Characterizing Learning Curves During Language Model Pre-Training: Learning, Forgetting, and Stability

Tyler A. Chang[1,2], **Zhuowen Tu**[1], **Benjamin K. Bergen**[1]

[1]Department of Cognitive Science
[2]Halıcıoğlu Data Science Institute
University of California San Diego, USA
{tachang, ztu, bkbergen}@ucsd.edu

Abstract

How do language models learn to make predictions during pre-training? To study this, we extract learning curves from five autoregressive English language model pre-training runs, for 1M unseen tokens in context. We observe that the language models generate short repetitive phrases before learning to generate longer and more coherent text. We also find that individual tokens often exhibit sudden increases or decreases in loss that are surprisingly consistent across pre-training runs. To better understand these fluctuations, we quantify the final surprisal, within-run variability, age of acquisition, forgettability, and cross-run variability of learning curves for individual tokens in context. More frequent tokens reach lower final surprisals, exhibit less variability within and across pre-training runs, are learned earlier, and are less likely to be ‘‘forgotten’’ during pre-training. Higher n-gram probabilities further accentuate these effects. Independent of the target token, shorter and more frequent contexts correlate with marginally more stable and quickly acquired predictions. Based on our results, we argue for the existence of sequential learning dependencies between different model capabilities, and we characterize language model learning as early n-gram learning before gradual refinement of tail n-gram predictions.

1 Introduction

Language models have received unprecedented attention in recent years due to impressive performance on natural language tasks (e.g., OpenAI, 2022; Google, 2023; Anthropic, 2023). However, these models are initialized as random word (token) generators, and it remains unclear how the models achieve complex linguistic abilities during pre-training. Previous work has investigated when syntactic, semantic, and reasoning abilities emerge (Liu et al., 2021; Evanson et al., 2023), quantified ages of acquisition for tokens averaged over contexts (Chang and Bergen, 2022b), and extracted learning curves for individual examples (Xia et al., 2023). However, features that influence individual learning curves have yet to be identified (e.g., n-gram probabilities and context lengths). Given any token in context, it is largely unknown when or how stably that token would be learned.

From a scientific perspective, understanding when examples are learned by language models can provide insights into possible mechanisms for language acquisition. Regardless of their similarity to human language processing, language models are exemplars of how learning from language statistics alone (i.e., ‘‘distributional’’ learning) can lead to complex linguistic abilities (Chang and Bergen, 2022b; Warstadt and Bowman, 2023; Mahowald et al., 2023). Notably, despite smoothly decreasing corpus-level loss and independent and identically distributed (i.i.d.) data throughout pre-training, individual text examples exhibit learning curves with sudden decreases and increases in loss (§5 and Xia et al., 2023). This highlights the importance of examining individual example learning curves for pre-training dynamics research; aggregate curves often do not capture the fluctuations exhibited by individual examples. Our work seeks to characterize these fine-grained convergence patterns in terms of simpler distributional statistics.

From a practical perspective, understanding language model learning curves can inform the pre-training and deployment of language models. Learning curve results might allow NLP practitioners to determine how much pre-training is necessary for different capabilities and what behaviors will remain stable after additional

Transactions of the Association for Computational Linguistics, vol. 12, pp. 1346–1362, 2024. https://doi.org/10.1162/tacl_a_00708
Action Editor: Dr. Xavier Carreras. Submission batch: 1/2024; Revision batch: 6/2024; Published 11/2024.

pre-training (e.g., ''continual learning'' on more recent data; Jin et al., 2022). Learning curve results can also help identify scenarios in which to expect high levels of variability among fully-trained models, or even develop better pre-training curricula. For example, better curricula might maximize the presence of tractable features that a language model can learn at different pre-training steps.

Thus, our work seeks to quantify convergence patterns for individual tokens in context during language model pre-training. We focus on learning curve convergence, including learning speed, forgetting, and stability. Rather than evaluate model performance on downstream tasks throughout pre-training, we study individual tokens in context (c.f. Liu et al., 2021; Xia et al., 2023). Specifically, we run five English language model pre-training runs, and we extract learning curves for 1M unseen tokens in context. We quantify the final surprisal, variability within and across pre-training runs, age of acquisition, and forgettability of each example. We report general learning curve patterns, and we assess the impact of token frequencies, n-gram probabilities, context lengths and likelihoods, and part-of-speech tags on the speed and stability of language model learning. Based on our results, we argue that there exist sequential dependencies between when language models acquire different capabilities (§7). We then characterize language model learning as early n-gram learning, before gradual refinement of low probability n-gram predictions based on longer context and more nuanced linguistic capabilities. Finally, we discuss implications of our work for informed language model deployment.

2 Related Work

Previous work has studied the pre-training dynamics of language models (Saphra and Lopez, 2019). Choshen et al. (2022) and Evanson et al. (2023) find that language models learn linguistic generalizations in similar stages regardless of model architecture, initialization, and data-shuffling. In masked language models, syntactic rules are learned early, but world knowledge and reasoning are learned later and less stably (Chiang et al., 2020; Liu et al., 2021). Olsson et al. (2022) find that copy mechanisms (''induction heads'' for in-context learning) appear at an inflection point during pre-training. These results establish when a variety of abilities emerge in language models. Our work studies more fine-grained learning trajectories by evaluating individual tokens in context.

Indeed, previous work has studied how individual tokens are learned during pre-training. For example, word learning is highly dependent on word frequency (Chang and Bergen, 2022b). Larger models memorize more examples during pre-training without overfitting (Tirumala et al., 2022), but the time step that a model sees an example does not affect memorization (Biderman et al., 2023). Most similar to our work, Xia et al. (2023) collect learning curves for individual tokens in context, finding that some examples exhibit a ''double-descent'' trend where they first increase then decrease in surprisal. All of the studies above collect language model learning curves during pre-training, either for individual examples or targeted benchmark performance. Here, we introduce metrics to characterize such curves, we identify general learning patterns, and we isolate text features that are predictive of learning speed and stability.

3 Language Model Learning Curves

We extract learning curves for 1M unseen tokens in context from five English language model pre-training runs. Similar learning curves are computed in Xia et al. (2023); we extend their work by defining metrics to characterize such learning curves (§5), and we identify text features that predict each metric (§6). In this way, we aim to demonstrate the connection between simple distributional statistics (e.g., n-gram probabilities) and language model learning.[1]

3.1 Models and Dataset

We run five autoregressive Transformer language model pre-training runs from scratch, following the GPT-2 architecture with 124M parameters (Radford et al., 2019). We run five pre-training runs in order to quantify variability in learning curves across runs (§5.2). For all runs, we use the same SentencePiece tokenizer trained on 10M lines of our pre-training dataset with vocabulary size 50K.

