, since the representations learned by the model are not grounded in perceptual experiences. It is useful to start from the assumption that, in principle, all meanings can be considered as abstract, referring to general classes capable of subsuming heterogeneous and always particular phenomena (Eco, 1979, §2.6). From this perspective, the value of an approach based on lexical concreteness does not lie in grounding meanings in perceptual experience (Søgaard, 2023). Despite the abstract character of meaning, it is widely accepted that the first words children learn tend to have a tight connection to their experience with referents (Schwanenflugel, 2013; Bergelson and Swingley, 2013). Early language acquisition typically involves words related to the child's surroundings, such as parents, pets, and daily routines objects. From these familiar meanings, children gradually expand their vocabulary to include words with more complex meanings that concern more abstract situations, require greater linguistic competence and larger cultural experience. Thus, concreteness rating can be understood as an index of the difficulty in acquiring a word. For instance, learning a term like "dog" requires less linguistic knowledge and semantic structuring than understanding a more abstract concept like "justice". By initially exposing the model to sentences containing words with an high concreteness rating, we attempt to simulate this learning trajectory, providing the model with simpler, more fundamental contexts before progressing to the acquisition of more complex meanings and linguistic situations. The method proposed in this paper is consistent with the findings of Abdou et al. (2021) and Patel and Pavlick (2022) which suggest that LM embeddings encode perceptual structures (e.g, meaningful spatial relations and colors) without requiring perceptual grounding. The hypothesis is that a curriculum based on lexical concreteness can facilitate the acquisition of these meaningful structures.

A potential objection to our method is why age of acquisition is not used directly as a feature for sorting the curriculum. The core principle of the Curriculum Learning approach is to establish a criterion that accurately assesses the difficulty of language items, thereby grouping items of similar difficulty together. In this context, a criterion such as age of acquisition alone does not serve this purpose effectively. Linguistically similar items — those that function similarly in speech and possess comparable levels of difficulty — can exhibit different ages of acquisition. In fact, consulting data from WordBank (Frank et al., 2017), specifically the British Oxford Communicative Development Inventory (CDI), we observe that linguistically similar items are acquired at different ages by children. For example, the proportion of children understanding the word *dog* at 12 months is slightly over 0.6, whereas the corresponding value for the word *lamb* is just under 0.1. This disparity remains relatively constant until 25 months, even though *lamb* does not appear to present any particular challenges compared to *dog*. Consequently, two words of similar difficulty may be sorted differently within the curriculum based solely on age of acquisition. In contrast, this issue is mitigated by the use of concreteness ratings, with *lamb* being rated at 4.97 and *dog* at 4.85 in Brysbaert et al. (2014).

3 Curriculum Design

Brysbaert et al. (2014) collected ratings from 4,237 native speakers for 37,058 English words and 2,896 two-word expressions. The ratings ranged from a minimum of 1, representing «something you cannot experience directly through your senses or actions», to a maximum of 5, indicating «something that exists in reality; you can have immediate experience of it through your senses (smelling, tasting, touching, hearing, seeing) and the actions you do. The easiest way to explain a word is by pointing to it or by demonstrating it» (Brysbaert et al., 2014). Based on these ratings, we assigned a concreteness score to each sentence in the dataset (10M Strict-Small dataset, Choshen et al. 2024). For each sentence, only adjectives, nouns, and verbs were considered in the score calculation. The concreteness ratings of the words in the sentence were summed, and the total was divided by the number of selected words. This resulting value corresponds to the **sentence concreteness score**. Once the sentence scores were obtained, the dataset was divided into four slices, each containing approximately 300,000 items, based on increasing concreteness, as shown in Figure 1.

Based on the dataset slicing, we devise two different curriculum strategies:

SEQUENTIAL – This strategy considers the slicing as-is, and the curriculum is based on their sequential ordering, from the most concrete to the most abstract.

MIXED – This strategy is more nuanced, and accounts for the fact that while sentence-level concreteness can be used as proxy for the natural order in which children acquire language, it is also likely that childrens will be exposed to more complex words as well. Thus, starting from the original slices, we redistribute part of each slice into the other ones. Specifically, each slice contains 50% of the data from the original slice, and 50% from the other three slices, in different proportions, to simulate an increasing percentage of progressively more abstract sentences in each slice. The exact proportions of sentences from each mixed slice are reported in Figure 2.

