

# Reconstructing Locality-Disrupted Text: Effects of Architecture and Scale

Evelien van Driel, Amirhossein Mohammadi, Robert A. Bagheri

Utrecht University  
e.vandriel2@students.uu.nl

## Abstract

Studies of *impossible languages* evaluate whether LLMs share human linguistic biases by testing their ability to learn or reconstruct systematically perturbed text. However, existing literature relies almost exclusively on the decoder-only architecture at a fixed scale. We ask whether reconstruction generalizes across architecture families and improves with model size, pretraining GPT-2 Medium and three T5-base variants (varying pretraining objective and decoder supervision) from scratch on four perturbations and fine-tuning each for reconstruction. GPT-2 Medium reconstructs all four (exact match 0.13–0.82); all three T5 variants collapse to  $\approx 0.00$ . Six diagnostics trace this to attention entropy collapse in the T5 encoder within 600 pretraining steps, a 96% reduction robust to pretraining objective, optimizer, and fine-tuning data: a trainability limit under low-resource from-scratch pretraining, not a capacity limit, leaving the bidirectional advantage untested. Within the decoder-only family, scaling helps unevenly (+0.25 exact match where local word order survives, +0.02 where it does not), and recoverable sentence length scales the same way. Both benefits are gated by how much local word order survives the perturbation, refining the account on which reconstruction difficulty tracks information locality. Architecture and scale are therefore experimental variables in this paradigm, not neutral defaults.

## 1 Introduction

Information locality, the tendency for words that share information to occur close together, is a core organizing property of human language (Futrell, 2019): natural languages keep related elements near one another, and comprehension difficulty grows as they are pulled apart. Perturbations that disrupt locality therefore yield variants of normal English outside the space of attested human languages. These variants, commonly termed *impossible languages*, are now a tool for probing

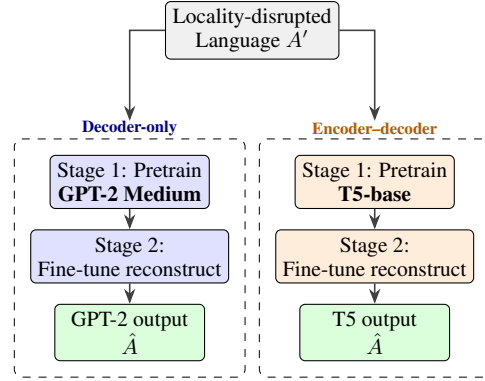


Figure 1: Decoder-only vs. encoder-decoder training pipeline for reconstruction from locality-disrupted text.

a central question in computational linguistics: whether large language models (LLMs) share human linguistic biases or learn any pattern equally well. Chomsky et al. (2023) argued on theoretical grounds that LLMs acquire both with equal ease and therefore tell us little about the structural properties of human language; Kallini et al. (2024) tested this by training GPT-2 Small (Radford et al., 2019) on both, finding possible languages easier to learn. Someya et al. (2025) traces that difference to locality itself: what makes an impossible language hard is largely how far it pulls related words apart. Whether such findings bear on human language remains debated (Ziv et al., 2026; Xu et al., 2025; Yang et al., 2025; Kallini and Potts, 2025; Tosolini and Blevins, 2025), but they have established locality as the property this literature manipulates.

Work in this area has focused on learnability, measured by perplexity on perturbed text (Kallini et al., 2024; Yang et al., 2025; Ziv et al., 2026; Xu et al., 2025). Mohammadi et al. (2026) introduced a second task, reconstruction, in which the model receives systematically corrupted input and must produce the original. This asks not how easily a pattern is learned but what linguistic information survives a perturbation and whether a model can

use it to reverse the damage. They report that reconstruction difficulty tracks the degree of locality disruption, and that longer sentences aid recovery where local structure is preserved but not where it is globally destroyed. What a model recovers from locality-disrupted input is thus governed by how much locality the perturbation leaves intact.

This conclusion rests on evidence from a single model. All reconstruction results come from GPT-2 Small: one decoder-only architecture, one scale, one left-to-right way of consuming disrupted input. Kallini et al. (2024) anticipated this limitation, presenting their paradigm as an invitation to test different LLM architectures on impossible languages. That invitation has gone largely unanswered: Xu et al. (2025) paired GPT-2 Small with an encoder-only masked language model on the learning task and found similar patterns across architectures, but reconstruction has been tested on no other model, and encoder-decoders on neither task. Claims about what LLMs recover from locality-disrupted input are, strictly speaking, claims about one model.

Whether a disruption is hard to undo may depend not only on the perturbation but on the model: how it accesses the disrupted input, and how much capacity it has. Reconstruction is unusually sensitive to the first, since deciding where a displaced word belongs draws on evidence lying on either side of its current position. A decoder-only model reads left to right; an encoder-decoder reads the input in full before generation, using both left and right context to place each word (Raffel et al., 2020). Bidirectional encoding is thus the natural candidate for undoing displacement, and reconstruction the natural task on which to test it, but any such comparison presupposes something untested: that an encoder-decoder can be trained on locality-disrupted input at all.

Capacity may similarly interact with locality. Scaling reliably improves language modeling (Kaplan et al., 2020), but reconstruction depends on more than representational capacity: severe disruption leaves less structure recoverable, and added capacity can only exploit signal that survives (Mohammadi et al., 2026). Scale should therefore improve reconstruction unevenly, and the same reasoning bounds sentence length, which supplies usable context only where local structure remains to anchor it. Testing scale thus asks not whether prior results replicate at a larger size, but whether the locality account predicts where model resources help.

Both axes bear on the wider debate: this literature reasons from model behavior to human linguistic bias, and if what a model recovers depends on how it reads its input and how large it is, such reasoning holds only with architecture and scale fixed.

We train a decoder-only model (GPT-2 Medium) and an encoder-decoder model (T5-base) from scratch on perturbed BabyLM text (Warstadt et al., 2023), then fine-tune both for reconstruction following Mohammadi et al. (2026). We use four perturbations from Kallini et al. (2024), two shuffles and two reversals, which disrupt locality to varying degrees; their third class, HOPs, is excluded because it leaves locality intact. T5-base is trained under three conditions varying the pretraining objective and decoder supervision, and scale is assessed against the published GPT-2 Small results of Mohammadi et al. (2026).<sup>1</sup>

Five questions follow, three on architecture and two on scale. **RQ1**: can an encoder-decoder learn locality-disrupted input from scratch under low-resource conditions, as a decoder-only model can, and can the comparison be run at all? **RQ2**: where does any failure originate, and does it reflect a limit on what the architecture can represent or on what training produces? **RQ3**: does it hold across pretraining objectives and supervision quantities, or is it an artifact of one configuration? **RQ4**: does one increase in model size improve reconstruction evenly across perturbations that disrupt locality to varying degrees? **RQ5**: does the sentence-length effect of Mohammadi et al. (2026) depend on disruption severity within a single corpus?

The experiments answer each in turn. (**RQ1**) No: GPT-2 Medium reconstructs all four locality-disrupted variants, while every T5 variant fails completely. (**RQ2**) A diagnostic chain locates the failure in attention entropy collapse in the encoder during pretraining, a trainability failure rather than a capacity limit, leaving the question of a bidirectional advantage open. (**RQ3**) The collapse is robust across pretraining objectives, supervision regimes, and optimizer and data ablations. (**RQ4**) No: scaling gains are uneven, and the gap between perturbations widens with scale. (**RQ5**) Within a single corpus, length aids reconstruction where local structure survives but not under global shuffling.

This paper contributes the first cross-architecture

<sup>1</sup>All data and code are available at [https://github.com/Evelien-bug/impossible\\_project\\_training](https://github.com/Evelien-bug/impossible_project_training).

and cross-scale evidence on reconstruction from locality-disrupted input, showing that single-model findings do not generalize and that for encoder-decoders the barrier is trainability rather than capacity (RQ1–RQ3); a refinement of the locality account, in which the benefits of both added capacity and longer input are gated by how much local structure survives the disruption (RQ4–RQ5); and a replicable diagnostic framework for separating trainability failures from capacity limits in low-resource pretraining (RQ2).

