A Syntactic Attention Structure Probe

We use the following simple probe, based on (Clark et al., 2019), to measure syntactic attention structure. First, we define a head-specific probe $f_{h,l}$ that predicts the parent word for a target word x_i from the attention map α in head h and layer l. Denoting $\alpha_{ij}^{(h,l)}$ as the attention weight between words i and j for attention head h in layer l, we define this probe as:

$$f_{h,l}(x_i) \coloneqq \arg\max_{x_j} \left[\max \left(\alpha_{ij}^{(h,l)}, \alpha_{ji}^{(h,l)} \right) \right], \tag{3}$$

In other words, given target word i and attention head h in layer l, $f_{h,l}$ predicts the other word that receives the maximum attention across both directions (e.g., from parent to child and child to parent). Since BERT$_{\text{Base}}$ uses byte-pair tokenization (Sennrich et al., 2016), we convert the token-level attention maps to word-level attention maps. Attention to a word is summed over its constituent tokens, and attention from a word is averaged over its tokens, as in Clark et al. (2019).

However, this probe is head-specific. To then acquire a parent predictor for a given dependency relation, we select the best-performing head (as determined by accuracy of the predicted parent words when compared against silver labels) for each relation. Denoting the set of all dependency relations as $\mathcal{R}$ and each relation $R \in \mathcal{R}$ as the set of all ordered word pairs (x, y) that have that relation (with y being the parent of x), the best-performing probe for each relation R is:

$$\hat{h}_R = \arg\max_{f_{h,l}} \frac{1}{|R|} \sum_{x \in \mathcal{D}} \sum_{(x_i, x_j) \in R} \mathbb{1}_R \Big((x_i, f_{h,l}(x_i)) \Big) \tag{4}$$

where $\mathcal{D}$ is the dataset, x_i and x_j represent the i-th and j-th words in example x, and $\mathbb{1}_R$ is the indicator function for set R—i.e., $\mathbb{1}_R\left((x_i, f_{h,l}(x_i)\right) = 1$ if $(x_i, f_{h,l}(x_i)) \in R$ (which occurs only when $f_{h,l}(x_i) = x_j$ since each word can only have one parent) and 0 otherwise. Lastly, we use these relation-specific probes to compute the overall accuracy across all dependency relations. The resulting accuracy is known as the **Unlabeled Attachment Score** (UAS):

$$\text{UAS} \coloneqq \frac{1}{\sum_{R \in \mathcal{R}} |R|} \sum_{R \in \mathcal{R}} \sum_{x \in \mathcal{D}} \sum_{(x_i, x_j) \in R} \mathbb{1}_R \left((x, \hat{h}_R(x)) \right) \tag{5}$$

We compute UAS on a random sample of 1000 documents from the Wall Street Journal portion of the Penn Treebank (Marcus et al., 1999) and use Stanford Dependencies (Schuster & Manning, 2016) parses as our silver labels of which word pairs (x_i, x_j) correspond to each dependency relation $R \in \mathcal{R}$.

B Syntactic Regularizer

We add a **syntactic regularizer** that manipulates the structure of the attention distributions. The regularizer adds a syntacticity score $\gamma(x_i, x_j)$ that is equal to the maximum attention weight (summed across the forward and backward directions) between words i and j, for all pairs (i, j) where there exists some dependency relation between i and j. Because heads tend to specialize in particular syntactic relations, we compute this maximum over all heads and layers. More precisely,

$$\gamma(x_i, x_j) = \max_{h,l} \alpha_{ij}^{(h,l)} + \max_{h,l} \alpha_{ji}^{(h,l)} \tag{6}$$

We use this regularizer to either penalize or reward higher attention weights on a token's syntactic neighbors by adding it to the MLM loss L_{MLM}, scaled by a constant coefficient λ. We set $\lambda < 0$ and $\lambda > 0$ to promote and suppress syntacticity, respectively. If we denote $\text{parent}(x)$ as the dependency parent of x and $D(x) \coloneqq \{y \mid x = \text{parent}(y)\}$ as the set of all dependents of x, then the entire loss objective is:

$$L(x) = \underbrace{L_{\text{MLM}}(x)}_{\text{Original loss}} + \underbrace{\lambda \sum_{i=1}^{|x|} \sum_{x_j \in D(x_i)} \gamma(x_i, x_j)}_{\text{Syntactic regularization}} \tag{7}$$

C Related Work

Our work links two parallel bodies of literature, one from the interpretability community, focusing on causal methods of interpretation; and the other from the training dynamics community, focusing on phase changes and on the influence of early training. We combine insights from both communities, studying the role of interpretable artifacts on model generalization by measuring both variables while intervening on training.

C.1 Simplicity Bias and Phase Changes

Models tend to learn simpler functions earlier in training (Hermann & Lampinen, 2020; Shah et al., 2020; Nakkiran et al., 2019; Valle-Pérez et al., 2019; Arpit et al., 2017). In LMs, Choshen et al. (2022) identify a trend in BLiMP (Warstadt et al., 2020a) grammatical tests during training: earlier on, LMs behave like n-gram language models, but later in training they diverge. Likewise, LMs learn early representations that are similar to representations learned for simplified versions of the language modeling task, like part-of-speech prediction (Saphra & Lopez, 2019). Despite gradual increases in complexity, SGD exhibits a bias towards simpler functions and features that are already learned (Pezeshki et al., 2021), so simplistic functions learned early in training can still shape the decisions of a fully trained model. Importantly, a large degree of simplicity bias can be disadvantageous to robustness, calibration, and accuracy (Shah et al., 2020), which inspires our approach of limiting access to interpretable—and thus simplistic, as interpretable behaviors must be simple enough to understand (Lipton, 2018)—solutions early in training.

In studying transitions between simplistic internal heuristics and more complex model behavior, we incorporate findings from the literature that identifies multiple phases during training (Jastrzebski et al., 2020; Shwartz-Ziv & Tishby, 2017a). While often, the performance of language models scales predictably (Kaplan et al., 2020; Srivastava et al., 2022), some tasks instead show breakthrough behavior where a single point in training shows a spike in performance (Srivastava et al., 2022; Wei et al., 2022; Caballero et al., 2023). One computational structure, the induction head, emerges in autoregressive language models at a discrete phase change (Olsson et al., 2022) and is associated with handling longer context sizes and in-context learning. In machine translation, Dankers et al. (2022) find a learning progression in which a Transformer first overgeneralizes the literal interpretations of idioms and then memorizes idiomatic behavior. When the training set makes grammatical rules ambiguous, Murty et al. (2023) show that language modelling eventually leads to phase transitions towards the hierarchical version of a rule over an alternative based on linear sequential order. Outside of NLP, phase changes are observed in the acquisition of concepts in strategy games (Lovering et al., 2022; McGrath et al., 2022) and arithmetic (Liu et al., 2022; Nanda et al., 2023). Our work also identifies a specific phase in MLM training, the SAS phase, and analyzes its role in performance and generalization behavior.

We also observe an alignment between phase changes in representational complexity and in generalization performance (Section 4.1), which parallels the observation of Thilak et al. (2022) that generalization time during grokking (Power et al., 2022) aligns with the timing of cycles of classifier weight growth. Lewkowycz et al. (2020) and Hu et al. (2023) report a similar alignment between classifier weight growth and the early transition when loss first begins to decline. Our results indicate that, in natural settings, such phase changes in complexity happen at intermediate points in training when a model first acquires particular representational strategies.

Finally, although we find the same phase transition occurs across multiple training runs, other work indicates that generalization capabilities are sensitive to random seed (Sellam et al., 2022; Juneja et al., 2023; Jordan, 2023). Furthermore, phase transitions may be primarily an artifact of poor hyperparameter settings (Liu et al., 2023), which lead to unstable optimization (Hu et al., 2023). Therefore, it is possible that these abrupt breakthroughs would vanish under the correct architecture and optimizer settings.

C.2 Interpreting Training

Recent work in interpretability has begun to take advantage of the chronology of training in developing a better understanding of models. In some of the first papers explicitly interpreting the training process, Raghu et al. (2017) and Morcos et al. (2018) use subspace methods to understand model convergence and representational similarity. Adapting their methods, Saphra & Lopez (2019) find that early in training, LSTM language models produce representations similar to other token level tasks, and only begin to model long range context later in training. Liu et al. (2021) use a diverse set of probes to observe that RoBERTa achieves high performance on most linguistic benchmarks early in pre-training, whereas more complex tasks require longer pre-training time.

