# The Grammar-Learning Trajectories of Neural Language Models

**Leshem Choshen**[†], **Guy Hacohen**[†‡], **Daphna Weinshall**[†], **Omri Abend**[†]
Department of Computer Science[†]
Department of Brain Sciences[‡]
Hebrew University of Jerusalem
{first.last}@mail.huji.ac.il

## Abstract

The learning trajectories of linguistic phenomena in humans provide insight into linguistic representation, beyond what can be gleaned from inspecting the behavior of an adult speaker. To apply a similar approach to analyze neural language models (NLM), it is first necessary to establish that different models are similar enough in the generalizations they make. In this paper, we show that NLMs with different initialization, architecture, and training data acquire linguistic phenomena in a similar order, despite their different end performance. These findings suggest that there is some mutual inductive bias that underlies these models' learning of linguistic phenomena. Taking inspiration from psycholinguistics, we argue that studying this inductive bias is an opportunity to study the linguistic representation implicit in NLMs.

Leveraging these findings, we compare the relative performance on different phenomena at varying learning stages with simpler reference models. Results suggest that NLMs exhibit consistent "developmental" stages. Moreover, we find the learning trajectory to be approximately one-dimensional: given an NLM with a certain overall performance, it is possible to predict what linguistic generalizations it has already acquired. Initial analysis of these stages presents phenomena clusters (notably morphological ones), whose performance progresses in unison, suggesting a potential link between the generalizations behind them.

## 1 Introduction

Children present remarkable consistency in their patterns of language acquisition. They often acquire linguistic phenomena in a similar order (Kuhl et al., 1992; Ingram, 1989), and make similar generalizations and over-generalizations (Kuczaj II, 1977; Pinker, 1995). This consistency provides an important starting point for linguistic study. For example, arguments in favor of single or dual system accounts of morphological representation are often backed by computational models of children learning trajectories (e.g., Rumelhart and McClelland, 1986; Pinker and Prince, 1988; Kirov and Cotterell, 2018). In this paper, we embrace this program for the study of computational language models, investigating learning trajectories. [1]

The representations that language models (LM) acquire have been studied extensively, including studying their learning dynamics to improve training (see §6). However, very little work aimed at drawing connections between the training dynamics and the learned representations. In this work we adopt a behavioral approach, thus revealing that NLMs share learning trajectories and generalize in similar ways during training. This implies that studying trajectories of NLMs is worthwhile, in the sense that results on one architecture or size are expected to be reproducible by others.

These findings call for a characterization of these trajectories, a new and promising territory for research. We take first steps to explore these directions, emphasizing their potential benefit to a better future understanding of what models learn.

Specifically, we train NLMs on next-word prediction, but evaluate and compare them by tracking their performance on grammar learning in English, using the BLIMP dataset (See 2.1). BLIMP is a dataset that consists of 67K minimal pairs, where each pair includes a grammatically correct and a grammatically erroneous sentence. NLMs are tested for their ability to assign higher probability to the correct one. See example in Table 1, and details of our experimental methodology in §2.

We begin (§3) by establishing that NLMs learn grammatical phenomena in a consistent order. We evaluate NLMs at different time points along their training, showing that the performance on linguis-

[1] Code is supplied in https://github.com/borgr/ordert



*Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*
*Volume 1: Long Papers*, pages 8281 - 8297

| Challenge | Correct | Erroneous |
|---|---|---|
| Animate subject | **Galileo** had talked to Bell. | **This car** had talked to Bell. |
| Drop argument | The groups **buy**. | The groups **dislike**. |

Table 1: BLIMP minimal pairs examples.

tic phenomena across initializations is highly correlated. We further find many similarities in the set of examples that they correctly classify.

Still, models of different architectures learn at a different pace, and hence cannot be directly compared at identical time points. In §3.3, we overcome this by re-scaling the timeline. We then show that despite architectural differences, NLMs present highly correlated performance trajectories. In §3.4, we further demonstrate that even the choice of training data has minor influence on the results. Finally, in §3.5 we show that the learning dynamics essentially follows a single dimension. Namely, where the average performance is similar, success on linguistic phenomena is also similar.

We proceed by analyzing the early stages of learning in §4. We find that, at first, NLMs rely mostly on local cues and not on word order. They thus resemble bag-of-words models over a window of the preceding tokens. Later stages seem to drift further away from bag-of-words models toward $n$-gram models, and with time seem to be more sensitive to structural cues. We also find evidence that some latent features that the model learns may not be related to linguistic phenomena.

Finally, in §5 we take the first steps in categorizing linguistic phenomena by their learning trajectories. We identify links between their representations by finding phenomena that progress in unison. For example, we find that morphological phenomena are mostly learned at similar stages. Of particular interest are cases where performance decreases with time, which may suggest either overgeneralization or biases in the BLIMP challenges.

## 2 Experimental Setup

### 2.1 The BLIMP Dataset

We use BLIMP (Warstadt et al., 2019) to assess the extent to which generalizations are made by the NLMs. BLIMP includes 67 grammatical *challenges* categorized into 13 *super-phenomena* (e.g., island-related or quantifiers) comprising of 4 broad *fields* (e.g., Syntax, Semantics). Each challenge consists of 1K minimal pairs of sentences. A minimal pair contains a sentence and a near-duplicate distractor that incorporates an error on a particular linguistic phenomenon, i.e., only the phenomenon in question is changed between the sentences in a pair (see Table 1). Each challenge includes pairs with the same linguistic phenomenon.

### 2.2 Training

LM details: as training multiple GPT2 instances (Radford et al., 2019) is computationally demanding, we train smaller NLMs. Following Turc et al. (2019), we trained 1 instance of GPT2$_{small}$ (width 768, 12 layers, 8 attention heads) and 4 instances of GPT2$_{tiny}$ (width 512, 4 layers, 4 attention heads), with different random seeds.

Similarly, we train a small TransformerXL (Dai et al., 2019), $XL_{small}$ (width 512, 4 layers, 8 attention heads) and a full-sized one (width 4096, 18 layers, 16 attention heads). We stop the full model after 600K steps, while the perplexity remained high. We use it for comparison to the early stages of learning of TransformerXL. All models' hyperparameters can be found in App. §B. We also use the results of the fully trained GPT2, TransformerXL, LSTM and human performance reported in Warstadt et al. (2019).

In §4, we compare NLMs with simpler models. To this end, we create two GPT2$_{tiny}$ variations, denoted *BOW* and *Window-5*. BOW replicates GPT2$_{tiny}$, but relies only on bag of words. This is achieved by removing the positional weights, and replacing the attention weights with a simple average.[2] Window-5 similarly ignores the positions, and additionally only attends to the last 5 words. Note that both are unidirectional LMs and consider only previously predicted words at each step.

Unless explicitly stated otherwise (as in §3.4), all models were trained on the WikiBooks dataset (Zhu et al., 2015), which contains the English Wikipedia ($2.1B$ words) and BookCorpus ($854M$ words). This dataset resembles BERT's training data (Devlin et al., 2019), except that current Wikipedia is used. Additionally, we trained models on the following datasets: English openSubtitles (Lison and Tiedemann, 2016), newsCrawl (Barrault et al., 2019), GigaWord (Napoles et al., 2012), and

[2] Supposedly, removing the positional embeddings would suffice. Empirically, it has little effect. Presumably, as embeddings only attend to previous positions, the network manages to represent positions by the difference between them. This is in line with the finding that GPT2's positional embeddings are not meaning-bearing (Wang and Chen, 2020).

a sample of openWebText (3B words; Gokaslan and Cohen, 2019) – a replication of GPT2 dataset.

Throughout this paper, we report Pearson correlation. Using Spearman correlation leads to qualitatively similar conclusions. When multiple models are correlated against each other, their average pairwise correlation is reported.