## 2 Background

### 2.1 Information locality and perturbations

Recent research shows that LLMs can learn and represent linguistic structures (Hewitt and Manning, 2019; Tenney et al., 2019; Warstadt et al., 2020). Information locality, the tendency for words that share information to occur close together, is one such property that shapes how efficiently language can be processed (Futrell, 2019). A perturbation is a function that systematically modifies natural text, disrupting information locality and producing a variant that falls outside the space of attested human languages (Moro et al., 2023). Such variants, the impossible languages of Chomsky et al. (2023) and Kallini et al. (2024), isolate the contribution of a single property: by degrading it and measuring the effect, one can test how much a model relies on it.

Kallini et al. (2024) introduced perturbations in three classes: shuffles, reversals, and hops. These break natural language in distinct ways. Shuffles and reversals leave verb morphology intact but disrupt word order, and with it locality, to varying degrees. Hops, by contrast, alter verb morphology through a count-based rule while leaving word order and locality intact. This study draws on the shuffles and reversals, the classes that disrupt information locality.

Building on these perturbations, Mohammadi et al. (2026) introduced the reconstruction task, in which a model recovers the original text from shuffled or reversed input. Reconstruction difficulty tracked the degree of locality disruption, which indicates that locality is a property the model draws on when recovering a sentence. Reconstruction also depended on sentence length: where local structure was preserved, longer sentences yielded stronger recovery despite the greater distance between related words. Difficulty therefore depends

both on how far related words are displaced and on how much context the model has to work with.

### 2.2 Model architecture

Findings on the reconstruction task are so far limited to GPT-2 Small. GPT-2 models are decoder-only Transformers trained on autoregressive next-token prediction (Radford et al., 2019): each token is predicted from its left context, with no access to what follows (Wang et al., 2022). This causal masking may influence the model’s ability to represent linguistic structures for reconstruction, since the information needed to recover a token’s original position can lie anywhere in the corrupted input, including to its right (Devlin et al., 2019; Fu et al., 2023). Furthermore, there is a risk of diminishing attention to early parts of the input as generation progresses, especially in long sequences (Fu et al., 2023).

Encoder-decoder architectures such as T5 (Raffel et al., 2020) are not subject to the same constraint. The encoder applies bidirectional self-attention over the whole corrupted input, and the decoder generates the reconstruction with access to that full encoding through cross-attention. Syntactic relations, including long-distance dependencies, are therefore available before generation begins (Zhong et al., 2024; Fu et al., 2023). On sequence-to-sequence tasks such as summarization and translation, encoder-decoders match or outperform decoder-only models of comparable scale (Chrabrowa et al., 2022; Araujo et al., 2024; Kumar et al., 2020), which makes them the natural candidate for undoing displacement.

This advantage, however, is not guaranteed to be available. An architecture is trained with a pretraining objective, and those objectives are designed to exploit the natural-language structures that the perturbations disrupt. Whether an encoder-decoder can build useful encoder representations from such input at all, under from-scratch pretraining, is untested. The potential of bidirectional encoding is therefore conditional on a prior question of trainability, which this study takes up directly.

### 2.3 Model size

Beyond architecture, model size may affect reconstruction. GPT-2 Small (124M parameters) already learns to recover corrupted input (Mohammadi et al., 2026), but how recovery changes at larger scale is unknown. Kaplan et al. (2020) showed that language modeling performance improves with

| Language       | Example 1           | Example 2                                                 |
|----------------|---------------------|-----------------------------------------------------------|
| Original text  | Fine as a fiddle.   | Ned pointed to the sky. There was not a cloud in sight.   |
| LocalShuffle   | a Fine as fiddle.   | to Ned pointed There sky. the a was not in sight. cloud   |
| FullShuffle    | fiddle. a Fine as   | the a was to sight. Ned cloud not sky. pointed in There   |
| PartialReverse | Fine R fiddle. a as | Ned pointed to R sight. in cloud a not was There sky. the |
| FullReverse    | fiddle. a as R Fine | sight. in cloud a not was There sky. the R to pointed Ned |

Table 1: Example sentences under each of the four perturbations used in this study. Colors indicate the position of each word in the original sentence to make movement under perturbation visible. For ease of comparison, we adopt the visualization style of Kallini et al. (2024) and Mohammadi et al. (2026).

increased parameter count, dataset size and compute. Furthermore, at increased scales, models can capture syntactic structure more reliably (Warstadt et al., 2020). Added capacity, however, can only help where the input still carries signal to exploit, suggesting that performance gains may be smaller for perturbations that disrupt locality most severely. This motivates comparing GPT-2 Small with a larger model in the same family, to test whether one step up in size improves reconstruction evenly across perturbations or only where local word order survives.

### 3 Method

This study applies the reconstruction task of Mohammadi et al. (2026) to two additional models. Because pretraining checkpoints on perturbed text exist only for GPT-2 Small, we train both from scratch following Kallini et al. (2024), in two stages: Stage 1 pretrains on the perturbed BabyLM 2026 Strict corpus (Warstadt et al., 2023); Stage 2 fine-tunes on (perturbed, original) sentence pairs from its *Gutenberg* subset. GPT-2 Medium versus T5-base gives the architecture comparison; GPT-2 Medium versus the published GPT-2 Small results of Mohammadi et al. (2026) gives the scale comparison.

#### 3.1 Models

*GPT-2 Medium* (355M) extends *GPT-2 Small* (124M) used by Kallini et al. (2024) and Mohammadi et al. (2026). All GPT-2 sizes share one architecture and differ only in depth and width (Radford et al., 2019), isolating scale as the only variable that changes between Small and Medium.

*T5-base* (220M, 12-layer encoder, 12-layer decoder) introduces an encoder-decoder (Raffel et al., 2020). We chose T5 over BART (Lewis et al., 2020), whose best-performing recipe combines text infilling with sentence permutation: permutation

overlaps conceptually with our shuffles and reversals, so a BART-pretrained model would already have to undo a structurally similar disruption before fine-tuning. T5’s span corruption is structurally distinct from our perturbations, and it is additionally documented under prefix language modeling, the second objective we use (§3.3).

### 3.2 Data generation

#### 3.2.1 Perturbations

We apply the four perturbations of Kallini et al. (2024) that disrupt word order and information locality, excluding Hops for the reason given in §2. Both reversals first insert a marker token R at a random position:

- **LOCALSHUFFLE**: shuffles words within consecutive non-overlapping windows of  $w$  words ( $w=3$ , matching Mohammadi et al., 2026), disrupting local structure while roughly preserving sentence-level order.
- **FULLSHUFFLE**: shuffles all words in the sentence, using the DeterministicShuffle algorithm of Kallini et al. (2024).
- **PARTIALREVERSE**: reverses the sequence after the marker, leaving the prefix in original order.
- **FULLREVERSE**: reverses the entire sentence.

Table 1 shows an example of each. We use the published seeds of Kallini et al. (2024), with one deviation from their reference code. Following Mohammadi et al. (2026), we sample an independent permutation per LOCALSHUFFLE window rather than reusing one across all windows in a sentence. Seeds are listed in Appendix B.

#### 3.2.2 BPE and word-level implementations

GPT-2 and T5 use different tokenizers (BPE and SentencePiece), which segment the same sentence differently. Prior work perturbs at BPE level, native to GPT-2 (Kallini et al., 2024; Mohammadi et al.,

2026); replicating this unchanged would train the two architectures on differently perturbed text, confounding their comparison. We therefore apply each perturbation at both levels. BPE level supports the GPT-2 Small vs. Medium scaling comparison; word level, being tokenizer-agnostic, yields the same sequence whichever model tokenizes it, and so supports the GPT-2 Medium vs. T5-base comparison. GPT-2 Medium is trained at both levels, T5-base at word level.

### 3.2.3 Data

Stage 1 uses the full BabyLM 2026 Strict corpus ( $\approx 100\text{M}$  tokens across train, dev, and test splits). Stage 2 uses 100k (perturbed, original) pairs from the *Gutenberg* subset, with 1k held out for evaluation; we restrict Stage 2 to *Gutenberg* because its longer, more complex sentences maximize the signal available for measuring reconstruction differences (Mohammadi et al., 2026). Perturbing both corpora at both levels yields eight Stage 1 and eight Stage 2 datasets.

### 3.3 Stage 1: impossible language pretraining

Each model learns the distributional statistics of one perturbation rather than any property of natural English, since the original text is never seen. For PARTIALREVERSE and FULLREVERSE, both tokenizers are extended with the marker token R before pretraining. Hyperparameter specifications are provided in Appendix B.