Some studies find that specific capabilities are often learned in a particular order. In autoregressive language models, Xia et al. (2023) show that training examples tend to be learned in a consistent order independent of model size. In MLMs, Chiang et al. (2020) find that different part of speech tags are learned at different rates, while Warstadt et al. (2020b) find that linguistic inductive biases only emerge late in training. Our work likewise finds that extrinsic grammatical capabilities emerge at a consistent point in training.

While our phase transition results mirror Murty et al. (2022)'s findings that the latent structure of autoregressive language models plateaus in its adherence to formal syntax, their work also finds the structure continues to become more tree-like long after syntacticity plateaus. Their results suggest that continued improvements in performance can still be attributed to interpretable hierarchical latent structure, which may be an inductive bias of some autoregressive model training regimes (Saphra & Lopez, 2020).

Although Appendix I precludes the impact of thresholding effects (Schaeffer et al., 2023; Srivastava et al., 2022) on our results, the relationship between the structure onset and capabilities onset does reflect a dependency pattern similar to the checkmate-in-one task, which Srivastava et al. (2022) consider to be precipitated by smooth scaling in the ability to produce valid chess moves. Even in cases where there is no clear dependency between extrinsic capabilities, there may be internal structures like SAS that emerge smoothly, which can be interpreted as progress measures (Barak et al., 2022; Nanda et al., 2023; Merrill et al., 2023).

C.2.1 Interpretation Through Intervention

Claims about model interpretations can be subject to causal tests, typically applied at inference time (Vig et al., 2020; Meng et al., 2023). Although causal *training* interventions are rare, some existing work has used them to support claims about interpretable model behavior. Leavitt & Morcos (2020), for example, control the degree of neuron selectivity during training in order to demonstrate that this transparent behavior was often, in fact, maladaptive. Follow-up work (Ranadive et al., 2023) shows that neuron selectivity is only transiently necessary early in training, implying that selective neurons are ultimately vestigial. Likewise, Olsson et al. (2022) modify the Transformer architecture to mimic induction head circuits, in order to test their claim that induction head formation was responsible for an early phase change. In considering the role of SAS in model performance, we likewise intervene during training to support and suppress this behavior.

Our work closely relates to the literature on critical learning periods (Achille et al., 2018), where biased data samples prevent the acquisition of particular features or other model behaviors early in training, leading to a finding that a model which fails to learn certain features early in training cannot easily acquire them later. While those experiments illustrate different phases by removing early features and damaging performance, our experiments elicit *positive* changes by removing certain early behaviors, in order to promote other strategies. Furthermore, the behaviors we suppress early in training are immediately learned as soon as they are permitted, so these phases would not be considered critical learning periods.

In all the preceding experimental work that infers causal relationships, it is possible that some related factor is also affected by the proposed intervention. Our work must likewise confront the possibility of an entangled factor that responds to our intervention. The standard approach to remedy this issue is to intervene in as *targeted* a way as possible, ensuring minimal entanglement between the targeted factor and other factors. Since our approach specifically targets internal syntax structure, we expect that the causal relationships we infer are a direct result of changes in internal syntax representations, even if these representations are not *exactly* SAS but *strongly associated with* SAS. We also observe that the capabilities onset consistently appears after the SAS onset, even as we adjust the timing of the SAS onset to arbitrary points, which supports the connection between SAS—or a closely related structural pattern—and grammatically capabilities.

D BLiMP IMPLEMENTATION DETAILS

BLiMP (Warstadt et al., 2020a) consists of 67 different challenges of 1000 minimal pairs each, covering a variety of syntactic, semantic, and morphological phenomena. To evaluate, we use MLM scoring from Salazar et al. (2020) to compute the pseudoperplexity (PPPL) of the sentences in each minimal pair, defined in terms of the pseudo-log-likelihood (PLL) score:

$$\mathrm{PPPL}(\mathcal{D}) \coloneqq \exp\left(-\frac{1}{N}\sum_{x\in\mathcal{D}} \mathrm{PLL}(x)\right), \tag{8}$$

where $\mathcal{D}$ is a corpus of text and N is the size of $\mathcal{D}$. Additionally, PLL is defined as

$$\mathrm{PLL}(x) \coloneqq \sum_{i=1}^{|x|} \log P_{MLM}(x_i|x_{\setminus i};\theta), \tag{9}$$

where $P_{MLM}(x_i|x_{\setminus i};\theta)$ is the probability assigned by the model parameterized by θ to token x_i, given only the context x with the i-th token masked out.

The BLiMP accuracy is computed as the proportion of acceptable sentences assigned a higher PLL (or lower PPPL) than the unacceptable alternatives. For example, consider the minimal pair consisting of the sentences "These patients do respect themselves" and "These patients do respect himself," where the former is linguistically acceptable and the latter is not. Suppose $\mathrm{BERT}_{\mathrm{Base}}$ assigns the former sentence an average PLL of -0.8, and the latter an average PLL of -6.0. Then we consider $\mathrm{BERT}_{\mathrm{Base}}$ to be correct in this case, since the average PLL of the acceptable sentence is higher than the PLL of its unacceptable counterpart.

E CORRELATION BETWEEN UAS AND CAPABILITIES

When we consider all 25 MultiBERTs seeds, we can measure the degree to which natural random variation yields a correlation between model quality and implicit parse accuracy (UAS) from SAS. We find tbhat UAS does not correlate with either the MLM test loss ($R^2 = -2821$) or the grammatical capabilities measured by BLiMP ($R^2 = -317$). This complete lack of significant correlation is clear in Fig. 6. Therefore, correlational results do not support the common assumption that SAS leads to grammatical capabilities.

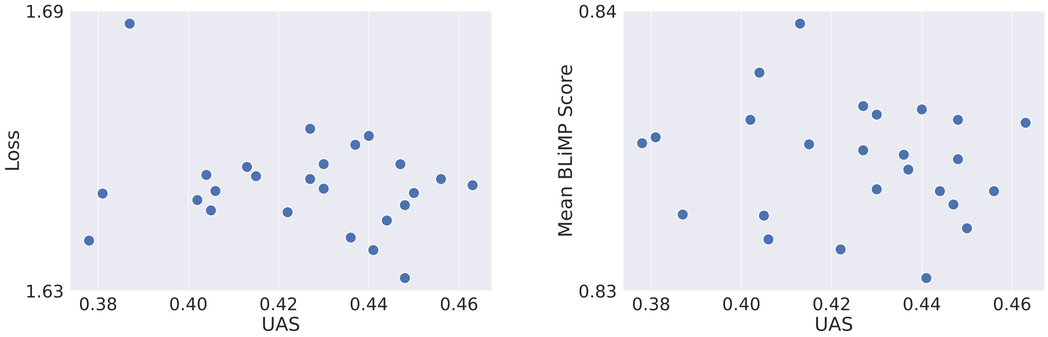


Figure 6: Across 25 MultiBERTs seeds, we do not find a significant correlation between implicit parse accuracy (UAS) and either (a) MLM test loss ($R^2 = -2821$) or (b) grammatical capabilities ($R^2 = -317$).

F MULTIBERTS DEVELOPMENTAL ANALYSIS

MultiBERTs (Sellam et al., 2022) is a public release of 25 BERT-base runs, 5 of which have intermediate checkpoints available. Although the intermediate checkpoints are not granular enough to show the timing of the abrupt spike in implicit parse accuracy, or to show a clear break in the accuracy curve for BLiMP, the results nonetheless align with ours (Fig. 7). UAS clearly shows a sharp initial spike followed by a plateau within 20K timesteps. It appears that the BLiMP increase also occurs within the first 20K steps. The loss drops precipitously initially and more slowly after the 20K step checkpoint, as we would expect from the other metrics.

The timing of the UAS plateau implies a slightly faster timeline for the acquisition of linguistic structure compared to our reproduction, and the loss is slightly lower than ours as well, with a slightly higher BLiMP average. Although MultiBERTs appears to be a closer reproduction of BERT with better results compared to our run, we find that it nonetheless is compatible with the same phase transition.

