

Word Order Does Matter (And Shuffled Language Models Know It)

***Mostafa Abdou**[†] ***Vinit Ravishankar**[‡] **Artur Kulmizev**[§] **Anders Søgaard**[†]

[†]Department of Computer Science, University of Copenhagen

[‡]Language Technology Group, Department of Informatics, University of Oslo

[§]Department of Linguistics and Philology, Uppsala University

† {abdou, soegaard}@di.ku.dk

‡ vinitr@ifi.uio.no

Abstract

Recent studies have shown that language models pretrained and/or fine-tuned on randomly permuted sentences exhibit competitive performance on GLUE, putting into question the importance of word order information. Somewhat counter-intuitively, some of these studies also report that position embeddings appear to be crucial for models’ good performance with shuffled text. We probe these language models for word order information and investigate what position embeddings learned from shuffled text encode, showing that these models retain information pertaining to the original, naturalistic word order. We show this is in part due to a subtlety in how shuffling is implemented in previous work – *before* rather than *after* subword segmentation. Surprisingly, we find even Language models trained on text shuffled *after* subword segmentation retain some semblance of information about word order because of the statistical dependencies between sentence length and unigram probabilities. Finally, we show that beyond GLUE, a variety of language understanding tasks do require word order information, often to an extent that cannot be learned through fine-tuning.

1 Introduction

Transformers (Vaswani et al., 2017), when used in the context of masked language modelling (Devlin et al., 2018), consume their inputs concurrently. There is no notion of inherent order, unlike in autoregressive setups, where the input is consumed token by token. To compensate for this absence of linear order, the transformer architecture originally proposed in Vaswani et al. (2017) includes a fixed, sinusoidal position embedding added to each token embedding; each token carries a different position embedding, corresponding to its position in the sentence. The transformer-based BERT (Devlin et al., 2018) replaces these fixed sinusoidal

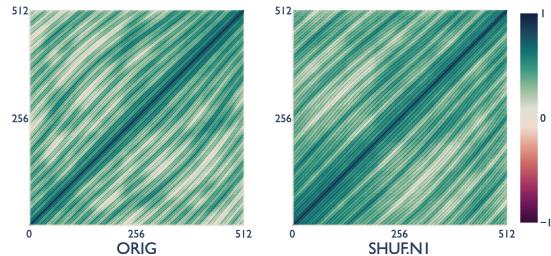


Figure 1: Pearson correlations between position embeddings for full-scale models; the patterns are similar to fully learnable absolute embeddings (Wang et al., 2021) and can be said to have learned something about position. We later demonstrate that this is not the case with post-BPE scrambling.

embeddings with unique, learned embeddings per position; RoBERTa (Liu et al., 2019), the model investigated in this work, does the same.

Position embeddings are the only source of order information in these models; in their absence, contextual representations generated for tokens are independent of the actual position of the tokens in a sentence, and the models thus resemble heavily overparameterised bags-of-words. Sinha et al. (2021) pre-trained RoBERTa models on shuffled corpora to demonstrate that the performance gap between these ‘shuffled’ language models and models trained on unshuffled corpora is minor (when fine-tuned and evaluated downstream on the GLUE (Wang et al., 2018) benchmark). They further show that this gap is considerably wider when a model is pre-trained without position embeddings. In this paper, we attempt to shed some light on why these models behave the way they do, and in doing so, seek to answer a set of pertinent questions:

- Do shuffled language models still have traces of word order information?
- Why is there a gap in performance between models *without* position embeddings and models trained on shuffled tokens, with the latter

*Equal contribution. Order was decided by a coin toss.

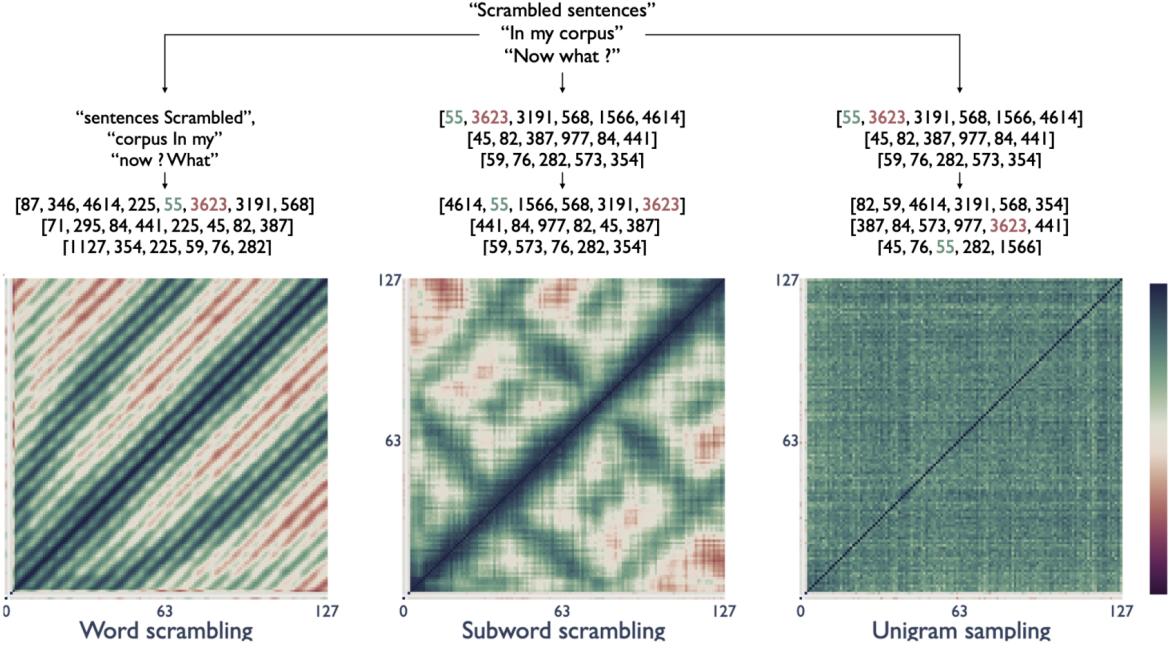


Figure 2: Correlations between position embeddings when shuffling training data *before* segmentation (left), i.e., at the word level, and *after* segmentation (middle), i.e., at the subword level, as well as when replacing all subwords with random subwords based on their corpus-level frequencies (right). The latter removes any dependency between subword probability and sentence length. The plots show that shuffling before segmentation retains more order information than shuffling after, and that even when shuffling after segmentation, position embeddings are meaningful because of the dependence between subword probability and sentence length.

performing better?

