# An empirical study on modelling Working Memory constraints in Transformers

Anonymous ACL submission

## Abstract

We investigate the integration of human-like working memory constraints into the Transformer architecture and implement several cognitively inspired attention variants, including fixed-width windows based and temporal decay based attention mechanisms. Our modified GPT-2 models are trained from scratch on developmentally plausible datasets (10M and 100M words). Performance is evaluated on grammatical judgment tasks (BLiMP) and alignment with human reading time data. Our results indicate that these cognitively-inspired constraints, particularly fixed-width attention, can significantly improve grammatical accuracy especially when when training data is scarce. These constrained models also tend to show a stronger alignment with human processing metrics. The findings suggest that such constraints may serve as a beneficial inductive bias, guiding models towards more robust linguistic representations, especially in data-limited settings.

## 1 Introduction

The dominant self-attention mechanism in Transformer-based models (Vaswani et al., 2017) diverges profoundly from established cognitive theories of human language processing. A key area of divergence lies in how information is accessed and maintained over time. Human comprehenders rely on working memory (WM), a short-term mental buffer, in order to temporarily store and manipulate information during cognitive tasks such as language comprehension. This memory system is understood to be capacity-limited (Miller, 1956; Cowan, 2001), subject to temporal decay (Baddeley, 2000), and influenced by serial position effects like primacy and recency (Glanzer and Cunitz, 1966). In stark contrast, standard Transformer self-attention allows near-uniform access to all tokens within a potentially very large context window, lacking inherent architectural biases that reflect these fundamental human cognitive constraints. While LLMs like GPT-2 have proven useful in cognitive modeling, correlating well with measures such as reading times and neural activity (Goodkind and Bicknell, 2018; Wilcox et al., 2020; Madhyastha et al., 2023), this success may occur despite, rather than because of, architectural alignment with human processing limitations.

This discrepancy motivates the central research question of our work where we investigate whether integrating architectural constraints inspired by human working memory can produce models that not only exhibit more human-like linguistic behaviour but also learn more effectively. To this end, our contribution is threefold. First, we implement and systematically compare four distinct, cognitively-motivated attention mechanisms: (1) a strict, fixed-width attention window to model capacity limits; (2) exponential and (3) logistic decay mechanisms to model recency bias; and (4) a primacy-recency model to capture serial position effects. Second, we train these models entirely from scratch on developmentally plausible datasets of 10 million and 100 million words, allowing these constraints to shape the learning process from its inception. Third, we conduct a multi-faceted evaluation, assessing not only grammatical competence on the BLiMP benchmark (Warstadt et al., 2020) but also alignment with human processing data and the nature of the internal representations learned by the models.

Recent research in computational psycholinguistics has begun to explore incorporating more cognitively plausible mechanisms in Transformer-like models. Studies leveraging pre-trained models like GPT-2, (e.g. Ryu and Lewis (2021)), have shown that these models can implicitly capture human-like processing phenomena, for example similarity-based interference, through their attention patterns and surprisal outputs, thereby validating GPT-2 as a tool for psycholinguistic analysis. However, the

inherent complexity of modern LLM architectures, particularly their multiple attention heads, raises questions about cognitive plausibility and may obscure direct alignment with human memory constraints. For instance, Timkey and Linzen (2023) advocate for simpler, more constrained models that are closely based on cue-based retrieval theories (Van Dyke and Lewis, 2003). In light of these considerations, researchers have sought more direct ways to integrate cognitive principles. A study by De Varda and Marelli (2024) demonstrated that applying a post-hoc exponential decay bias to GPT-2's attention weights improved its correlation with human reading times. These efforts highlight the ongoing investigation into how to better align LLM processing with human cognition, motivating approaches that explicitly integrate cognitive principles into model training.

Building on these research threads, our work moves beyond purely implicit learning or post-hoc modifications by explicitly integrating working memory constraints directly into the GPT-2 architecture from the outset of training. Specifically, we investigate how imposing constraints inspired by prominent findings in human working memory including capacity limitations, temporal decay, and serial position effects impacts model performance and language processing.

Our results show that imposing these constraints, particularly restrictive fixed-width attention windows, serves as a powerful inductive bias that yields significant benefits. In low-data settings, constrained models substantially outperform the standard GPT-2 baseline in grammatical accuracy, confirming that such biases are highly effective when data is scarce. We also see that these models produce surprisal values that are markedly more predictive of human processing data, suggesting a closer alignment with the cognitive dynamics of comprehension. While this performance gap narrows with more training data, our analysis of the models' internal mechanisms reveals why these constraints are so effective. We use attention visualisations and syntactic probing to show that limiting the context forces the model to develop specialised, linguistically interpretable attention heads and a more explicit internal encoding of syntactic structure a level of specialisation not observed in the diffuse patterns of the unconstrained baseline. Ultimately, this provides compelling evidence that architectural constraints inspired by human cognition are not merely a matter of plausibility, but a robust method for building more data-efficient, better-performing, and more interpretable language models.

## 2 Methods

We implement four distinct attention mechanisms directly into the GPT-2 architecture to simulate human-like working memory constraints. Each of these mechanisms is designed to model specific features of human working memory. The following sections detail the implementation of a fixed window attention, as well as formulations based on exponential/logistic decay and primacy-recency biases.

### 2.1 Fixed Window Attention

In this approach, the attention calculation for each token is restricted to a *fixed-size window* of a set of fixed preceding tokens. For a token at position $i$, attention is only computed over tokens in the range $[\max(0, i - W + 1), i]$, where $W$ is the fixed window size. This is implemented using an attention mask $M_{window}$ that prevents access to tokens outside the window: $M_{window}^{(i,j)} = \begin{cases} 0, & \text{if } \max(0, i - W + 1) \leq j \leq i \\ -\infty, & \text{otherwise} \end{cases}$

The attention weights are then calculated as:

$a'_{ij} = a_{ij} + M_{window}^{(i,j)}$ where $a_{ij}$ are the original attention weights. This sets the attention weights for tokens outside the window to $-\infty$ (and thus zero after `softmax` normalisation), while tokens inside the window retain their original weights unchanged. This hard windowing is a simplified approximation of working memory capacity limits prominent in cognitive psychology literature discussed in Section 1, forcing the model to focus on a local context.