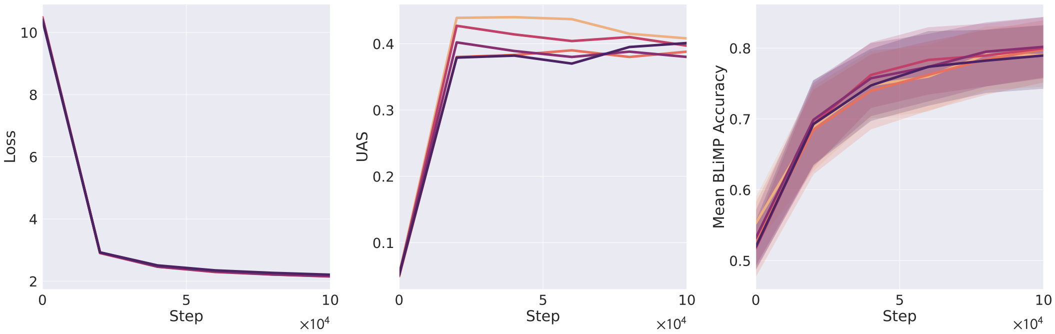


Figure 7: Metrics over the course of training for the 5 MultiBERTs seeds released with intermediate checkpoints. On y-axis: (a) MLM loss (b) Implicit parse accuracy (c) average BLiMP accuracy, with confidence intervals computed across tasks.

G Complexity and Compression

Interpretable behaviors such as SAS, by nature of their understandability, must be simplistic. Coincidentally, models tend to learn simpler functions earlier in training (Hermann & Lampinen, 2020; Shah et al., 2020; Nakkiran et al., 2019; Valle-Pérez et al., 2019; Arpit et al., 2017), a tendency often referred to as *simplicity bias*. However, too much simplicity bias can be harmful (Shah et al., 2020) — although simplistic predictors can be parsimonious, we may also lose out on the predictive power of more complex, nuanced features.

If we view SAS as an example of simplicity bias, then we can also view this phase transition through an information theoretic lens. The Information Bottleneck (IB) theory of deep learning (Shwartz-Ziv & Tishby, 2017b) states that the generalization capabilities of deep neural networks (DNNs) can be understood as a form of representation compression. This theory posits that DNNs achieve generalization by selectively discarding noisy and task-irrelevant information from the input, while preserving key features (Shwartz-Ziv, 2022). Subsequent research has provided generalization bounds that support this theory (Shwartz-Ziv et al., 2018; Kawaguchi et al., 2023). Similar principles have been conjectured to explain the capabilities of language models (Chiang, 2023; Cho, 2023; Sutskever, 2023). Current studies distinguish two phases: an initial *memorization* phase followed by a protracted representation *compression* phase (Shwartz-Ziv & Tishby, 2017b; Ben-Shaul et al., 2023). During memorization, SGD explores the multidimensional space of possible solutions. After interpolating, the system undergoes a phase transition into a diffusion phase, marked by chaotic behavior and a reduced rate of convergence as the network learns to compress information.

To validate this theory in MLM training, we analyze various complexity metrics as proxies for the level of compression (see Fig. 2(a) for TwoNN intrinsic dimension (Facco et al., 2017), and Appendix L.2 for additional complexity/information metrics). Our results largely agree with the IB theory, showing a prevailing trend toward information compression throughout the MLM training process. However, during the acquisition of SAS, a distinct memorization phase emerges. This phase, which begins with the onset of structural complexity, allows the model to expand its capacity for handling new capabilities. A subsequent decline in complexity coincides with the onset of advanced capabilities, thereby confirming the dual-phase nature postulated by the IB theory.

H Is the phase transition caused by abrupt changes in step size?

A possible alternative hypothesis to viewing breakthrough behavior as a conceptual "epiphany" would be that it is an artifact of varying training optimization scales. In other words, there may be some discrete factor in training that causes the optimizer's steps to lengthen, artificially compressing the timescale of learning. The step size decays linearly and the phase transition happens well after warmup ends at 10K steps, meaning that abrupt changes in the hyperparameters are unlikely to be the explanation. However, there may be other factors that affect the magnitude of a step. To confirm that the breakthrough is due to representational structure and not due to a change in the scale of optimization, we consider x-axis scales using a variety of measurements of the progress of optimization. Rather than considering the number of discrete time steps, we consider the following timescales for the weights w_t at timestep t:

Weight magnitude. Fig. 8(a) uses the Euclidian distance from the zero-valued origin, which is equivalent to the weight ℓ_2 norm or $\sqrt{\|w_t\|_2}$.

Distance from initialization. Fig. 8(b) uses the Euclidian distance from the random initialization, i.e., from the weights at timestep 0: $\sqrt{\|w_t - w_0\|_2}$.

Optimization path length. In Fig. 8(c), we approximate the distance traveled during optimization by adding together the lengths of each segment between weight updates, $\sum_{i=1}^{t} \sqrt{\|w_i - w_{i-1}\|_2}$. Because not every timestep is recorded as a checkpoint, we only offer an approximation of the path length by measuring the distance between the recorded sequential checkpoints.

We confirm the phase transitions occur across x-axis scales. Abrupt phase transitions occur whether we consider training timestep, Euclidian distance from initialization, magnitude of the weights, or the path length traveled during optimization.

I Is the phase transition an artifact of thresholding?

Apparent breakthrough capabilities often become linear when measured with continuous metrics instead of discontinuous ones like accuracy (Srivastava et al., 2022; Schaeffer et al., 2023). Is this the case with our accuracy metrics on SAS and BLiMP?

To the contrary, we find that even using a continuous alternative to the accuracy metric shows similar results. In the case of SAS, providing the attention value placed on the correct target, rather than accuracy based on whether attention is highest for that token, gives a continuous alternative to UAS. In the case of BLiMP, we give the relative probability given to the CLS token on the correct answer in the sequence pair, p, compared to the probability $\bar{p}$ of the incorrect CLS token. In other words, the continuous measurement of BLiMP performance is given by $\frac{p}{p+\bar{p}}$. In either case, we see that the phase transition remains clear (Fig. 9).

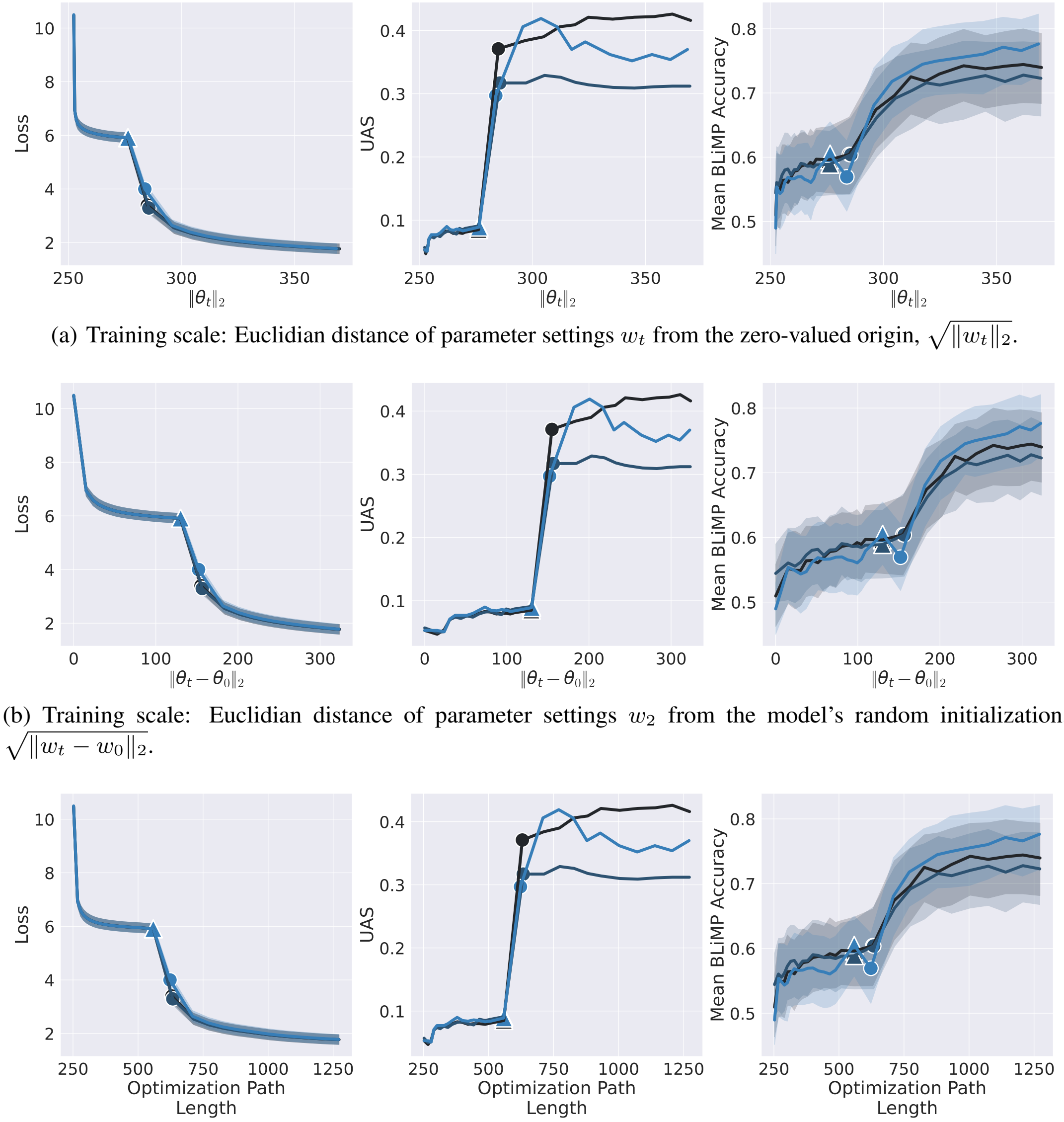
J GLUE Task Analysis

Fig. 10 shows the GLUE task breakdown while training BERT$_{\text{Base}}$. While not all tasks show a breakthrough in accuracy at the structure onset, most do.