- Are there NLU benchmarks, other than GLUE, on which shuffled language models perform poorly?

Contributions We first demonstrate, in Section 3, that shuffled language models *do* contain word order information, and are quite responsive to simple tests for word order information, particularly when compared to models trained without position representations. In Section 4, we demonstrate that pre-training is sufficient to learn this: position embeddings provide the appropriate inductive bias, and performing BPE segmentation after shuffling results in sensible n-grams appearing in the pre-training corpus; this gives models the capacity to learn word order within smaller local windows. Other minor cues - like correlations between sentence lengths and token distributions - also play a role. We further corroborate our analysis by examining attention patterns across models in Sec. 5. In Section 6, we show that, while shuffled models might be almost as good as their unshuffled counterparts on GLUE tasks, there exist NLU benchmarks that do require word order information to an extent that cannot be learned through

fine-tuning alone. Finally, in Section 7, we describe miscellaneous experiments addressing the utility of positional embeddings when added just prior to fine-tuning.

2 Models

Sinha et al. (2021) train several full-scale RoBERTa language models on the Toronto Book Corpus (Zhu et al., 2015) and English Wikipedia.¹ Four of their models are trained on shuffled text, i.e., sentences in which n -grams are reordered at random.² We dub the original, unperturbed model ORIG, and the scrambled models SHUF.N1, SHUF.N2, SHUF.N3 and SHUF.N4 depending on the size of the shuffled n -grams: SHUF.N1 reorders the unigrams in a sentence, SHUF.N2 reorders its bigrams, etc. For comparison, Sinha et al. (2021) also train a RoBERTa language model entirely *without* position embeddings (NOPOS), as well as a RoBERTa language model trained on a corpus drawn solely from unigram distributions of the original Book Corpus, i.e., a reshuffling of the entire corpus (SHUF.CORPUS).

¹Training reportedly takes 72 hours on 64 GPUs.

²The shuffling procedure does not reorder tokens *completely* at random, but moves a token in position i to a *new* position selected at random among positions $j \neq i$.

We experiment with their models, as well as with smaller models that we can train with a smaller carbon footprint. To this end, we downscale the RoBERTa architecture used in [Sinha et al. \(2021\)](#). Concretely, we train single-headed RoBERTa models, dividing the embedding and feed-forward dimensionality by 12, for 24 hours on a single GPU, on 100k sentences sampled from the Toronto Book Corpus. To this end, we train a custom vocabulary of size 5,000, which we use for indexing in all our subsequent experiments. While these smaller models are in no way meant to be fine-tuned and used downstream, they are useful proofs-of-concept that we later analyse.

3 Probing for word order

We begin by attempting to ascertain the extent to which shuffled language models are actually capable of encoding information pertaining to the naturalistic word order of sentences. We perform two simple tests on the full-scale models, in line with [Wang and Chen \(2020\)](#): the first of these is a classification task where a logistic regressor is trained to predict whether a randomly sampled token precedes another in an unshuffled sentence, and the second involves predicting the position of a word in an unshuffled sentence. The fact that we *do not* fine-tune any of the model parameters is noteworthy: the linear models can only learn word order information if it reflects in the representations the models generate somehow.

Pairwise Classification For this experiment, we train a logistic regression classification model on word representations extracted from the final layer of the Transformer encoder, mean pooling over sub-tokens when required. For each word pair x and y , the classifier is given a concatenation of our model m 's induced representations $m(x) \oplus m(y)$ and trained to predict a label indicating whether x precedes y or not. Holding out two randomly sampled positions, we use a training sets sized 2k, 5k, and 10k, from the Universal Dependencies English-GUM corpus ([Zeldes, 2017](#)) (excluding sentences with more than 30 tokens to increase learnability) and a test set of size 2,000. We report the mean accuracy from three runs.

Regression Using the same data, we also train a ridge-regularised linear regression model to predict the position of a word $p(x)$ in an unshuffled sentence, given that word's model-induced representa-

Model	Classification (acc.)			Regression (R^2)
	2k	5k	10k	
ORIG	81.50	81.74	80.40	0.68
SHUF.N1	65.96	64.98	71.82	0.60
NOPOS	50.41	53.35	50.22	0.03

Table 1: Pairwise classification and regression results.

tion $m(x)$. R^2 score is reported per model. To prevent the regressors from memorising word to position mappings, we perform 6-fold cross-validation, where the heldout part of the data contains no vocabulary overlap with the corresponding train set.

Results For both tasks (see Table 1), our results indicate that position encodings are particularly important for encoding word order: Classifiers and regressors trained on representations from ORIG and SHUF.N1 achieve high accuracies and R^2 scores, while those for NOPOS are close to random. Both ORIG and SHUF.N1 appear to be better than random given only 2k examples. These results imply that, given positional encodings and a modest training set of 2k or more examples, a simple linear model is capable of extracting word order information, enabling almost perfect extrapolation to unseen positions. Whether the position encodings come from a model trained on natural or shuffled text does not appear to matter, emphasizing that shuffled language models do indeed contain substantial information about the original word order.