Dataset and Training. We retrieve the first 128M lines of the deduplicated OSCAR English

[1] Code is available at https://github.com/tylerachang/lm-learning-curves.

corpus (Abadji et al., 2021). We tokenize the corpus, concatenating lines until each sequence has length 128. We sample 80% of the resulting dataset as our pre-training dataset (5.1B tokens), leaving the remainder for evaluation and testing. Models are trained for 1M steps with batch size 256 (Devlin et al., 2019; Chang and Bergen, 2022b). Each model is initialized with a different random seed and uses a different shuffle of the pre-training dataset. Pre-training details and hyperparameters are in §A.1.

Checkpoints. Previous work studying language models during pre-training has saved model checkpoints at inconsistent intervals (e.g., every 100 steps or every power of two up to step 1000, then every 1000 steps up to step 100K, etc.; Blevins et al., 2022; Chang and Bergen, 2022b; Sellam et al., 2022; Biderman et al., 2023; Xia et al., 2023). To obtain smoother changes between checkpoints, we save checkpoints such that the number of steps between checkpoints increases linearly as a function of the current step t. As a result, (1) we can define the checkpoint frequencies at the start and end of pre-training, (2) the checkpoint step is an exponential function of the checkpoint number, and (3) the number of steps per checkpoint is an exponential function of the checkpoint number. Checkpoint strategy details are in §A.2. We begin pre-training with 100 steps per checkpoint, and we end pre-training with 25K steps per checkpoint (ending at step 1M). Including a checkpoint at step zero, this results in 222 checkpoints per pre-training run. Sample outputs from different checkpoints are included in §4.

3.2 Surprisal Curves

For quantitative analyses of language model learning curves, we sample 100K sequences from the evaluation dataset in §3.1. We sample ten tokens per sequence, and we compute the surprisal $-\log_2(P(w))$ for each token w based on its preceding context (Levy, 2008), using each language model checkpoint. Surprisal is an established information-theoretic metric used to measure the ‘‘surprise’’ of a next token given a language model (Levy, 2008; Goodkind and Bicknell 2018; Futrell et al., 2019; Li et al., 2021; Chang and Bergen, 2022b; Oh and Schuler, 2023; Michaelov et al., 2024). We then have a learning curve for each token in context (i.e., each example) and each model, usually trending from higher surprisal (worse predictions) to lower surprisal (better predictions; Figure 1). Surprisal is equivalent to the language modeling loss function in log base two. In total, we collect surprisal curves for 1M examples per model.

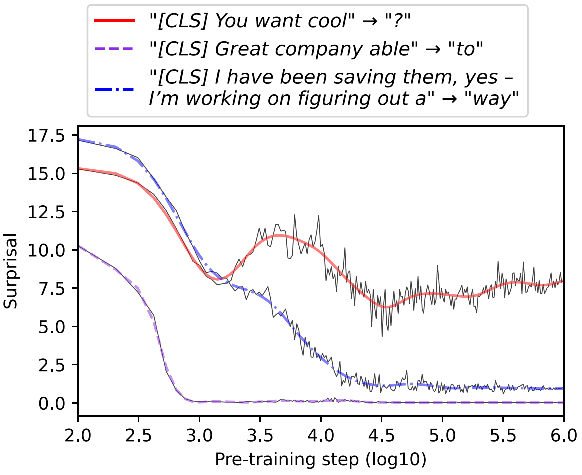


Figure 1: Learning curves for three evaluation examples from the OSCAR dataset during one pre-training run. Colored lines are fitted GAM curves.

4 Overall Learning Patterns

Before considering fine-grained learning patterns for individual surprisal curves, we observe several overall trends during language model pre-training. Many of these trends echo results from previous work (e.g., n-gram learning in Chang and Bergen, 2022b; Choshen et al., 2022) or intuitive results known by language model pre-training practitioners (e.g., the slow development of the ability to generate long coherent text), but these trends establish basic intuitions about how language models progress throughout pre-training.

Early in Pre-training, Models Generate Short Repetitive Phrases. Sample outputs from different model checkpoints are shown in Table 1. We manually inspect outputs from all five pre-training runs, generating text completions to 100 randomly sampled subsequences from the evaluation dataset in §3.1, using sampling temperature 0.3 (Holtzman et al., 2020). As expected, models initialize with random token predictions at step zero. By 100 steps, they repeatedly produce frequent tokens; at this stage, 99.8% of output tokens are ‘‘*the*’’, a comma, or a period. The remaining tokens are frequent words such as ‘‘*to*’’, ‘‘*of*’’, and ‘‘*and*’’. By 1000 steps, the models repeatedly produce frequent short phrases such as ‘‘*of the first*’’ or

Step	Training tokens	Model output
0	0	“This is469 gush liqueur Defense trophies Jakarta Sale Berlin deservingException validate jalapeno...”
100	3.3M	“This is,,,,,,,,,,,,,,,,,,,,,,,,,,,,,, the the the the,,,,,,,......”
1K	33M	“This is a few of the first of the same of the world’s the most of the first of the the same of the first of the world.”
10K	330M	“This is a great way to make a difference in your life.”
100K	3.3B	“This is a very important part of the process of getting your business off the ground.”
1M	33B	“This is a great opportunity to own a beautiful home in the desirable area of North Vancouver.”

Table 1: Sample model outputs completing the prompt “This is...” at different pre-training checkpoint steps, using sampling temperature 0.3. We also report the total number of tokens observed up to a given step; one epoch of our pre-training dataset is 5.1B tokens.

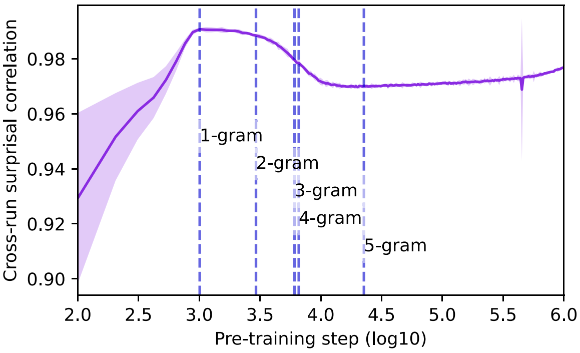


Figure 2: Mean pairwise correlation between model surprisals for different pre-training runs, at different pre-training steps.[2] Shaded regions indicate five standard deviations from the mean. Vertical lines indicate the pre-training steps where model surprisals are maximally correlated with n-gram surprisals.

‘‘*and the most*’’; 86.5% of completions contain the phrase ‘‘*of the first*’’, and 71.1% of completions include it at least twice. These observations align with previous work finding that language models overfit to unigram then bigram next-token predictions early in pre-training (Chang and Bergen, 2022b; see also Figure 2); here, we demonstrate these findings in longer sequences of generated text.

Models Later Generate Longer and More Coherent Text. By step 10K, the models generally produce coherent sentence completions, but they still contain repetitive phrases (10.8% of completions with a three-word phrase repeated at least three times). By step 100K, the repetition rate drops to 6.0%, and completions appear more specific to the context. By step 1M, the repetition rate is 4.7%, and the models can produce coherent multi-sentence completions. Still, due to our relatively small model size (124M parameters, the size of the original GPT model; Radford et al., 2018), we do not expect our models to exhibit text generation capabilities at the level of larger language models.