Most GLUE tasks, meanwhile, do not show marked improvements after brief early stage suppression of SAS (Fig. 11). The tasks that have more stability across finetuning seeds show a marked decline in performance as we continue to suppress SH past the alternative strategy onset.

K BLiMP Analysis

As seen in Fig. 12, most BLiMP tasks show similar responses to multistage SAS regularization: a dip in accuracy for the models that have their regularizer released at the alternative strategy onset and maximum accuracy for a model where the regularizer is released after brief suppression. The model released at the alternative strategy onset has the poorest performance for all tasks except ellipsis.



(a) Training scale: Euclidian distance of parameter settings w_t from the zero-valued origin, $\sqrt{\|w_t\|_2}$.

(b) Training scale: Euclidian distance of parameter settings w_2 from the model's random initialization, $\sqrt{\|w_t - w_0\|_2}$.

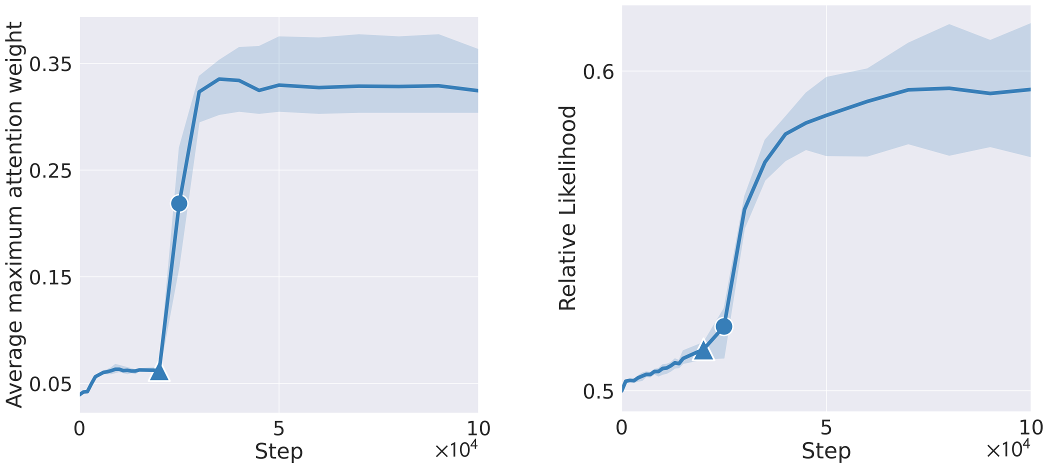
(c) Training scale: Total length of the optimization trajectory after initialization, $\sum_{i=1}^{t} \sqrt{\|w_i - w_{i-1}\|_2}$, with i given only at checkpoint intervals.

Figure 8: Learning trajectories of baseline BERT training, with x-axes reflecting various scales for determining the length of training for checkpoint parameters w_t at time t, as an alternative to counting discrete optimization steps. Each curve represents one of three random seeds. Each y-axis corresponds to, from left to right: MLM loss; implicit parse accuracy; and BLiMP average. Structure onset (▲) and capabilities onset (●) are both marked on each line.

We note that while training BERT$_{\text{Base}}$, for most BLiMP tasks a clear improvement occurs at the capabilities onset, though intriguingly, for some tasks there is a decline in performance at the structure onset (Fig. 13).

L Complexity metrics

The literature on model complexity provides an abundance of metrics which can provide radically different rankings between models (Pimentel et al., 2020). We consider some common complexity metrics during BERT$_{\text{Base}}$ training, primarily focusing on intrinsic dimension.



(a) Averaged maximum attention weight on syntactic neighbors, a continuous alternative to UAS accuracy.

(b) Relative likelihood given to the correct member of a minimal pair, a continuous alternative to BLiMP accuracy.

Figure 9: Continuous, non-thresholded metrics for SAS and BLiMP, across three seeds.

L.1 Intrinsic dimension

In order to measure the complexity of the model and its representations, we use **Two-NN intrinsic dimension** (Facco et al., 2017), with other common complexity metrics in Appendix L.2. Two-NN is a fractal measure of intrinsic dimension (ID) that estimates the ID d by computing the rate at which the number of data points within a neighborhood of radius r grows. If we assume that each of the points in the d-dimensional ball has locally uniform density, then d can be estimated as a function of the cumulative density of the ratio of distances to the two nearest neighbors of each data point. Two-NN has been used to study the ID of neural network data representations, and can sometimes identify geometric properties that are otherwise obscured by linear dimensionality estimates (Ansuini et al., 2019). In our analyses we compute the Two-NN intrinsic dimension on the `[CLS]` embeddings of our trained BERT$_{\text{Base}}$ models, using pair-wise cosine similarity as our distance metric. We also present the dynamics of several complexity metrics such as the empirical Fisher (EF) and weight norm.

L.2 Other complexity metrics

Weight magnitude: The norm of the classifier weights is an often-studied (Thilak et al., 2022; Lewkowycz et al., 2020) metric for model complexity during training. Shown in Fig. 14(a), this metric rises throughout training, with an inflection up at the capabilities onset. Note that this metric is equivalent to an x-axis scale used in Appendix H.

Fisher Information: Inspired by the approach of Achille et al. (2018), we approximate Fisher Information by $\|\nabla L_{MLM}\|_2^2$, the trend for which is shown in Fig. 14(b). Similar to TwoNN, the model experiences a sharp increase in complexity during SAS acquisition, marking the memorization phase. This phase ends abruptly at the capabilities onset, after which a slow decrease in complexity characterizes the compression phase and improved generalization.

M What is the alternative strategy?

Thus far, we refer to the acquisition of some competing opaque behaviors, defining them only as the useful behaviors *not* supported by SAS. We now argue that an alternative strategy is being learned in the absence of SAS, and characterize it as the use of long range semantic content rather than local syntactic structure.