4 Hidden word-order signals

In Section 3, we observed that [Sinha et al. \(2021\)](#)'s shuffled language models surprisingly exhibit information about naturalistic word order. That these models contain positional information can also be seen by visualizing position embedding similarity. Figure 1 displays Pearson correlations³ for position embeddings with themselves, across positions. Here, we see that the shuffled models satisfy the idealised criteria for position embeddings described by [Wang et al. \(2021\)](#): namely, they appear to be a) monotonous within smaller context windows, and b) invariant to translation. If position embedding correlations are consistent across offsets over the entire space of embeddings, the model can be said to have ‘learned’ distances between tokens. Since transformers process all positions in parallel,

³We see similar patterns with dot products for all our plots; we use Pearson correlations to constrain our range to $[-1, 1]$.

and since language models without position embeddings do not exhibit such information, position embeddings have to be the source of this information. In what follows, we discuss this apparent paradox.

Subword vs. word shuffling An important detail when running experiments on shuffled text, is *when* the shuffling operation takes place. When tokens are shuffled *before* BPE segmentation, this leads to word-level shuffling, in which sequences of subwords that form words remain contiguous. Such sequences become a consistent, meaningful signal for language modelling, allowing models to efficiently utilise the inductive bias provided by position embeddings. Thus, even though our pre-trained models have, in theory, not seen consecutive tokens in their pre-training data, they have learned to utilise positional embeddings to pay attention to adjacent tokens. The influence of this is somewhat visible in Figure 2: while models trained on text shuffled before and after segmentation both exhibit shifts in the *polarity* of their position correlations, only the former show bands of varying *magnitude*, similar to the full-scale models. Ravishankar and Søgaard (2021) discuss the implications of these patterns in a multilingual context; we hypothesise that in our context, the periodicity in magnitude is a visible artefact of the model’s ability to leverage position embeddings to enable offset attention. In Section 5, we analyse the effect of shuffling the pre-training data on the models’ attention mechanisms.

Accidental overlap In addition to the n -gram information which results from shuffling before segmentation, we also note that short sentences tend to include original bigrams with high probability, leading to stronger associations for words that are adjacent in the original texts. This effect is obviously much stronger when shuffling before segmentation than after segmentation. Figure 3 shows how frequent overlapping bigrams (of any sort) are, comparing word and subword shuffling over 50k sentences.

Sentence length Finally, we observe some preserved information about the original word order even when shuffling is performed *after* segmentation. We hypothesize that this is a side-effect of the non-random relationship between sentence length and unigram probabilities. That unigram probabilities correlate with sentence length follows from the fact that different genres exhibit different sentence

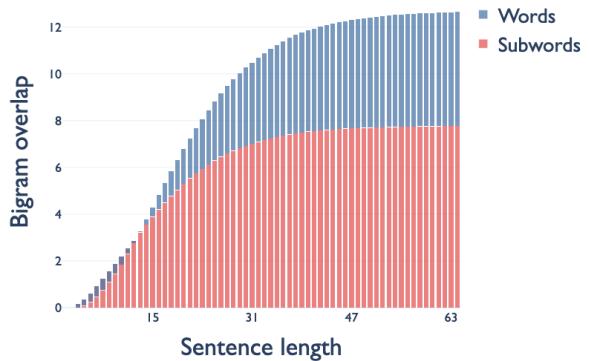


Figure 3: (Cumulative) plot showing subword bigram overlap after shuffling either words or subwords, as a percentage of the total number of seen bigrams. We see the overlap is significant, especially when performing shuffling before segmentation.

length distributions (Sigurd et al., 2004; Jin and Liu, 2017). Also, some words occur very frequently in formulaic contexts, e.g., *thank* in *thank you*. This potentially means that there is an approximately learnable relationship between the distribution of words and sentence boundary symbols.

To test for this, we train two smaller language models on unigram-sampled corpora: for the first, we use the first 100k BookCorpus sentences as our corpus, shuffling tokens at a corpus level (yet keeping the original sentence lengths). The stark difference in position embedding correlations between that and shuffling is seen in Figure 2. For the second, we sample from two different unigram distributions: one for short sentences and one for longer sentences (details in Appendix B). While the first model induces no correlations at all, the second does, as shown in Figure 4, implying that sentence length and unigram occurrences is enough to learn *some* order information.

5 Attention analysis

Transformer-based language models commonly have attention heads that attend to neighboring positions (Voita et al., 2019; Ravishankar et al., 2021). Such attention heads are positional and can only be learned in the presence of order information. We attempt to visualise the attention mechanism for pre-trained models by calculating, for each head and layer, the offset between a token and the token

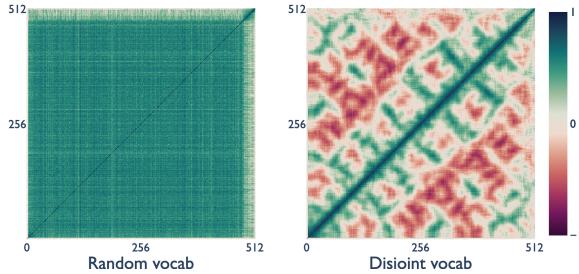


Figure 4: Similarity matrix between models with sentences sampled based on unigram corpus statistics; disjoint vocab implies a correlation between token choice and sentence length.

that it pays maximum attention to⁴. We then plot how frequent each offset is, as a percentage, over 100 Book Corpus sentences, in Figure 5, where we present results for two full-scale models, and two smaller models (see §2). When compared to NOPOS, SHUF.N1 has a less uniform pattern to its attention mechanism: it is likely, even at layer 0, to prefer to pay attention to adjacent tokens, somewhat mimicking a convolutional window (Cordonnier et al., 2020). We see very similar differences in distribution between our smaller models: Shuffling after segmentation, i.e., at the subword level, influences early attention patterns.

6 Evaluation beyond GLUE

SuperGLUE and WinoGrande Sinha et al. (2021)’s investigation is conducted on GLUE and on the Paraphrase Adversaries from Word shuffling (PAWS) dataset (Zhang et al., 2019). For these datasets, they find that models pretrained on shuffled text perform only marginally worse than those pretrained on normal text. This result, they argue can be explained in two ways: either a) these tasks do not need word order information to be solved, or b) the required word order information can be acquired during finetuning. While GLUE has been a useful benchmark, several of the tasks which constitute it have been shown to be solvable using various spurious artefacts and heuristics (Gururangan et al., 2018; Poliak et al., 2018). If, for instance, through finetuning, models are learning to rely on such heuristics as lexical overlap for MNLI (McCoy et al., 2019), then it is unsurprising that their performance is not greatly impacted by the

⁴This method of visualisation is somewhat limited, in that it examines only the *maximum* attention paid by each token. We provide more detailed plots over attention distributions in the Appendix.