**GPT-2 Medium** is pretrained with the causal language modeling objective (causal LM) (Radford et al., 2019), following the procedure Kallini et al. (2024) used for GPT-2 Small. Each of its eight Stage 1 models (four perturbations  $\times$  two levels) is trained once for 3,000 steps at sequence length 1,024.

GPT-2 Medium and T5-base are not parameter-matched (355M vs. 220M), and cross-architecture comparisons depend on what is held constant (Tay et al., 2022). Rather than rely on one comparison, we train T5-base under three conditions varying the pretraining objective and what is matched to GPT-2 Medium:

- **T5-SC**: span corruption, the default T5 objective (Raffel et al., 2020), which masks 15% of tokens and supervises only those, against GPT-2 Medium’s 100% supervision (see Appendix A).

- **T5-PLM-single**: prefix LM (Raffel et al., 2020), supervising  $\approx 50\%$  of tokens. Like causal LM, it predicts upcoming tokens from preceding context, making it the objective structurally closest to GPT-2 Medium’s. Exposure-matched, leaving decoder supervision unmatched.
- **T5-PLM-double**: the same objective trained to match decoder-supervised tokens rather than tokens seen. The cost is roughly twice the data exposure, though because the prefix split is resampled at random on each pass this is added variation over the same sentences, not repeated examples.

The two exposure-matched conditions run 6,000 steps to GPT-2 Medium’s 3,000 because T5-base uses sequence length 512 against 1,024, so both see  $\approx 1.57\text{B}$  tokens. Together the three conditions span two objectives and both matching schemes, so the outcome cannot be attributed to one training configuration. Appendix A gives per-condition supervision rates and an excluded fourth condition. Each of the twelve T5 Stage 1 models (four perturbations  $\times$  three conditions) is trained once; computational constraints limit Stage 1 to a single seed, where Kallini et al. (2024) used five.

### 3.4 Stage 2: reconstruction fine-tuning

Each Stage 1 model is fine-tuned to map the perturbed sentence back to its original. **GPT-2 Medium** uses the causal LM objective under a fixed prompt template following Mohammadi et al. (2026), with loss computed only on the target sentence. **T5-base** uses sequence-to-sequence cross-entropy: the encoder reads the perturbed sentence and the decoder generates the original, so no template is needed. T5’s Stage 2 objective therefore differs from both its Stage 1 objectives, whereas GPT-2 Medium uses causal LM throughout. All three T5 conditions are fine-tuned identically, giving eight fine-tuned GPT-2 Medium and twelve T5 models. Checkpoints are selected by best validation loss; Stage 2 is likewise single-seed. The template and hyperparameters are in Appendix B.

### 3.5 Evaluation metrics

Stage 1 is evaluated by perplexity (Bahl et al., 1983; Jurafsky and Martin, 2026) on 10,000 perturbed test sentences, reported as the geometric mean following Kallini et al. (2024). We report perplexity within each architecture but not across them: causal

LM, span corruption, and prefix LM are computed on different prediction targets and are not comparable.

Stage 2 is evaluated on 1,000 held-out *Gutenberg* pairs with four metrics, following [Mohammadi et al. \(2026\)](#). **Exact Match (EM)** is the percentage of predictions identical to the original ([Rajpurkar et al., 2016](#)), and **BLEU** their n-gram overlap ([Papineni et al., 2002](#)). **Average Dependency Length (ADL)** is the mean linear distance between syntactic heads and dependents ([Liu, 2008](#)), compared against a natural English baseline. **Dependency Triple F1** parses each sentence into (dependent, relation, head) triples and reports the F1 of exact overlap with the original ([Buchholz and Marsi, 2006](#); [Cai and Knight, 2013](#)). The first two capture surface recovery and the last two structural recovery, so that partial reconstructions are measured rather than scored as failures.

## 4 Experimental results

### 4.1 Architecture: decoder-only succeeds, encoder-decoder fails (RQ1)

GPT-2 Medium learns every perturbation (Appendix D, Table D1). Stage 1 perplexity falls steadily across all four (Appendix C), and Stage 2 reaches EM 0.82 on FULLREVERSE, 0.80 on PARTIALREVERSE, 0.30 on LOCALSHUFFLE and 0.13 on FULLSHUFFLE. Predicted ADL stays within 0.1 of the natural English value ( $\approx 3.16$ ) on all four. Outputs are shown in Appendix G.

All three T5 conditions fail on every perturbation. Stage 1 perplexity oscillates throughout training with no downward trend (Appendix C), and Stage 2 yields  $EM \approx 0.00$ ,  $BLEU \leq 0.006$ , and  $Dep-F1 \leq 0.001$ . These Dep-F1 scores fall below those of the perturbed input itself, which scores 0.14–0.53 against the original. The outputs consist of repeated tokens (“the”, “a”) rather than reconstructions (see Appendix G). The result is the same under span corruption and prefix LM and under both exposure- and supervision-matched regimes (RQ3). Because reconstruction never occurred, the cross-architecture and within-T5 comparisons the design was built to support (Appendix A) cannot be computed; the remainder of this section reports diagnostics that locate the failure.

### 4.2 Locating the failure (RQ2, RQ3)

The consistent failure of all three T5 variants across objectives and matching schemes motivates a closer

investigation. We conducted a series of diagnostic experiments on LOCALSHUFFLE that identify where and why T5’s encoder-decoder training fails under these conditions.

**OPTIMIZER ABLATION.** Retraining T5-SC with AdamW and a cosine schedule with warmup stabilized Stage 1; evaluation loss no longer oscillated, but plateaued immediately at  $\approx 4.62$  with no downward trend across 6,000 steps (Appendix D, Figure D1). The failure is therefore not an artifact of the training recipe of [Raffel et al. \(2020\)](#), nor is it resolved by replacing it.

**STAGE 2 OVERFIT TEST.** T5-SC could not overfit even 1,000 sentences over 200 epochs (loss plateaued at  $\approx 5.2$ ), so fine-tuning data quantity does not account for the failure.

**STAGE 1 OVERFIT TEST.** We overfit T5-PLM-single on 1,000 perturbed and, separately, 1,000 unperturbed sentences for 800 epochs. Both reduce loss to  $\approx 0.8$  by epoch 800, with oscillation throughout, and show no difference in convergence (Appendix D, Figure D2). T5 therefore fits perturbed text as readily as unperturbed text when given enough exposure per example.

**ZEROED ENCODER TEST.** Replacing the encoder output with a zero matrix changes the decoder’s output, confirming the decoder reads the encoder rather than bypassing it. This locates the failure in the encoder representations.

**ENCODER COSINE SIMILARITY.** We tested whether those representations distinguish inputs at all, computing cosine similarity between encoder hidden states for fifty randomly sampled sentence pairs after Stage 1; a value near 1.0 means the encoder returns nearly identical states for different inputs, passing the decoder no information about sentence content. T5-SC yields 0.989, T5-PLM-single 0.998, and T5-PLM-double 0.996, against 0.822 for a randomly initialized encoder (Appendix D, Table D2). The trained encoders are therefore *less* discriminative than an untrained one.

**ATTENTION ENTROPY.** To identify the reason for the encoder collapse, we measured the mean attention entropy per head across encoder layers after Stage 1. Maximum possible entropy varies with sequence length and averages 3.47 over the evaluation sentences; a randomly initialized encoder yields 3.65, close to that maximum. All three

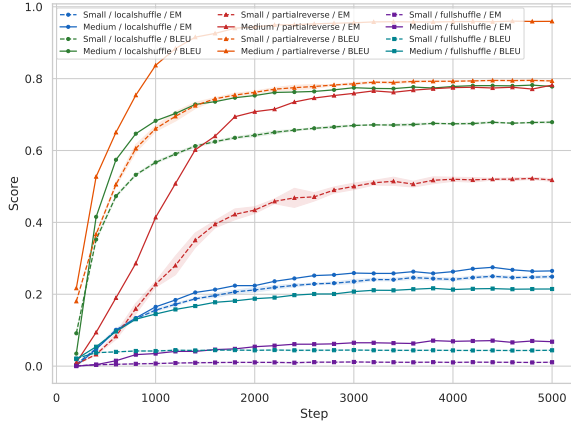


Figure 2: Reconstruction performance of GPT-2 Small (mean over 5 seeds) and GPT-2 Medium (single seed) on LOCALSHUFFLE, FULLSHUFFLE and PARTIALREVERSE at BPE-level perturbations. A table with the results can be seen in Appendix E, Table E3.

trained T5 models fall to between 0.10 and 0.22 by the first measured checkpoint (step 600) and stay flat thereafter (Appendix D, Figure D3), a 96% reduction relative to random initialization. At these values each head places nearly all of its weight on a few tokens regardless of input.

