Multilingual Pre-training with Self-supervision from Global Co-occurrence

Xi Ai
College of Computer Science
Chongqing University
barid.x.ai@gmail.com

Bin Fang
College of Computer Science
Chongqing University
fb@cqu.edu.cn

Abstract

Global co-occurrence information is the primary source of structural information on multilingual corpora, and we find that analogical/parallel compound words across languages have similar co-occurrence counts/frequencies (normalized) giving weak but stable self-supervision for cross-lingual transfer. Following the observation, we aim at associating contextualized representations with relevant (contextualized) representations across languages with the help of co-occurrence counts. The result is MLM-GC (MLM with Global Co-occurrence) pre-training that the model learns local bidirectional information from MLM and global co-occurrence information from a log-bilinear regression. Experiments show that MLM-GC pre-training substantially outperforms MLM pre-training for 4 downstream cross-lingual tasks and 1 additional monolingual task, showing the advantages of forming isomorphic spaces across languages.

1 Introduction

Empirical studies (Lample et al., 2018a; Conneau et al., 2020a,c) show multilinguality and cross-linguality emerge from MLM pre-training on multilingual corpora without any supervision. The model is trained/pre-trained as a generator that yields masked token probabilities over the vocabulary. To improve cross-lingual transfer, we present MLM-GC (MLM with Global Co-occurrence) with the combined objective of the generator and a global log-bilinear regression for multilingual pre-training. Our starting point is from two observations on multilingual MLM pre-training.

Language's structural information is every property of an individual language that is invariant to the script of the language. Conneau et al. (2020c); Karthikeyan et al. (2020); Sinha et al. (2021); Pires et al. (2019) show that structural similarities across languages can contribute to cross-lingual transfer. Co-occurrence information or n-gram is the primary source of structural information available to all methods. Some methods like span-based masking (Devlin et al., 2019; Joshi et al., 2020; Levine et al., 2021) now exist to leverage this information for new masking schemes in *monolingual* MLM pre-training, aiming at improving context understanding. However, in *multilingual* MLM pre-training, the question still remains as to how meaning is generated from these statistics on multilingual corpora, how the structural similarities could be learned from that meaning across languages, and how cross-lingual transfer might be improved from that meaning.

Furthermore, GloVe (Pennington et al., 2014) prove that leveraging global co-occurrence information can search for relevant information on *monolingual* embedding space. Inspired by GloVe, we assume that global co-occurrence information can also be used to search for relevant information across languages on *multilingual* corpora. This assumption underlies Zipf's law (Ha et al., 2002; Søgaard, 2020) that analogical words and compound words across languages might have similar frequencies/counts on the multilingual corpora. Our empirical studies further justify our assumption that analogical/parallel compound words across languages have similar co-occurrence counts (normalized). Meanwhile, in multilingual MLM pre-training, one of the ultimate goals is to form contextualized representations. Then, global co-occurrence information might be used to regularize representation learning in multilingual MLM pre-training, which allows for better contextualized representations in cross-lingual transfer.

In this work, we present MLM-GC to utilize global co-occurrence information. MLM-GC builds on MLM with an extra objective of global log-bilinear regression that minimizes the error between dot products of neighboring contextualized representations and the matrix of global co-

Findings of the Association for Computational Linguistics: ACL 2023, pages 7526–7543

occurrence counts. Since MLM only needs to predict masked tokens, we only consider the contextualized representations of the masked tokens and their neighbors, factorizing relevant global co-occurrence counts from the matrix. The model is pre-trained to learn bidirectional information from MLM and the global co-occurrence information from the global log-bilinear regression. On multilingual corpora, MLM-GC pre-training can improve cross-lingual transfer because analogical/parallel compound words across languages might have similar co-occurrence counts allowing for cross-lingual transfer, which is justified in our empirical studies on translation pairs.