The selection of fixed window sizes for our models is directly motivated by influential findings in cognitive science concerning the capacity limits of human working memory. A window size of $k = 4$ is informed by contemporary research, such as that of Cowan (2001), who hypothesised that short-term memory has a capacity of approximately four 'chunks' of information. The window sizes of $k \in 5, 7, 9$ are derived from Miller's (1956) seminal observation concerning "the magical number seven, plus or minus two", which represents the classic estimate for the number of items an individual can hold in immediate memory. In the context

of our experiments, we treat a token as a fundamental information chunk.

In the context of NLP, a form of fixed window attention is a core component of models designed for long documents, such as Longformer (Beltagy et al., 2020) and Block-Sparse Transformer (Child et al., 2019). While these models are motivated by efficiency, our fixed window attention is motivated by mirroring the limited capacity of human working memory, suggesting a cognitive basis for such architectural choices in NLP.

### 2.2 Primacy-Recency Attention

Human memory recall has been hypothesised to exhibit primacy and recency effects (Glanzer and Cunitz, 1966; Morrison et al., 2014, interalia). Items presented at the beginning (primacy) and end (recency) of a list are typically better recalled than items in the middle. In the context of language processing, this suggests that initial and final parts of a sequence might hold disproportionate importance in shaping the overall representation. We incorporate constraints of this kind through a primacy-recency attention mechanism. This mechanism adds a position-dependent bias to the attention weights that emphasise both the initial and final tokens in the sequence. It learns two parameters, $w_{\text{primacy}}$ and $w_{\text{recency}}$, which are initialized to 0.5 during training. We calculate primacy weights $p_i$ and recency weights $r_i$ for each position $i$ in a sequence of length $L$:

$$p_i = \frac{e^{-i/L}}{\sum_{j=0}^{L-1} e^{-j/L}} \tag{1}$$

$$r_i = \frac{e^{-(L-1-i)/L}}{\sum_{j=0}^{L-1} e^{-(L-1-j)/L}} \tag{2}$$

where $i$ is the position index (starting from 0). Primacy weights decay exponentially from the beginning of the sequence, while recency weights decay exponentially from the end. Both sets of weights are normalised to sum to one. The final bias $b_i$ for each position is a weighted combination of primacy and recency weights:

$b_i = w_{primacy} \cdot p_i + w_{recency} \cdot r_i$ where $w_{primacy}$ and $w_{recency}$ are learnable weights that control the relative contribution of primacy and recency biases. These biases are then added to the attention weights:

$a'_{ij} = a_{ij} + b_j$

We note here that the bias $b_j$ is added based on the *key* position $j$ (i.e., the position of the token being attended to in the attention mechanism). Our intention with this position-dependent bias is to encourage the model to attend more strongly to tokens at the beginning and end of the sequence, reflecting primacy and recency effects from psycholinguistic theories. We also separately run an ablation with exclusively primacy and recency based attention to understand the impact of each of these mechanisms. While less directly related to computational efficiency in long sequence processing, the primacy-recency attention mechanism aligns with the broader trend in NLP towards incorporating positional information in more sophisticated ways than simple positional embeddings. For instance, relative positional embeddings (Shaw et al., 2018) and complex positional encodings (Su et al., 2021) aim to capture richer positional relationships. In some way, this attention modification helps focus on more positional information to emphasize the structural importance of sequence beginnings and endings.

### 2.3 Exponential Decay Attention

Inspired by recent psycholinguistic theories that highlight the interplay between linguistic expectations and working memory constraints in human language processing (see Smith and Levy, 2013; Gibson, 1998; Hahn et al., 2022, interalia), we also consider an exponential decay attention mechanism. This modification is directly motivated by the work of De Varda and Marelli (2024), who propose biasing Transformer models to prioritize local linguistic context, simulating a lossy representation of distant contextual information in human sentence processing. Here, the exponential decay attention mechanism modulates the standard attention weights by incorporating a decay factor that diminishes the influence of tokens based on their temporal distance. The modified attention weight $a'_{ij}$ between token $i$ and token $j$ is calculated as:

$$a'_{ij} = (1 - \alpha)a_{ij} + \alpha e^{-|i-j| \cdot \lambda} \tag{3}$$

where $a_{ij}$ represents the original dot-product attention weight, $\lambda$ is the decay rate, and $\alpha$ is a mixing parameter. The exponential term $e^{-|i-j| \cdot \lambda}$ introduces a bias favouring attention to closer tokens, effectively implementing a recency effect by exponentially reducing the contribution of more distant tokens. Following De Varda and Marelli (2024),

we adopt the hyperparameters $\lambda = 82.86$ (corresponding to `decay_rate` in our implementation) and $\alpha = 0.37$. These values were identified as optimal in their grid search using GPT-2-small on the Provo corpus (Luke and Christianson, 2018). De Varda and Marelli (2024) demonstrated that applying a post-hoc exponentially decaying attention bias to a pretrained GPT-2 model improved its correlation with human reading times. While these results are useful, this approach modifies an already developed system rather than allowing constraints to shape the learning process from the outset. To remedy this, our methodology involves training the customised GPT-2 model from scratch with the exponential decay attention mechanism inherently integrated into its architecture. This approach will allow the model learns to process language under the constraint of locality-biased attention from the outset. We hypothesise that training with this constraint from the beginning may lead to a more concordant and effective integration of the psycholinguistic principle, as the model architecture is aligned with the intended processing mechanism throughout the learning process, rather than having the bias imposed after the model has already learned with a different attention paradigm. This approach has clear parallels within the field of NLP, where locality-sensitive attention is explored for efficiency reasons when handling long sequences. This is evident in models such as Performer (Choromanski et al., 2020) and in Transformers that use linear biases (Katharopoulos et al., 2020; Press et al., 2021).