Taken together, the encoder’s attention entropy collapses within the first 600 steps of Stage 1 and never recovers, and its hidden states are near-identical across inputs from that point onward. The collapse occurs under every pretraining objective, matching scheme, optimizer, and fine-tuning data quantity we tested, while GPT-2 Medium shows no comparable behavior under identical data and compute.

#### 4.3 Scale: uneven gains across perturbations (RQ4)

We compare the GPT-2 Small run of Mohammadi et al. (2026) against our GPT-2 Medium at BPE-level tokenization (Figure 2). The comparison covers LOCALSHUFFLE, FULLSHUFFLE and PARTIALREVERSE, the three perturbations for which they report reconstruction results. Scaling improves all three, but the size of the gain varies sharply. PARTIALREVERSE gains most (+0.25 EM), while LOCALSHUFFLE (+0.02) and FULLSHUFFLE (+0.06) gain little on Exact Match.

For FULLSHUFFLE, BLEU and Dependency Triple F1 rise (0.04  $\rightarrow$  0.22 and 0.15  $\rightarrow$  0.39) while EM stays near zero. The same pattern holds for LOCALSHUFFLE (EM +0.02, BLEU +0.10, Dep-F1 +0.07). PARTIALREVERSE, which already ex-

ceeded 0.50 EM at Small, reaches 0.77 EM at Medium, and its lead over LOCALSHUFFLE widens from 0.27 EM at Small to 0.50 at Medium (Figure 2). Predicted ADL remains close to the natural-English value ( $\approx 3.16$ ) across perturbations at both sizes. The EM gains exceed the standard deviations reported by Mohammadi et al. (2026).

#### 4.4 Sentence length (RQ5)

Mohammadi et al. (2026) reported a length effect between two corpora of different average length (*bnc-spoken* and *Gutenberg*). Because that comparison varies corpus and length together, we measure length within *Gutenberg* alone, comparing the lengths of sentences GPT-2 Medium reconstructs against those it fails on (Appendix F, Table F1).

Under every perturbation, reconstructed sentences are shorter than failed ones, but the length the model tolerates tracks how much locality survives. FULLSHUFFLE recovers only very short sentences, with a median of 36 characters and no reconstruction exceeding 85. LOCALSHUFFLE reconstructs to a median of 67 characters, and PARTIALREVERSE and FULLREVERSE to medians of 138 and 129 characters.

### 5 Discussion

We asked whether reconstruction of locality-disrupted input generalizes across architecture families and whether it improves with model size. The two axes gave very different answers. Reconstruction did not transfer to the encoder-decoder architecture under our conditions, and a single increase in decoder-only model size improved reconstruction unevenly, in a way that tracks how much local structure each perturbation leaves intact.

#### Architecture

The architecture axis was built to test whether an encoder-decoder, reading the input bidirectionally, reconstructs better than a decoder-only model. That prediction could not be tested, because every T5 condition failed to reconstruct at all while GPT-2 Medium recovered all four perturbations. The obstacle lies upstream of any bidirectional advantage, in whether the architecture can be trained on such input from scratch at this data scale. The diagnostic chain locates it: not the optimizer, not the fine-tuning data, and not the perturbations themselves, which T5 learns given

enough exposure per example, but an attention entropy collapse in the encoder during pretraining. [Zhai et al. \(2023\)](#) describe the same collapse as a failure mode of transformer training, where it manifests as instability; in our setting it leaves the encoder producing representations that carry no information about the input. GPT-2 shows no comparable failure under the same conditions, which is what one would expect of an architecture that does not depend on an encoder to build representations first, though our experiments do not test that explanation directly.

The three T5 conditions show that the failure is not an artifact of one configuration. Because all three failed at pretraining, the comparisons they were designed to support cannot isolate the effect of objective or supervision quantity; what they show instead is that no setting we tested let T5 escape collapse. If the pretraining objective were the main cause, prefix LM, being structurally closer to GPT-2 Medium’s causal LM, should have outperformed span corruption, and it did not. The failure is consistent with evidence that encoder-decoders need more data before the encoder becomes useful ([Khandelwal et al., 2019](#); [Ogundepo et al., 2022](#)), and with the encoder-decoder advantage of [Raffel et al. \(2020\)](#) having been obtained under far larger budgets than ours. The barrier is therefore one of trainability under these conditions, not a limit of the architecture itself, and whether bidirectional encoding aids reconstruction remains open.

### Scaling

Scaling the decoder-only model improved reconstruction, but not evenly. `PARTIALREVERSE`, which preserves the most local order, gained the most from added capacity, while `LOCALSHUFFLE` and `FULLSHUFFLE` barely moved on Exact Match. Their BLEU and Dependency Triple F1 rise sharply nonetheless, and predicted ADL stays close to natural English throughout. Where information locality is disrupted, then, extra capacity yields well-formed sentences rather than recovery; where more of the locality is retained, the same capacity translates into reconstruction. Scaling laws describe how performance improves with size without reference to what the input contains ([Kaplan et al., 2020](#)). Reconstruction does not follow that pattern: the gain depends on the perturbation, and the gap between the easiest and hardest widens rather than closing.

The length effect is bounded by the same constraint. Within a single corpus, the length a model can reconstruct scales with how much locality survives the perturbation: `FULLSHUFFLE` recovers only its shortest sentences, while the reversals recover ones several times longer. [Mohammadi et al. \(2026\)](#) observed a length effect between two corpora of different average length; our result locates it in the perturbation rather than the corpus, and offers an explanation for why their `FULLSHUFFLE` reconstruction collapsed on the longer corpus.

## 6 Conclusion

This study shows that reconstruction of locality-disrupted input does not generalize across architecture families under our training conditions, and that scaling within the decoder-only family improved reconstruction unevenly. The encoder-decoder result should be interpreted as a failure of trainability under low-resource conditions, not as evidence that the architecture lacks the capacity to reconstruct. Because the encoder never learned to build informative representations, coinciding with an early attention entropy collapse during pretraining, the question of whether bidirectional access benefits reconstruction performance remains open. The scaling result shows that added capacity helps more where local word order survives the perturbation than where it is destroyed.

These findings matter for the impossible-language line of research, where claims about language models, including statements on whether such models reflect properties of human language, have rested almost entirely on one architecture and one scale. Together, our findings show that architecture and scale should not be considered neutral methodological choices and conclusions should be framed accordingly. The diagnostic chain used to locate the encoder-decoder failure further provides a reusable procedure for distinguishing trainability failures from architectural and capacity limits in low-resource training.

## 7 Limitations

Our central negative result is bounded by the regime in which it was obtained. T5-base was pretrained from scratch on approximately 100M tokens, far below the budget under which [Raffel et al. \(2020\)](#) established the architecture, and we did not conduct a hyperparameter search: all three T5 conditions share one recipe, varied only by the

optimizer ablation reported in §4. We therefore cannot rule out that a different learning rate, warmup schedule, or initialization would avoid the collapse. The finding should be read as showing that the encoder-decoder family fails under conditions in which the decoder-only family succeeds, not that it must fail, and whether the collapse persists at larger data scale remains open.

Our diagnostics identify attention entropy collapse but do not test an intervention against it, as each candidate fix requires retraining from scratch. As the collapse may be attributed to an interaction between T5’s training dynamics and the conditions of this study, it is left open whether this failure persists at a larger data scale. An architectural intervention is proposed in the literature to prevent attention entropy collapse in a 100-layer encoder-decoder model. Zhai et al. (2023) show that spectral normalization of the linear layers prevents collapse. Whether this intervention would enable successful T5 pretraining on impossible languages at BabyLM data scale for a 12-layer model is an open question that we leave to future work.