## 3 The Learning Order of NLMs

In this section, we examine various aspects of NLMs, generally showing that their learning trajectories are similar.

We evaluate network similarity by adopting a behavioral approach. Accordingly, networks are viewed as functions, whose *latent features* manifest themselves only by their influence on the network's behavior. Latent features are the unobserved causes of the measured behavior. Consequently, parameters, activation patterns and representations can be completely different among *similar* models. This is unlike the approaches employed by Williams et al. (2018); Saphra and Lopez (2019); Liu et al. (2021), which analyze internal representations directly.

To formalize the above notion, let $L_t$ denote a checkpoint, the language model $L$ at time $t$. Let $pv(L_t)$ denote its *performance vector* – the accuracy obtained by $L$ on each BLIMP challenge $p$:

$$pv(L_t) = [acc(L_t, p)]_{p \in BLIMP} \in \mathbb{R}^{67} \quad (1)$$

Time $t$ is measured in training steps or perplexity. The trajectory of the performance vector as a function of $t$ reflects $L$'s training dynamics.

Given this behavioral definition, we focus on comparing the relative strength of models. Similarity between models is thus measured as the correlation between their performance vectors. Hence, models are similar if they rank phenomena in the same way. On the other hand, models of the same average performance can be dissimilar: consider two models that agree on everything except nouns. One generates only feminine nouns and the other plural nouns. The models' average performance is similar, but due to their biases, they are correct on different challenges. This dissimilarity suggests that the models rely on different latent features.

### 3.1 Consistent Order of Learning

We begin by showing that models produced by different initializations learn the same phenomena, in the same order. In terms of our definitions above, this may imply that despite converging to different parameter values, the learned latent features and the generalization patterns made are similar.

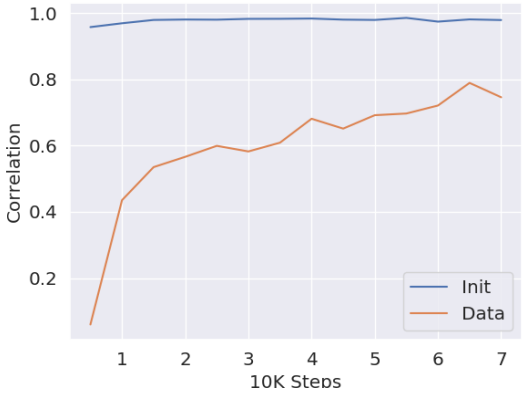



Figure 1: **High correlation** after warmup (5K steps). Correlation between the performance vectors (measured by steps) of $GPT2_{tiny}$ models with different initialization (blue) or training data (orange).

In order to examine the hypothesis empirically, we compute the correlation between 4 random initializations (Fig. 1). Results confirm the hypothesis, the correlation between $GPT2_{tiny}$ instances is extremely high. It is already high after 10K steps, and remains high throughout training. We note that the correlation at step 0 is 0 (not shown), and that after 10K warm-up steps the network's ability as a LM is still poor. For example, perplexity is 10.9 after 10K steps and 6.7 after 70K steps.

### 3.2 Effects of Architecture

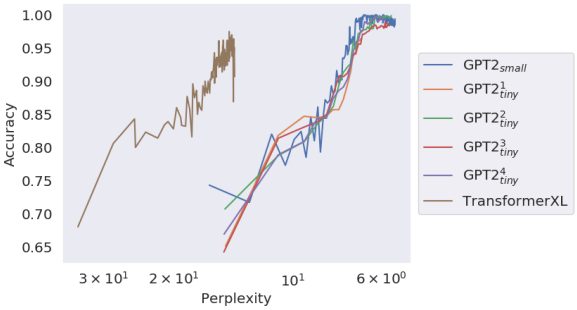



Figure 2: **Similar Accuracy** despite different initializations and sizes of the $GPT2_{small}$ models. TransformerXL perplexity is not computed on the same vocabulary, but still shows a (rescaled) similar trend. The graph depicts trajectories on an example phenomenon ("existential there"). y-axis is the accuracy during training and x-axis is the model's perplexity.

Next, we show that different architectures also present similar trajectories. As the learning pace is not comparable across models, computing correla-

tion in fixed and identical intervals is not informative. Instead, we choose $t$ to be the perplexity on the development set, comparing models at the same performance level. TransformerXL is not directly comparable as perplexity requires the vocabulary to be the same.

Following this paradigm, we see that GPT2$_{small}$ and GPT2$_{tiny}$ are highly correlated (>0.9), presenting similar learning order throughout training. Observing the trajectories per challenge qualitatively, we see that they align very well (cf. Fig. 2 and App. §A, §C). TransformerXL also seems to share the general tendencies of the GPT2 architectures.

Interestingly, we see that models behave similarly not only in terms of relative performance, but also at the example level (binary decision per minimal pair). We find that GPT2$_{small}$ and GPT2$_{tiny}$ have an average agreement of $\kappa = 0.83$ (Fleiss et al., 1969). This implies strong consistency in the order of learning of different examples also within phenomena. Henceforth, we focus on the phenomena-level as it is more interpretable, lending itself more easily to characterization. We discuss per-example similarity further in App. §D.

### 3.3 Comparison to Off-the-shelf Models

So far, we have observed the common trajectories presented by NLMs that are trained in parallel. We proceed to compare trajectories of one model to other models' performance vectors at a single point of interest in their learning, i.e. a checkpoint's performance vector. This allows us to analyze how similarities evolve, rather than whether two trajectories are synced. We compare fully trained off-the-shelf NLMs with the trajectory of GPT2$_{tiny}$ (Fig. 3a) and GPT2$_{small}$ (App. §E).

The observed similarity to off-the-shelf models is high (0.6-0.8), implying that NLMs in general share tendencies and biases. Moreover, similarity increases until the point of same performance and then (when relevant) decreases. This suggests that the small NLM approaches off-the-shelf tendencies as it improves and stops somewhere on the same trajectory of generalizations (cf. §3.5). Furthermore, we find considerable correlation with the performance levels of humans on the different challenges, but still, all NLMs correlate better with our model than humans correlate with it.

These results present a curious order imposed on the NLMs. Both GPT2$_{tiny}$ and GPT2$_{small}$ (App. §E) are more similar to the LSTM model than to TransformerXL, and even less similar to GPT2$_{large}$. Interestingly, our models are more similar to an RNN and a model with a different architecture, than to a larger model with the same architecture. Thus, it seems that the architecture type cannot explain the similarities in the relative order. We further examine this issue in the next section.

### 3.4 Effect of Training Data

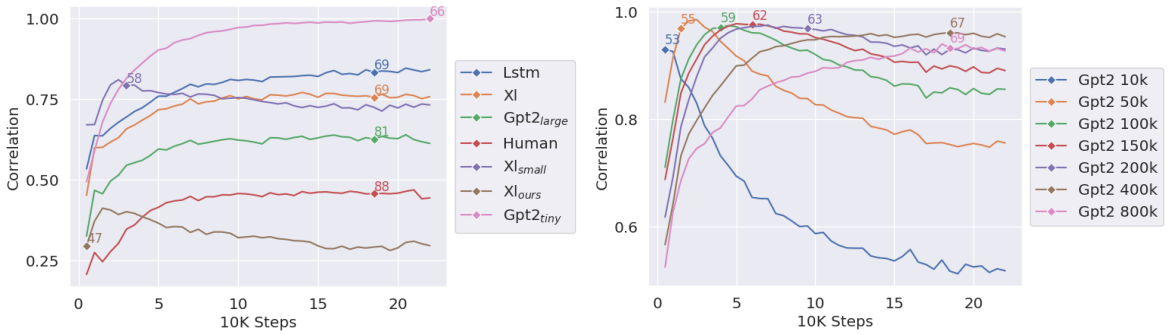
This section examines the possibility that the similarities reported in Fig. 3a can simply be explained by the similarity in the NLM's training data. More specifically, since the ranking by model similarity reported above fits the similarity between the training sets that the models were trained on, we view it as a potential confound and attempt to control for it. Our training data (WikiBooks) consists mostly of Wikipedia and so do the LSTM's and TransformerXL's training sets, which are trained on earlier versions of Wikipedia and WikiMatrix (Schwenk et al., 2019) respectively. GPT2, on the other hand, is trained on openWebText, which consists of scraped web pages.