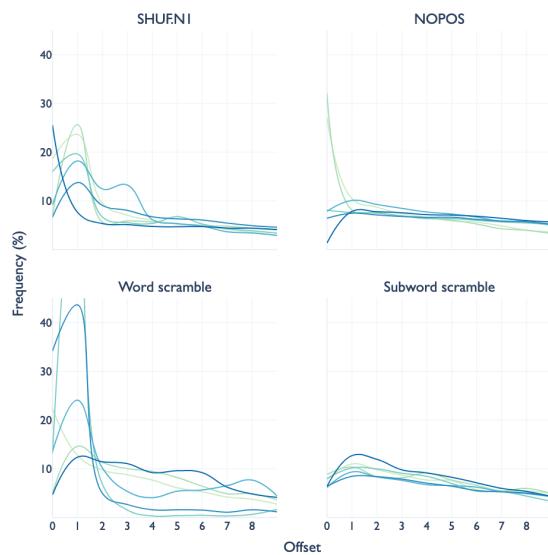


Figure 5: Relative frequency of offsets between token pairs in an attention relation; the y-axis denotes the percentage of total attention relations that occur at the offset indicated on the x-axis. We plot layers $l \in \{1, 2, 7, 8, 11, 12\}$ with increasing line darkness.

lack of word order information.

Evaluating on the more rigorous set of SuperGLUE tasks⁵ (Wang et al., 2019) and on the adversarially-filtered Winograd Schema examples (Levesque et al., 2012) of the WinoGrande dataset (Sakaguchi et al., 2020) produces results which paint a more nuanced picture compared to those of Sinha et al. (2021). The results, presented in Table 2, show accuracy or F1 scores for all models. For two of the tasks (MultiRC (Khashabi et al., 2018), COPA (Roemmele et al., 2011)), we observe a pattern in line with that seen in Sinha et al. (2021)’s GLUE and PAWS results: the drop in performance from ORIG to SHUF.N1 is minimal (mean: 1.75 points; mean across GLUE tasks: 3.3 points)⁶, while that to NOPOS is more substantial (mean: 10.5 points; mean across GLUE tasks: 18.6 points).

This pattern alters for the BoolQ Yes/No question answering dataset (Clark et al., 2019), the CommitmentBank (De Marneffe et al., 2019), the ReCoRD reading comprehension dataset (Zhang et al., 2018), both the Winograd Schema tasks,

⁵Results are reported for an average of 3 runs per task. The RTE task is excluded from our results as it is also part of GLUE; RTE results can be found in Sinha et al. (2021).

⁶CoLA results are excluded from the GLUE calculations due to the very high variance across random seeds reported by Sinha et al. (2021).

and to some extent the Words in Context dataset ([Pilehvar and Camacho-Collados, 2018](#)). For these tasks we observe a larger gap between ORIG and SHUF.N1 (mean: 8.1 points), and an even larger one between ORIG and NOPOS (mean: 19.78 points). We note that this latter set of tasks requires inferences which are more context-sensitive, in comparison to the two other tasks or to the GLUE tasks.

Consider the Winograd schema tasks, for example. Each instance takes the form of a binary test with a statement comprising of two possible referents (blue) and a pronoun (red) such as: [Sid](#) explained his theory to [Mark](#) but [he](#) couldn’t [convince](#) him. The correct referent of the pronoun must be inferred based on a special discriminatory segment (underlined). In the above example, this depends on a) the identification of “[Sid](#)” as the subject of “explained” and b) inferring that the pronoun serving as the subject of “convinced” should refer to the same entity. Since the Winograd schema examples are designed so that the referents are equally associated with their context⁷, word order is crucial⁸ for establishing the roles of “[Sid](#)” and “[Mark](#)” as subject and object of “explained” and “[he](#)” and “[him](#)” as those of “convinced”. If these roles cannot be established, making the correct inference becomes impossible.

A similar reasoning can be applied to the Words in Context dataset and the CommitmentBank. The former task tests the ability of a model to distinguish the senses of a polysemous word based on context. While this might often be feasible via a notion of contextual association that higher-order distributional statistics are sufficient for, some instances will require awareness of the word’s role as an argument in the sentence. The latter task investigates the projectivity of finite clausal complements under entailment cancelling operators. This is dependent on both the scope of the entailment operator and the identity of the subject of the matrix predicate ([De Marneffe et al., 2019](#)), both of which are sensitive to word order information.

A final consideration to take into account is dataset filtering. Two of the tasks where we observe

⁷e.g. [Sid](#) and [Mark](#) are both equally likely subjects/objects here. Not all Winograd schema examples are perfect in this regard, however, which could explain why scrambled models still perform above random. See [Trichelair et al. \(2018\)](#) for a discussion of the latter point.

⁸Particularly in a language with limited morphological role marking such as English.

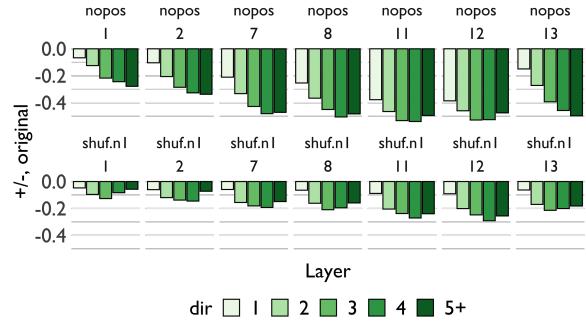


Figure 6: Δ , dependency arcs probing accuracy across lengths 1-5+, w.r.t. ORIG.

the largest difference between ORIG, SHUF.N1, and NOPOS — WinoGrande and ReCoRD — apply filtering algorithms to remove cues or biases which would enable models to heuristically solve the tasks. This indicates that by filtering out examples containing cues that make them solvable via higher order statistics, such filtering strategies do succeed at compelling models to (at least partially) rely on word order information.