## References

- Vladimir Araujo, Maria Mihaela Trusca, Rodrigo Tufiño, and Marie-Francine Moens. 2024. [Sequence-to-sequence Spanish pre-trained language models](#). *Preprint*, arXiv:2309.11259.
- Lalit R. Bahl, Frederick Jelinek, and Robert L. Mercer. 1983. [A maximum likelihood approach to continuous speech recognition](#). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5(2):179–190.
- Sabine Buchholz and Erwin Marsi. 2006. [CoNLL-X shared task on multilingual dependency parsing](#). In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 149–164, New York City. Association for Computational Linguistics.
- Shu Cai and Kevin Knight. 2013. [Smatch: an evaluation metric for semantic feature structures](#). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752, Sofia, Bulgaria. Association for Computational Linguistics.
- Noam Chomsky, Ian Roberts, and Jeffrey Watumull. 2023. [Noam Chomsky: The false promise of ChatGPT](#). *The New York Times*.
- Aleksandra Chrabrowa, Łukasz Dragan, Karol Grzegorzczak, Dariusz Kajtoch, Mikołaj Koszowski, Robert Mroczkowski, and Piotr Rybak. 2022. [Evaluation of transfer learning for Polish with a text-to-text model](#). *Preprint*, arXiv:2205.08808.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Zihao Fu, Wai Lam, Qian Yu, Anthony Man-Cho So, Shengding Hu, Zhiyuan Liu, and Nigel Collier. 2023. [Decoder-only or encoder-decoder? interpreting language model as a regularized encoder-decoder](#). *Preprint*, arXiv:2304.04052.
- Richard Futrell. 2019. [Information-theoretic locality properties of natural language](#). In *Proceedings of the First Workshop on Quantitative Syntax (Quasy, SyntaxFest 2019)*, pages 2–15, Paris, France. Association for Computational Linguistics.
- 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.
- Daniel Jurafsky and James H. Martin. 2026. [Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models](#), 3rd edition. Online manuscript released January 6, 2026.
- Julie Kallini, Isabel Papadimitriou, Richard Futrell, Kyle Mahowald, and Christopher Potts. 2024. [Mission: Impossible language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14691–14714, Bangkok, Thailand. Association for Computational Linguistics.
- Julie Kallini and Christopher Potts. 2025. [Language models as tools for investigating the distinction between possible and impossible natural languages](#). *Preprint*, arXiv:2512.09394.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. [Scaling laws for neural language models](#). *Preprint*, arXiv:2001.08361.
- Urvashi Khandelwal, Kevin Clark, Dan Jurafsky, and Lukasz Kaiser. 2019. [Sample efficient text summarization using a single pre-trained transformer](#). *Preprint*, arXiv:1905.08836.

- Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. [Data augmentation using pre-trained transformer models](#). In *Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems*, pages 18–26, Suzhou, China. Association for Computational Linguistics.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. [BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.
- Haitao Liu. 2008. [Dependency distance as a metric of language comprehension difficulty](#). *Journal of Cognitive Science*, 9(2):159–191.
- Amirhossein Mohammadi, Laurence E. Frank, Albert Gatt, and Robert A. Bagheri. 2026. [Language re-generation: An investigation into information locality effects on reconstruction](#). *Preprint*, arXiv:2607.10268.
- Andrea Moro, Matteo Greco, and Stefano F. Cappa. 2023. [Large languages, impossible languages and human brains](#). *Cortex*, 167:82–85.
- Odunayo Jude Ogundepo, Akintunde Oladipo, Mofetoluwa Adeyemi, Kelechi Ogueji, and Jimmy Lin. 2022. [AfriTeVA: Extending “small data” pretraining approaches to sequence-to-sequence models](#). In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 126–135, Hybrid. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#). Technical report.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Taiga Someya, Anej Svete, Brian DuSell, Timothy J. O’Donnell, Mario Giulianelli, and Ryan Cotterell. 2025. [Information locality as an inductive bias for neural language models](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 27995–28013, Vienna, Austria. Association for Computational Linguistics.
- Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Q. Tran, Dani Yogatama, and Donald Metzler. 2022. [Scaling laws vs model architectures: How does inductive bias influence scaling?](#) *Preprint*, arXiv:2207.10551.
- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. [BERT rediscovers the classical NLP pipeline](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.
- Alessio Tosolini and Terra Blevins. 2025. [Analyzing the linguistic priors of language models with synthetic languages](#). In *Proceedings of the 7th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 7–15, Vienna, Austria. Association for Computational Linguistics.
- Thomas Wang, Adam Roberts, Daniel Hesslow, Teven Le Scao, Hyung Won Chung, Iz Beltagy, Julien Launay, and Colin Raffel. 2022. [What language model architecture and pretraining objective work best for zero-shot generalization?](#) *Preprint*, arXiv:2204.05832.
- 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.
- Tianyang Xu, Tatsuki Kuribayashi, Yohei Oseki, Ryan Cotterell, and Alex Warstadt. 2025. [Can language models learn typologically implausible languages?](#) *Preprint*, arXiv:2502.12317.
- Xiulin Yang, Tatsuya Aoyama, Yuekun Yao, and Ethan Wilcox. 2025. [Anything goes? a crosslinguistic study of \(im\)possible language learning in LMs](#). *Preprint*, arXiv:2502.18795.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatuo Gu, and Josh Susskind. 2023. [Stabilizing transformer training by preventing attention entropy collapse](#). *Preprint*, arXiv:2303.06296.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2024. [E2S2: Encoding-enhanced sequence-to-sequence pretraining for language understanding and generation](#). *IEEE Transactions on Knowledge and Data Engineering*, 36(12):8037–8050.

Imry Ziv, Nur Lan, and Emmanuel Chemla. 2026. [Bias-less language models learn unnaturally: How LLMs fail to distinguish the possible from the impossible](#). In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5393–5403, Rabat, Morocco. Association for Computational Linguistics.

## A Experimental design

This appendix gives the full design of the T5-base pretraining conditions introduced in §3.3 and of the GPT-2 scaling comparison.

**Step counts.** Both models process approximately 1.57B tokens in Stage 1: GPT-2 Medium in 3,000 steps at sequence length 1,024, and T5-base in 6,000 steps at sequence length 512, the shorter length following from T5’s positional-embedding range. T5-PLM-double is trained for 12,000 steps, matching GPT-2 Medium on decoder-supervised tokens rather than tokens seen and therefore processing roughly twice the data.

**Prefix-LM split.** For the prefix-LM conditions, each sentence is split at a point  $k$  sampled uniformly at random; the encoder reads the tokens up to  $k$  as a bidirectional prefix and the decoder generates the remainder autoregressively. Because  $k$  is uniform, the decoder is supervised on approximately 50% of tokens in aggregate. For T5-PLM-double the split is resampled on each pass, so the additional data exposure consists of different prefix-target decompositions of the same sentences rather than repeated identical examples.

**Excluded condition.** A fourth condition, span corruption matched on decoder supervision (T5-SC-double), is excluded from this research due to computational constraints and overfitting risk, as it requires the model to train approximately  $6.7\times$  longer than GPT-2 Medium to compensate for the 15% supervision rate.

### Experimental design

Had the T5 models trained successfully, the

following comparisons would isolate each source of variation. Because all T5 conditions failed to reconstruct (§4), none of these comparisons can be evaluated here. We set out the design in full because it applies directly to future work: if the trainability barrier identified in §4 is overcome, these comparisons isolate the effects this study could not measure.

#### A.0.1 Effect of model architecture

This section sets out the experimental design for comparing the encoder-decoder architecture (T5-base) against the decoder-only architecture (GPT-2 Medium) on the task of reconstructing impossible languages. The three-condition T5 design introduced in §3.3 allows cross-architecture comparison to be examined under two matching schemes and additionally allows the effects of pretraining objective and decoder supervision quantity to be isolated within T5. The architecture comparison therefore decomposes into four experiments, each isolating one source of variation while holding the others constant.

#### Experiment 1: matched token exposure

##### 1A. GPT-2 Medium vs. T5-SC

The comparison between GPT-2 Medium and T5-SC holds total pretraining tokens constant across the two architectures. GPT-2’s causal LM supervises 100% of tokens, while T5’s span-corruption supervises approximately 15%. This contrast asks whether T5 is competitive with GPT-2 Medium when the two are matched on data exposure but T5 receives substantially less token supervision.

