Identifying Necessary Elements for BERT's Multilinguality

Philipp Dufter, Hinrich Schütze
Center for Information and Language Processing (CIS), LMU Munich, Germany
philipp@cis.lmu.de

Abstract

It has been shown that multilingual BERT (mBERT) yields high quality multilingual representations and enables effective zero-shot transfer. This is suprising given that mBERT does not use any kind of crosslingual signal during training. While recent literature has studied this effect, the exact reason for mBERT's multilinguality is still unknown. We aim to identify architectural properties of BERT as well as linguistic properties of languages that are necessary for BERT to become multilingual. To allow for fast experimentation we propose an efficient setup with small BERT models and synthetic as well as natural data. Overall, we identify six elements that are potentially necessary for BERT to be multilingual. Architectural factors that contribute to multilinguality are underparameterization, shared special tokens (e.g., "[CLS]"), shared position embeddings and replacing masked tokens with random tokens. Factors related to training data that are beneficial for multilinguality are similar word order and comparability of corpora.

1 Introduction

Multilingual models, i.e., models that are capable of processing more than one language with comparable performance, are central to natural language processing. They are useful as fewer models need to be maintained to serve many languages, resource requirements are reduced, and low- and mid-resource language can benefit from crosslingual transfer, thus achieving a higher performance. Further, having source and target representations in the same space, multilingual models are useful in machine translation, annotation projection and typological research. Given 7000+ languages in the world, the need for multilingual models seems obvious.

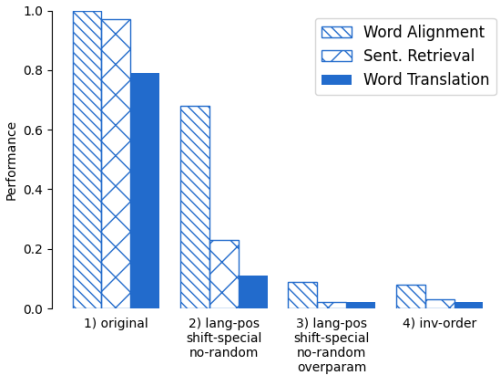


Figure 1: Three architectural modifications harm multilinguality (column 2 compared to 1). Together with overparameterization almost no multilinguality is left (see column 3). Pairing a language with its inversion (i.e., inverted word order) destroys multilinguality as well (column 4).

With the rise of static word embeddings countless multilingual embedding algorithms have been proposed (Mikolov et al., 2013; Hermann and Blunsom, 2014; Faruqui and Dyer, 2014); for a survey see (Ruder et al., 2019). Pretrained language models (Peters et al., 2018; Howard and Ruder, 2018; Devlin et al., 2019) have been proven to yield exceptional performance across a variety of tasks and generally outperform static word embeddings. A simple multilingual model is multilingual BERT[1] (*mBERT*). It is trained on the concatenation of the 104 largest Wikipedias with a shared subword vocabulary. There is no additional crosslingual signal. Still, mBERT yields high-quality multilingual representations (Pires et al., 2019; Wu and Dredze, 2019; Hu et al., 2020).

The reason for mBERT's multilinguality is – to the best of our knowledge – still unknown. Wang

[1] https://github.com/google-research/bert/blob/master/multilingual.md

et al. (2019) provide an extensive study which investigates properties of languages and model architecture. They conclude that a shared vocabulary is not necessary, but that the model needs to be deep and languages need to share a similar "structure". Artetxe et al. (2019) show that neither a shared vocabulary nor joint pretraininig is required for BERT to be multilingual.

We continue this line of research and investigate the reason for BERT's multilinguality. In total we find indications that six elements influence the multilinguality of BERT. Figure 1 provides a summary of our main findings.

Contributions: i) BERT is a large model that consumes tremendous resources. We propose an experimental setup with small BERT models and a mix of synthetic and natural data that allows for fast experimentation. **ii)** We hypothesize that BERT is multilingual because of a limited number of parameters. Thus the model is forced to use its parameters efficiently, share parameters across languages and thus exploit common structures by aligning representations. We provide experimental evidence for this hypothesis. **ii)** We show that shared special tokens, shared position embeddings and replacing masked tokens with random words contribute to multilinguality. **iii)** We show that structural similarity across languages in form of similar word order is essential for BERT to be multilingual. Further we provide an initial results of how word order similarity across languages influences performance. **iv)** We provide initial experiments showing that the degree of comparability of training corpora influences the degree of multilinguality. **v)** Using above insights we perform initial experiments to create better multilingual models.

Note that this paper is work in progress. Given our limited compute infrastructure we experimented in a setup that allows gaining insights quickly. We are currently working on transferring the experiments from the small laboratory setting to real-world data, and examing whether we can verify our findings there. Our code is publicly available.[2]

2 Setup and Hypotheses

2.1 Setup

BERT's size and training time hinders fast experimentation. Thus we propose a setup that allows for fast experimentation and for validating multilinguality. We hypothesize that we can gain insights in such a reduced setup faster. Our assumption is that those insights then transfer to a larger real world setup – this obviously needs to be verified.

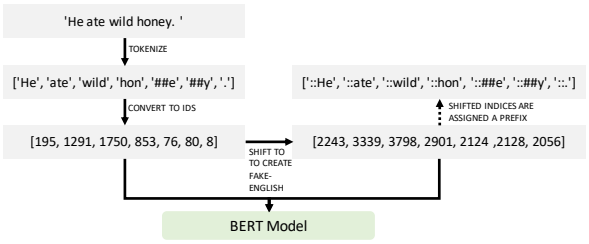


Figure 2: How to create a Fake-English sentence using an index shift of 2048.

Languages. Wang et al. (2019) propose to consider English and Fake-English, a language that is created by shifting unicode data points by a large constant. Fake-English in their case has the exact same linguistic properties as English, but is represented by different unicode data points.