Models Roughly Follow *n*-gram Learning. We compute the correlation between n-gram surprisals and model surprisals throughout pre-training.[3] Consistent with previous work (Chang and Bergen, 2022b; Karpathy et al., 2016 for LSTMs), the models overfit to unigram (token frequency) predictions then bigram predictions early in pre-training. Extending this up to 5-grams, the models reach maximal similarity to a unigram model around step 1K, before peaking in similarity to 2, 3, 4, and 5-grams, in that order (Figure 2). This is consistent with the hypothesis that language models at some specified level of performance make similar generalizations regardless of architecture (Choshen et al., 2022; Xia et al., 2023). Figure 2 demonstrates that as the models pre-train, their individual predictions pass through stages where they loosely match different n-gram models.

Models Are Maximally Similar Early and Late in Pre-training. We also compute the correlation between model surprisals across pre-training runs at different checkpoints (Figure 2). At any

[2] At approximately $10^{5.7}$ steps, one model exhibited a small temporary increase in loss, leading to a dip in the cross-run surprisal correlation.

[3] We compute n-gram probabilities directly from the pre-training dataset, as in §6.1. For unobserved n-grams, we use backoff to $(n-1)$-grams (Katz, 1987).

given checkpoint, the similarity between any two pre-training runs is both high (Pearson's $r > 0.95$ after step 1K) and consistent (extremely low standard deviations; Figure 2). The models are maximally similar almost exactly when they mirror the unigram distribution (i.e., predicting based on token frequency). The models then decrease slowly in cross-run similarity, reaching a local minimum as they approach the 5-gram distribution. This suggests that there is at least some variability in the generalizations that language models make beyond bigrams. Still, as demonstrated by the steady increase in similarity throughout the remainder of pre-training, language models eventually converge to similar solutions as their performance improves.

5 Characterizing Learning Curves

We then consider fine-grained analyses of learning curves for individual tokens in context. We introduce five metrics to characterize language model learning curves, each motivated by previous work.

5.1 Within-Run Metrics

First, we compute four metrics for each learning curve within a pre-training run (§3.2): final surprisal, variability across pre-training steps, age of acquisition, and forgettability.

Final Surprisal. Surprisal quantifies the quality of a language model's predictions for a token in context, with lower values corresponding to better predictions (Levy, 2008; §3.2). For each example, we compute the mean surprisal during the last 25% of pre-training. This is closely (and inversely) related to model confidence, which Swayamdipta et al. (2020) define as the mean probability assigned to the correct label for an example during language model fine-tuning. We use surprisals (i.e., negative log probabilities) instead of raw probabilities because the language modeling task has a much larger number of output labels (50K possible next tokens) than traditional classification tasks, leading to much lower output probabilities. Surprisal enables distinctions among lower probabilities, and it is commonly used for language modeling (§3.2).

Variability (steps). We then measure how much model performance for an example changes across steps within a pre-training run. Specifically, we consider variability late in pre-training, when a language model has largely converged. Longer term fluctuations in performance are captured by forgettability, defined later in this section. Motivated by Swayamdipta et al. (2020), who compute the standard deviation of model probabilities during fine-tuning, we compute the standard deviation of surprisal during the last 25% of pre-training.

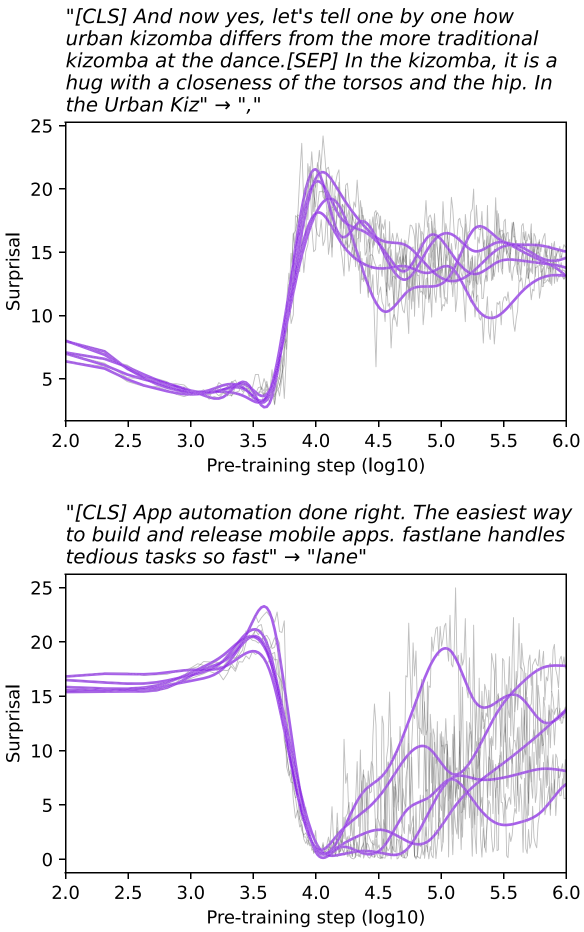
Age of Acquisition (AoA). We also measure when each example is learned during pre-training. Chang and Bergen (2022b) define a token's age of acquisition (AoA) in a language model as the log-pre-training step when the model's surprisal reaches 50% between random chance surprisal and the minimum surprisal attained by the model. Chang and Bergen (2022b) fit a sigmoid curve to the mean surprisal curve over all occurrences of the token. Because surprisal curves for individual examples are less stable than mean curves (e.g., sometimes exhibiting both peaks and dips in surprisal; Figures 1 and 3), we instead fit a GAM curve to each surprisal curve (surprisal $\sim$ log-pre-training step).[4] We define an example's age of acquisition as the log-pre-training step where the fitted GAM first passes 50% between random chance surprisal and the GAM's minimum surprisal.

Forgettability. Along with short-term surprisal spikes as quantified by variability (across steps), language models exhibit long-term increases in surprisal for some examples during pre-training (Xia et al., 2023). This process is described as ''forgetting''. To quantify long-term surprisal increases, we measure the total surprisal increase along the GAM curve fitted to each surprisal curve. Equivalently, this is the total surprisal difference between each relative maximum and its preceding relative minimum in the curve. Larger values indicate that an example is ''forgotten'' to a larger extent at some point during pre-training. Example curves with high forgettability scores are shown in Figure 3.

5.2 Across-Run Metrics

Individual Learning Curves Are Similar Across Pre-training Runs. Each of the metrics in §5.1 correlates across pre-training runs ($r = 0.652$ to 0.978; diagonal entries in

[4] We fit linear GAMs with 25 splines. These are smoothed piecewise functions with 25 linear segments (Wood, 2017; Servén and Brummitt, 2018).