To tease apart the effect of training data, we trained 3 additional GPT2$_{tiny}$ instances over the openWebText, openSubtitles and newsCrawl datasets. Results (Fig. 1) show that the dataset has more effect on the correlation than initialization. Hence, the choice of training data does affect the learning trajectory, but its effect decreases with training (correlation gets higher with more training steps). We also recompute the correlations from §3.3 after training GPT2$_{tiny}$ on the same data as GPT2$_{large}$ (App. §F), and find that the relative order between the NLMs remains the same, with GPT2$_{large}$ being the least similar.

We conclude that while the training data affects the learned generalizations, it only very partially explains the observed similarities between NLMs.

### 3.5 One Dimension of Learning

Based on the findings of the previous sub-sections, we hypothesize that current NLMs all learn in a similar order, where the effect of training data and architecture is secondary. In other words, training time, size and efficiency may affect what a model has learned, but not its learning order. This implies that stronger models may improve performance, but still follow a similar learning trajectory. If this hypothesis is correct, models should be most similar to models with the same performance; similarity should drop as the gap in performance widens.




(a) Off-the-shelf and human

(b) GPT2$_{small}$ checkpoint after $X$ steps

Figure 3: Reference models **correlate** the most with GPT2$_{tiny}$ **when** they have the most **similar performance** (or near it). Correlation during GPT2$_{tiny}$ training compared to off-the-shelf LMs and human performance (left) or to mid-training GPT2$_{small}$ checkpoints (right). Curves correspond to fixed performance vectors. Where the X-axis follows the training trajectory of gpt, each line represents similarity to a different checkpoint, either of different fully trained models (left) or to checkpoint during the training of a larger model (right). Numbers are the average performance of the checkpoint, and are placed over the step where this average performance is the most similar to that of GPT2$_{tiny}$. The best score of GPT2$_{tiny}$ is 67.

Controlled comparison supports this hypothesis. Fig. 3b presents the correlation of GPT2$_{tiny}$ training trajectory with several static checkpoints taken during GPT2$_{small}$ training. We observe that at the point in which the average performance of GPT2$_{tiny}$ is closest to that of the checkpoint, the correlation peaks, and then decreases again as GPT2$_{tiny}$ surpasses the checkpoint in average performance. So overall correlation peaks when average performance is most similar. Note that despite the different network sizes and convergence rates, the correlation's maximal value is very high (higher than 0.9).

Further experiments show similar trends. Fig. 3a presents a similar investigation, albeit with more varied architectures and training datasets. Here too the maximum correlation is obtained around the point of most similar performance.

### 3.6 Comparison to 5-gram

NLMs are most similar to other NLMs with the same performance. However, when compared to non-neural LMs, this is no longer the case.

More specifically, we compare GPT2$_{tiny}$ to two 5-gram LMs trained on the same dataset as the NLMs (WikiBooks) and another (GigaWord) dataset. Results are shown in Fig. 4, which is qualitatively different from Fig. 3a. Here, similarity in performance implies neither high correlation, nor the point of highest similarity. This serves both as a sanity check to our methodology, and as a reminder of model biases: In general, models may have different biases and tendencies, regardless of overall performance. In our case, it seems that NLMs share biases between them that are not necessarily shared with other LMs.

While not the main purpose of the analysis, our comparison reveals other noteworthy trends. For example, 5-gram LMs trained on different corpora have different correlations to the GPT2$_{tiny}$ trajectory. This is further discussed in App. §G.

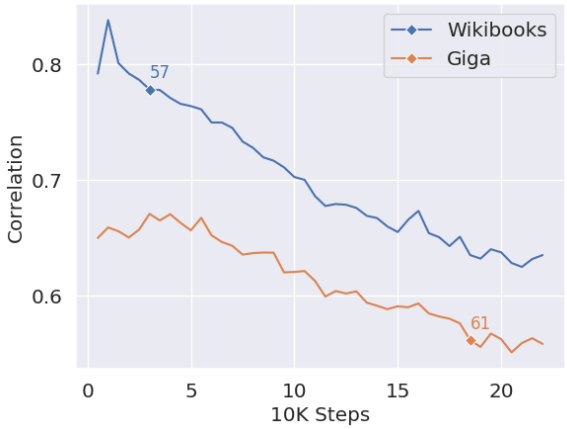



Figure 4: Correlation during training of GPT2$_{tiny}$ compared to a **5-gram** model trained on the same data (WikiBooks) and on GigaWord. On each curve, we mark the point at which the accuracy is most similar to GPT2$_{tiny}$, and additionally indicate the corresponding overall average accuracy of the reference models.

### 3.7 Discussion

We find that the order of learning is surprisingly stable across architectures, model sizes and training sets. Therefore, given a new NLM, the order

in which it will learn linguistic phenomena can be predicted by another model that achieves a similar average accuracy. When considering non-neural LMs, this observation does not always hold: inherently different architectures (such as 5-grams) have very different trajectories. Hence, future models with very different induced biases may present different orders.

## 4 Phases of Learning

Having established that different NLMs learn in a consistent order, we investigate the emerging learning trajectory by comparing it with simpler reference models. Our goal is to identify distinct learning phases that characterize NLM's training.

**Setup.** We compare GPT2$_{tiny}$ to fully trained LMs (same as §3.3), as well as to a variety of metrics. For each metric $m$ we compute the average score over each example for each of the 67 sets $\mathbb{E}_{p_i \in p}\left[m\left(p_i\right)\right] \in \mathbb{R}^{67}$. The results are replicated with GPT2$_{small}$ and TransformerXL and lead to similar conclusions (see App. §E).

**Sentence-level Metrics.** First, we consider two sentence-level metrics: sentence length (in tokens) and syntactic depth. Assuming a sentence parse tree, the depth is the longest path from a word to the root. Sentence length is often considered to be a source of challenge for infants (Brown, 1973) and networks (Neishi and Yoshinaga, 2019), regardless of the sentence's complexity. Syntactic depth (Yngve, 1960) is a measure used to assess how cognitively complex a sentence is. We leave the question of which measure of linguistic complexity (Szmrecsányi, 2004) correlates best with the trajectory exhibited by NLMs to future work.

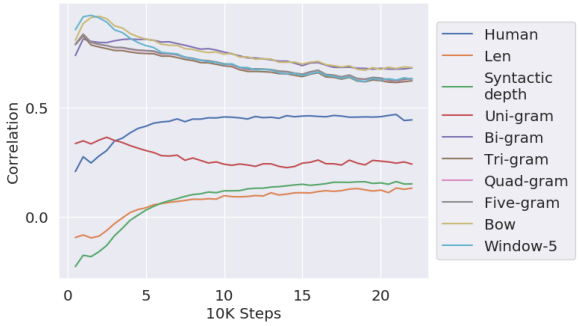



Figure 5: Correlation between the performance vectors of different **metrics** and **models** against the vector of GPT2$_{tiny}$ at different stages of learning.

Our results (Fig. 5) show that neither sentence-level metric (length and syntactic depth) can predict well what is difficult for the model. This is not surprising, as both measures only capture sentence complexity at a general level, and are not directly related to the linguistic phenomenon that is being tested. We do see that the syntactic depth starts off as a worse predictor of the NLM performance and ends as a better one. We provide a different perspective on this initial learning phase, before and after that switch, later in this section.