We follow a similar approach, but instead of shifting unicode datapoints we simply shift token indices after tokenization by a constant; shifted tokens are prefixed by "::" and added to the vocabulary. Shifting token indices by a large constant doubles the size of the vocabulary. See Figure 2 for an example of how to create a Fake-English sentence.

We prefer shifting token indices rather than unicode code points and argue that this is a cleaner setup. For example, the BERT tokenizer treats some punctuation as special symbols (e.g., "drycleaning" is tokenized as ["dry", "-", "##cleaning"], not as ["dry", "##-", "##cleaning"]). Thus, with a unicode shift, tokenizations of English and Fake-English might differ.

Training Data. For our setup, aimed at supporting fast experimentation, a small corpus with limited vocabulary and limited character range is desirable. We work on the English language and use the English Easy-to-Read version of the Parallel Bible Corpus (Mayer and Cysouw, 2014). The corpus is structured into verses and is word-tokenized. As each verse can contain multiple sentences we perform sentence splitting using NLTK (Loper and Bird, 2002). The final corpus has 17178 sentences, 228K words, a vocabulary size of 4449 and 71 distinct characters. The median sentence length is 12 words.

By creating a Fake-English version of this corpus

[2] https://github.com/pdufter/minimult

we get a shifted replica and thus a sentence-parallel corpus.

Vocabulary. We create a vocabulary of size 2048 from the English corpus with the wordpiece tokenizer.[3] We use the same vocabulary for English and Fake-English. Thus, our final vocabulary size is 4096.

Model. We use the BERT-Base architecture (Devlin et al., 2019) with only minor modifications to achieve a smaller model: we divide the hidden dimension, the intermediate dimensions of the feed forward layer as well as the number of attention heads by 12 to obtain hidden size: 64, intermediate size: 256, number of attention heads: 1. Note that this leaves us with just a single attention head. However, Wang et al. (2019) found that the number of attention heads is neither important for the overall performance nor for the multilinguality.

Training Parameters. We tuned the training parameters to achieve a reasonable perplexity on the training corpus. Unless indicated differently we use a batch size of 256, train for 100 epochs with AdamW (Loshchilov and Hutter, 2018) (learning rate 2e-3, weight decay 0.01, epsilon 1e-6), and use 50 warmup steps. We only use the masked-language-modeling objective and do not use next-sequence-prediction as it has been found to be harmful (Wang et al., 2019). With this setup we can train a single model in under 40 minutes on a single GeForce GTX 1080Ti. To accomodate for stochasticity we run each experiment with 5 different seeds and report the mean and standard deviation across runs.

2.2 Evaluation

We evaluate two properties of our trained language models: the *overall model fit* (i.e., is the trained language model of high quality) and the degree of *multilinguality*.

2.2.1 Model Fit

MLM Perplexity. To verify that BERT was successfully trained we evaluate the models with perplexity (using e as base) on the training data. Note that perplexity is only computed on randomly masked tokens. Given a set of masked tokens in a text $w_1, \ldots, w_n$ and probabilities $p_{w_1}, \ldots, p_{w_n}$ that the correct token was predicted by the model, perplexity is given by

$$\text{perpl.} = e^{-1/n \sum_{k=1}^{n} \log(p_{w_k})}.$$

In our corpus on average $n = 68K$ tokens are masked.

Our research goal is to investigate the reason for BERT's multilinguality. Creating language models that generalize well is only secondary. Therefore we do not see an immediate need to introduce validation or test data for computing the perplexity. While this might be easily critisized, we do not see major problems for the purpose of this study at the moment. Note that all multilingual evaluations are parameter-free and thus safe to perform on the training data.

2.2.2 Multilinguality

We evaluate the degree of multilinguality with three tasks. Representations from different layers of BERT can be considered. We evaluate representations from layer 0 (uncontextualized) and layer 8 (contextualized). Several papers have found layer 8 to work well for semantic and multilingual tasks (Tenney et al., 2019; Hewitt and Manning, 2019; Sabet et al., 2020). Note that representations from layer 0 include position and segment embeddings as well as the layer normalization which is applied in the initial embedding layer of BERT.

Word Alignment. The way we created the corpus, we have a sentence-aligned corpus with English and Fake-English. The gold word alignment between two sentences is simply the identity alignment. We can use this automatically created gold-alignment to evaluate BERT on a word alignment task. (Sabet et al., 2020) have found that mBERT performs generally well on the word alignment task.

To create word alignments using BERT we follow Sabet et al. (2020). Consider the parallel sentences $s^{(\text{eng})}, s^{(\text{fake})}$, with length n. We extract d-dimensional wordpiece embeddings from the l-th layer of BERT for the corresponding words to obtain the embedding matrices $\mathcal{E}(s^{(k)}) \in \mathbb{R}^{n \times d}$ for $k \in \{\text{eng}, \text{fake}\}$. By considering cosine similarity we obtain the similarity matrix $S \in [0,1]^{n \times n}$ induced by the embeddings where $S_{ij} :=$ cosine-sim $\left(\mathcal{E}(s^{(\text{eng})})_i, \mathcal{E}(s^{(\text{fake})})_j\right)$. We then align two wordpieces i and j if

$$(i = \arg\max_l S_{l,j}) \wedge (j = \arg\max_l S_{i,l}).$$

[3] We use `https://github.com/huggingface/tokenizers`

The alignments are then evaluated using precision, recall and F_1 as follows:

$$\text{prec.} = \frac{|P \cap G|}{|P|}$$

$$\text{rec.} = \frac{|P \cap G|}{|G|}$$

$$F_1 = \frac{2 \text{ prec. rec.}}{\text{prec.} + \text{rec.}}$$

where P is the set of predicted alignments and G the set of true alignments.