##### 1B. GPT-2 Medium vs. T5-PLM-single

The comparison between GPT-2 Medium and T5-PLM-single also holds total pretraining tokens constant but compares GPT-2’s causal LM to prefix-LM rather than span corruption. The decoder supervision rate gap is smaller: prefix-LM supervises approximately 50% of tokens, compared to span corruption’s 15%. This contrast asks whether T5’s performance shifts when the pretraining objective is closer in structure to GPT-2’s.

#### Experiment 2: matched token supervision

The comparison between GPT-2 Medium and T5-PLM-double holds total decoder supervision constant across the two architectures. Both models receive gradient signal on the same number of

tokens in absolute terms across training, but T5 sees the pretraining corpus approximately twice as many times to reach that supervision quantity. This contrast asks whether T5 is competitive with GPT-2 Medium when supervision is matched but data exposure is not.

The three cross-architecture comparisons are complementary because each holds a different quantity constant and leaves a different quantity confounded: T5-SC matches data but leaves a large supervision gap, T5-PLM-single matches data with a smaller supervision gap, and T5-PLM-double matches supervision at the cost of data exposure.

**Experiment 3: effect of pretraining objective (within T5)** The comparison between T5-SC and T5-PLM-single holds architecture, parameter count, and total exposure constant, and varies only in pretraining objective. This contrast isolates the effect of pretraining objective on the reconstruction task and tests whether closer alignment in pretraining and fine-tuning tasks translates into better performance.

**Experiment 4: effect of decoder supervision quantity (within T5)** The comparison between T5-PLM-single and T5-PLM-double holds architecture, parameter count, and pretraining objective constant, and varies only in the amount of decoder supervision. This contrast isolates the effect of supervision quantity on the reconstruction task and tests whether additional decoder supervision improves T5’s performance under the prefix LM objective.

#### A.0.2 Effect of model size

This section sets out the experimental design for comparing GPT-2 Small with GPT-2 Medium on the task of reconstructing impossible languages.

**Experiment 5: effect of model size** The comparison between GPT-2 Small and GPT-2 Medium holds architecture, objective, perturbation level (BPE), training recipe and data constant, and varies only in scale. This isolates the effect of model size on reconstruction performance and tests whether performance scales with model size. The comparison is restricted to LOCALSHUFFLE, FULLSHUFFLE and PARTIALREVERSE, the three perturbations for which Mohammadi et al. (2026) report reconstruction results.

Table A1 shows an overview of every model comparison.

| Comparison                         | Held constant             | Tests                                          |
|------------------------------------|---------------------------|------------------------------------------------|
| 1A. GPT-2 Medium vs. T5-SC         | Tokens seen               | Architecture & objective (exposure-matched)    |
| 1B. GPT-2 Medium vs. T5-PLM-single | Tokens seen               | Architecture & objective (exposure-matched)    |
| 2. GPT-2 Medium vs. T5-PLM-double  | Decoder supervision       | Architecture & objective (supervision-matched) |
| 3. T5-SC vs. T5-PLM-single         | Tokens seen, architecture | Pretraining objective within same architecture |
| 4. T5-PLM-single vs. T5-PLM-double | Architecture, objective   | Decoder supervision quantity                   |
| 5. GPT-2 Medium vs. GPT-2 Small    | Tokens seen               | decoder-only scaling (exposure-matched)        |

Table A1: Architecture comparisons in Stage 2 evaluation. Each row isolates one source of variation by holding the others constant.

## B Training details

### Perturbation details

We use published seeds from [Kallini et al. \(2024\)](#) for all four perturbations (0 for LOCALSHUFFLE, 57 for FULLSHUFFLE, 21 for both PARTIALREVERSE and FULLREVERSE). The implementations at BPE-level are identical to the reference code of [Kallini et al. \(2024\)](#) with one deviation. Their code applies the same permutation every window in a sentence rather than an independent permutation per window for LOCALSHUFFLE. We instead sample an independent permutation per window, matching the implementation of [Mohammadi et al. \(2026\)](#) while preserving reproducibility.

### Stage 1 pretraining implementation details

Stage 1 hyperparameters are shown in Table B1. All models were pretrained using transformer models from Hugging Face with PyTorch on NVIDIA H100 GPUs. Step counts are set so both models process approximately the same total tokens during Stage 1 (1.57B): GPT-2 Medium for 3,000 steps at sequence length 1,024, T5-base for 6,000 steps at sequence length 512. The difference in sequence length reflects the positional embedding range of each model.

GPT-2 Medium uses a linear warmup and decay schedule. T5-base uses a constant learning rate schedule with warmup, which deviates from the inverse square root schedule of [Raffel et al. \(2020\)](#). This setting was chosen for training stability at this scale. Warmup steps for T5-base are set to 120, matching the ratio of [Raffel et al. \(2020\)](#) of approximately 2% of total steps (10,000 of 524,288). Full configuration files are available on <https://anonymous.4open.science/r/Impossible-reconstruction-33ED>.

Table B1: Stage 1 hyperparameters. T5-PLM-double uses identical settings to T5-SC and T5-PLM-single except for 12,000 steps and 240 warmup steps.

|                 | GPT-2 Medium | T5-base   |
|-----------------|--------------|-----------|
| Optimizer       | AdamW        | Adafactor |
| Learning rate   | 6e-4         | 1e-2      |
| Warmup steps    | 300          | 120       |
| Weight decay    | 0.01         | 0.01      |
| Max grad norm   | 1.0          | 1.0       |
| Steps           | 3,000        | 6,000     |
| Sequence length | 1,024        | 512       |
| Batch size      | 512          | 512       |
| Precision       | bf16         | fp32      |

### Stage 2 fine-tuning implementation details

*GPT-2 Medium* is fine-tuned with the causal LM objective under a fixed prompt template that includes both the perturbed input ( $A'$ ) and its reconstruction target ( $A$ ):

```
Fix this text: <impossible_text>
Corrected: <possible_text><|endoftext|>
```

Following [Mohammadi et al. \(2026\)](#), prompt tokens up to and including “Corrected:” are masked with label  $-100$ . This ensures that the loss is computed only on the target sentence  $A$  and the *end-of-text* token.

All models were fine-tuned using the same infrastructure as Stage 1. Both architectures use AdamW with a cosine schedule. GPT-2 Medium settings are mostly based on [Mohammadi et al. \(2026\)](#). T5-base uses fp32 precision without label smoothing as T5 produces NaN with label smoothing enabled for fine-tuning. Stage 2 hyperparameters are shown in Table B2.

Table B2: Stage 2 hyperparameters. Settings are identical across all T5 models (T5-SC, T5-PLM-single, T5-PLM-double)

|                 | GPT-2 Medium | T5-base |
|-----------------|--------------|---------|
| Optimizer       | AdamW        | AdamW   |
| Learning rate   | 2e-5         | 1e-4    |
| Warmup steps    | 500          | 500     |
| Weight decay    | 0.01         | 0.01    |
| Max grad norm   | 1.0          | 1.0     |
| Steps           | 5,000        | 5,000   |
| Batch size      | 512          | 512     |
| Label smoothing | 0.1          | 0.0     |
| Precision       | bf16         | fp32    |
| Max length      | 512          | 512     |
| Data seed       | 123          | 123     |

## C Stage 1 perplexity curves

Stage 1 perplexity curves for GPT-2 Medium and all T5 variants across the four perturbations are shown in Figures C1, C2, C3, and C4. GPT-2 Medium shows a consistent downward trend across all conditions. All T5 variants show oscillating perplexity with no downward trend throughout training.

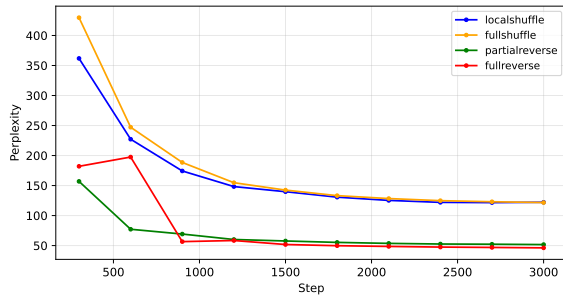


Figure C1: Stage 1 perplexity curves for GPT-2 Medium (word-level) across all four perturbations.