4 Model and Training

In our experiments, we use the GPT2 implementation from HuggingFace[1] as our base architecture, with its standard pretrained tokenizer. The model has 124M trainable parameters. To further limit the computational cost of training, we restrict the context length of the model to 128 tokens. This change is driven not only by concerns regarding computational resources, but also by theoretical considerations related to the development of working memory in humans, which appears to be limited during the early years of life (Swanson, 1996; Cowan, 2016). A reduced context length (though still larger than the number of tokens a child can process) better aligns with the cognitive plausibility criteria required by the challenge.

We experiment with a hybrid training procedure where the model is sequentially trained on each slice of the dataset for three epochs with the same hyperparameters. Note that we restart the training procedure each time, and that we randomized the batch sampling within the training slice. This means that data from each batch is randomly sampled (as customary for training LMs) from the training slice. Then, the resulting model is further trained on the entire dataset with a lower learning rate for an additional two epochs, again with random batch sampling. We follow this procedure for

[1] https://huggingface.co/openai-community/gpt2

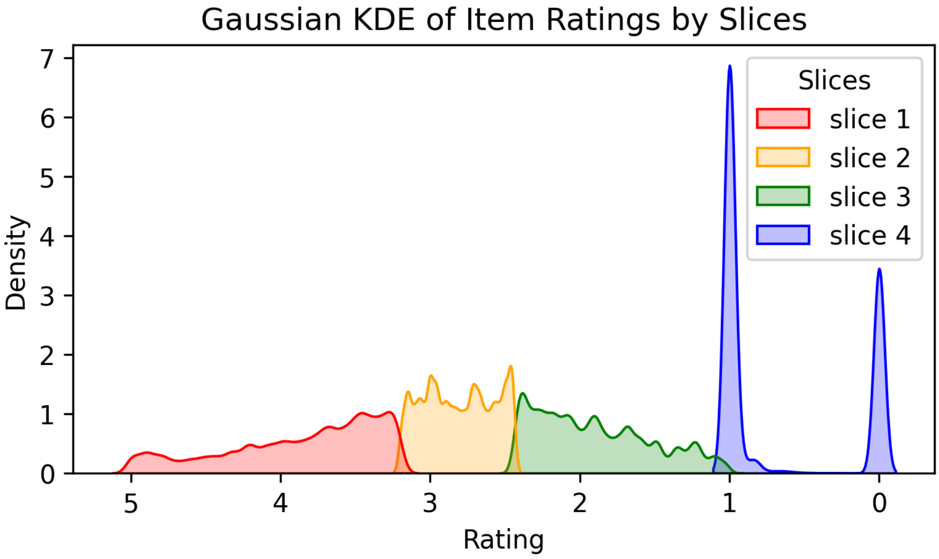

Figure 1: Distribution of sentences into slices. The dataset contains sentences with adjectives, verbs, and nouns that are not among the rated words (assigned a score of zero), as well as sentences that contain none of these word types (assigned a score of one). All such sentences are grouped into the final slice, representing the most abstract and complex sentences.

Model	**Data**	**Epochs**	**Init. LR**	**LR scheduler**	**Batch size**	**Grad. accum.**	**Warmup**
SEQUENTIAL	Slice 1	3	5e-4	Cosine	32	8	1000
	Slice 2	3	5e-4	Cosine	32	8	1000
	Slice 3	3	5e-4	Cosine	32	8	1000
	Slice 4	3	5e-4	Cosine	32	8	1000
	Full Dataset	2	2e-4	Cosine	32	8	1000
MIXED	Slice 1 - mix	3	5e-4	Cosine	32	8	1000
	Slice 2 - mix	3	5e-4	Cosine	32	8	1000
	Slice 3 - mix	3	5e-4	Cosine	32	8	1000
	Slice 4 - mix	3	5e-4	Cosine	32	8	1000
	Full Dataset	2	2e-4	Cosine	32	8	1000
SHUFFLE	Full Dataset	5	5e-4	Cosine	32	8	1000

Table 1: Pre-training parameters for each of the models. In the case of curriculum-based models, parameters are reported for each slice.