Next, we compare the performance vector with task difficulty for humans, as reported in the original BLIMP paper. We observe that correlation is fairly high after a sufficient number of steps. In fact, the network becomes more similar to humans as it improves: at the beginning, the network relies on different features than humans, but with time more of the hurdles are shared. However, correlation saturates at a mid-range correlation of under 0.5. This suggests that the network (partially) relies on features that are not used by human annotators. These may be valid generalizations not tested by BLIMP, or erroneous ones that are still beneficial to reduce the score on the task it was trained on (cf. McCoy et al., 2019). We revisit this issue in §5.

**Comparison with Limited Context and Locality.** Our methodology opens the door to examine other potential biases of LMs. We now do so, starting with context and locality.

We consider models that take into account different scopes of context: unigram, and 2-5 gram LMs that can exploit the order of preceding words. We argue that the correlation between NLMs and $n$-gram LMs may indicate that features based on limited context are also employed by NLMs.

Surprisingly, the unigram model, which doesn't use context, perfectly classifies 7 phenomena, achieves 98.1% accuracy on 1, and completely fails (0% accuracy) on 8. This suggests that high accuracy on some syntactic and semantic challenges (as defined by BLIMP) can be achieved by simple heuristics. Note, however, that the NLMs we test are not trained towards any specific phenomena and are not fine-tuned in any way. Hence, NLMs can only attain heuristics or biases (generalization errors) which are beneficial in general, not ones specific to our test challenges.

While NLMs initially present a strong correlation with the unigram model, this correlation quickly drops (see Fig. 5). From the outset, GPT2$_{tiny}$ succeeds on 6 of the 8 phenomena that are classified well by unigrams, and 4 of the 8 that

the unigram model utterly fails on. Interestingly, for 3 of the other phenomena on which the unigram failed, GPT2$_{tiny}$ initially achieves 0% accuracy (chance level is 50%), but its accuracy does climb during training (e.g., see App. §A). We conclude that, as expected, the NLM acquires a bias towards predicting frequent words early in training, but that this bias is weighed in against other (contextual) considerations later on in training.

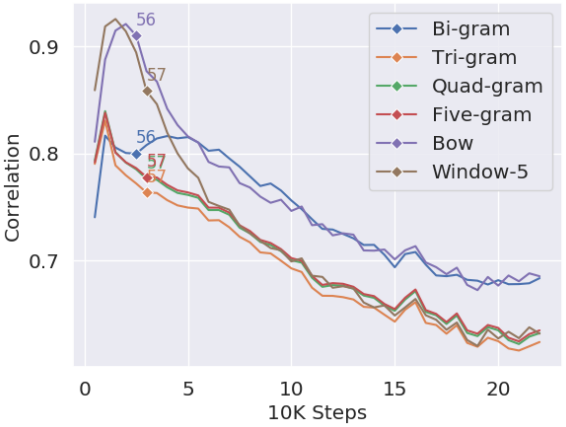



Figure 6: Correlation between the performance vectors of GPT2$_{tiny}$ throughout learning with simple LMs. The figure focuses on LMs found also on Fig. 5.

Comparing different scopes of context, our results (Fig. 6 and App. §E) show that throughout training, the network presents high correlation with $n$-gram models. From a certain point onward, the network becomes more similar to the bi-gram model than to the other $n$-gram LMs. We also note that similarity peaks early on, but with time the correlation decreases. This may suggest that initially, the NLMs acquire grammatical behavior that resembles a Markov model, or even a bi-gram model. Only later does the network rely more on global features. This is in line with our earlier findings, which show an increasing correlation with syntactic depth as compared to sentence length.

At the very beginning, NLMs often generate one word repetitions (e.g., "the" Fu et al., 2020). This seems to be at odds with our finding that grammar learning already begins at this early stage. However, while frequency may dictate the most probable predictions, comparing two options that differ only slightly may prove to depend more on context, as our results indicate.

**Limited Context and Word Order.** By comparing NLMs to $n$-grams, we examined the effect of context within a fixed window size. Now we examine the effect of word order, within a window and in general. To this end, we create two ablated GPT2$_{tiny}$ models. BOW is agnostic to the order between preceding tokens, while Window-5 is similar but relies only on 5 tokens (details in §2).

Our results suggest that initially, the identity of the preceding words is more important than their order. Both BOW and Window-5 better correlate with our NLM than the $n$-gram models. Later on, this trend reverses and the $n$-grams, that do exploit word order, become better correlated. Furthermore, the correlation with Window-5 is significantly smaller than with BOW at later stages of learning, suggesting that the network gradually learns to rely on more context (cf. Saphra and Lopez, 2019).

## 5 Classifying the Learning Trajectories

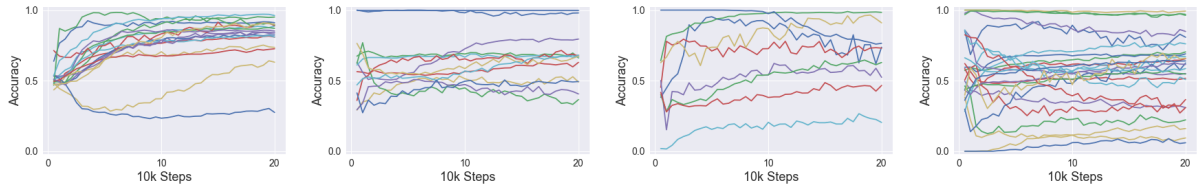
To understand the latent features learned by NLMs, we categorize linguistic phenomena through the lens of their learning trajectories. We ask whether linguistically similar phenomena are learned in a similar fashion, and whether what is learned similarly is defined by linguistic terms.

We inspect linguistic categories by comparing the learning trajectories of their phenomena. In the Morphology field, we find that they display similar gradual curves, ultimately reaching high performance (median accuracy 0.85, see Fig. 7a). This may indicate that some latent features learned are morphological, and affect performance on almost all 'Morphology' phenomena.

Syntax-semantics phenomena also present unique behavior: their scores plateau near chance performance (see Fig 7b), suggesting that the learned features are insufficient to correctly represent phenomena in this field. The other fields, "semantics" and "syntax" (Figs 7c,7d), do not present prototypical learning curves, suggesting that they are too broad to correspond to a single learning pattern. This, in turn, may suggest that they do not all correspond to a well-defined set of latent features.

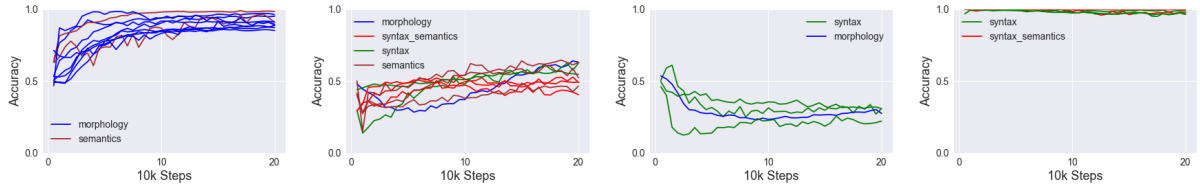
Next, we follow the reverse direction and cluster the learning curves of GPT2$_{tiny}$. We use spectral clustering with 10 clusters and sklearn default parameters, by projecting the learning curves into a normalized Laplacian and applying k-means. Intuitively, learning curves with similar values along the principal directions, are clustered together. Other clustering methods show similar results.

The clusters (Fig. 8 and App. §H) reflect several




(a) Morphology (b) Syntax-Semantics (c) Semantics (d) Syntax

Figure 7: Morphology and Syntax-Semantics (left) characterize NLM learning well, while semantics and syntactic phenomena show little similarity (between lines). Learning curves of GPT2$_{tiny}$ per challenge (lines), clustered according to different fields (graphs) and colored by super-phenomena.