"[CLS] And now yes, let's tell one by one how urban kizomba differs from the more traditional kizomba at the dance.[SEP] In the kizomba, it is a hug with a closeness of the torsos and the hip. In the Urban Kiz" → ","

"[CLS] App automation done right. The easiest way to build and release mobile apps. fastlane handles tedious tasks so fast" → "lane"

Figure 3: Learning curves for two evaluation examples from the OSCAR dataset with high forgettability scores, for the five pre-training runs. Purple lines are fitted GAM curves, one per pre-training run.

Table 2). Curves for a given example even exhibit similar peaks and dips across pre-training runs (Figure 3). Concretely, we quantify the distance between learning curves for two pre-training runs using the Euclidean distance between their fitted GAM curves. Given an example curve in one pre-training run, the curve for the same example in another pre-training run is on average (median) closer than the curve for 99.93% of other examples.[5]

Variability (runs). However, learning curves are not identical across runs. To quantify the cross-run variability of learning curves for a given example, we compute the mean pairwise distance (squared Euclidean distance) between the fitted GAM curves for different pre-training runs. This metric is correlated when computed using different three-run subsets of the five pre-training runs ($r = 0.798$; Table 2). Our final cross-run variability metric is computed over all five pre-training runs.

	Surprisal	Var. (steps)	AoA	Forgettability	Var. (runs)
Surprisal	**0.98**	0.46	0.31	0.62	0.45
Variability (steps)		**0.65**	0.38	0.43	0.57
AoA			**0.84**	0.14	0.43
Forgettability				**0.79**	0.51
Variability (runs)					**0.80**

Table 2: Pearson correlations between learning curve metrics. Diagonal entries indicate the mean correlation for that metric across pre-training runs. For variability across runs, the diagonal entry is the mean correlation between cross-run variability scores computed from different three-run subsets of the five pre-training runs.

5.3 Correlations Between Metrics

Surprisal Correlates with All Learning Curve Metrics. Correlations between metrics are reported in Table 2. All five metrics are positively correlated with one another. High-surprisal examples exhibit more variability across pre-training steps, are learned later, are more likely to be forgotten during pre-training, and exhibit more cross-run variability. Some of these correlations are unsurprising based on our metric definitions; for example, forgettability is quantified using surprisal curve increases during pre-training, which likely lead to higher final surprisals. However, the correlation between final surprisal and forgettability is far from perfect ($r = 0.622$), suggesting that some examples can be forgotten and then re-learned (high forgettability, low surprisal) or simply never learned (low forgettability, high surprisal). Indeed, upon manual inspection, we observe both of these types of curves. Of the 269 examples in both the top 5% of forgettability and bottom 5% of surprisal, 92% exhibit a sudden (greater than 2.5) surprisal increase in the fitted GAM curve that is later recovered. Of the 32 examples in both the bottom 5% of forgettability and top 5% of surprisal, 78% never deviate from their starting surprisal by more than 20%.

[5] We obtain similar results using distances between raw surprisal curves. Raw surprisal curve distances are highly correlated with fitted GAM curve distances ($r = 0.964$).

6 Predicting Learning Curve Metrics

In the previous section, we defined five metrics to characterize language model learning curves. Next, we predict each metric from specific features of each example, including n-gram probabilities, context likelihoods, and part-of-speech tags. We use a linear regression to quantify effects over all 1M examples, providing evidence that simple text features can predict language model learning patterns.

6.1 Predictors and Regressions

Each text example consists of an input context and a target token (§3.2). We consider six predictors (text features) that may be predictive of learning curve metrics:

- Target token log-frequency: We compute the log-frequency (i.e., unigram log-probability) of the target token in the pre-training dataset.
- Target token 5-gram log-probability: To capture the likelihood of the target token based on local context, we compute the log-probability of the target token conditioned only on the previous four tokens (i.e., a 5-gram model). We compute probabilities directly from the pre-training dataset, and we use backoff to $(n-1)$-grams when an n-gram is not observed in the dataset (Katz, 1987). Because 5-gram log-probability is roughly linearly related to target token log-frequency ($r = 0.632$), we compute the 5-gram log-probability residuals after regressing over target log-frequency. This captures the 5-gram log-probability after accounting for target token log-frequency.
- Context log-length: We compute the log of the number of context tokens.
- Context log-probability: We also compute the likelihood of the context, independent of the target token. We compute the mean log-frequency of all context tokens, equal to the negative log-perplexity of the context using a unigram language model. We use a unigram model to capture context frequency independent of word order within the context (Blei et al., 2003); longer n-gram models are more likely to capture probabilities of specific local constructions, even when they are distant from the target token.
- Target token contextual diversity: The diversity of contexts in which a word appears influences word learning in people, with beneficial effects in adults but potentially hindering effects in young children (Hills et al., 2010; Johns et al., 2016; Rosa et al., 2022; Chang and Bergen, 2022a). As in Hills et al. (2010), we count the number of unique tokens that appear within 30 tokens of the target token in the pre-training dataset.[6] To remove a nonlinear effect of token frequency on this raw diversity metric, we compute the residuals after fitting a GAM curve predicting a token's contextual diversity from its log-frequency (Chang and Bergen, 2022a). These residuals serve as a frequency-adjusted measure of a token's contextual diversity.
- Target token part-of-speech (POS): We annotate each example with POS tags (e.g., nouns, verbs, and adjectives; §A.3) using spaCy (Honnibal et al., 2020), and we consider the POS tag of the target token. Because words can span multiple tokens, we include a feature indicating whether the target token is the first token, intermediate token, last token, or only token in a word.

We fit separate linear regressions predicting each learning curve metric from the predictors above, iteratively adding predictors in the order listed.[7] We fit each regression to all 1M examples, predicting the mean value of each learning curve metric over all pre-training runs. We run likelihood ratio tests to assess whether each predictor is predictive of the target metric after accounting for all previous predictors, but we find that every test is highly significant ($p < 0.0001$). This is likely because the large number of examples (1M) makes even small effects statistically significant. Thus, we report adjusted R^2 values that capture the magnitude of effect of each predictor, after accounting for previous predictors (Table 3).

[6] Because our language models are autoregressive, we only consider context tokens that appear before the target token. We restrict our counts of co-occurring tokens to the 10K most frequent tokens in the dataset (Hills et al., 2010).

[7] We exclude interaction terms, which we find do not substantially improve predictions. Adjusted R^2 values increase by less than 0.03 even when including an interaction term between every pair of continuous predictors. We clip each predictor to five standard deviations from the mean.