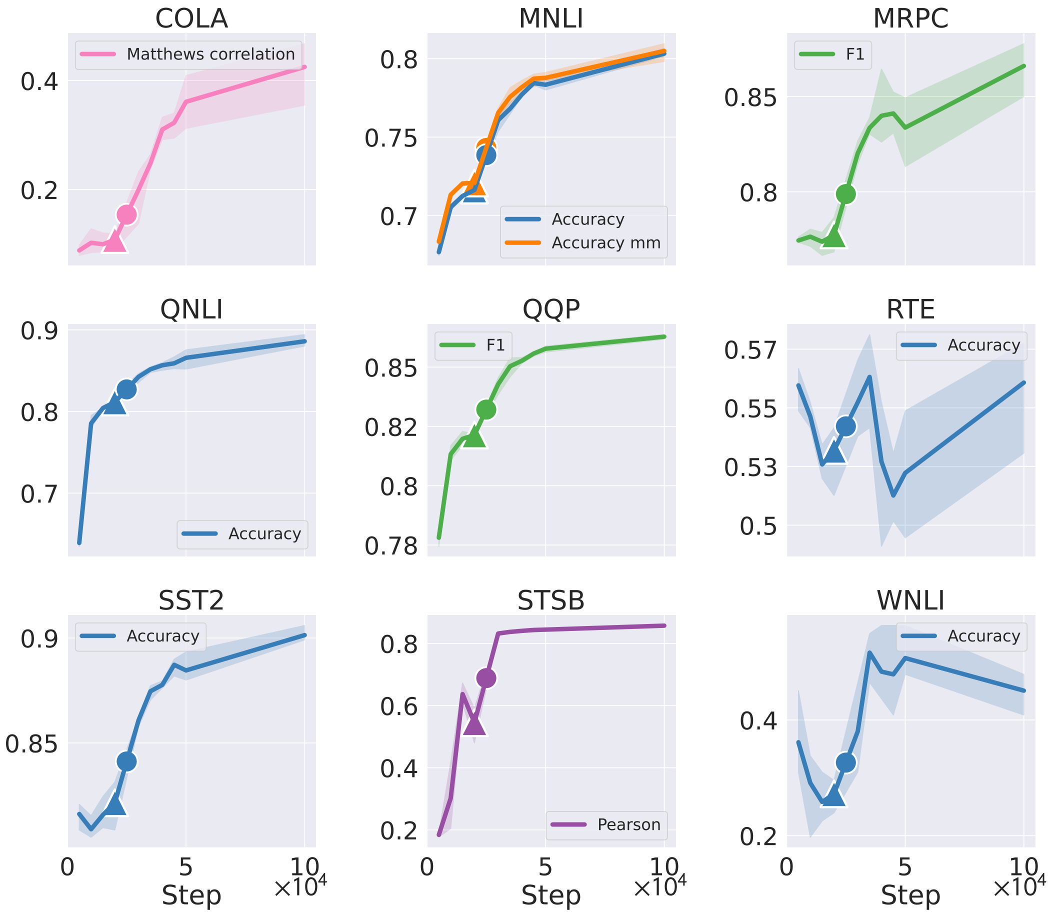


Figure 10: GLUE performance across training for BERT$_{\text{Base}}$, broken down by task. Structure onset (▲) and capabilities onset (●) are marked.

The first piece of evidence for the use of long-range context is that the onset of the alternative strategy's break in the loss curve coincides with the start of an increase in performance for longer n-gram contexts on the task of predicting a masked word within a fixed length context (Fig. 15(b)). For 1000 documents randomly sampled from our validation dataset, we randomly select a segment of $n + 1$ tokens from each document and mask a randomly selected token in each segment. We then compute the average likelihood that the model assigns to the masked token. For example, if $n = 3$ and the randomly selected segment is "a b c d," then we might input the sequence "`[CLS]` a `[MASK]` c d `[SEP]`" to the model and compute the likelihood that the model assigns to the token 'b' at index 2 in the sequence. As training continues, we see spikes in performance for increasingly small contexts while suppressing SAS, over a small window.

When training BERT$_{\text{Base}}$, we also see (Fig. 15(a)) a breakthrough in modeling n-gram contexts across lengths during the structure onset. However, we cannot assess whether a similar set of consecutive phase changes occurs in BERT$_{\text{Base}}$, as the difference in timing may occur at a smaller scale than the frequency of saved checkpoints. If all phase transitions in BERT$_{\text{Base}}$ are simultaneous or close to simultaneous, the gradual acquisition of increasingly local structure in BERT$_{\text{SAS-}}$ may account for the more gradual onset of the accompanying loss drop (Fig. 3(a)). The noteworthy difference that we can confirm is that BERT$_{\text{SAS-}}$ shows a faster initial increase in performance on n-gram modeling compared to BERT$_{\text{Base}}$, suggesting that its break in the loss curve relates to unstructured n-gram modeling, particularly using long range context.

Another piece of evidence is found in the attention distribution. The attention distribution for BERT$_{\text{SAS-}}$ is less predictable based on the relative position of the target word (Fig. 16(a)), so unlike

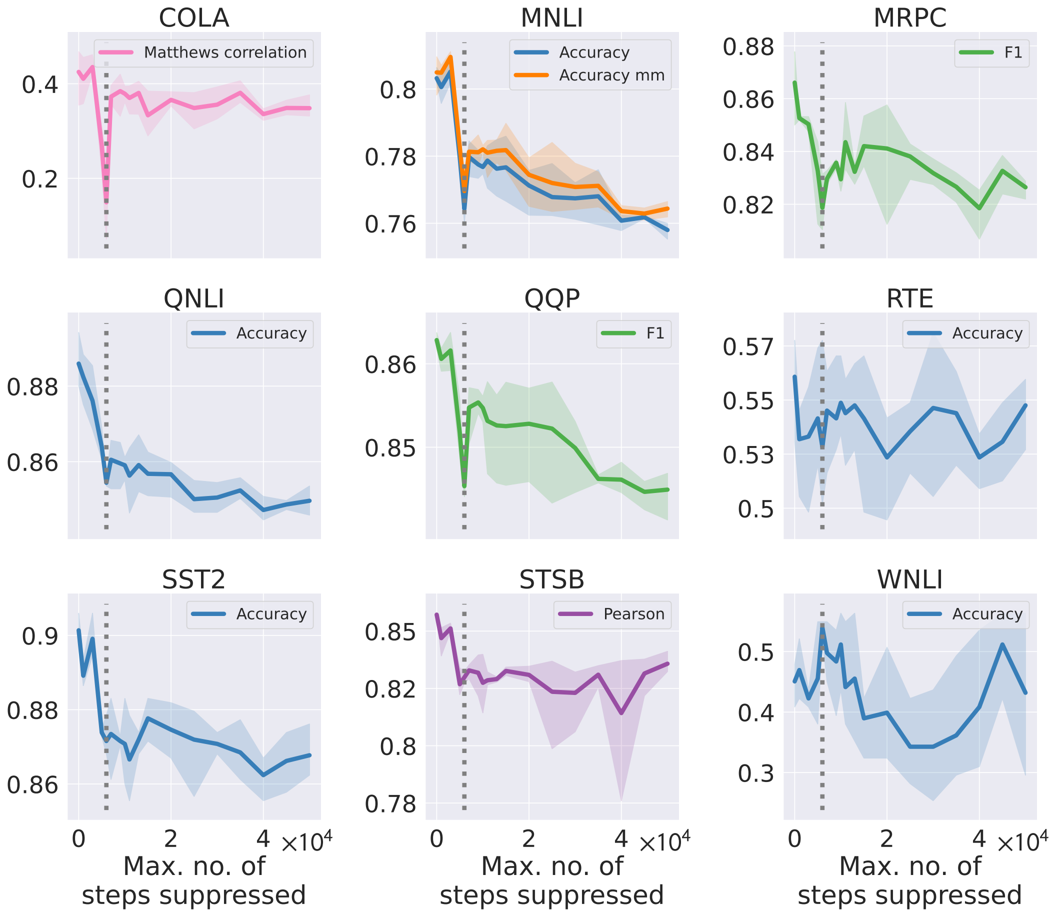


Figure 11: GLUE performance for multi-stage regularized models after 100K timesteps, as a function of the number of steps suppressed and broken down by task. Vertical line marks the BERT$_{\text{SAS-}}$ alternative strategy onset.

in BERT$_{\text{Base}}$, the nearest words no longer take the bulk of attention weight. However, on any given sample, the attention distribution is actually higher-entropy for BERT$_{\text{SAS-}}$ than BERT$_{\text{Base}}$ (Fig. 16(b)), indicating that a small number of tokens still retain the model's focus, although they are not necessarily the nearest tokens. Therefore, BERT$_{\text{SAS-}}$ may rely on other semantic factors, and not on position, to determine where to attend. Note that this evidence is weakened by the fact that attention cannot by directly applied as an importance metric (Ethayarajh & Jurafsky, 2021).

M.1 One big breakthrough or many small breakthroughs?

The loss curve after the alternative phase transition under SAS suppression display appears quite different from the baseline after the SAS onset, because the former trajectory declines far more gradually. A clue to this distinction may be in Fig. 15(a), where we see BERT$_{\text{Base}}$ exhibit simultaneous breakthroughs in n-gram modeling at every context length. In contrast, BERT$_{\text{SAS-}}$ is characterized by a sequence of consecutive breakthroughs starting from the longest context length and gradually reflecting more local context (although this may not account for the difference in transitions—the structure onset in BERT$_{\text{Base}}$ happens later in training when the checkpoints are sampled less frequently, and this may account for the apparent simultaneity). Although the performance at the phrase level shows clear phase transitions, the i.i.d. validation loss appears smooth and gradual from the start of the alternative phase transition. This observation is suggestive that a smooth i.i.d. loss curve can elide many phase transitions under various distribution shifts, possibly reflecting the conjecture of Nanda & Lieberum (2022) that "phase transitions are everywhere."