(a) (b) (c) (d)

Figure 8: Some phenomena are learned, others (c) deteriorate, implying the network (that learns language modelling, not phenomena) learns orthogonal features. Learning curves of GPT2$_{tiny}$ on BLIMP challenges, obtained by spectral clustering and colored by fields.

learning profiles, some more expected than others. For some, accuracy improves as learning progresses (see Fig. 8a). Some are barely learned, and accuracy remains at near-chance level (see Fig. 8b). Perhaps more surprisingly, some clusters deteriorate, and accuracy drops to nearly 0 as learning progresses (see Fig. 8c). Notably, some challenges are quite easy – NLMs instantly reach perfect accuracy (see Fig. 8d), while some are confusing – NLMs performance is worse than chance (see Fig. 8c). In the latter cases, the NLMs presumably learn unrelated, harmful generalizations.

When inspecting the emerging clusters, many (but not all, see Fig. 8b) contain a shared prominent field, but often varied super-phenomena (see Fig. 8a). Thus, while the categorization in BLIMP reflects a common linguistic organization of grammatical phenomena, from the perspective of learning trajectories only few of the super-phenomena in BLIMP show consistent behavior. We cautiously conclude that there is some discrepancy between the common linguistic categorization of grammatical phenomena and the categorization induced by the learning trajectories of NLMs. An interesting direction for future work would therefore be the development of a theory that can account for the patterns presented by NLMs' learning trajectories.

We manually inspect a few phenomena with strong initial performance that then deteriorates. We find that some of these challenges are solvable by a simple rule, easily learnable by an $n$-gram model. For example, in "principle A case 1", always preferring subjective pronouns (e.g., "she" or "he") over reflexive ones (e.g., "himself", "herself") is sufficient to obtain a perfect score, and preferring "not ever" over "probably/fortunately ever" solves "sentential negation NPI licensor present". The fact that NLM performance deteriorates, fits our finding that nascent NLMs resemble an $n$-gram model.

## 6 Related Work

Characterizing what networks learn is a long-standing challenge. Recently, studies suggested methods to analyze trained models such as probing (Tenney et al., 2019; Slobodkin et al., 2021), analyzing attention heads (Voita et al., 2019; Abnar and Zuidema, 2020) and neurons (finding also correlations across epochs; Bau et al., 2018) and assessing the extent to which LMs represent syntax (van Schijndel et al., 2019). Other works compare outputs, like us, to assess network generalizations (Choshen and Abend, 2019; Ontan'on et al., 2021), look for systematic biases (Choshen and Abend, 2018; Stanovsky et al., 2019) or evaluate characteristics of outputs (Gehrmann et al., 2021; Choshen et al., 2020). McCoy et al. (2020) fine-tuned BERT and tested generalizations on the adversarial dataset HANS (McCoy et al., 2019), finding models to make inconsistent generalizations. Their results differ from ours, but so is their setup, which in-

volves fine-tuning for inference.

Characterizing the features learned by networks according to the order in which examples and phenomena are learned is a relatively new topic. Recently, Hacohen et al. (2020); Hacohen and Weinshall (2021); Pliushch et al. (2021) showed that classifiers learn to label examples in the same order. While their focus was on computer vision, it provided motivation for this work. Other studies use learning dynamics as a tool, rather than a topic of study. They choose training examples (Toneva et al., 2018), categorize examples (Swayamdipta et al., 2020) or characterize the loss-space (Xing et al., 2018). Little research on NLM learning dynamics and generalization types was previously conducted.

Perhaps the closest to this work is Saphra and Lopez (2019), which compared LSTM representations with 3 types of linguistic tagger outputs, finding that correlation is low and that later in training, more context is used. The latter is reminiscent of our findings in §4.

In parallel work, Liu et al. (2021) probe models during training. They show that, early in training, information required for linguistic classifications is found somewhere in the layers of the model. Our work supports their findings by showing that grammar learning experiments conducted with one model are likely to replicate on another. Our methodology differs from theirs in requiring the information the model learnt to manifest itself in behavior rather than to be extractable with a dedicated classifier.

Studying the trajectories of language learning is a mostly untapped area in NLP, but is a long-established field of research in linguistics and psychology. Such lines of research study topics such as acquisition of phonemes (Kuhl et al., 1992), morphology (Marcus et al., 1992), complex constructions (Gropen et al., 1991; Qing-mei, 2007) and innate learning abilities (Tomasello, 2003). Considerable computational work was also done on constructing models that present similar learning trajectories to those of infants (McClelland and Rumelhart, 1981; Perfors et al., 2010; Abend et al., 2017, among many others).

Our work suggests that the generalizations NLMs make are coupled with the bottom-line performance. This gives a new angle and opens avenues of research when combined with previous work about bottom-line performance. For example, the bottom-line performance of small models could predict the performance of larger models (Ivgi et al., 2022). In such cases, the type of generalizations made might also be predicted from the smaller models.

Our work is also closely related to fields such as curriculum learning (Bengio et al., 2009; Hacohen and Weinshall, 2019), self-paced learning (Kumar et al., 2010; Tullis and Benjamin, 2011), hard data mining (Fu and Menzies, 2017), and active learning (Krogh and Vedelsby, 1994; Hacohen et al., 2022; Ein-Dor et al., 2020). In these fields, the order in which data should be presented to the learner is investigated. On the other hand, in our work, we study the order of the data in which the learner is learning – which may shed some light on the advancement of such fields.

## 7 Summary and Conclusions

We showed that NLMs learn English grammatical phenomena in a consistent order, and subsequently investigated the emerging trajectory. Our findings suggest that NLMs present consistent and informative trends. This finding suggests a path for studying NLMs' acquired behavior through their learning dynamics, as a useful complementary perspective to the study of final representations.

Future work will consider the impact of additional factors, architectures and learning phases that appear only later in training. We hope that this work will increase the affinity between the knowledge and methodologies employed in developmental studies, and those used for studying NLMs. Our goal is to obtain a better understanding of what makes linguistic generalization complex or simple to learn, for both humans and NLMs.

## Acknowledgments

We thank Prof. Inbal Arnon for her helpful discussions. This work was supported in part by the Israel Science Foundation (grant no. 2424/21), by a grant from the Israeli Ministry of Science and Technology, and by the Gatsby Charitable Foundations.

## References


Omri Abend, T. Kwiatkowski, N. Smith, S. Goldwater, and Mark Steedman. 2017. Bootstrapping language acquisition. *Cognition*, 164:116–143.

Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, Online. Association for Computational Linguistics.

Loïc Barrault, Ondrej Bojar, Marta R. Costa-jussà, C. Federmann, M. Fishel, Y. Graham, B. Haddow, Matthias Huck, Philipp Koehn, S. Malmasi, Christof Monz, M. Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (wmt19). In *WMT*.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2018. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Roger Brown. 1973. *A first language: The early stages.* Harvard U. Press.

Leshem Choshen and Omri Abend. 2018. Inherent biases in reference-based evaluation for grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 632–642, Melbourne, Australia. Association for Computational Linguistics.

Leshem Choshen and Omri Abend. 2019. Automatically extracting challenge sets for non-local phenomena in neural machine translation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 291–303, Hong Kong, China. Association for Computational Linguistics.

Leshem Choshen, Dmitry Nikolaev, Yevgeni Berzak, and Omri Abend. 2020. Classifying syntactic errors in learner language. In *CONLL*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Liat Ein-Dor, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim. 2020. Active Learning for BERT: An Empirical Study. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7949–7962, Online. Association for Computational Linguistics.

J. Fleiss, J. Cohen, and B. Everitt. 1969. Large sample standard errors of kappa and weighted kappa. *Psychological Bulletin*, 72:323–327.