Sentence Retrieval. Sentence retrieval is a popular way in evaluating crosslingual representations (e.g., (Artetxe and Schwenk, 2019)). In a parallel corpus, sentence retrieval is another possible way of evaluating multilinguality. Again, we obtain the embeddings $\mathcal{E}(s^{(k)})$ as before. Now, we obtain the sentence embedding $e_s^{(k)}$ simply by averaging vectors across all tokens (ignoring CLS and SEP tokens) in a sentence. Computing cosine similarities between English and Fake-English sentences yields the similarity matrix $(r_{ij})_{i,j\in[m]} = R \in \mathbb{R}^{m\times m}$ for m sentences ($m = 17178$ in our case).

Given a query sentence $s^{(\text{eng})}$, we obtain the retrieved sentences in Fake-English by ranking them according to similarity. We compute precision as

$$p = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{\arg\min_l r_{il} = i\}.$$

Note that we can do the retrieval step in two directions: using English as query language and using Fake-English as query. We report the mean precision of those two directions and call the measure *retr.*

Word Translation. Last, we consider word translation. Again, by having shifted the vocabulary we have a ground-truth dictionary by construction. We obtain word vectors by feeding each word in the vocabulary individually to BERT. That is, we input "[CLS] {token} [SEP]".

We evaluate word translation in the same way as sentence retrieval and again average the translation scores of the two directions (English and Fake-English as query language). We call this measure *trans.*

2.3 Architectural Properties

In this section we formulate hypotheses as to which components of BERT's architecture contribute to multilinguality.

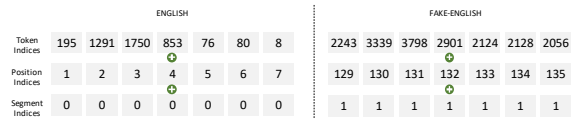


Figure 3: Input indices to BERT with language specific position and segment embeddings.

Overparameterization: *overparam*. We hypothesize that if BERT is severely overparameterized the model should have enough capacity to model each language separately without creating a multilingual space. Conversely, if the number of parameters is small, the model has a need to use parameters efficiently. In that case, it would be beneficial for BERT to identify common structures common among languages, align their "embedding manifolds" and model them together, consequently creating a multilingual space.

To test this we train a version of BERT, that has the same configuration as BERT-base (i.e., hidden size: 768, intermediate size: 3072, number of attention heads: 12) and is thus much larger than our standard configuration. Given our small training corpus size and the low number of languages we argue that this model is severely overparameterized. Note that for this overparameterized version we use a learning rate of 1e-4 (which was also used for the original BERT training). With the larger learning rate that we use for the small model we found that the overparameterized model does not converge.

Shared Special Tokens: *shift-special*. It has been found that a shared vocabulary is not essential for mBERT to be multilingual (Wang et al., 2019; Artetxe et al., 2019). Similarly, in our setting the vocabulary is not shared. However in prior studies, special tokens are usually shared across languages. Those include [UNK], [CLS], [SEP], [MASK], [PAD]. We investigate whether having shared special tokens contributes to multilinguality. This could be the case as those tokens are very frequent and could serve as "anchor points" for the languages. To investigate this, we shift the special tokens with the same shift as applied to token indices (*shift-special*).

Shared Position Embeddings: *lang-pos*. Position and segment embeddings are usually shared across languages. We investigate their contribution to multilinguality by using language-specific position (*lang-pos*) and segment embeddings. For an example see Figure 3.

Random Word Replacement: *no-random*.

The masked language modeling task as proposed by Devlin et al. (2019) masks out 15% of tokens in a sentence. These tokens are replaced with "[MASK]" with probability $p_{\text{[mask]}} = 80\%$, remain unchanged with $p_{\text{[id]}} = 10\%$ and are replaced with a random word of the vocabulary with $p_{\text{[rand]}} = 10\%$. Replacement with a random word is not constrained to be from the same language: the randomly sampled token can come from any language. Thus Fake-English tokens could appear in an English sentence and vice-versa. We hypothesize that this random replacement contributes to multilinguality.

We denote the triple $p = (p_{\text{[mask]}}, p_{\text{[id]}}, p_{\text{[rand]}})$ and investigate how setting $p = (0.9, 0.1, 0.0)$ (*no-random*) affects multilinguality.

Note that this affects the perplexity evaluation as well (as masked words are never replaced with random words). Thus perplexity numbers are not comparable with other approaches. The BERT models coming out of this approach are most likely less robust towards random noise in input sentences. We plan to investigate this effect in future work.

2.4 Linguistic Properties

Inverted Word Order: *inv-order*. Wang et al. (2019) shuffled word order in sentences randomly and found that word order has some, but not a severe effect on multilinguality. They conclude that "structural similarity" across languages is important without further specifying this term.

We investigate the extreme case where word order is completely flipped (*inv-order*). To this end we invert each sentence in the Fake-English corpus such that the last word becomes the first word. Note that, besides the reading order, all properties of the languages are preserved (including ngram statistics). Thus we argue that the structural similarity among English and inverted Fake-English is very high.

Degree of Comparability of Corpora: *no-parallel*. We argue that the similarity of the training corpora influences "structural similarity" of the languages as well. That is, if we train on a parallel corpus we expect the language structures to be more similar than when we train on two independent corpora, potentially coming from different domains. For the original mBERT one can argue that Wikipedias across languages are at least in the same domain, share some articles and thus are comparable, yet not parallel data. To test this hypothesis we train on a non-parallel corpus (*no-parallel*). We achieve this by splitting the Bible into two halves, using the first half for English and the second half for Fake-English, thus avoiding any parallel sentences. We still evaluate this model on the complete Bible.