### 2.4 Logistic Decay Attention

The logistic decay mechanism modulates attention weights based on the temporal distance between tokens using a psychometric function. For tokens at positions $i$ and $j$, the attention weight modification is computed as: $w_{ij} = \frac{1}{1+e^{k \cdot (d_{ij} - m)}}$ where $d_{ij} = \max(1, i - j + 1)$ represents the distance between tokens, $k$ is the steepness parameter controlling the sharpness of the decay curve, and $m$ is the midpoint parameter determining the distance at which attention weight equals 0.5. The final attention weights are calculated by multiplicatively combining the original attention scores with the logistic prior: $a'_{ij} = a_{ij} \cdot w_{ij}$ We set $k = 0.4$ and $m = 12.0$ as default parameters, establishing a psychologically motivated attention profile where tokens within approximately 5 positions maintain relatively strong (high) attention weights, while more distant tokens experience rapid attention decay. The logistic decay attention mechanism exhibits several key characteristics that distinguish it from other approaches. First, it maintains relatively stable attention weights for nearby tokens (distance $< m$), followed by a rapid transition to low attention weights for distant tokens (distance $> m$). This creates a more pronounced boundary between accessible and inaccessible memory, aligning with accounts of discrete WM span. Second, unlike fixed window attention which implements a hard cutoff, logistic decay provides a smooth but steep transition, avoiding the potential discontinuities associated with binary attention masking.

In the following sections, we will present the empirical results where we test out the above methods.

## 3 Experimental Setup

This section details our experimental setup. We introduce the set of models evaluated, the datasets employed for training, and the overall framework for our analysis, which focuses on training language models from scratch with modified attention mechanisms. Our experiments are conducted on the GPT-2-small architecture (Radford et al., 2019). Our core models are based on the standard GPT-2 configuration, but incorporate custom attention mechanisms implemented by modifying the `GPT2Attention` module. We also train two baseline models using the default GPT-2 configuration provided by the Hugging Face `transformers` library (Wolf et al., 2019).

### 3.1 (Pre-)Training Corpora

Our primary interest lies in evaluating language models under conditions that are more cognitively plausible in terms of data scale than typical large language model pretraining. Therefore, we utilize the BabyLM dataset (Warstadt et al., 2023). The BabyLM Challenge itself drew inspiration from the scale of data available during human language acquisition. Specifically, we use the training portions of "Strict-Small" (10 million words) and "Strict" (100 million words) training subsets provided by the BabyLM Challenge (Warstadt et al., 2023). These datasets comprise text from sources considered potentially relevant to child language exposure, including Simple English Wikipedia, children's books from Project Gutenberg, CHILDES transcripts, the British National Corpus, Open-

Subtitles, and the Switchboard Dialog Act Corpus. We note that all of our models also use GPT2Tokenizers which are trained specifically on 10 million and 100 million words based corpora separately.

### 3.2 Training Configuration

For all models, we used the GPT-2 small architecture as a base. Training was performed using the AdamW optimizer with a learning rate of $5e^{-5}$, a batch size of 64, and a weight decay of 0.01. Models were trained for 5 epochs with a batch size of 50. Gradient clipping was applied with a maximum norm of 1.0. These settings were kept consistent across all model variants to ensure a fair comparison. These hyperparameters are similar to the range of empirical setups common in Warstadt et al. (2023).

### 3.3 Tasks

We evaluate the trained models on two distinct tasks designed to probe different aspects of their linguistic capabilities and cognitive plausibility.

**BLiMP** The first task assesses the sensitivity of the models to English grammatical structure using the Benchmark of Linguistic Minimal Pairs (BLiMP) (Warstadt et al., 2020). BLiMP consists of numerous sub-tasks, each targeting a specific linguistic phenomenon. Every example in BLiMP presents a "minimal pair": one sentence that is grammatically acceptable and another that is unacceptable, with the two differing only minimally (often by a single word or morpheme). A language model is considered correct on a given minimal pair if it assigns a *higher probability score* (hence low surprisal) to the acceptable sentence compared to the unacceptable one. Success across BLiMP tasks indicates that the model has learned representations consistent with fine-grained grammatical distinctions in English.

**Psychometric Benchmark** The psychometric data we use are drawn from established experimental paradigms (detailed in de Varda et al. 2024) with measurements averaged across several neural and behavioural indices of cognitive processing on a set of 1725 sentences. This includes a) Eye-Tracking Data with measures such as First Fixation Duration, which reflects early lexical access, and later-stage integrative measures like Gaze Duration, Go-Past Time, and Right-Bounded Time; b) Self-Paced Reading Time which is a controlled measure of reading speed, influenced by a range of semantic and syntactic factors; c) Event-Related Potentials which are neural signals that provide a fine-grained temporal view of language processing. These include the N400 component, which is modulated by meaning processing; the P600, indicative of syntactic reanalysis and integration; and various Left Anterior Negativity components (LAN and ELAN) associated with phrase structure building.

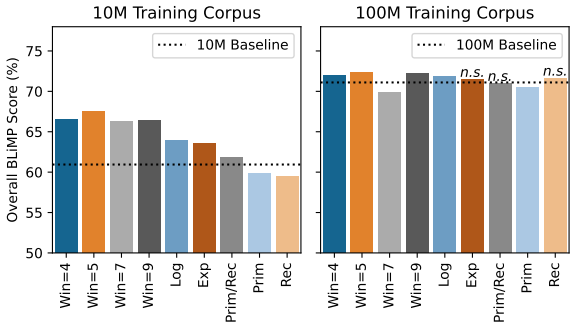



Figure 1: Performance comparison of GPT-2 self-attention modifications on BLiMP tasks for 10M (left) and 100M (right) parameter models. Dashed lines indicate the performance of the baseline GPT-2 model (without any attention modifications) trained from scratch on the corresponding dataset size. All models show statistically significant differences from baseline ($p < 0.001$) except where marked as not significant (n.s.).