Dependency Tree Probing Besides GLUE and PAWS, [Sinha et al. \(2021\)](#)’s analysis also includes several probing experiments, wherein they attempt to decode dependency tree structure from model representations. They show, interestingly, that the SHUF.N4, SHUF.N3 and SHUF.N2 models perform only marginally worse than ORIG, with SHUF.N1 producing the lowest scores (lower, in fact, than SHUF.CORPUS). Given the findings of Section 3, we are interested in taking a closer look at this phenomenon. Here, we surmise that *dependency length* plays a crucial role in the probing setup, where permuted models may succeed on par with ORIG in capturing local, adjacent dependencies, but increasingly struggle to decode longer ones. To evaluate the extent to which this is true, we train a bilinear probe (used in [Hewitt and Liang \(2019\)](#)) on top of all model representations and evaluate its accuracy across dependencies binned by length, where length between words w_i and w_j is defined as $|i - j|$. We opt for using the bilinear probe over the Pareto probing framework ([Pimentel et al., 2020](#)), as the former learns a transformation directly over model representations, while the latter adds the parent and child MLP units from [Dozat et al. \(2017\)](#) – acting more like a parser. We train probes on the English Web Treebank ([Silveira et al., 2014](#)) and evaluate using UAS, the standard parsing

Model	BoolQ	CB	COPA	MultiRC	ReCoRD	WiC	WSC	WinoGrande
ORIG	77.6	88.2 / 87.4	61.6	67.8 / 21.9	73.5 / 72.8	67.4	73.5	62.9
SHUF.N1	72.4	79.7 / 82.5	59.7	66.2 / 15.0	61.1 / 60.4	63.0	62.9	55.7
SHUF.N2	73.1	86.6 / 85.5	60.3	64.8 / 16.1	63.1 / 62.4	63.0	65.3	57.6
SHUF.N4	73.5	87.9 / 87.1	60.8	66.2 / 18.2	64.6 / 63.9	62.4	65.3	59.53
NOPOS	66.0	63.5 / 75.0	55.6	52.8 / 3.8	23.8 / 23.5	55.4	63.09	52.73
SHUF.CORPUS	66.7	65.6 / 73.8	56.1	52.6 / 6.4	31.0 / 30.3	57.3	65.14	51.68

Table 2: SuperGLUE and WinoGrande results for all models. Scores displayed are: Avg. F1 / Accuracy for CB; F1a / Exact Match for MultiRC; F1 / Accuracy for ReCoRD ; accuracy for the remaining tasks.

metric.

Figure 6 shows Δ probing accuracy across various dependency lengths for NOPOS and SHUF.N1, with respect to ORIG⁹; we include detailed Δ s for all models in Appendix C. For NOPOS, parsing difficulty increases almost linearly with distance, often mimicking the actual frequency distribution of dependencies at these distances in the original treebank (Appendix C); for SHUF.N1, the picture is a lot more nuanced, with dependencies at a distance of 1 consistently being closer in terms of parseability to ORIG, which, we hypothesise, is due to its adjacency bias.

7 Other Findings

Random position embeddings are difficult to add post-training We tried to quantify the degree to which the inductive bias imparted by positional embeddings can be utilised, solely via fine-tuning. To do so, for a subset of GLUE tasks (MNLI, QNLI, RTE, SST-2, CoLA), we evaluate NOPOS, and a variant where we randomly initialised learnable position embeddings and add them to the model, with the rest of the model equivalent to NOPOS. We see no improvement in results, except for MNLI, that we hypothesise stems from position embeddings acting as some sort of regularisation parameter. To test this, we repeat the above set of experiments, this time injecting Gaussian noise instead; this has been empirically shown to have a regularising effect on the network (Bishop, 1995; Camuto et al., 2021). Adding Gaussian noise led to a slight increase in score for just MNLI, backing up our regularisation hypothesis.

Models learn to expect specific embeddings Replacing the positional embeddings in ORIG with fixed, sinusoidal embeddings before fine-tuning significantly hurts scores on the same subset of

GLUE tasks, implying that the models expect embeddings that resemble the inductive bias imparted by random embeddings, and that fine-tuning tasks do not have sufficient data to overcome this. The addition of fixed, *sinusoidal* to NOPOS also does not improve model performance on a similar subset of tasks; this implies, given that sinusoidal embeddings are already meaningful, that model weights also need to learn to fit the embeddings they are given, and that they need a substantial amount of data to do so.

8 On Word Order

In Humans It is generally accepted that a majority of languages have “canonical” or “base” word orderings (Comrie, 1989) (e.g. Subject-Verb-Object in English, and Subject-Object-Verb in Hindi). Linguists consider word order to be a *coding property* — mechanisms by which abstract, syntactic structure is encoded in the surface form of utterances. Beyond word order, other coding properties include, e.g. subject-verb agreement, morphological case marking, or function words such as adpositions. In English, word order is among the most prominent coding properties, playing a crucial role in the expression of the main verb’s core arguments: subject and object. For more morphologically complex languages, on the other hand, (e.g. Finnish and Turkish), word order is primarily used to convey pragmatic information such as topicalisation or focus. In such cases, argument structure is often signalled via case-marking, where numerous orderings become possible (shift in topic or focus notwithstanding). We refer the reader to Kulkarni and Nivre (2021) for a broader discussion of these topics and their implications when studying syntax through language models.