Predictor	Surprisal	Var. (steps)	AoA	Forgettability	Var. (runs)
Target token log-frequency	$R^2 = 0.268$	$R^2 = 0.248$	$R^2 = 0.763$	$R^2 = 0.083$	$R^2 = 0.195$
+ Target 5-gram log-prob	(–) + 0.325	(–) + 0.050	(+) + 0.001	(–) + 0.149	(–) + 0.042
+ Context log-length	(–) + 0.007	(+) + 0.005	(+) + 0.001	(+) + 0.002	(+) + 0.005
+ Context 1-gram log-prob	(+) + 0.001	(–) + 0.006	(–) + 0.001	(–) + 0.010	(–) + 0.012
+ Target contextual diversity	(+) + 0.003	(+) + 0.000	(+) + 0.000	(+) + 0.001	(+) + 0.001
+ Target part-of-speech	+ 0.009	+ 0.006	+ 0.014	+ 0.028	+ 0.026
Total variance accounted	61.2%	31.5%	78.1%	27.2%	28.1%

Table 3: Increases in adjusted R^2 values when predicting each learning curve metric, iteratively adding predictors to a linear regression. The $(+)$ symbol indicates a positive coefficient for that predictor, evaluated in three regressions (§6.1). All other coefficients are negative. Coefficients for different part-of-speech tags are described in §6.2. In the bottom row, we report the total variance accounted for in each learning curve metric using all six predictors.

To assess the direction of effect for each continuous predictor on each learning curve metric, we consider the coefficient for that predictor in (1) a regression containing all predictors, (2) a regression containing that predictor alone, and (3) a regression containing that predictor alone but accounting for token log-frequency in the target metric (i.e., predicting learning curve metric residuals after the log-frequency regression). In all but one case, we obtain the same direction of effect in all three regressions.[8] Furthermore, the Pearson correlation between each pair of predictors is less than $r = 0.2$, and the variance inflation factor (VIF) for each predictor is less than 1.1. This indicates that the signs of our regression coefficients are safely interpretable. For effects of POS (a categorical variable), we consider the regression coefficient for different POS tags after accounting for all other predictors, by predicting learning curve metric residuals after regressing over the other predictors.

6.2 Results

The following conclusions are based on the regression results quantifying effects over all 1M examples. Results are reported in Table 3, including the direction of effect for each predictor and the variance accounted for in each learning curve metric.

Target Token Log-frequency. Frequent target tokens reach lower surprisals, are acquired faster, exhibit less variability within and across pre-training runs, and are less likely to be forgotten during pre-training. This is consistent with previous work showing that language models are highly reliant on token frequencies for syntactic rule learning (Wei et al., 2021), numerical reasoning (Razeghi et al., 2022), and overall word learning (Chang and Bergen, 2022b). Our work indicates that this effect persists at the individual example level.

Target 5-gram Log-probability. Unsurprisingly, 5-gram log-probabilities correlate with lower final surprisals after accounting for target token frequency; in other words, predictions from a 5-gram model and a Transformer model are correlated beyond the effects of token frequency. More notably, higher 5-gram log-probabilities are predictive of lower learning variability both within and across pre-training runs, along with lower forgettability. The added effect of 5-gram log-probability on forgettability ($+0.149$ R^2) is even stronger than the effect of target token frequency alone (0.083 R^2), suggesting that conditional token probabilities play a more significant role in language model forgetting than raw token frequencies.

Less intuitively, higher 5-gram log-probabilities are correlated with marginally later ages of acquisition. We hypothesize that this is because 5-grams do take time to learn (Figure 2), but low probability 5-grams are more likely to never be learned at all, reaching their minima early in training (e.g., during the unigram learning phase). This could drive the small effect where low probability 5-grams appear to be learned earlier. Indeed, of the 122 examples in both the bottom 1% of

[8] We obtain a negative coefficient for contextual diversity in one of three cases when predicting within-run variability. All other coefficients for contextual diversity are positive (§6.2).

5-gram log-probabilities and the earliest 1% of AoAs, 89% reach their minimum surprisal during the first 1K steps but then exhibit substantial (greater than 2.5) increases and fluctuations in surprisal for the remainder of pre-training. Notably, 96% never improve from random chance surprisal by more than 5%. In other words, low 5-gram probability examples may appear to exhibit early AoAs, but this is primarily because they are never learned particularly well, not due to early learning curve convergence. This reflects the fact that surprisal curves are not always accurate measures of ''learning'' (§7). An early drop in surprisal does not always indicate that an example is ''learned''.

Context Log-length. The remaining predictors account for far less variance in learning curve metrics than target log-frequency and 5-gram log-probability. Longer contexts correlate with lower surprisals, indicating that models successfully incorporate information from preceding context. However, longer contexts also correlate with higher variability within and across pre-training runs, higher forgettability, and later AoAs. This may be because predictions for a highly specific context are less generalizable and are thus learned less robustly by the models. This instability for long-context predictions is particularly notable as language models are increasingly used with long contexts (e.g., full conversations; OpenAI, 2022).

Context Log-probability. More frequent contexts are predictive of lower variance within and across pre-training runs, earlier acquisition, and lower forgettability. When models are repeatedly exposed to a context, regardless of the target token, their predictions stabilize earlier and with less variability. However, more frequent contexts also correlate with higher surprisals, indicating overall ''worse'' predictions. This may be because frequent contexts (e.g., descriptions of common situations) on average impose fewer constraints on the next token, leading to more ambiguous ground truth distributions and thus higher surprisals. If this is the case, the optimal surprisal values are simply higher in frequent contexts, but the models still learn faster and more stably given these contexts.

We note that the directions of effect for context log-probability remain stable for different window sizes of preceding context. After regressing out target token log-frequency, every coefficient sign for context log-probability remains the same for all window sizes in $\{1, 2, 4, \ldots, 128\}$. However, despite these consistent effects, context log-probability accounts for less than 3% of the variance in each learning curve metric in all cases, even before accounting for other predictors. Frequent contexts consistently correlate with faster and more stable learning, but with only small effects.

Target Contextual Diversity. Effects of contextual diversity are extremely small but statistically significant (§6.1). Tokens that appear in diverse contexts have higher final surprisals, are learned later, have greater variability within and across pre-training runs, and are more likely to be forgotten. This aligns with findings that contextual diversity hinders word learning in young children (Chang and Bergen, 2022a), contrasting with results in older children and adults (Johns et al., 2016; Rosa et al., 2022). Diverse contexts are thought to add noise to the early word learning process, introducing an excess of possible interpretations for a word.

Target Part-of-speech (POS). After accounting for other predictors, the POS tag of the target token has a small effect on each learning curve metric. Coefficients for all POS tags are reported in §A.3. Nouns, pronouns, and punctuation symbols reach lower final surprisals than verbs, adjectives, adverbs, and interjections. However, nouns are learned slower and with more variability (within and across pre-training runs) than adjectives, adverbs, and verbs, and they are more likely to be forgotten. Similarly, punctuation symbols exhibit high variability and forgettability, although they are learned early and reach low surprisals. Despite their high surprisals, interjections are learned early and stably. These results indicate that POS tags with lower surprisals are not necessarily learned more stably. Additionally, we find that different types of function words (e.g., conjunctions, prepositions, and determiners) have inconsistent effects, but they overall tend to be learned with high variability and forgettability.