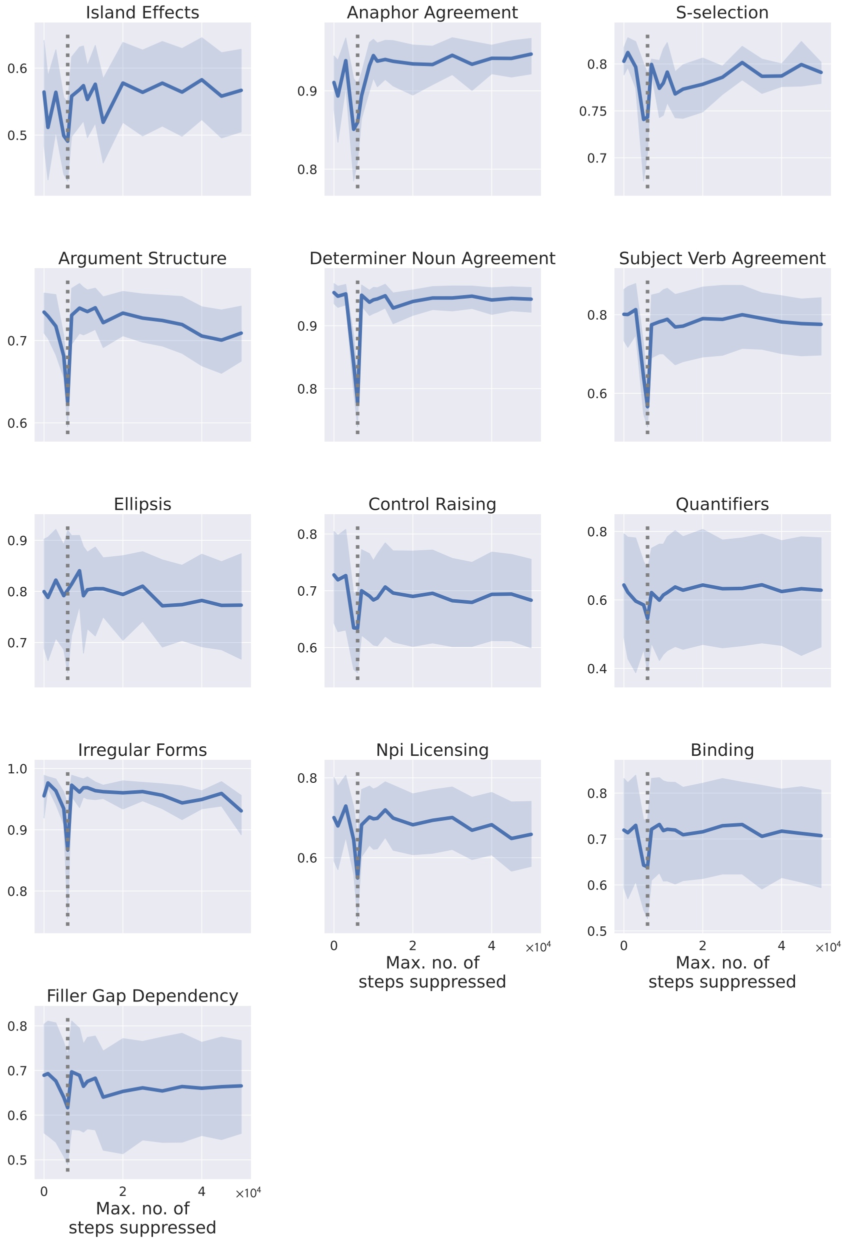


Figure 12: BLiMP accuracy for multistage models at 100k timesteps, broken down by task. Vertical line marks the alternative strategy onset during BERT$_{\text{SAS-}}$ training.

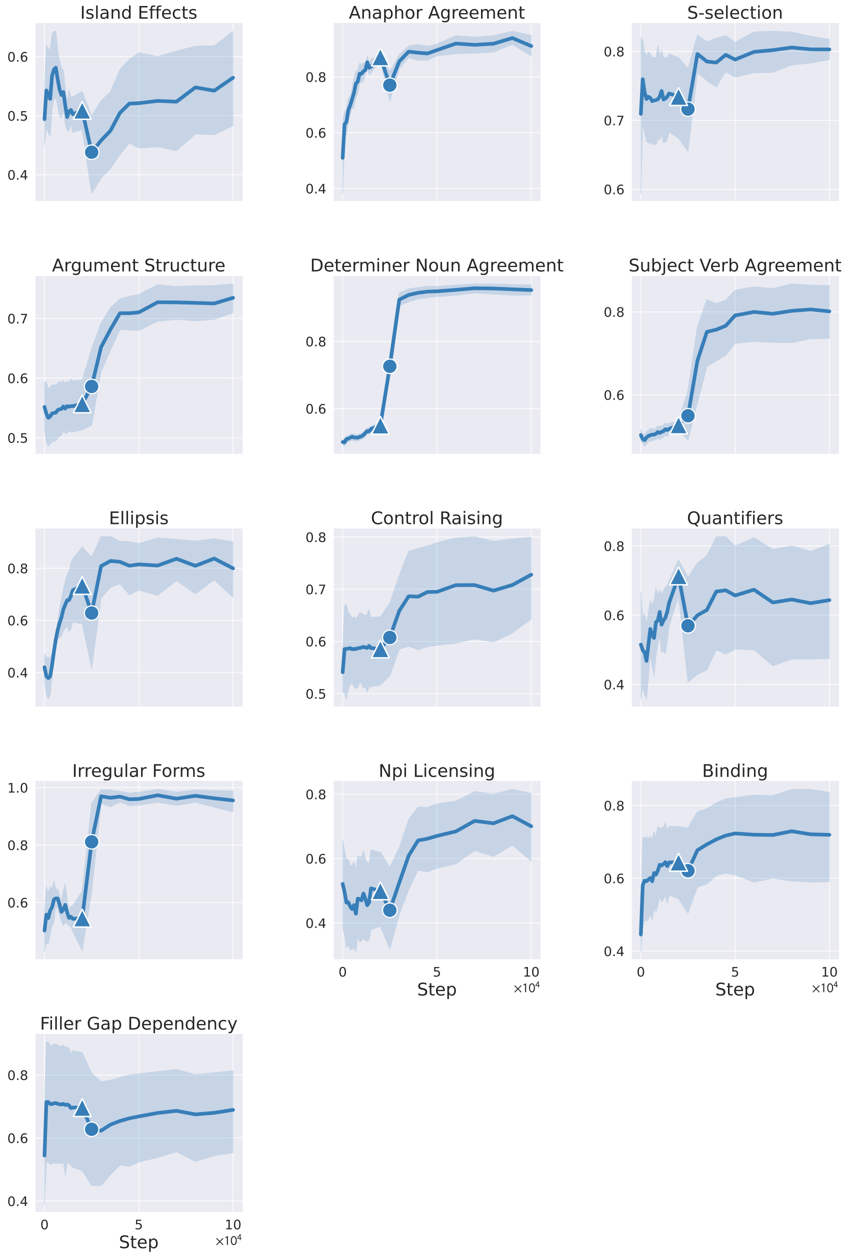
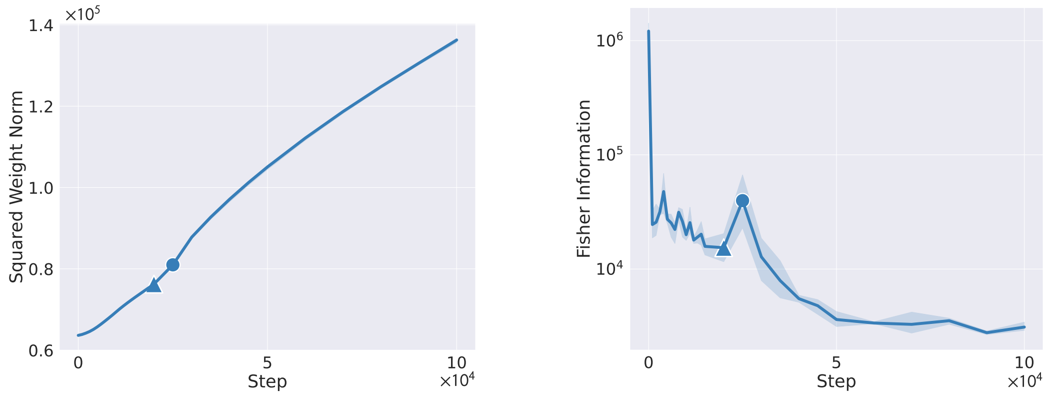