We have three contributions. **1)** We present MLM-GC pre-training for multilingual tasks. The model is additionally supervised by co-occurrence counts on multilingual corpora. **2)** MLM-GC pre-training outperforms MLM pre-training on 4 multilingual/cross-lingual tasks. The objective of MLM-GC can be adapted to encoder-decoder-based MLM models, e.g., MASS (Song et al., 2019) and encoder-based MLM models, e.g., XLM (Lample and Conneau, 2019). MLM-GC pre-training can also work on monolingual corpora for language understanding tasks. **3)** MLM-GC pre-training can help the model to form isomorphic embedding spaces across languages, which is potentially useful for cross-lingual and multilingual tasks. Our empirical study shows that analogical compound words across languages have similar co-occurrence counts (normalized) contributing to structural similarities across languages for cross-lingual transfer.

2 Related Work and Comparison

Structural Similarity and Zipf's Law Zipf's law (Zipf, 1949, 2013; Søgaard, 2020) indicates that words or phrases appear with different frequencies, and one may suggest analogical words or phrases appear with relatively similar frequencies in other languages. In multilingual MLM pre-training, Conneau et al. (2020c); Karthikeyan et al. (2020); Pires et al. (2019); Karthikeyan et al. (2020); Sinha et al. (2021) shed light on studying structural information and find that structural similarities across languages are essential for multilinguality, where in this paper, structural similarities mean similar counts as Zipf's law indicated. We follow this line, consider structural similarities from co-occurrence counts, and provide an empirical study to observe how the model learns structural similarities from global co-occurrence counts on multilingual corpora. Meanwhile, GloVe (Pennington et al., 2014) report that co-occurrence counts can provide regularities for embeddings to understand word analogies for monolingual tasks. We extend the scope of GloVe to contextualized representations and multilingual tasks, helping the model form isomorphic spaces across languages in multilingual MLM pre-training.

N-gram, Co-occurrence, and Regularity in MLM pre-training Studying co-occurrence or *n-gram* is not a novel idea in MLM pre-training. Whole Word Masking (Devlin et al., 2019), SpanBERT (Joshi et al., 2020), and PMI-Masking (Levine et al., 2021) suggest *n-gram* spans across several sub-tokens for masking to improve context understanding in monolingual tasks because the model may only learn from easier multi-tokens instead of usefully hard context, where easier multi-tokens are in a subset of the context and result in sub-optimization. In contrast, we show that co-occurrence counts can refine contextualized representations for improving context understanding and allow for better cross-lingual transfer, suggesting a new objective for MLM pre-training instead of a new masking scheme to capture global co-occurrence information in multilingual pre-training. On the other hand, for cross-lingual transfer, the contextualized representations could be further regularized and refined by aligning cherry-picked pairs after MLM pre-training on multilingual corpora (Ren et al., 2019; Chaudhary et al., 2020; Wang et al., 2020; Cao et al., 2020; Aldarmaki and Diab, 2019; Artetxe et al., 2020; Ai and Fang, 2021). Compared to that, MLM-GC pre-training does not require dictionaries, translation tables, or statistical machine translation models.

3 Approach

3.1 Global Regression Modeling in Monolingual Embedding Space

GloVe (Pennington et al., 2014) presents a log-bilinear regression model:

$$\mathcal{L} = \sum_{i,j=1}^{V} f(X_{w_i w_j})(E_{w_i}^T E_{w_j} - \log X_{w_i w_j})^2, \tag{1}$$

where $f(x) = \begin{cases} (x/x_{max})^\alpha, x < x_{max} \\ 1, otherwise \end{cases}$, V is the vocabulary, E_w is the embedding of token w, X

stands for the matrix of token-token co-occurrence counts, entries $X_{w_i w_j}$ tabulate the number of times token w_j occurs in the context of token w_i, and x_{max} is empirically set to 100. The model is able to distinguish relevant embeddings from irrelevant embeddings and discriminate between the two relevant embeddings.