## 4 Results

### 4.1 Overall observations on grammaticality

We first examine the overall performance of our modifications on attention mechanisms across 10 million and 100 million word corpora (Figure 1). We observe a clear trend in the low-data setting. We see that all models with our modified attention mechanisms demonstrate a substantial and statistically significant improvements ($p \leq 0.001$) in average BLiMP accuracy compared to the baseline model. While the baseline scores approximately 61% average accuracy on the task, models with fixed attention windows, which impose the most stringent constraints, achieve markedly higher scores of around 68%. This result highlights the benefit of architectural inductive biases derived from working memory principles, particularly when training data is scarce.

However, this advantage diminishes when models are trained on the larger 100M-word dataset. In this higher-data regime, the baseline's performance improves markedly to an accuracy of ap-

proximately 71%, narrowing the performance gap considerably. Notably, models with exponential or primacy-and-recency constraints on attention show no statistically significant difference from the baseline. With a sufficient volume of data, the standard attention mechanism evidently recovers much of the capability required for this task. Despite this, the best-performing constrained models maintain a small but statistically significant edge, indicating that their inductive biases remain beneficial even at a larger scale.

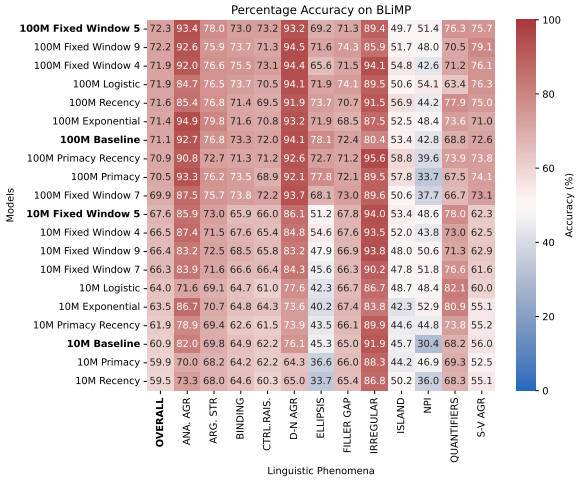



Figure 2: Model performance across linguistic phenomena evaluated on the BLiMP task. Models are sorted by Overall score in a descending order; 50% indicates chance performance. Best performing 10M and 100M models are highlighted along with the corresponding Baselines.

### 4.2 Performance across linguistic phenomena

In order to understand how our modifications to attention influences performance on specific linguistic tasks, we analysed the results across individual BLiMP sub-tasks. In Figure 2, we categorise the different tasks into a set of classes motivated by the analyses in Warstadt et al. (2020). Consistent with previous findings, our models perform best on phenomena related to morphological agreement, which often rely on local dependencies. Across both 10 million and 100 million data regimes, all models achieve reasonably high accuracy on Determiner-Noun Agreement (which focuses on number agreement, e.g., that chair vs. that chairs), Subject-Verb Agreement, and Anaphor Agreement (where the focus is on reflexive pronoun agreement, e.g., girls insulted themselves vs. herself). We note that the fixed-window models, which explicitly enforce locality, excel here. We also find that most models suffer on more abstract syntactic and semantic constraints. Performance on Island Effects (restrictions on syntactic movement) and NPI Licensing (the requirement for words like ever to be in a negative context) is the lowest across the board, often only marginally better than chance. This confirms that these phenomena, which predominantly require sensitivity to complex structural and logical scope, represent a persistent challenge for language models, and our architectural modifications do not offer a simple solution to assist models on these tasks.

On the other hand, the results for Argument Structure, which governs a verb's ability to appear with certain arguments (e.g., disturbing a person vs. boasting a person), are particularly interesting. Performance is generally boosted by locality constraints across both data regimens. We see that the models that have fixed window size of about 5 and exponentially decaying attention constraints (which tend to promote highly local attention structures) seem to generally perform better[1]. We also observe that Ellipsis emerges as a phenomenon highly dependent on data scale. At 10M, all models perform exceptionally poorly, with accuracies in the 30-51% range. However, at 100M words, performance improves dramatically into the 65-75% range across all models. This sharp increase suggests that the generalisations required to correctly resolve ellipsis are not readily captured with limited data, but indeed become accessible with more exposure.

A striking overall finding is that highly constrained, simple models often outperform the less-constrained baseline, particularly on complex syntactic and semantic challenges. Our null hypothesis before experimentation was that heavily constrained models may significantly hamper performance in transformers, especially since a large body of work in contemporary NLP looks into methods for moving away from locality bias (Tay et al., 2020; Zaheer et al., 2020, interalia). We find the performance of small, rigid attention windows on phenomena that are not strictly local to be the most surprising result. For instance, fixed window 5 is broadly one of the best performing models in complex sub categories. Especially in Argument Structure where smaller fixed window models obtain significantly higher performance, where the

[1]We also see that the performance on Argument Structure especially is remarkably similar to GPT2-large pretrained model (Warstadt et al., 2020).

sentences usually focus on structures which govern a verb's relationship with its arguments. This is noteworthy because these relationships can be complex and are not solely determined by adjacent words, yet this highly local model captures them effectively. Similarly, these models perform exceptionally well on Binding, which involves the structural relationship between a pronoun and its antecedent. One might expect this to require a wider context, but the small fixed window proves highly effective, outperforming the baseline. On the other hand, leaky models, both Exponential and Logistic bias based models, which allow attention to "leak" across the entire context while prioritising recent information, show interesting trends, however these models are inferior in performance compared to stricter and smaller fixed window based models.