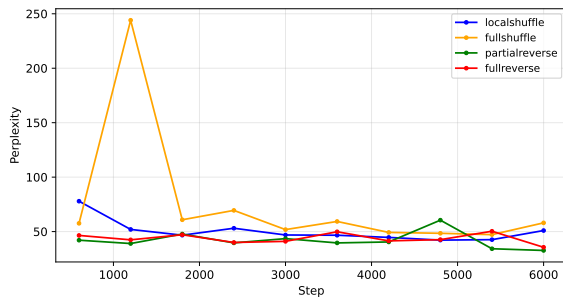


Figure C2: Stage 1 perplexity curves for T5-SC across all four perturbations.

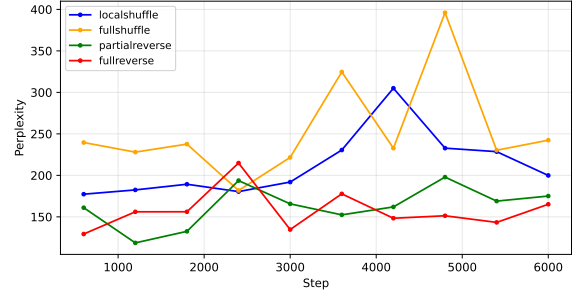


Figure C3: Stage 1 perplexity curves for T5-PLM-single across all four perturbations.

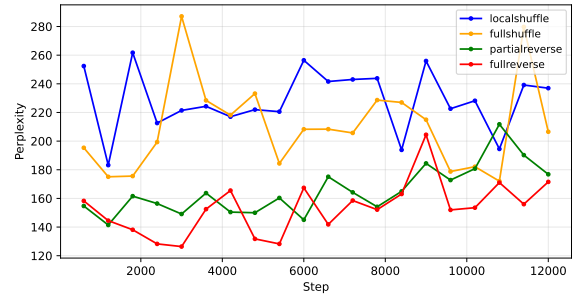


Figure C4: Stage 1 perplexity curves for T5-PLM-double across all four perturbations.

## D Diagnostic experiments T5 failure

This appendix reports the full results of the diagnostic experiments described in §4. Table D1 shows the failure of every T5 model.

| Pert.           | Model    | EM   | BLEU | Dep-F1 | ADL  |
|-----------------|----------|------|------|--------|------|
| Local Shuffle   | GPT-2 M  | 0.30 | 0.80 | 0.86   | 3.23 |
|                 | T5-SC    | 0.00 | 0.00 | 0.00   | 3.83 |
|                 | T5-PLM-s | 0.00 | 0.01 | 0.00   | 2.99 |
|                 | T5-PLM-d | 0.00 | 0.01 | 0.00   | 8.44 |
| Full Shuffle    | GPT-2 M  | 0.13 | 0.33 | 0.46   | 3.18 |
|                 | T5-SC    | 0.00 | 0.00 | 0.00   | 3.74 |
|                 | T5-PLM-s | 0.00 | 0.01 | 0.00   | 8.26 |
|                 | T5-PLM-d | 0.00 | 0.01 | 0.00   | 4.53 |
| Partial Reverse | GPT-2 M  | 0.80 | 0.97 | 0.98   | 3.15 |
|                 | T5-SC    | 0.00 | 0.01 | 0.00   | 3.47 |
|                 | T5-PLM-s | 0.00 | 0.01 | 0.00   | 4.94 |
|                 | T5-PLM-d | 0.00 | 0.01 | 0.00   | 4.36 |
| Full Reverse    | GPT-2 M  | 0.82 | 0.97 | 0.97   | 3.25 |
|                 | T5-SC    | 0.00 | 0.01 | 0.00   | 3.86 |
|                 | T5-PLM-s | 0.00 | 0.00 | 0.00   | 5.57 |
|                 | T5-PLM-d | 0.00 | 0.01 | 0.00   | 5.57 |

Table D1: Architecture comparison at word-level perturbations. (ADL of natural English  $\approx 3.16$ ). Abbreviations are used: GPT-2 M = GPT-2 Medium, T5-PLM-s = T5-PLM-single, T5-PLM-d = T5-PLM-double

### T5-SC optimizer ablation

T5-SC was retrained on LOCALSHUFFLE with

AdamW, learning rate  $1e-3$ , cosine schedule and 300 warmup steps, with all other settings identical to T5-SC (Appendix B). T5-SC v2 stabilizes but plateaus immediately at approximately 4.62 with no learning across 6,000 steps. This optimizer ablation shows that training instability is not the main cause of T5’s failure.

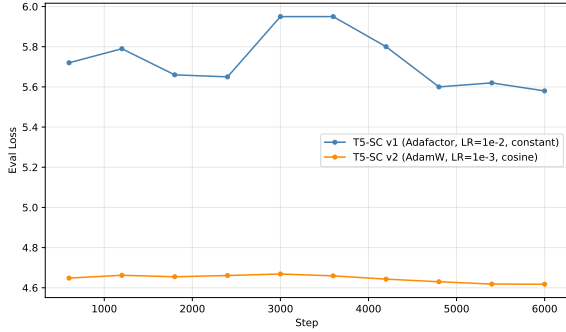


Figure D1: Stage 1 evaluation loss curves for T5-SC v1 (Adafactor,  $LR=1e-2$ , constant schedule) and T5-SC v2 (AdamW,  $LR=1e-3$ , cosine schedule) on LocalShuffle. T5-SC v1 oscillates throughout training.

### Encoder cosine similarity

Mean cosine similarity between encoder hidden states for fifty randomly sampled sentence pairs, computed after Stage 1 training. A randomly initialized T5 encoder is included as a baseline.

Table D2: Mean encoder cosine similarity after Stage 1 training.

| Model          | Mean   | Std    |
|----------------|--------|--------|
| Random init T5 | 0.8222 | 0.0921 |
| T5-SC          | 0.9889 | 0.0125 |
| T5-PLM-single  | 0.9982 | 0.0016 |
| T5-PLM-double  | 0.9958 | 0.0060 |

### Stage 1 overfit test

Figure D2 shows training loss over 800 epochs for T5-PLM-single trained on 1,000 LocalShuffle sentences and 1,000 natural English sentences without dropout or weight decay to overfit the models. Both conditions reduce training loss to approximately 0.8 with oscillation throughout, with no clear visible difference between the two conditions.

### Attention entropy trajectories

The mean attention entropy per head across encoder layers at each Stage 1 checkpoint for T5-PLM-single is visualized for LOCALSHUFFLE.

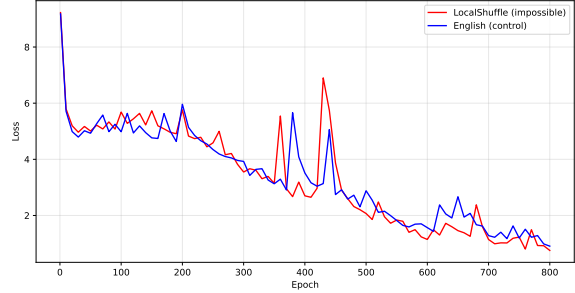


Figure D2: Stage 1 overfit test: training loss over 800 epochs for T5-PLM-single on 1,000 impossible and 1,000 natural English sentences.

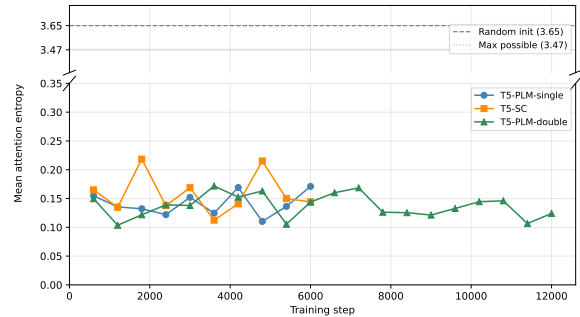


Figure D3: The mean attention entropy per head across encoder layers at each Stage 1 checkpoint for T5-PLM-single for LOCALSHUFFLE. The maximum possible entropy varies with sequence length and averages 3.47 across the sentences used in this evaluation. A randomly initialized encoder yields 3.65.