Wei Fu and Tim Menzies. 2017. Easy over hard: A case study on deep learning. In *Proceedings of the 2017 11th joint meeting on foundations of software engineering*, pages 49–60.

Zihao Fu, Wai Lam, Anthony Man-Cho So, and Bei Shi. 2020. A theoretical analysis of the repetition problem in text generation. *arXiv preprint arXiv:2012.14660*.

Sebastian Gehrmann, Tosin P. Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Aremu Anuoluwapo, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna Clinciu, Dipanjan Das, Kaustubh D. Dhole, Wanyu Du, Esin Durmus, Ondrej Dusek, Chris C. Emezue, Varun Gangal, Cristina Garbacea, Tatsunori B. Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Rubungo Andre Niyongabo, Salomey Osei, Ankur P. Parikh, Laura Perez-Beltrachini, Niranjan Rao, Vikas Raunak, Juan Diego Rodríguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The gem benchmark: Natural language generation, its evaluation and metrics. *ArXiv*, abs/2102.01672.

Aaron Gokaslan and Vanya Cohen. 2019. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

J. Gropen, S. Pinker, M. Hollander, and R. Goldberg. 1991. Affectedness and direct objects: The role of lexical semantics in the acquisition of verb argument structure. *Cognition*, 41:153–195.

Guy Hacohen, Leshem Choshen, and D. Weinshall. 2020. Let's agree to agree: Neural networks share classification order on real datasets. *International Conference of Machine Learning*.

Guy Hacohen, Avihu Dekel, and Daphna Weinshall. 2022. Active learning on a budget: Opposite strategies suit high and low budgets. *arXiv preprint arXiv:2202.02794*.

Guy Hacohen and Daphna Weinshall. 2019. On the power of curriculum learning in training deep networks. In *International Conference on Machine Learning*, pages 2535–2544. PMLR.

Guy Hacohen and Daphna Weinshall. 2021. Principal components bias in deep neural networks. *arXiv preprint arXiv:2105.05553*.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *WMT@EMNLP*.

David Ingram. 1989. *First language acquisition: Method, description and explanation*. Cambridge university press.

Maor Ivgi, Yair Carmon, and Jonathan Berant. 2022. Scaling laws under the microscope: Predicting transformer performance from small scale experiments. *ArXiv*, abs/2202.06387.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Christo Kirov and Ryan Cotterell. 2018. Recurrent neural networks in linguistic theory: Revisiting pinker and prince (1988) and the past tense debate. *Transactions of the Association for Computational Linguistics*, 6:651–665.

Anders Krogh and Jesper Vedelsby. 1994. Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7.

Stan A Kuczaj II. 1977. The acquisition of regular and irregular past tense forms. *Journal of Verbal Learning and Verbal Behavior*, 16(5):589–600.

Patricia K Kuhl, Karen A Williams, Francisco Lacerda, Kenneth N Stevens, and Björn Lindblom. 1992. Linguistic experience alters phonetic perception in infants by 6 months of age. *Science*, 255(5044):606–608.

M Kumar, Benjamin Packer, and Daphne Koller. 2010. Self-paced learning for latent variable models. *Advances in neural information processing systems*, 23.

P. Lison and J. Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *LREC*.

L. Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A. Smith. 2021. Probing across time: What does roberta know and when? In *EMNLP*.

G. Marcus, S. Pinker, M. Ullman, M. Hollander, T. Rosen, and F. Xu. 1992. Overregularization in language acquisition. *Monographs of the Society for Research in Child Development*, 57 4:1–182.

James L McClelland and David E Rumelhart. 1981. An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychological review*, 88(5):375.

R. Thomas McCoy, Junghyun Min, and Tal Linzen. 2020. Berts of a feather do not generalize together: Large variability in generalization across models with similar test set performance. *ArXiv*, abs/1911.02969.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Courtney Napoles, Matthew R Gormley, and Benjamin Van Durme. 2012. Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 95–100.

Masato Neishi and Naoki Yoshinaga. 2019. On the relation between position information and sentence length in neural machine translation. In *CoNLL*.

Santiago Ontan'on, Joshua Ainslie, Vaclav Cvicek, and Zachary Kenneth Fisher. 2021. Making transformers solve compositional tasks. *ArXiv*, abs/2108.04378.

Amy Perfors, Joshua B. Tenenbaum, and Elizabeth Wonnacott. 2010. Variability, negative evidence, and the acquisition of verb argument constructions. *Journal of Child Language*, 37(3):607–642.

Steven Pinker. 1995. Language acquisition. *Language: An invitation to cognitive science*, 1:135–82.

Steven Pinker and Alan Prince. 1988. On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition*, 28(1):73 – 193.

Iuliia Pliushch, Martin Mundt, Nicolas Lupp, and Visvanathan Ramesh. 2021. When deep classifiers agree: Analyzing correlations between learning order and image statistics. *ArXiv*, abs/2105.08997.

Ren Qing-mei. 2007. The cognitive and psychological interpretation of construction acquisition: A survey. *Journal of foreign languages*.

A. Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

David E. Rumelhart and James L. McClelland. 1986. On learning the past tenses of english verbs. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, 2:216–271.

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.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia.

Aviv Slobodkin, Leshem Choshen, and Omri Abend. 2021. Mediators in determining what processing BERT performs first. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 86–93, Online. Association for Computational Linguistics.

Gabriel Stanovsky, Noah A. Smith, and Luke Zettlemoyer. 2019. Evaluating gender bias in machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1679–1684, Florence, Italy. Association for Computational Linguistics.

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.

Benedikt Szmrecsányi. 2004. On operationalizing syntactic complexity. In *Le poids des mots. Proceedings of the 7th international conference on textual data statistical analysis. Louvain-la-Neuve*, volume 2, pages 1032–1039.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

M. Tomasello. 2003. Constructing a language: A usage-based theory of language acquisition.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. 2018. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*.

Jonathan G Tullis and Aaron S Benjamin. 2011. On the effectiveness of self-paced learning. *Journal of memory and language*, 64(2):109–118.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: On the importance of pre-training compact models. *arXiv: Computation and Language*.

Marten van Schijndel, Aaron Mueller, and Tal Linzen. 2019. Quantity doesn't buy quality syntax with neural language models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5835–5841.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *arXiv preprint arXiv:1905.09418*.

Yu-An Wang and Yun-Nung Chen. 2020. What do position embeddings learn? an empirical study of pre-trained language model positional encoding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6840–6849, Online. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2019. Blimp: The benchmark of linguistic minimal pairs for english. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Adina Williams, Andrew Drozdov, and Samuel R. Bowman. 2018. Do latent tree learning models identify meaningful structure in sentences? *Transactions of the Association for Computational Linguistics*, 6:253–267.

Chen Xing, Devansh Arpit, Christos Tsirigotis, and Yoshua Bengio. 2018. A walk with sgd. *arXiv preprint arXiv:1802.08770*.

Victor H Yngve. 1960. A model and an hypothesis for language structure. *Proceedings of the American philosophical society*, 104(5):444–466.

Y. Zhu, Ryan Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

## A Per challenge Graphs

We include behaviours of each model trained over the main dataset used (Wikipedia and books) on each BLIMP challenge by perplexity. In general, accuracy is similar despite different initialization and size of the GPT2 models. TransformerXL shows a similar trend, despite the uncomparable Perplexity. We supply several examples here and leave the rest to the data accompanying this paper.