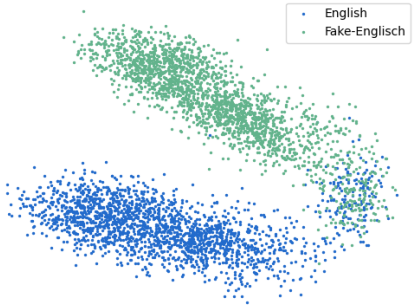


Figure 4: Principal component analysis of the **token embeddings** for the original model for a single seed. One can clearly see that the representations of the two languages are separated yet have a similar structure. The nearest neighbor of almost all tokens are the respective tokens in the other language (i.e., the nearest neighbor of "go" is "::go"). This is quantitatively confirmed in Table 1

2.5 Baseline: *untrained*

As a consistency check for our experiments we consider random embeddings in the form of an randomly initialized but untrained BERT model.

3 Results

3.1 Architectural Properties

Table 1 shows results for vectors extracted from layer 0 (uncontextualized) and from layer 8 (contextualized). One can see that the original model (line/ID 0) fits the data well (perpl = 7.42). As expected it shows a high degree of multilinguality. Alignment is an easy task with shared position embeddings (as the gold standard alignment is the identity alignment): Align. = 1.00. Retrieval works better with contextualized representations on layer 8 (0.97 vs. 0.54), whereas word translation works better on layer 0 (0.88 vs. 0.79), as expected. Overall the embeddings seem to capture the similarity of English and Fake-English exceptionally well (see Figure 4 for a PCA of the token embeddings). The random BERT models perform poorly (lines 18 and 19), except for the alignment task with shared

position embeddings. We need to keep this artefact of the alignment task in mind when comparing results.

When applying our **architectural modifications** (lang-pos, shift-special, no-random) individually we see slight decreases in multilinguality (lines 1, 2, 4), but for no-random we even see an increase (line 4). Apparently, applying just a single modification is not enough, as this can be compensated by the model. Indeed, when using two modifications at a time (lines 5–7) multilinguality goes down and when using all three modifications (line 8) one can see that the degree of multilinguality is drastically lowered. Note that the language model quality in terms of perplexity is stable across all models (6.60 – 7.89). Keep in mind that perplexity is not directly comparable when considering the *no-random* modification.

Compared to the original, the **overparameterized** model (ID 15) exhibits lower scores for word translation, but higher ones for retrieval. Generally it fits the data better by showing even lower perplexity (2.10). However, when we apply our three architectural modifications (ID 17), multilinguality drops to 0. We thus conclude that by decoupling languages with the proposed modifications and severely increasing the number of parameters, it is possible to get a language model with the comparable performance that is not multilingual. Conversely, this result indicates that the four architectural properties are necessary for BERT to be effectively multilingual.

3.2 Linguistic Properties

As mentioned above we assume that having the same "positional structure" in two languages helps BERT to align embedding spaces. Table 1 shows the effect of inverting Fake-English (lines 3, 9). Multilinguality breaks almost completely – independently of any architectural modifications (ID 3 vs. 9). Having a language with the exact same structure (same ngram statistics, vocabulary size etc.), only with inverted order, seems to block BERT from creating a multilingual space – even in the most favorable conditions. Most importantly though, the language model fit is unchanged: perplexity is almost the same as for the original model. We conclude that having a similar word order structure is necessary for BERT to create a multilingual space. Our hypothesis is that the positional encoding hinders the learning of a multilingual space,

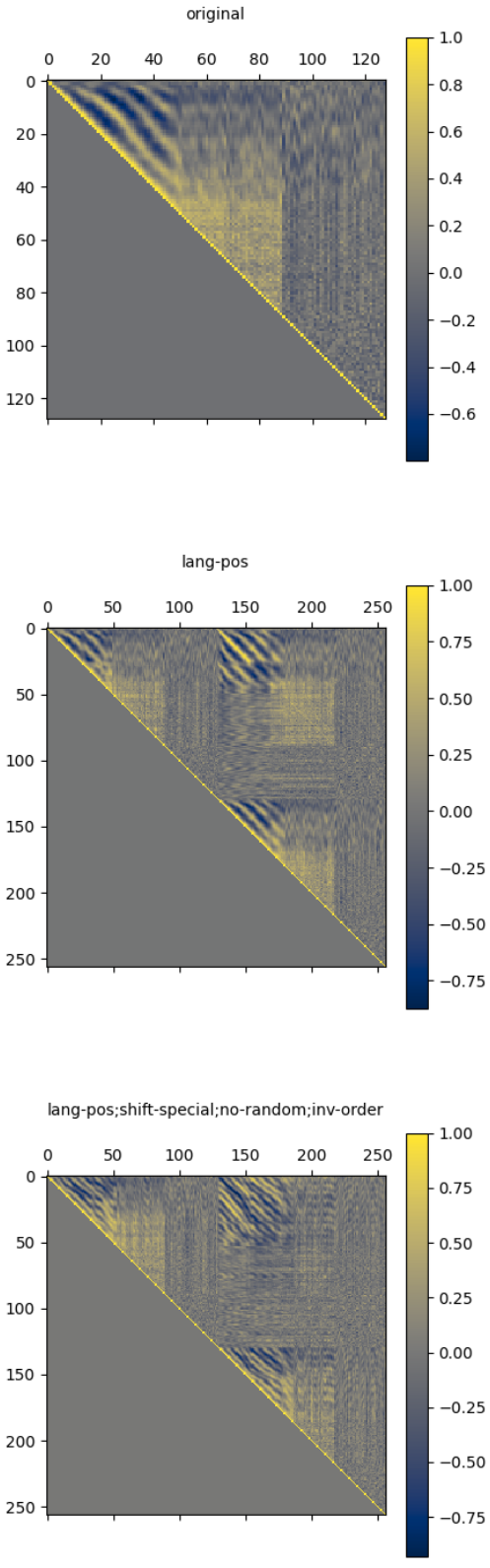


Figure 5: Cosine similarity matrices of position embeddings. The maximum length after tokenization in our experiments is 128. If language specific position embeddings are used, the range 0-127 is used by English and 128-255 by Fake-English. Note that similarity matrices are symmetric by definition and thus we only show the upper triangular matrix.