3.2 Global Co-occurrence Modeling for Contextualized Representations

In MLM pre-training, when w_t at the position t is replaced by the artificial masking token $[\mathcal{M}]_t$, *the final hidden state or the contextualized representation* $H_{[\mathcal{M}]_t}$ of position t is factorized from the final sequence representation of the input sentence to predict w_t (with a $softmax$ operation). We further factorize a neighboring contextualized representation H_{w_k} (of w_t) at position k for the neighboring token w_k. Note that w_k could be masked (if span-based masking strategies are applied, e.g., MASK (Song et al., 2019)) or unmasked (e.g., XLM (Lample and Conneau, 2019)), and we test both scenarios in our experiments. Then, similar to global regression modeling in monolingual embedding space, we consider a regression model [1]:

$$\mathcal{L}_{[\mathcal{M}]_t w_k} = f(X_{w_t w_k})(H^T_{[\mathcal{M}]_t} H_{w_k} - \log X_{w_t w_k})^2. \tag{2}$$

For all the neighboring tokens $w_{t\pm n}$ of the input sentence at position $[t-n, ..., t) \cup (t, ..., t+n]$, i.e., excluding position t, we have the model $\mathcal{L}_{GC}$. Then, we employ the new global log-bilinear regression model in MLM pre-training. Formally, given the factorized $H_{w_{t\pm n}}$ and $H_{[\mathcal{M}]_t}$ from H and $X_{w_t w_{t\pm n}}$ from X, we have the model:

$$\mathcal{L}_{GC} = \frac{1}{2n} \sum_{n} f(X_{w_t w_{t\pm n}})(\frac{H^T_{[\mathcal{M}]_t} H_{w_{t\pm n}}}{\sqrt{d}} - \log X_{w_t w_{t\pm n}})^2, \tag{3}$$

where d is the model dimension. Compared to Eq.1, we add scaling $\sqrt{d}$ and weight $\frac{1}{2n}$ to make training stable, where $\sqrt{d}$ is inspired by scaled dot-product attention (Vaswani et al., 2017) to prevent the dot products get large. They serve as principled hyper-parameters.

To obtain the matrix of token-token co-occurrence counts on multilingual corpora for multilingual tasks, we follow GloVe's suggestion that a distance weight scheme is employed. Specifically, in a context window of size $2n+1$, we calculate the token-token co-occurrence counts for positions $[t-n, ..., t, ..., k+n]$ with the rule $[c_{lang}/(n+1), ..., c_{lang}/2, 0, c_{lang}/2, ..., c_{lang}/(n+1)]$ over the shared vocabulary, which means we do not calculate the unigram counts or self-co-occurrence $X_{w_t w_t}$ for the centric token w_t at position t. Meanwhile, we are aware that the probability is not normalized and equivalent to token-token co-occurrence counts on the multilingual corpora. However, not all the languages have the same amount of samples in the corpora (e.g., low-resource vs. high-resource). Considering this, we use the language-wise constant $c_{lang} = C_{En}/C_{lang}$, where C_{En} is the total number of tokens in English corpora, and C_{lang} is the total number of tokens in the language $lang$, i.e., *co-occurrence counts are normalized by* c_{lang}.

3.3 Multilingual MLM-GC Pre-training

In multilingual pre-training, we have a combined objective of MLM and global co-occurrence modeling[2], attempting to train the model to understand the masked tokens from bidirectional information and linguistic structures surrounding the masked tokens from global co-occurrence counts, and the result is our MLM-GC pre-training:

$$\mathcal{L}_{MLM-GC} = \mathcal{L}_{MLM} + \lambda \mathcal{L}_{GC}. \tag{4}$$

In the early experiment, we experiment with $\lambda \in \{0.1, 0.5, 1, 2\}$. We find $\lambda = 1$ is a general choice for experiments. On the other hand, we find $warm\_up$ (Vaswani et al., 2017) of lr, $\sqrt{d}$, and $\frac{1}{2n}$ (Eq. 3) are significant. The model might collapse to $\mathcal{L}_{GC}$ without $warm\_up$, $\sqrt{d}$, or $\frac{1}{2n}$ because $\mathcal{L}_{GC}$ converges too fast and is unstable. In this situation, the model ignores the objective of MLM. Then, the model can only learn co-occurrence information and does not learn the language knowledge. The result is presented in Table 1.