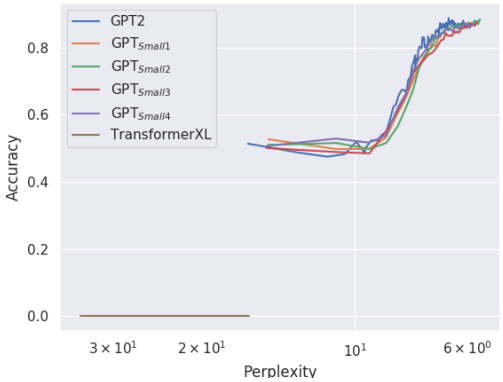



Figure 9: The accuracy on determiner noun agreement during training. Accuracy is similar despite different initialization and size of the GPT2 models. TransformerXL shows a similar trend, despite the uncomparable Perplexity.

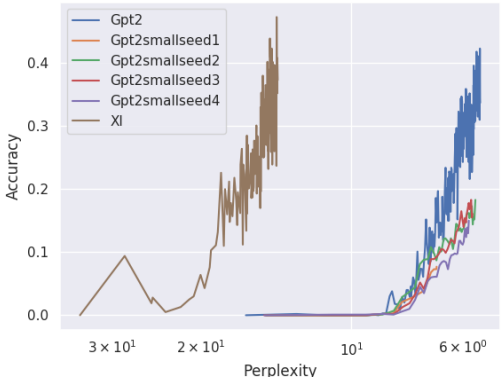



Figure 10: The accuracy on wh vs that with gap during training.

## B Details on experimental settings

We include further settings to ensure reproduciblity of the results. Parameters shared by all the trained NLMs include $32K$ tokens in the vocabulary, $5 \cdot 10^{-5}$ learning rate, max gradient norm of 1, Adam optimizer (Kingma and Ba, 2015), and $10K$ warm-up steps. TransformerXl vocabulary is kept to its

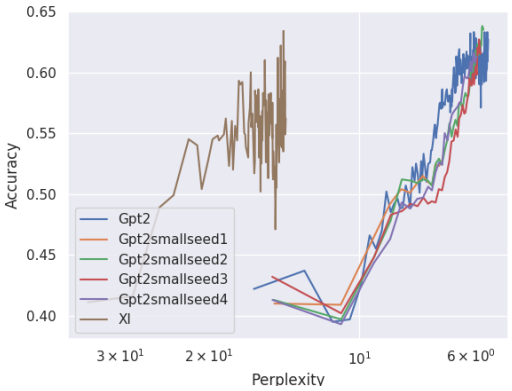



Figure 11: The accuracy on causative during training.

default. All other parameters, including GPT2$_{small}$ size parameters, are the defaults according to the HuggingFace transformers library.

Our 2-5 grams are KenLM (Heafield, 2011) trained on WikiBooks. A second 5-gram model trained on GigaWord corpus (Graff et al., 2003), as reported by BLIMP. The Uni-gram LM is defined according to the frequency of a word in WikiBooks. Sentence probability is normalized by the number of words, which is helpful for the rare cases where the minimal pairs are of different lengths.

## C Correlation during training

We see that tendencies during training are not only similar between instances of the same architecture but also between different architectures. On comparable stages of learning, the GPT2$_{tiny}$ and GPT2$_{small}$ correlate well (>0.9) with respect to their performance vectors. We present the correlations of GPT2$_{tiny}$ compared to GPT2$_{small}$ in Fig. 12. We find the two learn in a similar order throughout their training.

We manually compare the results to TransformerXL. Qualitatively, observing the trajectories per challenge (Trajectories are found in Supp. §A and the supplied data) of TransformerXL, it seems to share the general tendencies of the GPT2 architectures. However, reaching a lower stage of training, it never improves on some challenges (e.g., determiner-noun agreement).

## D Models are consistent on per example level

We compute the binary score of every example by each model. We reframe the question as an annotator agreement problem and ask whether the models agree on the right answer for each example.

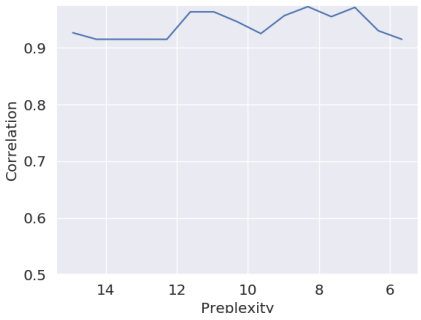



Figure 12: Correlation between the performance vectors of GPT2$_{tiny}$ and GPT2$_{small}$, aligned by perplexity.

Framed this way, the methodology is clear. We compute Fleiss kappa (Fleiss et al., 1969) and find the per example correlation. The full results per step and challenge are added as a supplemental file. The average overall kappa is 0.83, models not only agree on the order of learning phenomena but also on the order of learning examples within each per-phenomenon (if learnt at all). While there are phenomena with lower and higher agreement, there are only two phenomena in the range of 0.5-0.6 agreement. Meaning even the most different ones have high example correlation and there is little variance between models to explain.

Our main aim in this work is to compare models acquisition. However, we see the per example order of acquisition as less informative, unless we can cluster or name the examples learnt. The reason to choose the phenomena was to extract such names, and we hence focus in our work on them.

Note, that consistency per example was shown before in the scope of computer vision (Hacohen et al., 2020). However, a critical difference is that they deal with classification and check whether which examples are learnt first. We however, aim to ask about generalization, given that you learn one task (language modelling), what type of generalizations do you make, tested on another. For example, while learning to predict the next word, the network understands after X steps that the verb should be in agreement with the subject.

## E Reproducing with other models

We provide the GPT2$_{small}$ correlation with other models and with various metrics and models in Fig. 13 and 14 respectively. We also supply the average BLIMP accuracies of the models we trained in Fig. 15.

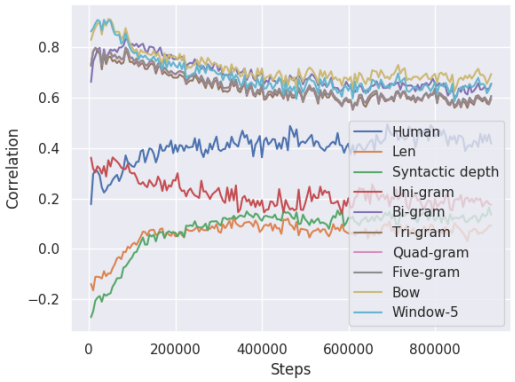



Figure 13: Correlation between the difficulty of GPT2 and of other models for each phenomena in each training step.

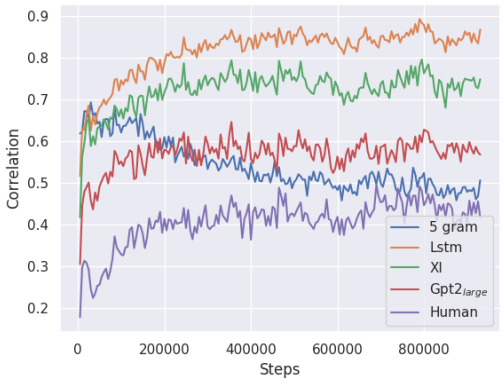



Figure 14: Correlation between the difficulty predicted by BLIMP models and the difficulties for the model for each phenomena in each training step.

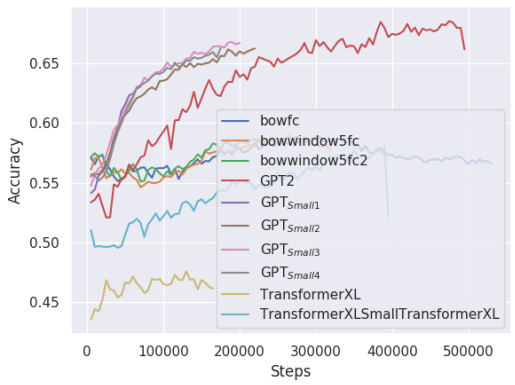



Figure 15: Overall BLIMP accuracy by step.

### E.1 Results mainly replicate in TransformerXL

We replicate the same experiment over the training of the TransformerXL instance. The TransformerXL seems to reach a lower stage of learning, probably due to the vast vocabulary and model.