More generally, evidence for the saliency of word order in linguistic processing and comprehension comes from a variety of studies using acceptability judgements, eye-tracking data, and neu-

⁹Note that Layer 13 refers to a linear mix of all model layers, as is done for ELMo (Peters et al., 2018).

ral response measurements (Bever, 1970; Danks and Glucksberg, 1971; Just and Carpenter, 1980; Friederici et al., 2000, 2001; Bahlmann et al., 2007; Lerner et al., 2011; Pallier et al., 2011; Fedorenko et al., 2016; Ding et al., 2016). Psycholinguistic research has, however, also highlighted the robustness of sentence processing mechanisms to a variety of perturbations, including those which violate word order restrictions (Ferreira et al., 2002; Gibson et al., 2013; Traxler, 2014). In recent work, Mollica et al. (2020) tested the hypothesis that composition is the core function of the brain’s language-selective network and that it can take place even when grammatical word order constraints are violated. Their findings confirmed this, showing that stimuli with shuffled word order where local dependencies were preserved — as is, roughly speaking, the case for many dependencies in the sentences SHUF.N4 is trained on — elicited a neural response in the language network that is comparable to that elicited by normal sentences. When interword dependencies were disrupted so combinable words were so far apart that composition among nearby words was highly unlikely — as in SHUF.N1, neural response fell to a level compared to unconnected word lists.

In Machines Recently, many NLP researchers have attempted to investigate the role of word order information in language models. For example, Lin et al. (2019) employ diagnostic classifiers and attention analyses to demonstrate that lower (but not higher) layers of BERT encode word order information. Papadimitriou et al. (2021) find that Multilingual BERT is sensitive to morphosyntactic alignment, where numerous languages (out of 24 total) rely on word order to mark subjecthood (English among them). Alleman et al. (2021) implement an input perturbation framework (n-gram shuffling, phrase swaps, etc.), and employ it towards testing the sensitivity of BERT’s representations to various types of structure in sentences. They report a sensitivity to larger constituent units of sentences in higher layers, which they deduce to be influenced by hierarchical phrase structure. O’Connor and Andreas (2021) examine the contribution of various contextual features to the ability of GPT-2 (Radford et al., 2019) to predict upcoming tokens. Their findings show that several destructive manipulations, including in-sentence word shuffling, applied to mid- and long range contexts lead only to a modest increase in *usable information* as defined according

to the V-information framework of Xu et al. (2020).

Similarly, word order information has been found not to be essential for various NLU tasks and datasets. Early work showed that Natural Language Inference tasks are largely insensitive to permutations of word order (Parikh et al., 2016; Sinha et al., 2020). Pham et al. (2020) and Gupta et al. (2021) discuss this in greater detail, demonstrating that test-time word order perturbations applied to GLUE benchmark tasks have little impact on LM performance. Following up on this, Sinha et al. (2021), which our work builds on, found that *pretraining* on scrambled text appears to only marginally affect model performance. Most related to this study, Clouatre et al. (2021) introduce two metrics for gauging the local and global ordering of tokens in scrambled texts, observing that only the latter is altered by the perturbation functions found in prior literature. In experiments with GLUE, they find that local (sub-word) perturbations show a substantially stronger performance decay compared to global ones.

In this work, we present an in-depth analysis of these results, showing that LMs trained on scrambled text can actually retain word information and that — as for humans — their sensitivity to word order is dependent on a variety of factors such as the nature of the task and the locality of perturbation. While performance on some “understanding” evaluation tasks is not strongly affected by word order scrambling, the effect on others such as the Winograd Schema is far more evident.

9 Conclusion

Much discussion has resulted from recent work showing that scrambling text at different stages of testing or training does not drastically alter the performance of language models on NLU tasks. In this work, we presented analyses painting a more nuanced picture of such findings. Primarily, we demonstrate that, as far as altered pre-training is concerned, models still do retain a semblance of word order knowledge — largely at the local level. We show that this knowledge stems from cues in the altered data, such as adjacent BPE symbols and correlations between sentence length and content. The order in which shuffling is performed — before or after BPE tokenization — is influential in models’ acquisition of word order, which calls for caution in interpreting previous results. Finally, we show that there exist NLU tasks that are far more

sensitive to sentence structure as expressed by word order.

Acknowledgements

We thank Stephanie Brandl, Desmond Elliott, Yova Kementchedjhieva, Douwe Kiela and Miryam de Lhoneux for their feedback and comments. We acknowledge the CSC-IT Centre for Science, Finland, for providing computational resources. Vinit worked on this paper while on a research visit to the University of Copenhagen. Mostafa and Anders are supported by a Google Focused Research Award.

References

- Matteo Alleman, Jonathan Mamou, Miguel A Del Rio, Hanlin Tang, Yoon Kim, and SueYeon Chung. 2021. Syntactic perturbations reveal representational correlates of hierarchical phrase structure in pretrained language models. *arXiv preprint arXiv:2104.07578*.
- Jörg Bahlmann, Antoni Rodriguez-Fornells, Michael Rotte, and Thomas F Münte. 2007. An fmri study of canonical and noncanonical word order in german. *Human brain mapping*, 28(10):940–949.
- Thomas G Bever. 1970. The cognitive basis for linguistic structures. *Cognition and the development of language*.
- Chris M. Bishop. 1995. Training with noise is equivalent to tikhonov regularization. *Neural Computation*, 7(1):108–116.
- Alexander Camuto, Matthew Willetts, Umut Şimşekli, Stephen Roberts, and Chris Holmes. 2021. Explicit Regularisation in Gaussian Noise Injections. *arXiv:2007.07368 [cs, stat]*.
- Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.
- Louis Clouatre, Prasanna Parthasarathi, Amal Zouaq, and Sarath Chandar. 2021. Demystifying neural language models’ insensitivity to word-order. *arXiv preprint arXiv:2107.13955*.
- Bernard Comrie. 1989. *Language universals and linguistic typology: Syntax and morphology*. University of Chicago press.
- Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. 2020. On the Relationship between Self-Attention and Convolutional Layers. *arXiv:1911.03584 [cs, stat]*.
- Joseph H Danks and Sam Glucksberg. 1971. Psychological scaling of adjective orders. *Journal of Memory and Language*, 10(1):63.
- Marie-Catherine De Marneffe, Mandy Simons, and Judith Tonhauser. 2019. The commitmentbank: Investigating projection in naturally occurring discourse. In *proceedings of Sinn und Bedeutung*, volume 23, pages 107–124.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Nai Ding, Lucia Melloni, Hang Zhang, Xing Tian, and David Poeppel. 2016. Cortical tracking of hierarchical linguistic structures in connected speech. *Nature neuroscience*, 19(1):158–164.
- Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford’s graph-based neural dependency parser at the CoNLL 2017 shared task. In *Proceedings of the Conll 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada. Association for Computational Linguistics.
- Evelina Fedorenko, Terri L Scott, Peter Brunner, William G Coon, Brianna Pritchett, Gerwin Schalk, and Nancy Kanwisher. 2016. Neural correlate of the construction of sentence meaning. *Proceedings of the National Academy of Sciences*, 113(41):E6256–E6262.
- Fernanda Ferreira, Karl GD Bailey, and Vittoria Feraro. 2002. Good-enough representations in language comprehension. *Current directions in psychological science*, 11(1):11–15.
- Angela D Friederici, Axel Mecklinger, Kevin M Spencer, Karsten Steinhauer, and Emanuel Donchin. 2001. Syntactic parsing preferences and their online revisions: A spatio-temporal analysis of event-related brain potentials. *Cognitive Brain Research*, 11(2):305–323.
- Angela D Friederici, Martin Meyer, and D Yves Von Cramon. 2000. Auditory language comprehension: an event-related fmri study on the processing of syntactic and lexical information. *Brain and language*, 74(2):289–300.
- Edward Gibson, Leon Bergen, and Steven T Piantadosi. 2013. Rational integration of noisy evidence and prior semantic expectations in sentence interpretation. *Proceedings of the National Academy of Sciences*, 110(20):8051–8056.
- Ashim Gupta, Giorgi Kvernadze, and Vivek Sriku-
mar. 2021. Bert & family eat word salad: Experiments with text understanding. *arXiv preprint arXiv:2101.03453*.

- Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. [Annotation artifacts in natural language inference data](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.
- John Hewitt and Percy Liang. 2019. [Designing and interpreting probes with control tasks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.
- Huiyuan Jin and Haitao Liu. 2017. [How will text size influence the length of its linguistic constituents?](#) *Poznan Studies in Contemporary Linguistics*, 53.
- Marcel A Just and Patricia A Carpenter. 1980. A theory of reading: From eye fixations to comprehension. *Psychological review*, 87(4):329.
- Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking beyond the surface:a challenge set for reading comprehension over multiple sentences. In *NAACL*.
- Artur Kulmizev and Joakim Nivre. 2021. Schrödinger’s Tree – On Syntax and Neural Language Models. *arXiv preprint arXiv:2110.08887*.
- Yulia Lerner, Christopher J Honey, Lauren J Silbert, and Uri Hasson. 2011. Topographic mapping of a hierarchy of temporal receptive windows using a narrated story. *Journal of Neuroscience*, 31(8):2906–2915.
- Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*.
- Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open sesame: Getting inside bert’s linguistic knowledge. *arXiv preprint arXiv:1906.01698*.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*.
- Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. [Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.
- Francis Mollica, Matthew Siegelman, Evgeniia Diacheck, Steven T Piantadosi, Zachary Mineroff, Richard Futrell, Hope Kean, Peng Qian, and Evelina Fedorenko. 2020. Composition is the core driver of the language-selective network. *Neurobiology of Language*, 1(1):104–134.
- Joe O’Connor and Jacob Andreas. 2021. What context features can transformer language models use? *arXiv preprint arXiv:2106.08367*.
- Christophe Pallier, Anne-Dominique Devauchelle, and Stanislas Dehaene. 2011. Cortical representation of the constituent structure of sentences. *Proceedings of the National Academy of Sciences*, 108(6):2522–2527.
- Isabel Papadimitriou, Ethan A Chi, Richard Futrell, and Kyle Mahowald. 2021. Deep subjecthood: Higher-order grammatical features in multilingual bert. *arXiv preprint arXiv:2101.11043*.
- Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Thang M Pham, Trung Bui, Long Mai, and Anh Nguyen. 2020. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? *arXiv preprint arXiv:2012.15180*.
- Mohammad Taher Pilehvar and Jose Camacho-Collados. 2018. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. *arXiv preprint arXiv:1808.09121*.
- Tiago Pimentel, Naomi Saphra, Adina Williams, and Ryan Cotterell. 2020. [Pareto probing: Trading off accuracy for complexity](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3138–3153, Online. Association for Computational Linguistics.
- Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. [Hypothesis only baselines in natural language inference](#). In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

- Vinit Ravishankar, Artur Kulmizev, Mostafa Abdou, Anders Søgaard, and Joakim Nivre. 2021. [Attention can reflect syntactic structure \(if you let it\)](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3031–3045, Online. Association for Computational Linguistics.
- Vinit Ravishankar and Anders Søgaard. 2021. [The impact of positional encodings on multilingual compression](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 763–777, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.
- Bengt Sigurd, Mats Eeg-Olofsson, and Joost Van Weijer. 2004. [Word length, sentence length and frequency – zipf revisited](#). *Studia Linguistica*, 58(1):37–52.
- Natalia Silveira, Timothy Dozat, Marie-Catherine De Marneffe, Samuel R Bowman, Miriam Connor, John Bauer, and Christopher D Manning. 2014. A gold standard dependency corpus for english. In *LREC*, pages 2897–2904. Citeseer.
- Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. 2021. [Masked language modeling and the distributional hypothesis: Order word matters pre-training for little](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2888–2913, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Koustuv Sinha, Prasanna Parthasarathi, Joelle Pineau, and Adina Williams. 2020. Unnatural language inference. *arXiv preprint arXiv:2101.00010*.
- Matthew J Traxler. 2014. Trends in syntactic parsing: Anticipation, bayesian estimation, and good-enough parsing. *Trends in cognitive sciences*, 18(11):605–611.
- Paul Trichelaire, Ali Emami, Jackie Chi Kit Cheung, Adam Trischler, Kaheer Suleman, and Fernando Diaz. 2018. On the evaluation of common-sense reasoning in natural language understanding. *arXiv preprint arXiv:1811.01778*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.
- Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. [Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.
- Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. [GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding](#). In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.
- Benyou Wang, Lifeng Shang, Christina Lioma, Xin Jiang, Hao Yang, Qun Liu, and Jakob Grue Simonsen. 2021. ON POSITION EMBEDDINGS IN BERT. page 21.
- Yu-An Wang and Yun-Nung Chen. 2020. [What do position embeddings learn? an empirical study of pre-trained language model positional encoding](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6840–6849, Online. Association for Computational Linguistics.
- Yilun Xu, Shengjia Zhao, Jiaming Song, Russell Stewart, and Stefano Ermon. 2020. A theory of usable information under computational constraints. *arXiv preprint arXiv:2002.10689*.
- Amir Zeldes. 2017. The gum corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.
- Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. Record: Bridging the gap between human and machine commonsense reading comprehension. *arXiv preprint arXiv:1810.12885*.
- Yuan Zhang, Jason Baldridge, and Luheng He. 2019. [PAWS: Paraphrase adversaries from word scrambling](#). 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 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*.