## E Scale comparison results

Table E3 gives the results of the scaling comparison visualized in Figure 2.

| Perturbation   | Model        | EM               | BLEU             | Dep-F1           | ADL              |
|----------------|--------------|------------------|------------------|------------------|------------------|
| LocalShuffle   | GPT-2 Small  | $0.25 \pm 0.002$ | $0.68 \pm 0.001$ | $0.77 \pm 0.002$ | $3.16 \pm 0.009$ |
|                | GPT-2 Medium | 0.27             | 0.78             | 0.84             | 3.18             |
| FullShuffle    | GPT-2 Small  | $0.01 \pm 0.001$ | $0.04 \pm 0.001$ | $0.15 \pm 0.002$ | $3.19 \pm 0.006$ |
|                | GPT-2 Medium | 0.07             | 0.22             | 0.39             | 3.20             |
| PartialReverse | GPT-2 Small  | $0.52 \pm 0.004$ | $0.79 \pm 0.001$ | $0.84 \pm 0.002$ | $3.16 \pm 0.008$ |
|                | GPT-2 Medium | 0.77             | 0.96             | 0.97             | 3.16             |

Table E3: Scaling comparison: reconstruction performance of GPT-2 Small (Mohammadi et al. (2026); mean  $\pm$  std over five seeds) and GPT-2 Medium (single seed) at BPE-level tokenization. The comparison is restricted to the three perturbations that Mohammadi et al. (2026) reported.

## F Sentence length and reconstruction

For each perturbation, we split the 1,000 Gutenberg evaluation sentences into those GPT-2 Medium reconstructed exactly (EM=1) and those it did not (EM=0), and report the length distribution of each group. Lengths are in characters of the original sentence. Table F1 shows that reconstructed sentences are consistently shorter than failed ones, and that the length tolerated under success grows as the perturbation preserves more locality.

Table F1: Character length of reconstructed (EM=1) vs. failed (EM=0) sentences per perturbation, on the 1,000 Gutenberg evaluation pairs (GPT-2 Medium, BPE level).

| Perturbation   | Group   | $n$ | Median | Mean | Min | Max  |
|----------------|---------|-----|--------|------|-----|------|
| FULLSHUFFLE    | Success | 71  | 36     | 38   | 13  | 85   |
|                | Failure | 929 | 168    | 273  | 9   | 4209 |
| LOCALSHUFFLE   | Success | 275 | 67     | 91   | 9   | 400  |
|                | Failure | 725 | 219    | 320  | 17  | 4209 |
| FULLREVERSE    | Success | 658 | 129    | 177  | 9   | 1098 |
|                | Failure | 342 | 270    | 410  | 17  | 4209 |
| PARTIALREVERSE | Success | 782 | 138    | 193  | 9   | 1525 |
|                | Failure | 218 | 320    | 486  | 19  | 4209 |

## G Reconstruction examples

### G.1 T5 examples

Table G2 shows the predictions of the T5 models for one example sentence across the four perturbations. The predictions are shown to be repetitive and uninformative. The output also shows that the models memorized tokens from pretraining: MOT and CHI are speaker labels from the CHILDES transcripts in the BabyLM pretraining corpus, where MOT stands for the mother’s turn to speak and CHI for child’s turn to speak.

### G.2 GPT-2 Medium examples

Table G3 shows the predictions of GPT-2 Medium on BPE level perturbations. This shows the difference in reconstruction difficulty.

| Model         | LocalShuffle                                                                                                     | FullShuffle                                                                                                      | PartialReverse                                                                                                                                                                   | FullReverse                                                                                                                                                                      |
|---------------|------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Original      | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any one I know to see it." | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any one I know to see it." | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any one I know to see it."                                                                 | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any one I know to see it."                                                                 |
| Perturbed     | it's "Well, all the there in now, house but you'd have to good a be than smarter deal one any I to see know it." | one "Well, to any you'd house there good a the now, in all it's have smarter but than be I it." to see deal know | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any <span style="border: 1px solid black; padding: 0 2px;">R</span> it." see to know I one | it." see to know I one <span style="border: 1px solid black; padding: 0 2px;">R</span> any than smarter deal good a be to have you'd but now, house the in there all it's "Well, |
| T5-SC         | T ., the a a. a a                                                                                                | Relevan Relevan Relevan Relevan Relevan Relevan Relevan Relevan Relevan. Relevan Relevan Relevan Relevan         | R . R R a.                                                                                                                                                                       | a R. R R . RT: R a a R. a. R                                                                                                                                                     |
| T5-PLM-single | a a a the to, the and a, he a, was a a the of a. a the a the of the of the and                                   | a a a the the a and a the the a a was the the the a the a *MOT: ah, a a                                          | . R a a a a R a's a a was a a to the a R a's a was                                                                                                                               | a *MOT: R *MOT: R . R *MOT: R *MOT: R a R *MOT:                                                                                                                                  |
| T5-PLM-double | a a a a a he he a a a a a a *MOT: a a a a                                                                        | a a. a you a a he he's, a. a *CHI: a a                                                                           | aaa R aahe't. R ah. R a ah. R a., R a                                                                                                                                            | a a and R *MOT: R *MOT: R *MOT: R a *CHI: R                                                                                                                                      |

Table G2: T5 reconstruction predictions at the best EM checkpoint for one example sentence across all four perturbations. All T5 variants achieve EM  $\approx 0.00$  on every perturbation. **Green** marks contiguous spans of words matching the original sentence; **red** marks differences.

|            | LocalShuffle                                                                                                       | FullShuffle                                                     | PartialReverse                                                                                   | FullReverse                                                                                                                   |
|------------|--------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------|--------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Original   | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any one I know to see it."   | "I never believed there was such a good woman in this world!"   | "Thank 'ee kindly, sir, but don't 'ee go against the law. The law be a terrible creature."       | "Dear me! Then I must ask the driver to come to the house. My notes are in my dressing-case. One cannot be too careful."      |
| Perturbed  | Well," it all's the there in now house, but you'd to be have a good deal any smarter than one know I it see to."   | never a wasI there good such believed woman this world! in      | "Thank 'ee kindly, sir, but don't 'ee go against the law. The R." creature terrible a be law     | ." careful too be cannot One .case- dressing R my in are notes My . house the to come to driver the ask must I Then ! meDear" |
| Prediction | "Well, it's the in bed-room, and you'd better get a good deal of it."<br>"What's the matter, anyway?"              | "I was a very good thing to be a very good thing."              | "Thank you'ee kindly, sir, sir, and don't 'ee go against the law. The law is law. The law is law | "Oh, I shall go to me to the cab-room. And I am not go to the kitchen-room. If I can't be a little little pocket.             |
| Checkpoint | step 200                                                                                                           | step 200                                                        | step 200                                                                                         | step 200                                                                                                                      |
| Prediction | "Well, it's all in the house now, but you'd have to be a good deal any than smarter one I know to see it."         | "I never knew in such a good world!" was there a woman.         | "Thank 'ee kindly, sir, but don't 'ee go against the law. The law is a terrible creature."       | "Dear me! Then I must ask the driver to come to the house. My notes are in my dressing-case. One cannot be too careful."      |
| Checkpoint | step 600                                                                                                           | step 800                                                        | step 600                                                                                         | step 600                                                                                                                      |
| Prediction | "Well, it's all there in the house, now but you'd be to have a good deal than any smarter one I know it to see."   | "I never believed such a good woman in this world!"             | "Thank 'ee kindly, sir, but don't 'ee go against the law. The law is a terrible creature."       | "Dear me! Then I must ask the driver to come. My notes are in my dressing-case. One cannot be too careful."                   |
| Checkpoint | step 800                                                                                                           | step 1400                                                       | step 800                                                                                         | step 800                                                                                                                      |
| Prediction | "Well, it's all there in the house now, but you'd have to be a good deal smarter than any one I know to see it." ✓ | "I never believed there was such a good woman in this world!" ✓ | "Thank 'ee kindly, sir, but don't 'ee go against the law. The law be a terrible creature." ✓     | "Dear me! Then I must ask the driver to come to the house. My notes are in my dressing-case. One cannot be too careful." ✓    |
| Checkpoint | step 1000                                                                                                          | step 2000                                                       | step 1200                                                                                        | step 1000                                                                                                                     |

Table G3: GPT-2 Medium (BPE) reconstruction at selected checkpoints for one example sentence per perturbation type. **Green** marks contiguous spans of words matching the original sentence; **red** marks differences.