ID	Experiment	MLM Perpl.	Layer 0 Align.	Layer 0 Retr.	Layer 0 Trans.	Layer 8 Align.	Layer 8 Retr.	Layer 8 Trans.
0	original	7.42 $_{0.15}$	1.00 $_{0.00}$	0.54 $_{0.08}$	0.88 $_{0.02}$	1.00 $_{0.00}$	0.97 $_{0.00}$	0.79 $_{0.03}$
1	lang-pos	7.73 $_{0.11}$	0.98 $_{0.02}$	0.68 $_{0.15}$	0.40 $_{0.09}$	0.99 $_{0.00}$	0.61 $_{0.20}$	0.09 $_{0.05}$
2	shift-special	7.47 $_{0.13}$	1.00 $_{0.00}$	0.49 $_{0.03}$	0.88 $_{0.01}$	1.00 $_{0.00}$	0.97 $_{0.00}$	0.63 $_{0.13}$
4	no-random	*6.83 $_{0.35}$	1.00 $_{0.00}$	0.58 $_{0.05}$	0.87 $_{0.02}$	1.00 $_{0.00}$	0.96 $_{0.01}$	0.82 $_{0.04}$
5	lang-pos;shift-special	7.89 $_{0.40}$	0.84 $_{0.17}$	0.36 $_{0.30}$	0.27 $_{0.20}$	0.88 $_{0.16}$	0.38 $_{0.32}$	0.05 $_{0.04}$
6	lang-pos;no-random	*7.42 $_{0.29}$	0.99 $_{0.01}$	0.47 $_{0.21}$	0.36 $_{0.12}$	0.99 $_{0.01}$	0.47 $_{0.26}$	0.25 $_{0.12}$
7	shift-special;no-random	*6.60 $_{0.19}$	1.00 $_{0.00}$	0.58 $_{0.04}$	0.85 $_{0.01}$	1.00 $_{0.00}$	0.97 $_{0.00}$	0.79 $_{0.04}$
8	lang-pos;shift-special;no-random	*7.51 $_{0.43}$	0.63 $_{0.34}$	0.22 $_{0.31}$	0.18 $_{0.22}$	0.68 $_{0.38}$	0.23 $_{0.36}$	0.11 $_{0.13}$
15	overparam	2.10 $_{0.03}$	1.00 $_{0.00}$	0.68 $_{0.06}$	0.63 $_{0.05}$	1.00 $_{0.00}$	0.96 $_{0.01}$	0.47 $_{0.06}$
16	lang-pos;overparam	2.40 $_{0.02}$	0.33 $_{0.14}$	0.00 $_{0.00}$	0.01 $_{0.00}$	0.41 $_{0.15}$	0.00 $_{0.00}$	0.00 $_{0.00}$
17	lang-pos;shift-special;no-random;overparam	*1.65 $_{0.02}$	0.09 $_{0.05}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.09 $_{0.07}$	0.00 $_{0.00}$	0.00 $_{0.00}$
3	inv-order	8.71 $_{0.19}$	0.08 $_{0.01}$	0.00 $_{0.00}$	0.01 $_{0.00}$	0.08 $_{0.02}$	0.01 $_{0.00}$	0.00 $_{0.00}$
9	lang-pos;inv-order;shift-special;no-random	*7.66 $_{0.16}$	0.12 $_{0.03}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.10 $_{0.04}$	0.00 $_{0.00}$	0.00 $_{0.00}$
18	untrained	3479.62 $_{31.49}$	0.98 $_{0.01}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.98 $_{0.01}$	0.00 $_{0.00}$	0.00 $_{0.00}$
19	untrained;lang-pos	3482.45 $_{65.13}$	0.03 $_{0.01}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.03 $_{0.01}$	0.00 $_{0.00}$	0.00 $_{0.00}$

Table 1: The table shows model fit and evaluation of multilinguality for architectural and linguistic modifications. We report the mean and standard deviation (subscript) across 5 different random seeds. Models are trained with the same hyperparameters. *: perplexity only comparable to other methods marked with star.

as BERT needs to learn that the word on position 0 in English is similar to word on position n in Fake-English. However, n (the sentence length) varies from sentence to sentence. This might be an argument that relative position embeddings rather than absolute position embeddings might benefit multilinguality

Figure 5 shows the similarity among position embeddings for three models. The top panel shows the similarity for the original model without modifications. The middle panel shows the similarity with language specific position embeddings (positions 0-127 are for English 128-255 for Fake-English).

1) One can see that despite having language specific position embeddings the embeddings live in a similar space and exhibit a similar structure: in the upper right quadrant there is a clear yellow diagonal at the beginning which gets weaker at the end (barely visible here, see appendix for a large version). The bottom panel shows that for a non-multilingual model where Fake-English has been inverted the position embeddings live in different spaces: there is no diagonal visible in the similarity between English and Fake-English position embeddings.

2) Another effect one can spot is that smaller position embeddings are trained much more than larger ones (which occur less frequently). Especially in the range from 90-128 the similarities look almost random (a sentence length which occurs rarely). We suspect that embedding similarity correlates with the number of gradient updates a single position embedding receives. The positions 0 and 1 receive a gradient update in every step and can thus be considered an average of all gradient updates (disregarding the random initialisation). Smaller position embeddings receive the majority of gradient updates as well. This might be the main reason for the clear diagonal pattern in the upper left corners.