The position of a token within a word also impacts learning curve metrics. Sub-word tokens after the first token in a word have low final surprisals, but they exhibit high forgettability and cross-run variability. Single-token words are the least likely to be forgotten and have the lowest cross-run variability. Compared to the POS tag of a word, a token's position within a word has only tiny effects on within-run variability and AoA (judged by the R^2 increase from including within-word position vs. only POS tag itself). These results underline the importance of tokenizer quality in language model pre-training (Rust et al., 2021); sub-word tokens are more likely to exhibit unstable learning despite low surprisals.

7 Discussion

In the previous sections, we report general patterns during language model pre-training (§4), define ways to characterize learning curves (§5), and isolate specific features that predict the speed and stability of learning for individual tokens in context (§6). Our results contribute to ongoing work studying language model pre-training dynamics, with implications for robust model deployment.

Sequential Learning. Previous work has demonstrated that language models exhibit fine-grained learning patterns that are not captured by the corpus-level loss curve (related work in §2). In particular, sudden increases and decreases in example loss (§5 and Xia et al., 2023) may be somewhat surprising given that the pre-training text is i.i.d. for all pre-training steps. By demonstrating that many of these sudden changes are consistent regardless of random initialization and data shuffling (§5.2), our work indicates that some instances of sudden learning and ''forgetting'' are not due to random chance or the specific examples observed in a given step.[9] Rather, they reflect some change in model processing that consistently occurs partially into pre-training (roughly step $t \neq 0$). Because such a sudden change cannot be attributed to the specific examples observed (robust to random shuffling) or any change in the pre-training distribution at time t (the data is always i.i.d.), the primary remaining explanation is that the models' sudden ''learning'' at step $t \neq 0$ is made possible by some systematic difference between models (and their optimizers) just before step t vs. at step 0.

Framed from a potentially more interesting perspective, some types of language model ''learning'' appear to be dependent on previous learning and the linguistic abilities already present in the model. This aligns with previous work showing that language models acquire linguistic abilities in a systematic order during pre-training (Liu et al., 2021; Choshen et al., 2022), although not necessarily due to sequential dependencies. For example, Evanson et al. (2023) show that despite similar acquisition orders across models, different syntactic abilities are learned in parallel; performance for most individual abilities increases from the onset of pre-training. Our work provides evidence that there exist other capabilities or types of generalizations (e.g., non-syntactic abilities or even more fine-grained syntactic sub-abilities) that can only be learned after others, or at least only once the model reaches some particular state. Isolating these sequential dependencies is an exciting direction for future work.

***N*-gram Learning and Refinement.** As a further step towards understanding fine-grained learning patterns in language models, our work investigates whether simple statistical regularities can explain learning patterns such as the sudden loss changes discussed above. We demonstrate that learning curves are more stable and converge faster for frequent tokens, n-gram probable tokens, and frequent contexts (§6.2). High probability n-grams in particular are less likely to be ''forgotten'', suggesting that evolving model generalizations throughout pre-training have larger effects on low-probability n-grams. Combined with findings that language models roughly follow n-gram learning early in pre-training and only later produce longform coherent text (§4; Chang and Bergen 2022b), language model learning might be characterized as early n-gram learning, then gradual refinement of the tail n-gram probabilities based on longer contexts and more nuanced linguistic capabilities (e.g., world knowledge and reasoning; Liu et al., 2021).

Robust Model Deployment. Our work also has implications for robust model deployment. High token frequencies and n-gram probabilities are by far the most influential predictors of early and stable learning in language models (§6.2, with marginal additional effects of context lengths and likelihoods). As language models are

[9] In our case, ''forgetting'' does not always indicate a decrease in model quality, but rather that the model has changed its output distribution such that a given ground truth token is less likely. The model distribution might still be a better reflection of text distributions overall.

deployed in domains with highly-specific vocabulary terms (e.g., healthcare, law, and finance; Yang et al., 2024), the accurate prediction of infrequent domain-specific terms during text generation is likely to require extensive pre-training (late acquisition, likely mitigated by large pre-training datasets). Such domain-specific text generation is also likely to be unstable across models and pre-training steps (high variability, potentially more difficult to mitigate). Even if model deployment in these areas is beyond researchers' control, realistic expectations of when models might behave unstably are important to facilitate safe use by the public. Of course, it is also possible that fine-tuning or careful prompting may reduce instability across models and training steps in these domains.

Finally, our work demonstrates that loss curves for individual examples often fluctuate in ways that are not evident from aggregate loss curves. Even as models appear to converge (smoothly plateauing loss), models may still be adjusting predictions for tail examples in substantial ways. Our work provides insights and methods to identify examples that are likely to exhibit late fluctuations in language model pre-training; for example, low probability n-grams correlate with high variability and forgettability metrics. When determining whether a model is sufficiently and stably trained for a given use case, convergence for these types of examples should be considered.

Limitations and Scaling. Our work has several limitations. First, surprisal is an imperfect proxy for language model learning. A model might achieve the same surprisal at different points during pre-training by using different internal prediction strategies (e.g., predicting the same token based on frequency vs. more nuanced reasoning). Additionally, reaching some minimum surprisal does not mean that an example is ''learned''; it simply indicates the best performance achieved by a model. The optimal surprisal is not necessarily zero due to the nondeterminism of language. That said, surprisal remains a common measure of language model behavior (Futrell et al., 2019; Li et al., 2021), performance (Hoffmann et al., 2022), and learning (Chang and Bergen, 2022b; Xia et al., 2023), and it requires no annotated text data to compute.

Second, we only consider language models with 124M parameters trained on 5.1B tokens. Previous work has demonstrated that learning curves differ across model sizes (Xia et al., 2023); larger models are able to ''learn'' some examples (usually late in pre-training) for which smaller models reach non-optimal local minima or even diverge. Larger models also exhibit less forgetting of pre-training examples (Tirumala et al., 2022), although it remains unclear whether similar mechanisms are responsible for evaluation example forgetting (i.e., surprisal increases for seen vs. unseen examples). Further research is necessary to determine the effects of model size on learning speed, variability, and forgetting; with a larger compute budget, the methods presented in our work can easily be applied to larger models. Nonetheless, previous work has documented similar behaviors for different model sizes when they achieve similar perplexities (Choshen et al., 2022; Xia et al., 2023), suggesting that pre-training dynamics in smaller models may be similar to the early dynamics of larger models. A particularly exciting direction for future work is to characterize the examples (e.g., based on types of reasoning, world knowledge, or commonsense) that fluctuate at different points during pre-training across model sizes.

8 Conclusion

In this work, we identify learning patterns during language model pre-training, including concrete features that predict when and how stably individual examples are acquired. We assess the impact of n-gram probabilities, context lengths and likelihoods, and part-of-speech tags on the speed and stability of language model learning. We propose a high-level characterization of language model learning based on simple distributional statistics, and we discuss implications for deploying robust language models in practice.