Improved Contextualized Representation $\mathcal{L}_{GC}$ considers the correspondence in the context $[t-n, ..., t) \cup (t, ..., t+n]$ with an explicit objective. In this way, the model is encouraged to learn from usefully hard context instead of easier multi-tokens under the supervision from co-occurrence information, where easier multi-tokens are in a subset of the context and result in sub-optimization

[1] We provide an alternative in Appendix D

[2] We discuss the scope and limitation in §Limitation.

#	Model	MUSE cosine
1	MUSE Lample et al. (2018a)	0.38
2	XLMLample and Conneau (2019)	0.55
3	XLM + OURS $\lambda = 1$ (default)	0.60
4	XLM + OURS $\lambda = 0.1$,	0.59
5	XLM + OURS $\lambda = 0.5$	0.60
6	XLM + OURS $\lambda = 2$	0.58
7	XLM + OURS $\lambda = 1$ and no $warm\_up$ of learning rate	fail
8	XLM + OURS $\lambda = 1$ and no $\sqrt{d}$	fail
9	XLM + OURS $\lambda = 1$ and no $\frac{1}{2n}$	fail

Table 1: Early Experiments on MUSE tasks.

(Levine et al., 2021), as discussed in §Related Work. Meanwhile, co-occurrence counts help the model disambiguate word representations (Ai and Fang, 2022) in language modeling by distinguishing relevant information from irrelevant information and discriminating between the two relevant information in the language.

Improved Cross-lingual Transfer With the objective of $\mathcal{L}_{GC}$, we aim at associating $H^T_{[\mathcal{M}]_t} H_{w_{t\pm n}}$ with $H^T_{[M]_{\tilde{t}}} H_{\tilde{w}_{\tilde{t}\pm n}}$ of different languages if $X_{w_t w_{t\pm n}} = X_{\tilde{w}_{\tilde{t}}\tilde{w}_{\tilde{t}\pm n}}$, where compound words $w_t w_{t\pm n}$ and $\tilde{w}_{\tilde{t}}\tilde{w}_{\tilde{t}\pm n}$ are analogical in different languages. In this way, it underlies the basic assumption that analogical compound words across languages have similar co-occurrence counts (normalized by c_{lang}), i.e., $w_t w_{t\pm n}$ and $\tilde{w}_{\tilde{t}}\tilde{w}_{\tilde{t}\pm n}$ are analogical compound words $\implies X_{w_t w_{t\pm n}} = X_{\tilde{w}_{\tilde{t}}\tilde{w}_{\tilde{t}\pm n}}$. Although Zipf's law supports this assumption (Ha et al., 2002; Søgaard, 2020) in linguistics, we are still interested in the questions: *how it reflects on the multilingual corpora we use* and *whether analogical pair of* $w_t w_{t\pm n}$ *and* $\tilde{w}_{\tilde{t}}\tilde{w}_{\tilde{t}\pm n} \implies X_{w_t w_{t\pm n}} = X_{\tilde{w}_{\tilde{t}}\tilde{w}_{\tilde{t}\pm n}}$. To answer these questions, we extract all the pairs of parallel compound words in En and De from the open-source translation tables (OPUS, Wikipedia v1.0)$\diamond$, e.g., "ist die" (De) and "is the" (En), and compute co-occurrence counts on $\{De, En\}$ Wikipedia dumps (the same dataset we use in our experiment). For any pair, we compute the absolute difference $|log(De) - log(En)|$, the sum $log(De) + log(En)$ (sorted into bins), and the ratio $|log(De) - log(En)|/(log(De) + log(En))$ for statistics in Figure 1. The figure tells us that the absolute difference avg and the ratio avg for all the pairs are relatively small and have narrow confidence (95%) intervals. Although the absolute difference avg is proportional to the sum, the ratio avg has no proportional relationship with the sum and is small throughout all the bins. Note that some pairs have low translation scores resulting in large absolute differences. The absolute difference avg is not 0, i.e., an exact match for any pair. However, it still confirms that analogical compound words across languages have similar (but not identical) co-occurrence counts, which might give weak (not 0) but stable (relatively small with high confidence) self-supervision for cross-lingual transfer. Meanwhile, the model is encouraged to distinguish relevant information from irrelevant information and to discriminate between the two relevant information across languages from co-occurrence counts and refine contextualized representations accordingly, which is beneficial for cross-lingual transfer. For example, in our experiment ($n = 2$), given the translation pair "ist die" (De) and "is the" (En), the relevant pair "ist die" and "is a" (En), the irrelevant pair "ist die" and "locally known" (En), we find $|log(ist\ die) - log(is\ the)| = 0.67 < |log(ist\ die) - log(is\ a)| = 1.73 < |log(ist\ die) - log(locally\ known)| = 5.45 < |log(ist\ die) - log(En\ avg)| = 5.58$, where $log(En\ avg)$ is the avg co-occurrence counts in En.