**Primacy/Recency Ablation** We wanted to further understand the efficacy of primacy and recency based formulation and carried out an experiment where we trained models separately with primacy and recency biases. We present these results in Figures 1 and 2. We notice, unsurprisingly, that Recency based bias in attention tends to perform better with local structures while Primacy tends to have a slight edge on tasks with dependency structures that require access to longer distances. While both Primacy and Recency generally tend to perform worse in comparison to the baseline model, however, intriguingly the increasing data tends to substantially help the model with Recency based bias. This tends to correlate with fixed window models with strict local attention constraints. Finally, while the model based on Primacy and Recency is not among the best models, however in most cases, it has a tendency to even out the divergence between models trained separately with Primacy or Recency.

### 4.3 Analysing the model behaviour with human processing data

Would our cognitively inspired attention constraints correlate with human processing data? To answer this we examine the alignment of surprisal and psychometric data using averaged difference of log-likelihood (Figure 3). Here, ΔLog-Likelihood measures the additional explanatory power offered by surprisal values over a base model without surprisal. We then average this across psychometric corpora. That is, the difference in log-likelihood

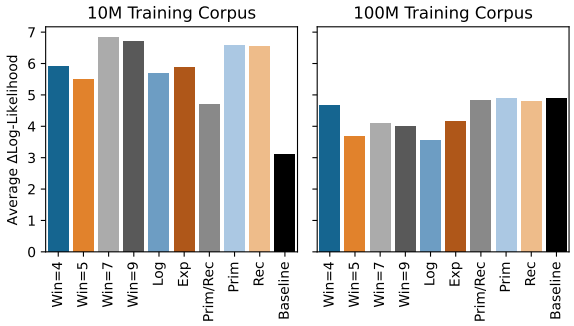
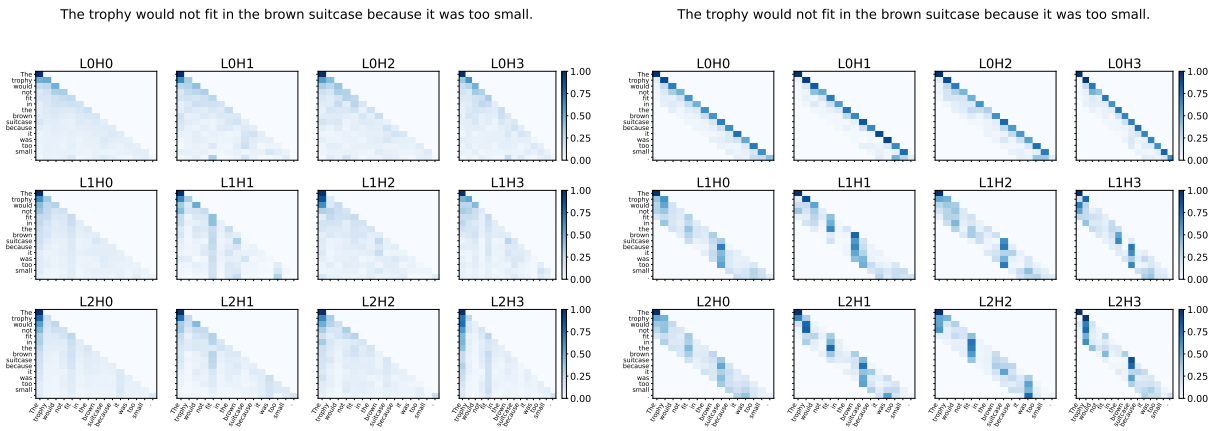



Figure 3: ΔLog-Likelihood averaged across psychometric measures for each model.

(ΔLog-likelihood) represents the improvement in statistical model fit when surprisal is added as a predictor, compared to a base statistical model that includes only covariates like word length and frequency. Therefore, a higher ΔLog-likelihood indicates that a language model's surprisal values are a better predictor of human cognitive effort during sentence comprehension. The 'Baseline' bar in the figure refers to the ΔLog-Likelihood achieved by GPT-2 model with unmodified attention trained on corresponding corpora, serving as our primary model for comparison.

We observe a clear and compelling trend emerging in the low data regimen. Nearly all models with modified attention mechanisms produce surprisal values that are substantially more predictive of human processing data than the unmodified Transformer baseline. The baseline model achieves a relatively low average ΔLog-Likelihood of approximately 3.2, whereas the top-performing models with fixed-window attention (especially 7 and 9), as well as Primacy and Recency biases, achieve scores nearly twice as high. This result suggests that in data-constrained settings, architectural constraints that mimic principles of human working memory serve as a powerful and effective inductive bias, guiding the model to learn representations that are more closely aligned with human cognitive processes.

On the other hand, the advantage of these explicit architectural biases diminishes considerably when the models are trained on the larger, 100M-word dataset. The standard baseline model's performance improves, with its average ΔLog-Likelihood increasing marginally. Consequently, the performance gap between the baseline and the modified models narrows significantly. This pattern suggests that with an order of magnitude more




(a) 10M Baseline

(b) 10M Fixed Window 5

Figure 4: Attention weights distributions for the sentence "The trophy would not fit in the brown suitcase because it was too small." across individual heads in the first three layers of the 10M Baseline and Fixed Window 5 models.

data, the standard self-attention mechanism is better able to approximate the necessary behaviours and produce surprisal values that correlate well with human processing metrics. However, we note here that even for all the constrained attention models and the baseline (with unmodified attention) the $\Delta$Log-likelihood is lower in the 100M setting compared to the 10M setting. A compelling hypothesis for this seemingly counterintuitive trend relates to the divergence between the language modelling objective and the cognitive processes underlying human comprehension. As models are trained on more data, they become increasingly specialised at predicting the statistical patterns of the training corpus. This high degree of specialisation may cause their expectations to diverge from the more generalised predictions that humans make which is also illustrated in previous work (Oh and Schuler, 2023; de Varda et al., 2024; Shain et al., 2024, interalia).