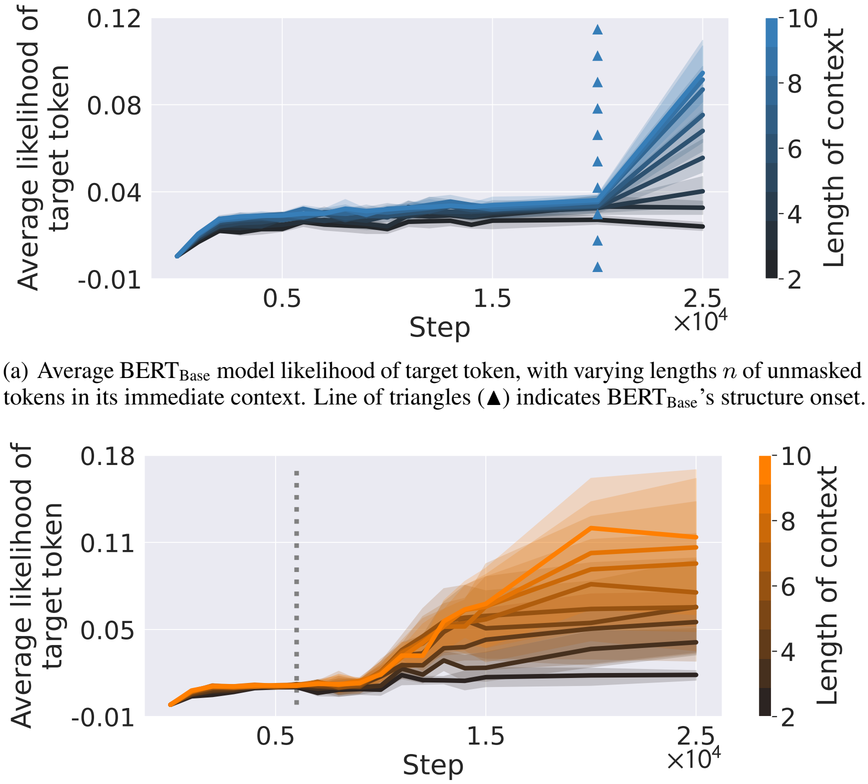
Figure 13: BLiMP accuracy during $\text{BERT}_{\text{Base}}$ training broken down by task. Structure and capabilities onsets are marked.



(a) Weight norm.

(b) Approximate Fisher Information.

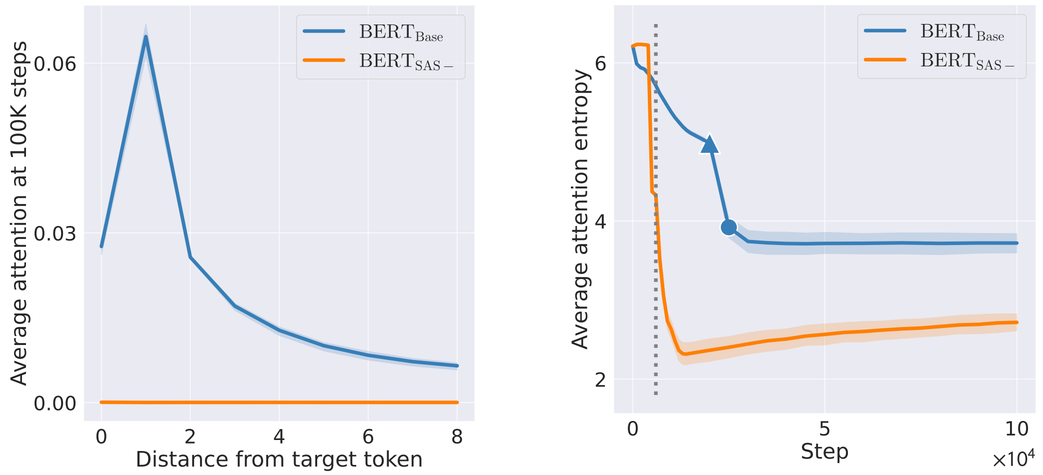
Figure 14: Complexity metrics over time for $\mathrm{BERT}_{\mathrm{Base}}$. Structure and capabilities onsets are marked.



(a) Average $\mathrm{BERT}_{\mathrm{Base}}$ model likelihood of target token, with varying lengths n of unmasked tokens in its immediate context. Line of triangles (▲) indicates $\mathrm{BERT}_{\mathrm{Base}}$'s structure onset.

(b) Average $\mathrm{BERT}_{\mathrm{SAS-}}$ model likelihood of target token, with varying n lengths of unmasked tokens in its immediate context. Dotted line indicates the alternative strategy onset.

Figure 15: The alternative strategy onset, i.e., the break in loss for $\mathrm{BERT}_{\mathrm{SAS-}}$, is associated with improvements in n-gram modeling with longer-range contexts. Meanwhile, the structure onset for $\mathrm{BERT}_{\mathrm{Base}}$ is associated with an improvement in modeling phrases, possibly simultaneously, for all lengths.



(a) Average attention placed on the target token, as a function of distance (in tokens) from the target token.

(b) Entropy of the attention distribution, averaged across heads and samples.

Figure 16: The alternative strategy in $\mathrm{BERT}_{\mathrm{SAS\text{-}}}$ is associated with sparser attention compared to the attention distributions in $\mathrm{BERT}_{\mathrm{Base}}$. However, the average attention of $\mathrm{BERT}_{\mathrm{SAS\text{-}}}$ (as a function of position) is overall low, indicating that $\mathrm{BERT}_{\mathrm{SAS\text{-}}}$ does not focus attention on nearby tokens as much as $\mathrm{BERT}_{\mathrm{Base}}$ does, despite the lower entropy.

N Early-suppression training curves

Fig. 17 illustrates the general trend that, when we briefly suppress SAS early in training, we can recover and even augment the corresponding UAS spike and loss drop. As we continue to suppress SAS, we lose these benefits and further weaken the transition to SAS. The best timing for hyper-parameter release is $\mathrm{BERT}^{(3k)}_{\mathrm{SAS\text{-}}}$, and the training dynamics in Fig. 18 confirm that the multistage approach accelerates and arguments the structure onset and improves model quality during the first 100K steps.

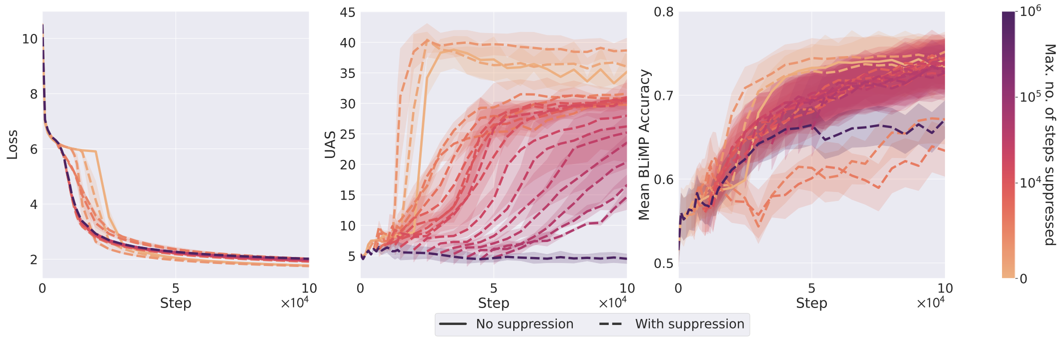
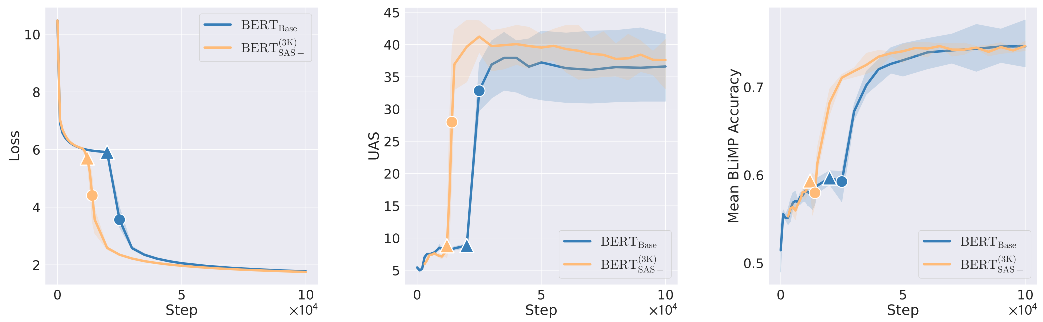


Figure 17: Metrics over the course of training for multistage SAS-regularized models stopped at various points. On y-axis: (a) MLM loss (b) Implicit parse accuracy (c) average BLiMP accuracy. For all multistage training runs, visualized curves begin only after the regularizer is released, i.e., if we suppress SAS for the first 10k steps, the curve begins at 10k. Curve for $\mathrm{BERT}_{\mathrm{Base}}$ is presented as a solid line, with all suppressed models using dashed lines.