The model replicates some of the general notions seen on GPT2$_{small}$. It correlates most with simpler models, then with humans and then with global features. At first, sentence length makes a sentence more challenging than its actual structure, 5 window BOW starts as more relevant than BOW over all the sentence.

We do see that the overall graph is quite straight. With that, the increase in correlation with humans is quite small, the BOW models don't drop and the evidence of relying on more abstract knowledge in late stages is less apparent. This might be expected, as we know the network reached an early step on the performance scale.

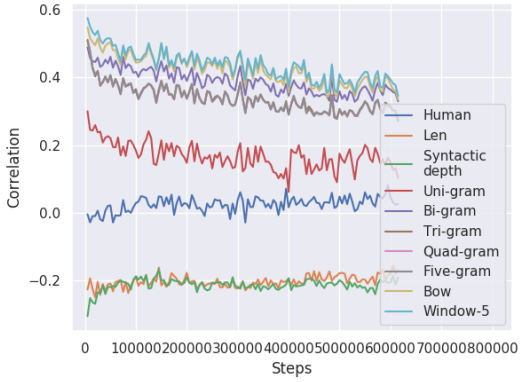



Figure 16: Correlation between the difficulty predicted by metrics and the difficulties for the model for each phenomena in each time step.

## F Reproducing with other data

As comparison to the correlations with our main model, we provide the correlations of GPT2$_{tiny}$ trained on OpenWebText with the two 5-gram models, one on WikiBooks and one on Giga word (Fig. 17). We see that the higher resemblence to WikiBook trained model is kept despite being trained on the same data, but the difference is lower at the beginning and more stable. It might be the case that over reliance on the specific data is shown at those first steps where the difference is large, but it would require further evidence.

We also compare the model to several other trained models in Fig. 18.

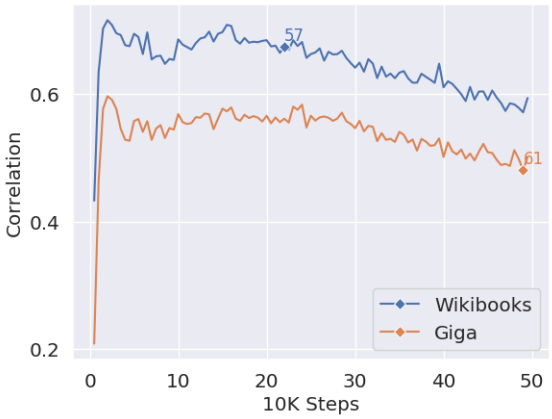



Figure 17: Correlation during training of GPT2$_{tiny}$ on OpenWebText compared to 5-gram model trained on WikiBooks and on GigaWord. Correlation is over BLIMP challenges. Numbers indicate the overall average of the reference models over BLIMP and are found over the step with most similar accuracy on GPT2$_{tiny}$. GPT2$_{tiny}$ best score is 67.

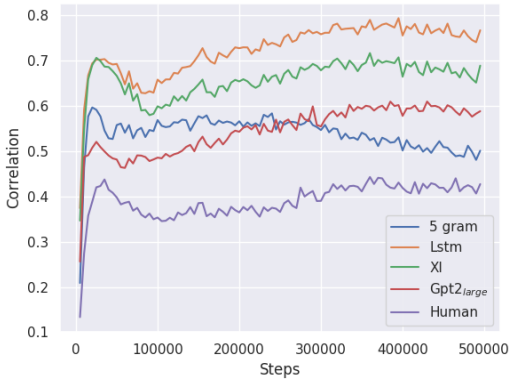



Figure 18: Correlation during training of GPT2$_{tiny}$ trained on OpenWebText data compared to Off-the-shelve models and XL smaller models. The correlation with itself during training is shown in gray. Correlation is over BLIMP challenges. Numbers indicate the overall average of the reference models over BLIMP and are found over the step with most similar accuracy on GPT2$_{tiny}$.

## G 5-grams notes

The gap between the correlation with the two 5-grams decreases during the first 50K steps or so, and then remains constant. This suggests that the choice of a dataset is more important during early NLM training. Because, at the beginning the network learn generalizations which are more common to counts of one (huge, general domain) dataset than another, and this effect diminishes. Possibly, this is because at this point NLMs rely more on word identity, rather than on abstract generalizations, that are shared to a greater extent across corpora (see §4). We observe that the 5-gram trained on WikiBooks correlates better with GPT2$_{tiny}$, even when GPT2$_{tiny}$ is not trained on it (not reported). We cannot offer a simple explanation for this trend.

## H Clustering BLIMP

We include the learning curves of GPT2$_{tiny}$ on BLIMP dataset, clustered according to fields (Fig. 19),, super-phenomena (Fig. 20), and the spectral clustering (Fig. 21). Due to restrictions on appendix files the figures are found in corresponding folders in the supplied data.

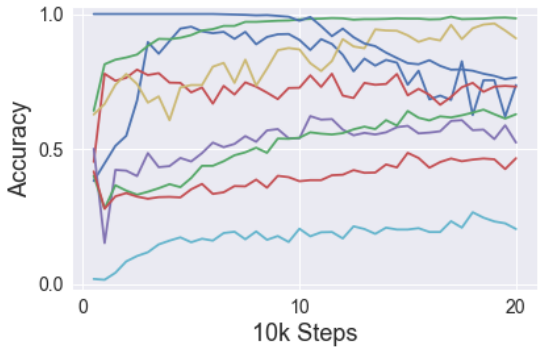



Figure 19: Cluster of semantic phenomena, each line is the trajectory of learning of a phenomenon.

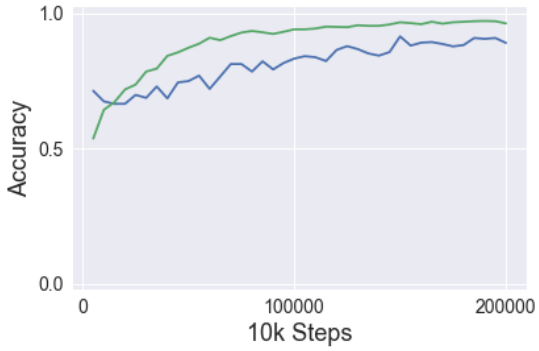



Figure 20: Anaphor agreement super phenomena trajectories.

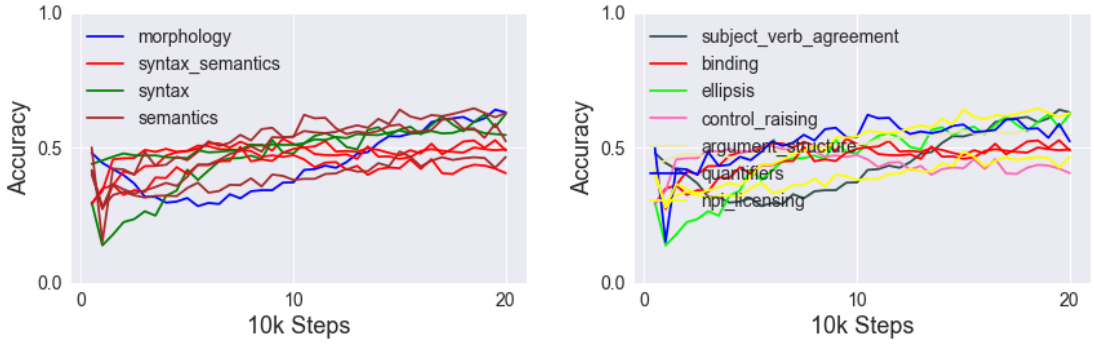



Figure 21: Cluster of phenomebna chosen by spectral clustering. The phenomena behave similarly but do not follow the same linguistic categorizations.