### 4.4 Understanding attention distribution

To better understand the inductive biases developed within our models, we analysed their internal attention distributions. In Figure 4, we present a comparison of two models trained on the 10 million word corpus: a baseline GPT-2 with standard unmodified self-attention and a constrained model with a fixed-window attention mechanism (here window size is 5). The visualisations render the attention patterns for the sentence: "The trophy would not fit in the brown suitcase because it was too small", a classic example used to test pronoun resolution. While we focus in the figure on early layers, the pattern is consistent across higher layers too. We primarily observe how architectural constraints foster specialisation. The fixed-window model demonstrates an immediate and sharp focus on local context; even in its initial layer, heads concentrate on the immediately preceding token, unlike the diffuse patterns of the baseline. This specialisation becomes highly pronounced in later layers. For instance, heads `L2H0` and `L2H1` learn to focus on the core subject-verb-object structure of the sentence ('trophy', 'fit', 'suitcase'), a pattern reminiscent of the telegraphic speech observed in children between the ages of 18 and 36 months (Brown, 1973). Other heads develop distinct roles, with `L2H2` specialising in verbs (e.g., 'fit', 'was') and `L2H3` in nouns (e.g., 'trophy', 'suitcase'). In stark contrast, the baseline model's attention remains diffused, with its heads struggling to specialise on clear linguistic functions, focusing instead on less interpretable combinations of function and content words. It is seemingly evident that the fixed-window constraint acts as a potent inductive bias, compelling a structured division of labour amongst the attention heads to represent syntax in a way not seen in the unconstrained model, at least as explicitly as in fixed window models.

## 5 Conclusions

Our work demonstrates that imposing architectural constraints inspired by human working memory produces superior language models, particularly when data is limited. These constraints, especially fixed-width attention, act as a potent inductive bias, leading to significantly improved grammatical accuracy and a closer alignment with human processing data.

## 6 Limitations

While our findings provide compelling evidence for the utility of cognitive inductive biases in data-constrained settings, we acknowledge several limitations to the current study. Firstly, our investigation is explicitly confined to the developmentally plausible regime (10M and 100M words); consequently, it remains an open question whether the architectural advantages of fixed-width and decay-based attention persist or are subsumed by scale in Large Language Models trained on trillions of tokens. Secondly, our implementation of working memory constraints represents a simplified abstraction of human cognitive processes. While we model capacity and decay, human working memory is also dynamic and content-addressable, features not fully captured by our rigid, position-based masks. Thirdly, our empirical evaluation is restricted to English. Given that the efficacy of locality constraints may vary depending on a language's morphosyntactic density and word-order flexibility, future work must verify these effects across typologically diverse languages. Finally, while constrained models excel at local dependency tasks, our results on Island Effects indicate that strict locality biases may hinder the acquisition of phenomena requiring the tracking of dependencies over extensive global contexts.

## References


Alan Baddeley. 2000. The episodic buffer: a new component of working memory? *Trends in cognitive sciences*, 4(11):417–423.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

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

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, and 1 others. 2020. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*.

Nelson Cowan. 2001. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and brain sciences*, 24(1):87–114.

Andrea De Varda and Marco Marelli. 2024. Locally biased transformers better align with human reading times. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 30–36, Bangkok, Thailand. Association for Computational Linguistics.

Andrea Gregor de Varda, Marco Marelli, and Simona Amenta. 2024. Cloze probability, predictability ratings, and computational estimates for 205 english sentences, aligned with existing eeg and reading time data. *Behavior Research Methods*, 56(5):5190–5213.

Edward Gibson. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68(1):1–76.

Murray Glanzer and Anita R Cunitz. 1966. Two storage mechanisms in free recall. *Journal of verbal learning and verbal behavior*, 5(4):351–360.

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

Michael Hahn, Richard Futrell, Roger Levy, and Edward Gibson. 2022. A resource-rational model of human processing of recursive linguistic structure. *Proceedings of the National Academy of Sciences*, 119(43):e2122602119.

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

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR.

Steven G Luke and Kiel Christianson. 2018. The provo corpus: A large eye-tracking corpus with predictability norms. *Behavior research methods*, 50:826–833.

Pranava Madhyastha, Ye Zhang, and Gabriella Vigliocco. 2023. Are words equally surprising in audio and audio-visual comprehension? *arXiv preprint arXiv:2307.07277*.

George A Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81.

Alexandra B Morrison, Andrew RA Conway, and Jason M Chein. 2014. Primacy and recency effects as indices of the focus of attention. *Frontiers in human neuroscience*, 8:6.

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

Ofir Press, Noah A Smith, and Mike Lewis. 2021. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*.

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

Soo Hyun Ryu and Richard Lewis. 2021. Accounting for agreement phenomena in sentence comprehension with transformer language models: Effects of similarity-based interference on surprisal and attention. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 61–71, Online. Association for Computational Linguistics.

Cory Shain, Clara Meister, Tiago Pimentel, Ryan Cotterell, and Roger Levy. 2024. Large-scale evidence for logarithmic effects of word predictability on reading time. *Proceedings of the National Academy of Sciences*, 121(10):e2307876121.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Nathaniel J Smith and Roger Levy. 2013. The effect of word predictability on reading time is logarithmic. *Cognition*, 128(3):302–319.

Taiga Someya, Ryo Yoshida, Hitomi Yanaka, and Yohei Oseki. 2025. Derivational probing: Unveiling the layer-wise derivation of syntactic structures in neural language models. *arXiv preprint arXiv:2506.21861*.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. 2021. Roformer: Enhanced transformer with rotary position embedding. *Preprint*, arXiv:2104.09864.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2020. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*.