A Subword vs. word scrambling

B On biased sampling

We first split our vocab of size 5,000 into two halves, both of size 2500, such that the sum total of unigram frequencies of tokens in each half is roughly equivalent. Next, iterating over 100k BookCorpus sentences, we determine the sentence length l , for which there are an equivalent number of tokens in sentences with length $< l$ and sentences with length $\geq l$. We then sample tokens from the first vocab half for sentences $< l$, and from the second vocab half for sentences with length $\geq l$, 80% of the time; for the other 20%, we sample from the opposite half to introduce some overlap.

C Full UD results

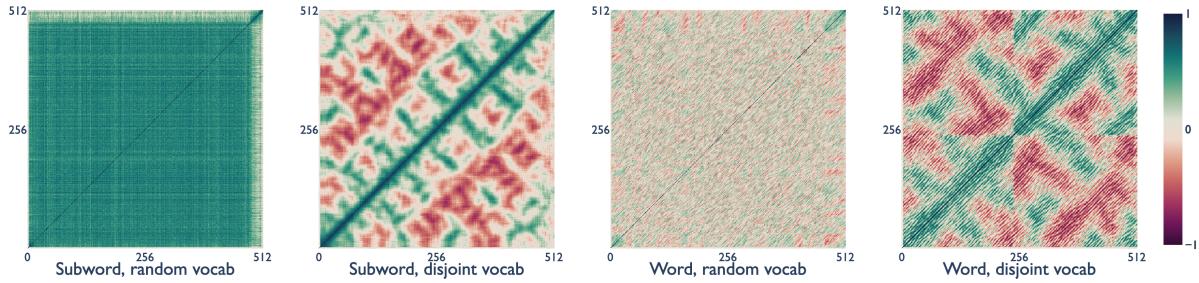


Figure 7: Pearson correlations, when scrambling by subword/word, with/without disjoint vocabularies. Disjoint vocabularies appear to induce patterns in position-position correlations, while scrambling at a word level induces ‘stripes’ of oscillating magnitude; this is likely due to position embeddings learning connections to adjacent tokens.

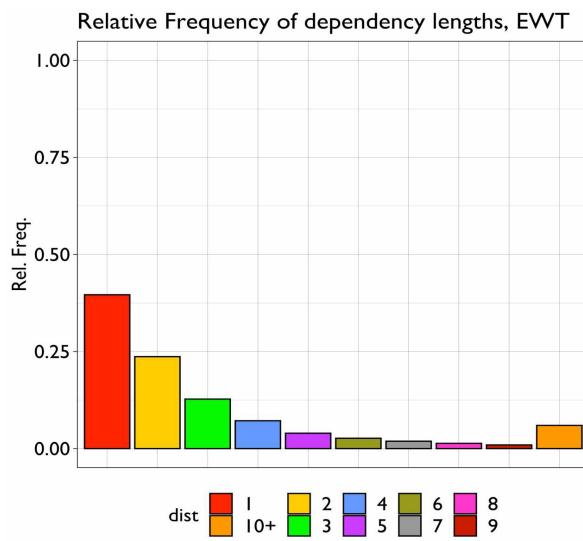


Figure 8: Relative frequencies of dependency relations in $UD_{English-EWT}$, at a dependency lengths indicated by the x-axis

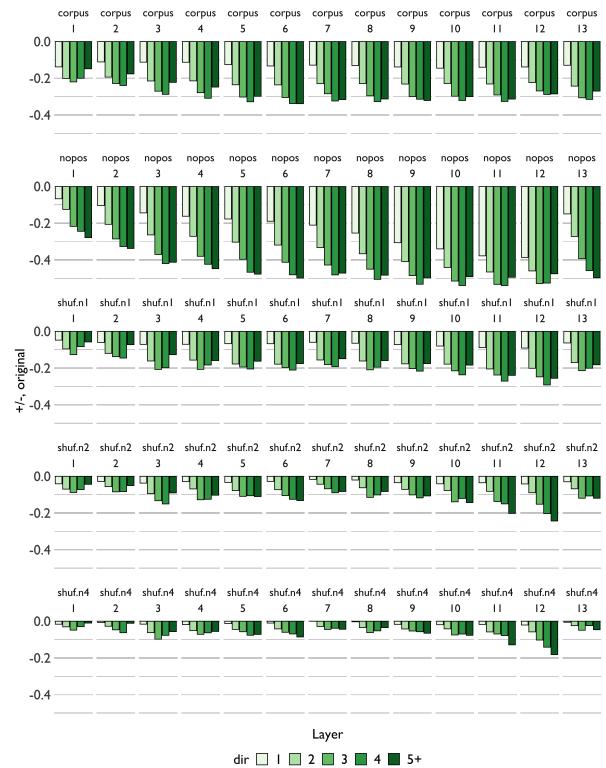


Figure 9: Δ UAS, all models and layers across dependency lengths 1-5+, w.r.t. ORIG. Layer 13 represents a linear mix of all model layers.