Experiment	MLM Perpl.	Layer 0 Al.	Layer 0 Retr.	Layer 0 Trans.	Layer 8 Al.	Layer 8 Retr.	Layer 8 Trans.
original	7.42 $_{0.15}$	1.00 $_{0.00}$	0.54 $_{0.08}$	0.88 $_{0.02}$	1.00 $_{0.00}$	0.97 $_{0.00}$	0.79 $_{0.03}$
no-p.	11.48 $_{0.28}$	0.99 $_{0.00}$	0.21 $_{0.02}$	0.28 $_{0.01}$	0.99 $_{0.00}$	0.70 $_{0.05}$	0.15 $_{0.03}$
lang-pos;no-p.	12.65 $_{0.45}$	0.86 $_{0.09}$	0.15 $_{0.15}$	0.07 $_{0.05}$	0.92 $_{0.07}$	0.16 $_{0.18}$	0.02 $_{0.01}$

Table 2: Results showing the effect of having a sentence-parallel training corpus.

3.3 Corpus Comparability

So far we have always trained on a sentence-parallel corpus. To train on a non-parallel corpus we use the first half of our parallel corpus for English and the second half for Fake-English. To alleviate the reduction of training data we train for twice as many epochs.

Table 2 shows that indeed multilinguality decreases as the training corpus becomes non-parallel. However, perplexity almost doubles indicating that the model fit deteriorates drastically. This is not unexpected.

This might be an indication that the more comparable a training corpus is across languages the higher the multilinguality. However, as we show in §3.4 model fit correlates drastically with multilinguality. Thus we should take these results with a grain of salt and verify them in future work.

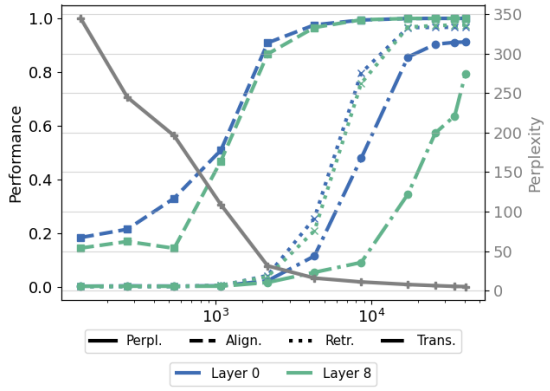


Figure 6: The longer a model is trained, the more multilingual it gets. x-axis shows training steps. This model uses language specific position embeddings. We show the result for a single seed.

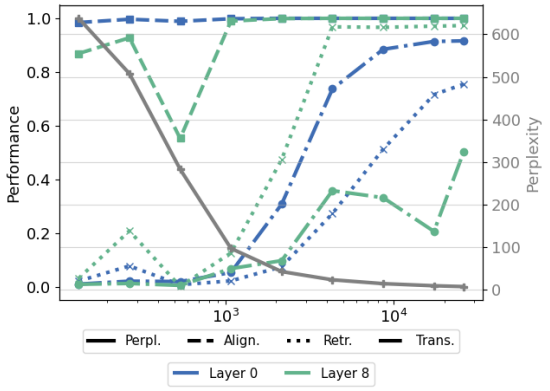


Figure 7: Showing the performance over training steps for a model with language specific position embeddings and $p = (0.4, 0.06, 0.54)$. Multilinguality starts to kick in much earlier, but has chaotic drops in the middle.

3.4 Multilinguality during Training

During experimentation we noticed that multilinguality is correlated with perplexity. We investigate this effect more closely in Figure 6. We plot evaluation measures and perplexity for our original model configuration with language specific position embeddings over training time. Clearly multilinguality rises over time. We hypothesize that once a model needs to fit the training data, efficient parameter usage is beneficial. Thus the model identifies shared structures across the languages in order to use parameters more efficiently, and therefore becomes multilingual. Again one can see that word alignment is the easiest task, followed by retrieval and then word translation.

This might raise suspicion that our overaparameterized models in Table 1, especially ID 17 is simply not trained enough and are therefore not multilingual. Note, that perplexity for the overparameterized models is already much lower compared to our standard model. To provide additional evidence that multilinguality does not appear at a later training stage we train this model configuration for 250 epochs and report results in Table 3.

4 Improving Multilinguality

Up to now we tried to destructively break the multilinguality of BERT to identify the reason why it produces multilingual representations. The overall objective, however, is to identify components that are important for multilinguality and steer them in a way to create better multilingual models.

The standard BERT architecture already has shared position embeddings, and shared special tokens and we cannot change linguistic properties of languages. Thus our takeaways reduce to training smaller BERT models in order to get multilingual spaces – however, BERT-base might already be underparameterized given that it is trained on 104 large Wikipedias with 104 languages.

We suspect that increasing $p_{[rand]}$, i.e., how often a masked word is replaced with a random word increases multilinguality. In our experiments, we could not get an overall increase, but Figure 7 compared to Figure 6 clearly shows that a model with increased $p_{[rand]}$ becomes multilingual much earlier in the training process. This is particularly important for training BERT on large amounts of data. Given how expensive training is, it may not be possible train a model long. In general, BERT has been found to be undertrained by several studies (Liu et al., 2019; Rönnqvist et al., 2019). Thus achieving multilinguality early in the training process most probably yields higher quality models.

On the downside, training does seem to be unstable (with unexplainable drops in multilinguality). Random replacements might be too noisy for a trained model. In future work, we thus plan to modify this by not replacing masked words randomly, but with nearest neighbours in a different language in the initial embedding space.