William Timkey and Tal Linzen. 2023. A language model with limited memory capacity captures interference in human sentence processing. *arXiv preprint arXiv:2310.16142*.

Julie A Van Dyke and Richard L Lewis. 2003. Distinguishing effects of structure and decay on attachment and repair: A cue-based parsing account of recovery from misanalyzed ambiguities. *Journal of Memory and Language*, 49(3):285–316.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

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

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

Ethan Wilcox, Peng Qian, Richard Futrell, Ryosuke Kohita, Roger Levy, and Miguel Ballesteros. 2020. Structural supervision improves few-shot learning and syntactic generalization in neural language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4640–4652, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and 1 others. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and 1 others. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

# A Appendix

## A.1 Core Architectural Parameters

We rely on the GPT-2 Small model which is defined by the following core parameters:

**Number of Layers ($n_{\text{layer}}$):** This parameter defines the depth of the model, representing the number of stacked Transformer decoder blocks. Our models have 12 layers.

**Embedding Dimension ($n_{\text{embd}}$):** This specifies the dimensionality of the token embeddings and the hidden states used throughout the model. It essentially defines the "width" of the network's layers. The dimensionality in our setup is 768.

**Number of Attention Heads ($n_{\text{head}}$):** In each Transformer block, the multi-head self-attention mechanism is split into a number of parallel "heads," allowing the model to jointly attend to information from different representation subspaces. All our models use 12 attention heads per layer.

**Context Window Size ($n_{\text{ctx}}$):** This specifies the maximum number of tokens the model can process in a single input sequence. The model's positional embeddings are trained for this length. The maximum context length for the baseline version is 1024 tokens.

## A.2 Total Parameter Count

The combination of these architectural choices results in a model with approximately **124 million** trainable parameters. This is calculated based on the dimensions of the weight matrices in the embedding layer, the Transformer blocks (including self-attention and feed-forward networks), and the final output layer.

# B Reproducibility Statement

We will release all code and accompanying data for full reproducibility in the final version of this paper.

# C Semantic Probes

To further investigate the syntactic knowledge encoded within the models' internal representations, we use derivational probing (Someya et al., 2025), a methodology which builds on top of Structural Probing technique (Hewitt and Manning, 2019). Derivational Probing assesses how explicitly a model learns hierarchical linguistic structure by training a simple diagnostic classifier which is a structural probe to reconstruct syntactic dependency trees directly from the model's word embeddings. The success of this reconstruction is measured by the Unlabeled Unrooted Attachment Score (UUAS), where a higher score indicates that the model's internal geometry of representations more accurately reflects the grammatical relationships between words in a sentence. This method provides a fine-grained view of a model's latent syntactic knowledge, moving beyond surface-level behaviour to an analysis of its learned representational space. We focus here again on the same two models from above – the unconstrained baseline and the fixed window model with a 5-window constraint.

The results of this probing analysis reveal a pronounced advantage for the fixed-window model over the baseline in encoding core syntactic dependencies. We observe that the fixed-window architecture consistently achieves a higher UUAS early on across all five relations tested, indicating a more robust and accessible representation of syntax. The performance gap is particularly pronounced for fundamental grammatical relations such as `nsubj` (nominal subject) and `dobj` (direct object), where the architectural constraint appears to significantly aid in learning the relationship between a verb and its arguments. This suggests that limiting or constraining the model's attention to a local context, the fixed-window mechanism not only improves performance on tasks requiring ambiguity resolution but also fosters the development of more explicit and coherent internal representations of syntactic structure itself. We present these results in the appendix (Figure 5).