Hyperparameter	**Value**
Initial learning rate	3e-4
Batch size	64
Maximum epochs	32
Evaluate every (epochs)	1
LoRA alpha	16
LoRA rank	8
LoRA dropout	0.1

Table 2: Parameters for fine-tuning with LoRA on the GLUE tasks.

both the SEQUENTIAL and MIXED models, only changing the composition of the slices as described in Section 3. For training the comparison model (i.e., the model without curriculum learning), that we call SHUFFLE, we aimed to use the same amount of computing, and thus to show the model each data point the same number of times. Therefore, we trained it for 5 epochs on the entire dataset with random sampling.

Table 1 summarizes the training parameters. All models were trained using half precision (fp16). No direct hyperparameter optimization was performed. However, we experimented with several configurations, specifically varying the initial Learning Rate and its scheduler, and found the chosen configuration to work best. As for batch size and gradient accumulation steps, the values were chosen to best fit the available computational resources. All models were trained using a Nvidia

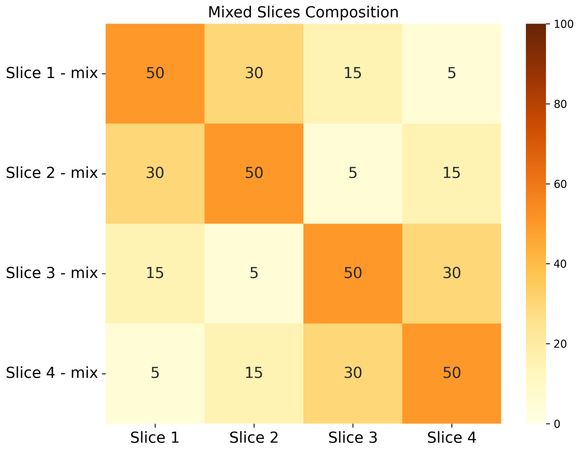

Figure 2: Percentage of sentences from each slice for training the MIXED model.

A100 40GB GPU. Notably, we used the same random seed for all training runs, to ensure that the starting condition was the exact same for all of the trained models. We are aware that averaging the results of multiple training runs would have yielded more reliable results. However, it would have also drastically increased the computational cost of our experiments. The pre-training procedure was handled with the HuggingFace Trainer.[2] For the fine-tuning, we train a LoRA for each of the GLUE tasks using the script provided by the challenge organisers. As for the hyperparameters, we left the default ones provided by the challenge organisers (Choshen et al., 2024). For the sake of completeness, we report the LoRA fine-tuning parameters in Table 2.

5 Results and Discussion

The models are evaluated on two distinct sets of tasks: one requiring fine-tuning and the other performed in a zero-shot setting. Fine-tuning was conducted using the script provided by the organisers (see Section 4). The baselines are two models trained by the organizers, and inspired by the 2023 edition winning systems: LTG-BERT (Charpentier and Samuel, 2023) and BabyLlama (Timiryasov and Tastet, 2023).

For the fine-tuning task, models are fine-tuned on the GLUE benchmark tasks (Wang et al., 2018). Table 3 shows results on all the tasks for each of our trained models, namely SEQUENTIAL, MIXED, and SHUFFLE. While the differences in performance across the models are not substantial, the curriculum-based models (SEQUENTIAL and MIXED) consistently outperform the non-curriculum one (SHUFFLE), with the exception of the CoLA and RTE tasks. Among curriculum-based models, the MIXED model outperform the SEQUENTIAL model on 7 out of 10 tasks, and for 2 out of 10 taks they achieve the same level of performances.

For the zero-shot tasks, results are reported in Table 4, and are less clear-cut. For the Ewok task (Ivanova et al., 2024), the MIXED model perform slightly better than SHUFFLE and SEQUENTIAL models, and achieve a score on par with the best baseline. For the Blimp tasks (Warstadt et al., 2020) the scenario is different: in Blimp Filtered, none of our models manage to match the BabyLlama baseline, although the two models trained with curriculum learning come very close. Nevertheless, they significantly outperform LTG-BERT. For Blimp Supplement, none of the models reach the baseline, and the best-performing model is the non-curriculum one (SHUFFLE). Nevertheless, in two out of three zero-shot tasks curriculum-based models outperform, albeit slightly, the non-curriculum based one. In Blimp Filtered, both models perform the same, while for Ewok the best performing model is again the MIXED model. Table 4 also report the average on the fine-tuning GLUE tasks. The MIXED model significantly outperform both the other proposed models as well as the baselines.