Efficiency *1)* Computing the co-occurrence matrix is laborious on large corpora. However, it requires a single pass through the entire corpora to collect the statistics, which is a one-time up-front cost and is easy to obtain new information from new corpora for updating. *2)* For memories, the co-occurrence matrix is huge, e.g., $\approx$ 11 G for a 60k BPE vocabulary with float 32. However, it is somewhat trivial because the memory is allocated to CPUs, not GPUs. This can be automatically finished by DL platforms like TensorFlow. Also, the matrix can be formatted to float 16 or even float 8 by pre-$logging$ the co-occurrence counts, which will significantly reduce the memory. *3)* Meanwhile, we save the token-token co-occurrence matrix as dictionaries {(w_i, w_j): token-token co-occurrence counts} so that querying the co-occurrence counts for $X_{w_i w_j}$ is O(1).

Tokenization Sub-token-level vocabularies may impact the co-occurrence counts. In extreme cases, several connective tokens of co-occurrence may only come from one word. However, As discussed in §Improved Contextualized Representation, even in this scenario, the model can be improved from the co-occurrence counts in MLM-GC pre-training. See experiments in Appendix B.

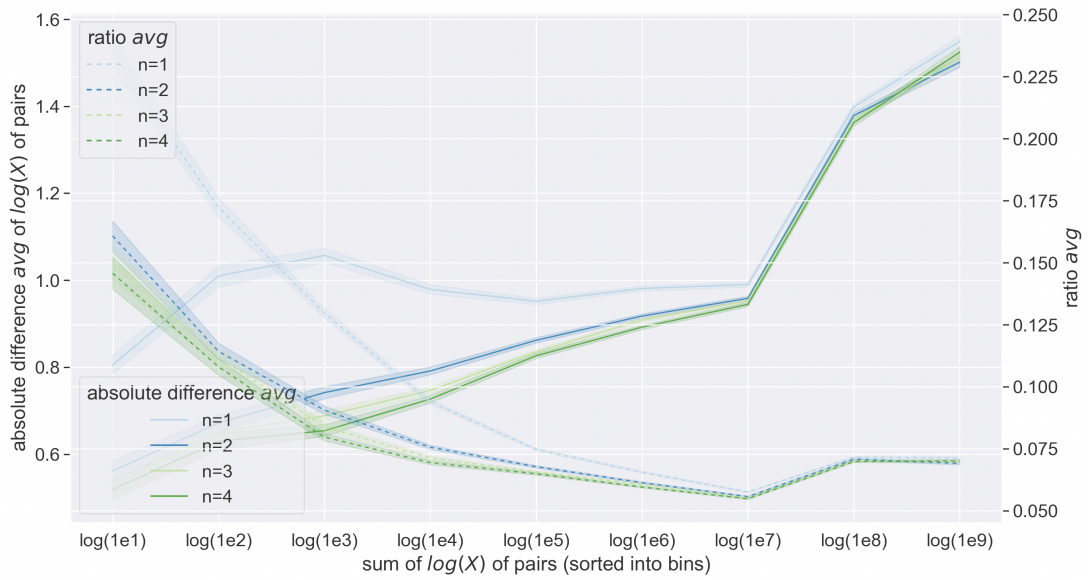


Figure 1: Study of co-occurrence counts of pairs across languages on $\{De, En\}$ corpora. We report absolute difference avg between (word-word) parallel co-occurred words across languages and 95% confidence interval.