Acknowledgments

We thank the UCSD Language and Cognition Lab for valuable discussion, and the anonymous reviewers for insightful comments. Some models were trained on hardware provided by the NVIDIA Corporation as part of an NVIDIA Academic Hardware Grant. Zhuowen Tu is supported by NSF IIS-2127544. Tyler Chang is partially supported by the UCSD HDSI graduate fellowship.

References

Julien Abadji, Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2021. Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora*, pages 1–9.

Anthropic. 2023. Introducing Claude. *Anthropic Blog*.

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.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Terra Blevins, Hila Gonen, and Luke Zettlemoyer. 2022. Analyzing the mono- and cross-lingual pretraining dynamics of multilingual language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3575–3590, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.234

Tyler A. Chang and Benjamin K. Bergen. 2022a. Does contextual diversity hinder early word acquisition? In *Proceedings of the 44th Annual Conference of the Cognitive Science Society*.

Tyler A. Chang and Benjamin K. Bergen. 2022b. Word acquisition in neural language models. *Transactions of the Association for Computational Linguistics*, 10:1–16. https://doi.org/10.1162/tacl_a_00444

Cheng-Han Chiang, Sung-Feng Huang, and Hung-yi Lee. 2020. Pretrained language model embryology: The birth of ALBERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6813–6828, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.553

Leshem Choshen, Guy Hacohen, Daphna Weinshall, and Omri Abend. 2022. The grammar-learning trajectories of neural language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8281–8297, Dublin, Ireland. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.568

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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. https://doi.org/10.18653/v1/2023.findings-acl.773

Richard Futrell, Ethan Wilcox, Takashi Morita, Peng Qian, Miguel Ballesteros, and Roger Levy. 2019. Neural language models as psycholinguistic subjects: Representations of syntactic state. 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 32–42, Minneapolis, Minnesota. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1004

Adam Goodkind and Klinton Bicknell. 2018. Predictive power of word surprisal for reading times is a linear function of language model quality. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 10–18, Salt Lake City, Utah. Association for Computational Linguistics. https://doi.org/10.18653/v1/W18-0102

Google. 2023. PaLM 2 technical report. *arXiv*.

Thomas Hills, Josita Maouene, Brian Riordan, and Linda Smith. 2010. The associative structure of language: Contextual diversity in early word learning. *Journal of Memory and Language*, 63(3):259–273. https://doi.org/10.1016/j.jml.2010.06.002, PubMed: 20835374

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

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. SpaCy: Industrial-strength natural language processing in python. SpaCy.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2022. Lifelong pretraining: Continually adapting language models to emerging corpora. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4764–4780, Seattle, United States. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.351

Brendan Johns, Melody Dye, and Michael Jones. 2016. The influence of contextual diversity on word learning. *Psychonomic Bulletin and Review*, 23:1214–1220. https://doi.org/10.3758/s13423-015-0980-7, PubMed: 26597891

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. 2016. Visualizing and understanding recurrent networks. *arXiv*.

Slava Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):400–401. https://doi.org/10.1109/TASSP.1987.1165125

Roger Levy. 2008. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177. https://doi.org/10.1016/j.cognition.2007.05.006, PubMed: 17662975

Bai Li, Zining Zhu, Guillaume Thomas, Yang Xu, and Frank Rudzicz. 2021. How is BERT surprised? Layerwise detection of linguistic anomalies. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4215–4228, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.325

Zeyu Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A. Smith. 2021. Probing across time: What does RoBERTa know and when? In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 820–842, Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.1007/978-3-030-84186-7

Kyle Mahowald, Anna A. Ivanova, Idan A. Blank, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. 2023. Dissociating language and thought in large language models: A cognitive perspective. *arXiv*. https://doi.org/10.1016/j.tics.2024.01.011, PubMed: 38508911

James A. Michaelov, Megan D. Bardolph, Cyma K. Van Petten, Benjamin K. Bergen, and Seana Coulson. 2024. Strong Prediction: Language Model Surprisal Explains Multiple N400 Effects. *Neurobiology of Language*, pages 1–29. https://doi.org/10.1162/nol_a_00105, PubMed: 38645623

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection.

In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Byung-Doh Oh and William Schuler. 2023. Why does surprisal from larger transformer-based language models provide a poorer fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11:336–350. https://doi.org/10.1162/tacl_a_00548

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Benjamin Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, John Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom B. Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. 2022. In-context learning and induction heads. *arXiv*.

OpenAI. 2022. ChatGPT: Optimizing language models for dialogue. *OpenAI Blog*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *OpenAI*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Technical Report*.

Yasaman Razeghi, Robert L. Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot numerical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 840–854, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.findings-emnlp.59

Eva Rosa, Rafael Salom, and Manuel Perea. 2022. Contextual diversity favors the learning of new words in children regardless of their comprehension skills. *Journal of Experimental Child Psychology*, 214. https://doi.org/10.1016/j.jecp.2021.105312, PubMed: 34753015

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? On the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.243

Naomi Saphra and Adam Lopez. 2019. Understanding learning dynamics of language models with SVCCA. 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 3257–3267, Minneapolis, Minnesota. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1329

Thibault Sellam, Steve Yadlowsky, Jason Wei, Naomi Saphra, Alexander D'Amour, Tal Linzen, Jasmijn Bastings, Iulia Turc, Jacob Eisenstein, Dipanjan Das, et al. 2022. The MultiBERTs: BERT reproductions for robustness analysis. *International Conference on Learning Representations*.

Daniel Servén and Charlie Brummitt. 2018. pyGAM: Generalized additive models in Python. *pyGAM*.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.746

Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems*.

Alex Warstadt and Samuel R. Bowman. 2023. *What Artificial Neural Networks Can Tell*

Us About Human Language Acquisition. Taylor & Francis, chap. Algebraic Structures in Natural Language. https://doi.org/10.1201/9781003205388-2

Jason Wei, Dan Garrette, Tal Linzen, and Ellie Pavlick. 2021. Frequency effects on syntactic rule learning in transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 932–948, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.72

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. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Simon Wood. 2017. *Generalized Additive Models: An Introduction with R*. CRC Press. https://doi.org/10.1201/9781315370279

Mengzhou Xia, Mikel Artetxe, Chunting Zhou, Xi Victoria Lin, Ramakanth Pasunuru, Danqi Chen, Luke Zettlemoyer, and Veselin Stoyanov. 2023. Training trajectories of language models across scales. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13711–13738, Toronto, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.767

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. 2024. Harnessing the power of LLMs in practice: A survey on ChatGPT and beyond. *ACM Transactions on Knowledge Discovery from Data*, 18(6). https://doi.org/10.1145/3649506

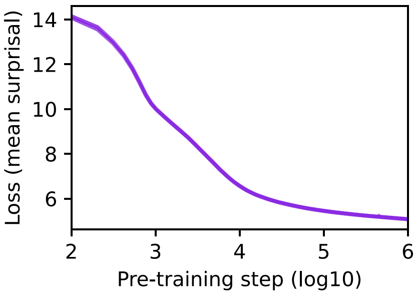


Figure 4: Loss curves (mean surprisal) for all five pre-training runs. Loss curves are nearly identical across runs. To align with other figures, pre-training steps are reported in log10.