On average, across all tasks, the three models consistently outperform the worse baseline (LTG-BERT), but do not exceed the performance of the best baseline (BabyLlama) except for GLUE. Of the three, the MIXED model performs the closest to BabyLlama overall. However, this result is largely influenced by the poorer performance in Blimp, especially in Blimp Supplement. In this case the curriculum learning strategy appears to negatively affect performance on acceptability tasks, as indicated by the weaker results observed in CoLA and Blimp. A potential explanation is that curriculum learning based on lexical concreteness enhances performance in tasks with a stronger semantic component, such as MRPC, SST-2, and MNLI, where the curriculum-trained models demonstrate superior performances.

However, these findings appear to corroborate the idea that, given the same (and limited) amount of data and training compute, employing a cognitively plausible training strategy that leverages lexical concreteness as a proxy for a plausible or-

[2] https://huggingface.co/docs/transformers/v4.44.2/en/main_classes/trainer.

model	mrpc	boolq	qqp	sst2	qnli	wsc	cola[matt_corr]	rte	mnli	multirc
Sequential	0.80	0.64	**0.77**	0.80	0.74	0.58	0.59 [0.02]	**0.57**	0.67	**0.65**
Mixed	**0.82**	**0.66**	**0.77**	**0.85**	**0.78**	**0.65**	0.61 [0.04]	0.56	**0.68**	**0.65**
Shuffle	0.79	**0.66**	0.74	0.79	0.76	0.62	0.59 [**0.08**]	**0.57**	0.64	0.64

Table 3: Fine-tuning results. As specified in the evaluation pipeline documentation (github.com/babylm/evaluation-pipeline-2024), we use accuracy as the evaluation metric for all tasks except QQP and MRPC, for which we report F1 scores, and CoLA, for which we use the Matthews correlation coefficient (we also report the evaluation loss for this task).

Model	Zero Shot			Fine tuning	Macro Avg.
	blimp_supp	blimp_filt	ewok	Glue Avg.	
Sequential	55.9	**68.6**	50.2	62.4	59.3
Mixed	55.9	**68.6**	**50.7**	**64.6**	**60.0**
Shuffle	**57.1**	67.8	50.5	62.9	59.6
BabyLlama	59.5	**69.8**	**50.7**	63.3	**60.8**
LTG-BERT	**60.8**	60.6	48.9	60.3	57.7

Table 4: Overall results and comparison with baselines.

dering to acquire language is probably beneficial. In addition to this, it is also relevant to point out that our model, albeit larger in terms of number of parameters, was not trained until convergence and in any case was trained with less computing than the strongest baseline represented by the BabyLlaMA model, but it still either reach its performances or surpasses them in 2 out of the 4 evaluations.

6 Conclusion and Future Work

In this paper we propose two models for the Strict-Small track of the BabyLM Challenge 2024 (Choshen et al., 2024). The models were trained using a Curriculum Learning strategy designed to optimise performance. The dataset provided by the organisers was divided into four slices based on increasing levels of lexical concreteness. From this division, two models were trained: the Sequential was trained on the slices in order of decreasing concreteness, while the Mixed incorporated a progressively higher percentage of abstract and complex sentences at each epoch. For comparison, the same architecture was trained using a standard training procedure on the entire dataset with the same amount of compute (Shuffle model).

The Shuffle model outperforms the curriculum-trained models only in the Blimp Supplement (for zero-shot) and in CoLA (for fine-tuned) tasks. In all other tasks however curriculum learning based on lexical concreteness, particularly the Mixed model, demonstrates improved performance. Compared to the baselines provided by the organisers, the Mixed model exhibits comparable or lower performance on zero-shot tasks but performs well in fine-tuning tasks. These results are notable, especially given the relatively small amount of training compute provided to the model.

Our findings suggest that in low resources and/or low compute scenarios, cognitively plausible training strategies, specifically using concreteness, may help the model learn effective representation faster than with traditional training methods. Nevertheless, we must point out that the proposed approach does not systematically outperform the strong baselines provided by the challenge organisers, especially in zero-shot tasks. Possible explanations are that i.) our concreteness-based approach still requires some refinement, and that ii.) our models may be undertrained with respect to the baselines.