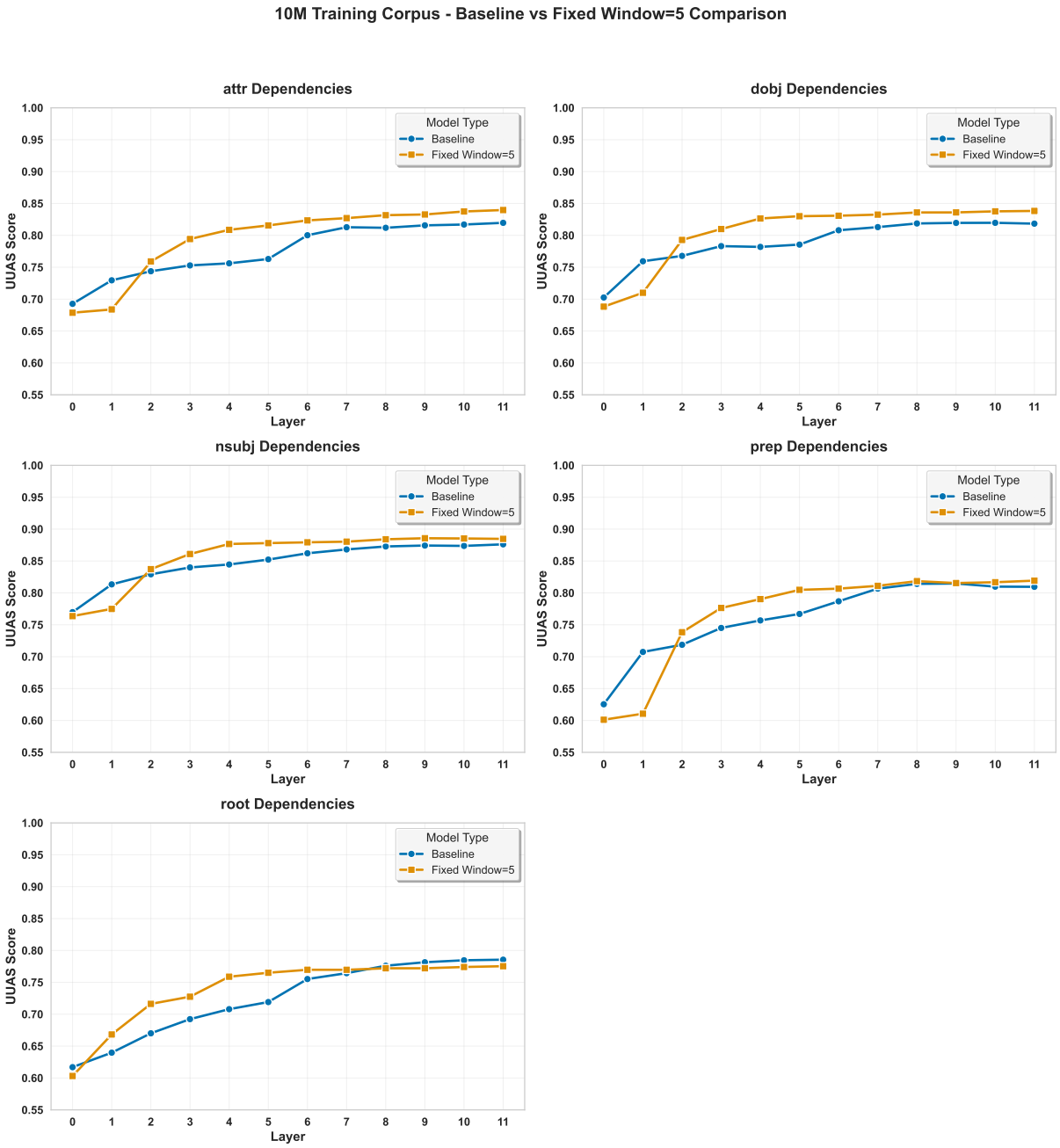



Figure 5: UUAS Score comparison between the Baseline and Fixed Window 5 models trained on 10M words.

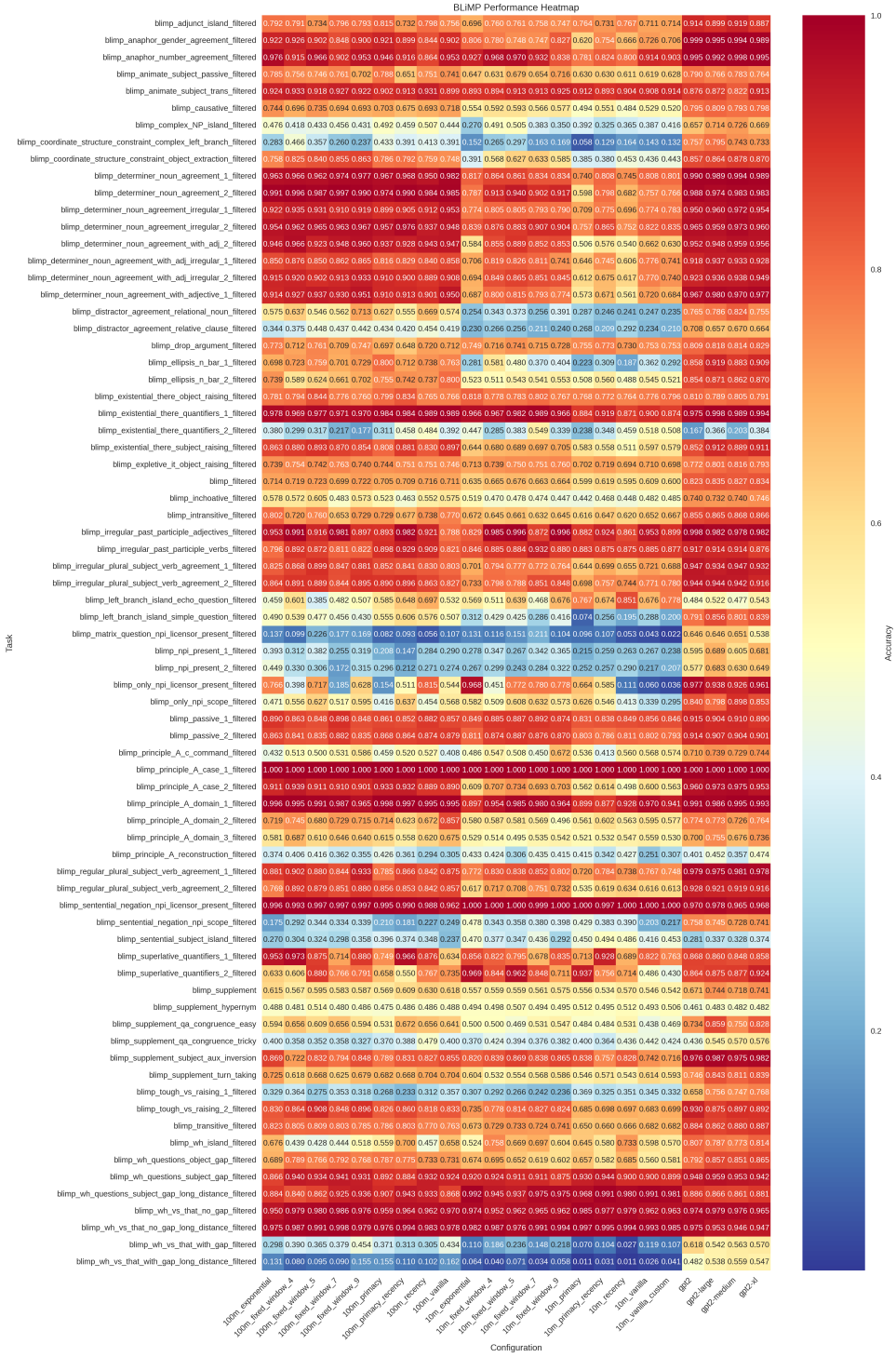



Figure 6: Model performance on BLiMP across individual tasks and models.