4.1 Real World Word Order

To verify whether similar word order across languages influences the multilinguality we propose to compute a word reordering metric and correlate this metric with the performance of 0-shot transfer

	Experiment	N Epochs	MLM Perpl.	Layer 0 Align.	Layer 0 Retr.	Layer 0 Trans.	Layer 8 Align.	Layer 8 Retr.	Layer 8 Trans.
0	original	100	7.42 $_{0.15}$	1.00 $_{0.00}$	0.54 $_{0.08}$	0.88 $_{0.02}$	1.00 $_{0.00}$	0.97 $_{0.00}$	0.79 $_{0.03}$
17	lang-pos;shift-special;no-random;overparam	50	*3.51 $_{0.05}$	0.09 $_{0.04}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.09 $_{0.08}$	0.00 $_{0.00}$	0.00 $_{0.00}$
17	lang-pos;shift-special;no-random;overparam	100	*1.65 $_{0.02}$	0.09 $_{0.05}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.09 $_{0.07}$	0.00 $_{0.00}$	0.00 $_{0.00}$
17	lang-pos;shift-special;no-random;overparam	250	*1.13 $_{0.00}$	0.09 $_{0.05}$	0.00 $_{0.00}$	0.00 $_{0.00}$	0.09 $_{0.07}$	0.00 $_{0.00}$	0.00 $_{0.00}$

Table 3: Even when continuing the training for a long time and most likely overfitting the training data, overparameterized models with architectural modifications do not become multilingual. *: perplexity only comparable to other methods marked with star

capabilities of mBERT. To this end we consider the performance of mBERT on XNLI. We follow Birch and Osborne (2011) in computing word reordering metrics between parallel sentences (XNLI is a parallel corpus). More specifically we compute the Kendall's tau distance. To this end, we compute word alignments between two sentences using the Match algorithm by Sabet et al. (2020), which directly yield a permutation between sentences as requried by the distance metric. We compute the metric on the test data of XNLI and average Kendall's tau distance across sentences to get a single score per language.

The Pearson correlation between Kendall's tau distance and the XNLI classification accuracy in a zero-shot scenario (mBERT only finetuned on English and tested on all other languages) is 46%.

While this value is lower than expected, there is still a clear correlation visible. We will extend this experiment to more results as reported in (Hu et al., 2020) and examine this effect more closely in future work.

5 Related Work

There is a range of work analyzing the reason for BERT's multilinguality. Singh et al. (2019) use canonical correlation analysis to show that BERT stores language representations in different subspaces rather than creating a single common space. They investigate how subword tokenization influences multilinguality. Artetxe et al. (2019) show that neither a shared vocabulary nor joint pretraining is essential for multilinguality. The initial embedding layer of mBERT can be retrained for different languages given that all other weights are fixed. Wang et al. (2019) provide an extensive study to find the reason for multilinguality. In terms of architecture they research the depth, number of parameters, number of attention heads, learning objectives, adding language identify markers as well as character vs. subword vs. word processing. In their conclusion deepness is essential, subword processing and omitting the next-sequence-prediction task helps slightly and language identity markers do not show any effect. They also investigate language properties such as word order similarity and unigram frequencies, and conclude that structural similarity across languages is important, without further defining this term. Last, Wu et al. (2019) find that a shared vocabulary is not required. They claim that shared parameters in the top layers are required for multilinguality. Further they show that different monolingual BERT models exhibit a similar structure and thus conclude that mBERT somehow aligns those isomorphic spaces.

We base much of our setup on (Wang et al., 2019) and try to continue their analysis. In contrast to the mentioned work we focus on architectural aspects that have not been explored before.

There is another line of work focusing on creating better multilingual models. Conneau and Lample (2019) introduce the translation modeling objective and propose XLM, a crosslingual model outperforming mBERT. Conneau et al. (2019) propose XLM-R, a multilingual Roberta model. We plan to investigate the multilinguality of this model in future work. Cao et al. (2020) improve the multilinguality of mBERT by introducing a regularization term in the objective, quite similar to the creation of static multilingual embedding spaces. Huang et al. (2019) extend mBERT pretraining with 3 additional tasks and show an improved overall performance.

Finally, there is a lot of research that analyzes how multilingual mBERT actually is. Many papers find that mBERT yields competitive performance across a huge range of languages and tasks, such as parsing and NER (Pires et al., 2019; Wu and Dredze, 2019). Rönnqvist et al. (2019) investigate language generation capabilities and conclude that mBERT is undertrained with respect to individual languages – an aspect that we can confirm with our experiments. Hu et al. (2020) provide an extensive evaluation benchmark covering 40 languages and 9 tasks and show that mBERT indeed yields good

performance.

6 Conclusion

In this work-in-progress report we showed which architectural and lingustic properties are a *necessary requirement* for BERT to yield crosslingual representations. The main takeaways are threefold: **i)** Shared position embeddings, shared special tokens, replacing masked tokens with random tokens and having a limited amount of parameters are necessary elements for a BERT model to be multilingual. **ii)** Word order is essential. BERT does not yield a multilingual space with normal English and inverted Fake-English. **iii)** The comparability of training corpora contributes to multilinguality.

The objective of the paper was to identify those components and use them in a second step to create a better multilingual BERT model. Initial experiments in this regard have been presented. All experiments have been made in a laboratory setting consisting of a mix of natural and synthetic data and we plan to validate the findings on parallel corpora with several natural languages.

We gratefully **acknowledge** funding through a Zentrum Digitalisierung.Bayern fellowship awarded to the first author. This work was supported by the European Research Council (# 740516). We thank Mengjie Zhao and Nina Pörner for fruitful discussions and valuable comments.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856*.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Alexandra Birch and Miles Osborne. 2011. Reordering metrics for MT. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1027–1035, Portland, Oregon, USA. Association for Computational Linguistics.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. *arXiv preprint arXiv:2002.03518*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*.

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.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 462–471, Gothenburg, Sweden. Association for Computational Linguistics.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 58–68, Baltimore, Maryland. 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.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *arXiv preprint arXiv:2003.11080*.

Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A universal language encoder by pre-training with multiple cross-lingual 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)*.

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 preprint arXiv:1907.11692*.