Based on these findings, we propose several directions for future work. First, training the model on a larger dataset and for more epochs would allow us to test whether the performance gap scales with additional data, potentially by further refining the progression of the slices in the Mixed strategy. Second, applying this curriculum learning approach to a multimodal model would help assess whether it also facilitates mapping between language and images. Finally, it would be valuable to further investigate the differences in performance on acceptability tasks (which are more syntactic in nature) versus tasks focused on semantics and inference, to better understand the robustness of this trend.

Acknowledgments

We acknowledge financial support under the PRIN 2022 Project Title "Computational and linguistic benchmarks for the study of verb argument structure" – CUP I53D23004050006 - Grant Assignment Decree No. 1016 adopted on 07/07/2023 by the Italian Ministry of University and Research (MUR). This work was also supported under the PNRR—M4C2—Investimento 1.3, Partenariato Esteso PE00000013—"FAIR—Future Artificial Intelligence Research"—Spoke 1 "Human-centered AI," funded by the European Commission under the NextGeneration EU programme, and partially supported by the Italian Ministry of University and Research (MUR) in the framework of the PON 2014-2021 "Research and Innovation" resources – Innovation Action - DM MUR 1062/2021 - Title of the Research: "Modelli semantici multimodali per l'industria 4.0 e le digital humanities."

References

Mostafa Abdou, Artur Kulmizev, Daniel Hershcovich, Stella Frank, Ellie Pavlick, and Anders Søgaard. 2021. Can language models encode perceptual structure without grounding? a case study in color. *arXiv preprint arXiv:2109.06129*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning (icml). *Google Scholar Google Scholar Digital Library Digital Library*.

Elika Bergelson and Daniel Swingley. 2013. The acquisition of abstract words by young infants. *Cognition*, 127(3):391–397.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46:904–911.

Lucas Georges Gabriel Charpentier and David Samuel. 2023. Not all layers are equally as important: Every layer counts bert. *arXiv preprint arXiv:2311.02265*.

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 2nd babylm challenge: Sample-efficient pretraining on a developmentally plausible corpus. *arXiv preprint arXiv:2404.06214*.

Nelson Cowan. 2016. Working memory maturation: Can we get at the essence of cognitive growth? *Perspectives on Psychological Science*, 11(2):239–264.

Umberto Eco. 1979. *A theory of semiotics*, volume 217. Indiana University Press.

Michael C Frank, Mika Braginsky, Daniel Yurovsky, and Virginia A Marchman. 2017. Wordbank: An open repository for developmental vocabulary data. *Journal of child language*, 44(3):677–694.

Anna A Ivanova, Aalok Sathe, Benjamin Lipkin, Unnathi Kumar, Setayesh Radkani, Thomas H Clark, Carina Kauf, Jennifer Hu, RT Pramod, Gabriel Grand, et al. 2024. Elements of world knowledge (ewok): A cognition-inspired framework for evaluating basic world knowledge in language models. *arXiv preprint arXiv:2405.09605*.

Richard Diehl Martinez, Zebulon Goriely, Hope McGovern, Christopher Davis, Andrew Caines, Paula Buttery, and Lisa Beinborn. 2023. Climb: Curriculum learning for infant-inspired model building. *arXiv preprint arXiv:2311.08886*.

Maggie Mi. 2023. Mmi01 at the babylm challenge: Linguistically motivated curriculum learning for pretraining in low-resource settings. *Proceedings of the BabyLM Challenge. Association for Computational Linguistics (ACL)*.

Mattia Opper, J Morrison, and N Siddharth. 2023. On the effect of curriculum learning with developmental data for grammar acquisition. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 346–355.

Roma Patel and Ellie Pavlick. 2022. Mapping language models to grounded conceptual spaces. In *International conference on learning representations*.

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.

Paula J Schwanenflugel. 2013. Why are abstract concepts hard to understand? In *The psychology of word meanings*, pages 223–250. Psychology Press.

Anders Søgaard. 2023. Grounding the vector space of an octopus: Word meaning from raw text. *Minds and Machines*, 33(1):33–54.

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. 2022. Curriculum learning: A survey. *Int. J. Comput. Vision*, 130(6):1526–1565.

H Lee Swanson. 1996. Individual and age-related differences in children's working memory. *Memory & Cognition*, 24(1):70–82.

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. *arXiv preprint arXiv:2308.02019*.

Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. 2024. Position: Will we run out of data? limits of llm scaling based on human-generated data. In *Forty-first International Conference on Machine Learning*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, et al. 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*.

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.