A Appendix

A.1 Pre-Training Details

Language models are pre-trained using the HuggingFace Transformers library (Wolf et al., 2020). Hyperparameters are reported in Table 4, and loss curves are in Figure 4.

Hyperparameter	Value
Layers	12
Embedding size	768
Hidden size	768
Intermediate hidden size	3072
Attention heads	12
Attention head size	64
Activation function	GELU
Vocab size	50004
Max sequence length	128
Position embedding	Absolute
Batch size	256
Train steps	1M
Learning rate decay	Linear
Warmup steps	10000
Learning rate	1e-4
Adam ϵ	1e-6
Adam β_1	0.9
Adam β_2	0.999
Dropout	0.1
Attention dropout	0.1

Table 4: Language model pre-training hyperparameters (Devlin et al., 2019; Chang and Bergen, 2022b).

Each model takes 2.1 weeks to train on four NVIDIA TITAN Xp GPUs or 2.5 weeks to train on one NVIDIA RTX A6000 GPU. Including pre-training and inference (evaluation surprisals), our experiments take approximately 2220 hours in A6000 GPU hours. Computing fitted GAM curves, distances between curves, n-gram probabilities, contextual diversities, and POS tags takes approximately 2990 CPU core hours.

A.2 Checkpoint Strategy

Assume that the steps per checkpoint $s(t)$ increases linearly as a function of the current step t. Assume we start with some $s(0) = s_0$ and end with $s(t_1) = s_1$ steps per checkpoint. Then:[10]

$$s(t) = s_0 + \frac{s_1 - s_0}{t_1} t$$

The rate of checkpoints per step is the inverse of steps per checkpoint, or $1/s(t)$. Excluding the checkpoint at step zero (i.e., $\text{checkpoints}(0) = 0$), the number of checkpoints at step t is:

$$\begin{aligned} \text{checkpoints}(t) &= \text{checkpoints}(0) + \int_0^t \frac{1}{s(t)} dt \\ &= \int_0^t \frac{t_1}{s_0 t_1 + (s_1 - s_0)t} dt \\ &= \frac{t_1}{s_1 - s_0} \ln \left(1 + \frac{s_1 - s_0}{s_0 t_1} t\right) \end{aligned}$$

By solving for t, we can compute the time steps where the number of checkpoints is equal to $n = 0, 1, 2, 3, \text{etc.}$ Formally, we can compute the time step t for the n^{th} checkpoint:

$$\begin{aligned} n &= \text{checkpoints}(t) \\ n &= \frac{t_1}{s_1 - s_0} \ln \left(1 + \frac{s_1 - s_0}{s_0 t_1} t\right) \end{aligned}$$

$$t = \frac{s_0 t_1}{s_1 - s_0} \left(\boldsymbol{e}^{n\left(\frac{s_1 - s_0}{t_1}\right)} - 1 \right)$$

Note that t increases exponentially as a function of the checkpoint number n. For our experiments, we start with $s_0 = s(0) = 100$ steps per checkpoint. We end with $s_1 = s(1000000) = 25000$ steps per checkpoint at step 1M. Then, the time step t for the n^{th} checkpoint is:

$$\text{step}(n) = \frac{100 * 1000000}{24900} \left(\boldsymbol{e}^{n\left(\frac{24900}{1000000}\right)} - 1 \right)$$

We round each $\text{step}(n)$ to the nearest integer, and we save model checkpoints at the selected steps until reaching 1M steps. Concretely, we save checkpoints at steps: $\text{step}(1) = 101$, $\text{step}(2) = 205, \ldots, \text{step}(221) = 981536$. We also save one checkpoint at step zero. In total, we save 222 checkpoints per pre-training run.

A.3 Part-of-Speech (POS) Coefficients

In Table 5, we report coefficients for all POS tags when predicting each learning curve metric, after accounting for other predictors (§6.1).

[10] Assume $t_1 > 0$ and $s_1 > s_0 > 0$.

Surprisal		Var. (steps)		AoA		Forgettability		Var. (runs)	
Tag	**Coef.**	**Tag**	**Coef.**	**Tag**	**Coef.**	**Tag**	**Coef.**	**Tag**	**Coef.**
PART	−1.28	INTJ	−0.03	INTJ	−0.30	INTJ	−0.36	INTJ	−0.23
AUX	−1.19	PART	−0.02	PUNCT	−0.25	SCONJ	−0.25	NUM	−0.15
NOUN	−1.17	X	−0.01	DET	−0.23	ADV	−0.22	VERB	−0.08
PUNCT	−1.16	AUX	−0.01	NUM	−0.19	VERB	−0.19	ADV	−0.05
PRON	−1.13	SCONJ	−0.01	X	−0.18	NUM	−0.12	SCONJ	−0.05
X	−1.12	NUM	−0.01	VERB	−0.17	PART	−0.11	AUX	−0.02
SYM	−0.89	VERB	0.00	ADJ	−0.17	X	−0.09	ADJ	−0.02
PROPN	−0.81	PRON	0.00	ADV	−0.15	ADJ	−0.08	PART	0.02
ADP	−0.75	ADV	0.00	SYM	−0.15	PRON	−0.03	PRON	0.03
VERB	−0.64	DET	0.00	PROPN	−0.13	AUX	−0.01	SYM	0.04
NUM	−0.60	PROPN	0.00	PART	−0.12	ADP	0.05	DET	0.07
SCONJ	−0.53	PUNCT	0.00	CCONJ	−0.11	NOUN	0.09	X	0.07
CCONJ	−0.49	ADJ	0.00	SCONJ	−0.09	CCONJ	0.14	CCONJ	0.08
INTJ	−0.47	ADP	0.00	NOUN	−0.07	PUNCT	0.22	PUNCT	0.11
DET	−0.41	CCONJ	0.01	PRON	−0.06	PROPN	0.27	ADP	0.12
ADJ	−0.35	SYM	0.01	AUX	−0.04	SYM	0.29	NOUN	0.15
ADV	−0.11	NOUN	0.01	ADP	0.01	DET	0.50	PROPN	0.15
L	−0.56	B	−0.01	U	0.00	U	0.00	U	0.00
I	−0.26	L	0.00	I	0.00	B	0.43	B	0.11
U	0.00	U	0.00	L	0.03	L	0.47	L	0.30
B	0.82	I	0.02	B	0.03	I	0.67	I	0.41

Table 5: Part-of-speech (POS) tag coefficients when predicting each learning curve metric, after accounting for other predictors (§6.1). POS tags use the Universal POS tags (Nivre et al., 2020), and we include a feature indicating whether a token is the first token (B), intermediate token (I), last token (L), or only token (U) in a word.