Edward Loper and Steven Bird. 2002. NLTK: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 63–70, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Thomas Mayer and Michael Cysouw. 2014. Creating a massively parallel bible corpus. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

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.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Samuel Rönnqvist, Jenna Kanerva, Tapio Salakoski, and Filip Ginter. 2019. Is multilingual bert fluent in language generation? In *Proceedings of the First NLPL Workshop on Deep Learning for Natural Language Processing*.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631.

Masoud Jalili Sabet, Philipp Dufter, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. *arXiv preprint arXiv:2004.08728*.

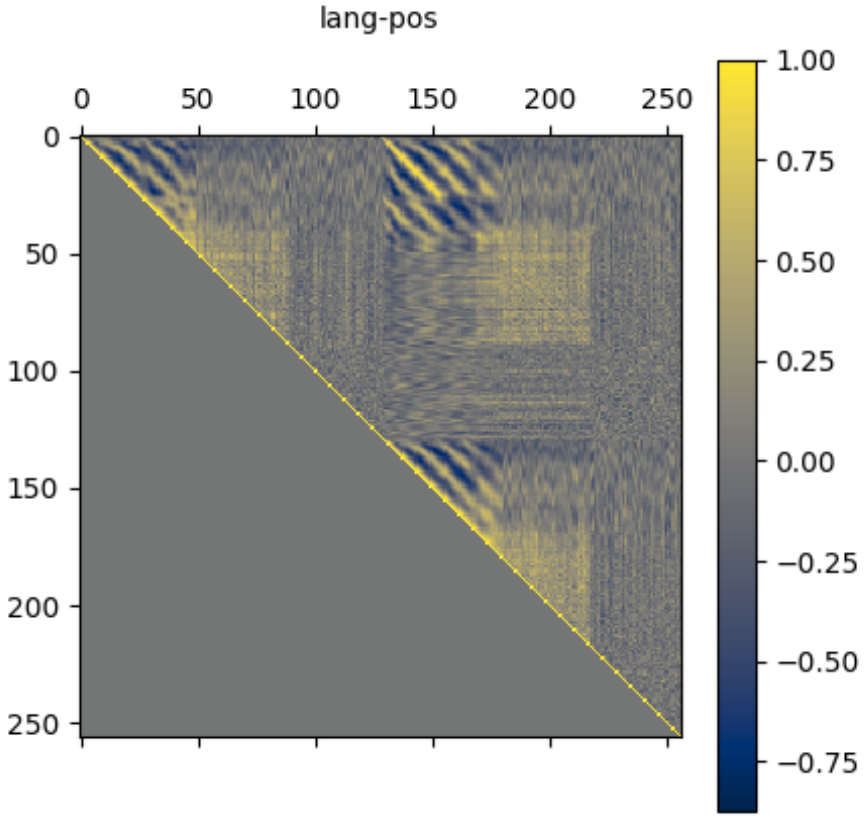
Jasdeep Singh, Bryan McCann, Richard Socher, and Caiming Xiong. 2019. Bert is not an interlingua and the bias of tokenization. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*.

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.

Zihan Wang, Stephen Mayhew, Dan Roth, et al. 2019. Cross-lingual ability of multilingual bert: An empirical study. *arXiv preprint arXiv:1912.07840*.

Shijie Wu, Alexis Conneau, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Emerging cross-lingual structure in pretrained language models. *arXiv preprint arXiv:1911.01464*.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of bert. 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 833–844.



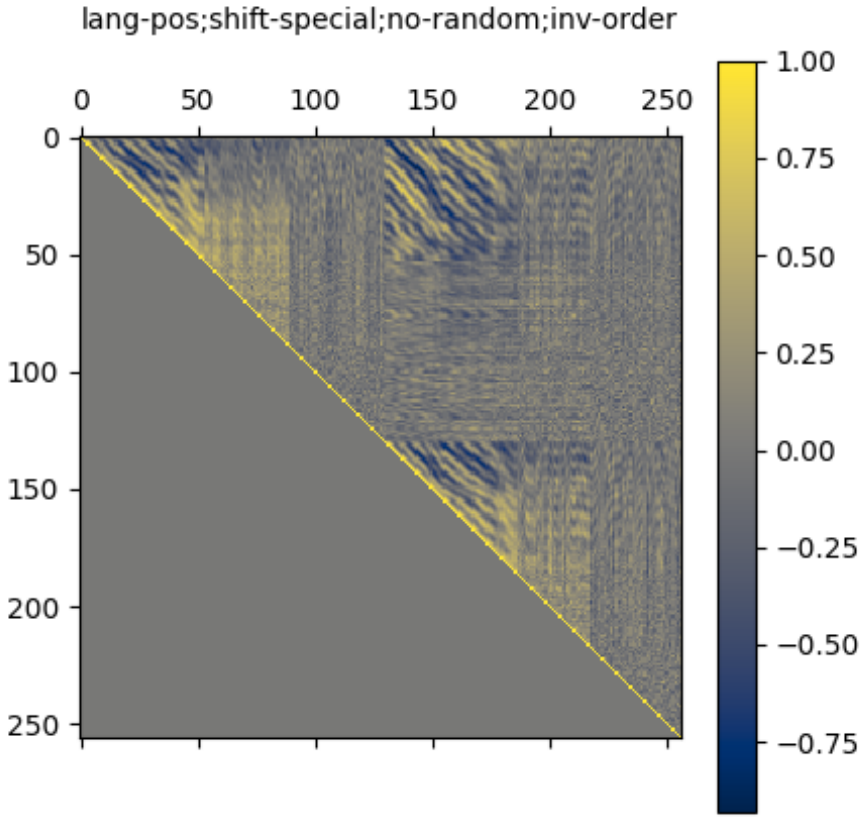


Figure 8: Larger version of Figure 5.