(a) MLM validation loss.

(b) Implicit parse accuracy.

(c) Mean BLiMP challenge accuracy.

Figure 18: Briefly suppressing SAS results in improvements of MLM loss, implicit parse accuracy, and linguistic capabilities early on in training.

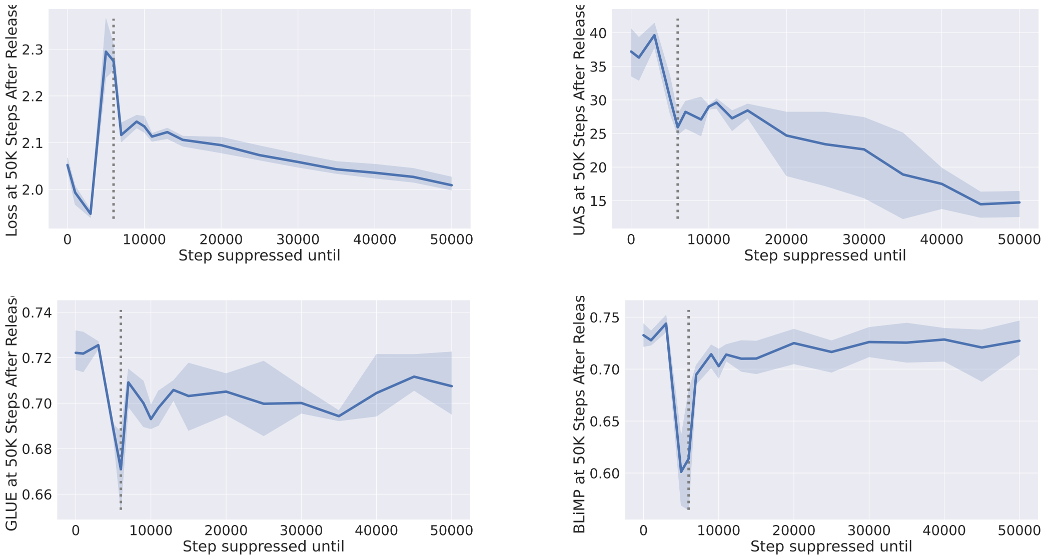


Figure 19: Metrics for the checkpoint 50K steps after the regularizer is removed. X-axis is timestep when regularizer with $\lambda = 0.001$ is removed. On y-axis: (a) MLM loss shown with standard error of the mean across batches (b) Implicit parse accuracy (c) GLUE average (d) BLiMP average. Vertical line marks $\text{BERT}_{\text{SAS-}}$ alternative strategy onset.

O CONTROLLING FOR TIME ALLOWED TO ACQUIRE SAS

Here, we present the same results given in Fig. 4, while controlling for the length of time a model trains after releasing the regularizer in a multistage setting, instead of total training time. We therefore measure performance at exactly 50K steps after setting λ to 0, instead of measuring at 100K steps overall. This allows a shorter overall training time which varies across the models. Therefore, the loss of models with shorter first stages improves less compared to the models with longer first stages. Otherwise, the overall patterns remain consistent with the results at a fixed 100K time steps.

P LONG TERM IMPACT OF MULTISTAGE SUPPRESSION

To investigate whether the models eventually converge in their biases and structures, we look at functional differences in the form of total variation distance (TVD, or the maximum difference in

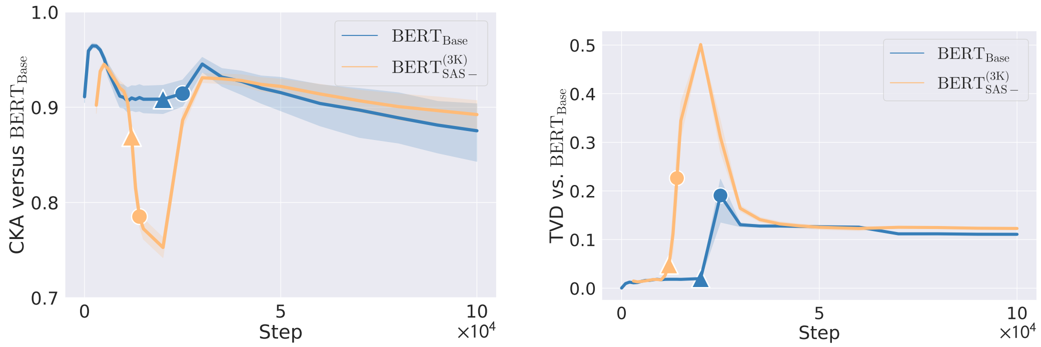
Table 2: Evaluation metrics, with standard error, after training for 300K steps (∼ 39M tokens), averaged across three random seeds for each regularizer setting. We selected $\text{BERT}^{(3k)}_{\text{SAS-}}$ as the best multistage hyperparameter setting based on MLM test loss at 100K steps. We selected 300K as the checkpoint to evaluate longer term performance on because longer runs often destabilized, requiring restarts or re-quantization, which force artificial phase transitions late in training.

	MLM Loss ↓	GLUE average ↑	BLiMP average ↑
$\text{BERT}_{\text{Base}}$	$\mathbf{1.55 \pm 0.00}$	0.74 ± 0.01	0.75 ± 0.05
$\text{BERT}_{\text{SAS+}}$	2.17 ± 0.04	0.63 ± 0.04	$\mathbf{0.77 \pm 0.01}$
$\text{BERT}_{\text{SAS-}}$	1.76 ± 0.02	0.73 ± 0.00	0.62 ± 0.05
$\text{BERT}^{(3k)}_{\text{SAS-}}$	$\mathbf{1.55 \pm 0.01}$	$\mathbf{0.75 \pm 0.01}$	0.76 ± 0.02

probabilities that two distributions can assign to the same event) between the output distributions and representational similarity in the form of centered kernel alignment (CKA; Kornblith et al., 2019).

Although the models initially become more similar in their output functions, their distance eventually stabilizes with the average TVD between $\text{BERT}_{\text{Base}}$ and $\text{BERT}^{(3k)}_{\text{SAS-}}$ falling above the average between pairs of $\text{BERT}_{\text{Base}}$ seeds, suggesting that in the absence of another phase transition, the models will remain distinct in their behavior. Meanwhile, the average CKA($\text{BERT}_{\text{Base}}$, $\text{BERT}^{(3k)}_{\text{SAS-}}$) similarity diverges during the structure and capabilities onsets but converges again towards the average CKA between different $\text{BERT}_{\text{Base}}$ seeds later on. This suggests that even if $\text{BERT}_{\text{Base}}$ and $\text{BERT}^{(3k)}_{\text{SAS-}}$ have high representational similarity, their outputs may still have noticeable differences.

Ultimately, in the long run of training, the difference in quality between $\text{BERT}^{(3k)}_{\text{SAS-}}$ and $\text{BERT}_{\text{Base}}$ ceases to be statistically significant (Tables 1 and 2). While avoiding SAS early in training accelerates convergence and leads to some functional differences in the final models, the critical learning period (Achille et al., 2018) we observe does not continue to hold at scale.



(a) CKA similarity for the activations of $\text{BERT}_{\text{Base}}$ and $\text{BERT}^{(3k)}_{\text{SAS-}}$, compared to the average CKA similarity between different runs of $\text{BERT}_{\text{Base}}$.

(b) Average divergence between output distributions of $\text{BERT}_{\text{Base}}$ and $\text{BERT}^{(3k)}_{\text{SAS-}}$, compared to the average divergence between different runs of $\text{BERT}_{\text{Base}}$.

Figure 20: Representational and functional similarity over time for $\text{BERT}_{\text{Base}}$ and $\text{BERT}^{(3k)}_{\text{SAS-}}$. Structure (▲) and capabilities (●) onsets are marked on each line. Shaded regions are 95% confidence intervals over different pairs of models.