4 Experiment

All the links of datasets, libraries, scripts, and tools marked with ⋄ are listed in Appendix E. A preview version of the code is submitted, and we will open the source code on GitHub.

4.1 MLM Instance, Configuration, Data Preprocessing and Pre-training

We use XLM (Lample and Conneau, 2019) and MASS (Song et al., 2019) as the MLM instances, where XLM is a token-based encoder model, and MASS is a span-based encoder-decoder model (see Appendix §A.1 for more details). The Transformer configuration is identical to XLM and MASS, where word embeddings, hidden states, and filter sizes are 1024, 1024, and 4096 respectively (**default**). To be fair, we reimplement all the baseline models with our configurations, using official XLM⋄, Tensor2Tensor⋄, and HuggingFace⋄ as references. We compare the results of our reimplementation with the reported results on the same test set to ensure the difference less than 2% in overall performance (Appendix C). For the context window size $2n+1$ of the co-occurrence counts and Eq.4, **we set** $n = 2$ **for all the experiments**, which is decided by our *dev experiment*.

Data preprocessing is identical to XLM and Mass. Specifically, we employ fastBPE⋄ to learn BPE (Sennrich et al., 2016b) with a sampling criterion from Lample and Conneau (2019) for all the experiments. To tokenize $\{Zh, Th, Ne\}$, we use Stanford Word Segmenter⋄, PyThaiNLP⋄, and Indic-NLP Library⋄, respectively. For the others, we use the Moses tokenizer⋄ with default rules.

Our code is implemented on Tensorflow 2.6 (Abadi et al., 2016). We use Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9$,$\beta_2 = 0.999$, $\epsilon = 1e-8$, $warm\_up = 10000$ (Vaswani et al., 2017) and $lr = 1e-4$. We set dropout regularization with a drop rate $rate = 0.1$. The mini-batch size is set to 8192 tokens.

4.2 Multilingual Task

Readers can refer to Appendix §A.2 or references for more introductions to these tasks.

Cross-lingual Embedding We attempt MUSE⋄ (Lample et al., 2018a) tasks that measure similarities between two paired words to generally evaluate the degree of the isomorphism of languages' embedding spaces. As discussed in Lample and Conneau (2019); Wang et al. (2020) and our preliminary experiment, the performance of the isomorphism is potentially proportional to the performance of cross-lingual transfer. We treat this experiment as our *dev experiment* to search for n.

UNMT UNMT (unsupervised neural machine translation) (Lample and Conneau, 2019; Lample et al., 2018b; Song et al., 2019; Liu et al., 2020) tackles bilingual translation (Bahdanau et al., 2015; Vaswani et al., 2017) on non-parallel bilingual corpora without any cross-lingual signal.

#	Model	MUSE cosine
1	MUSE	0.38
2	XLM (reported on 15 languages)	0.55
	12-layer Transformer encoder, 60K BPE and Wikipedia dumps in $\{De, En\}$.	
3	XLM (reimplemented on 2 languages)	0.55
4	XLM + OURS $n = 2$ (default)	0.60
5	XLM + OURS $n = 1$	0.58
6	XLM + OURS $n = 3$	0.60
7	XLM + OURS $n = 4$	0.59

Table 2: Results on MUSE tasks. This is our *dev experiment* for n.

Cross-lingual Classification We test XNLI⋄ (Conneau et al., 2020b) on 15 languages (including English) under the cross-lingual transfer setting. The model is pre-trained on multilingual corpora and fine-tuned on the English dataset, aiming at zero-shot classification for other languages.

Cross-lingual Question Answering MLQA⋄ (Lewis et al., 2020b) on 7 languages (including English) requires identifying the answer to a question as a span in the corresponding paragraph. We pre-train the model on multilingual corpora and fine-tune it on the English dataset, and then we attempt zero-shot prediction for other languages.

4.3 Secondary Monolingual Task

Recall that, as presented in Eq.2, $H_{[\mathcal{M}]_t}$ is the contextualized representation or the final hidden state. Therefore, MLM-GC pre-training is general and can work for other MLM instances such as BERT (Devlin et al., 2019), mBART (Liu et al., 2020), SpanBERT (Joshi et al., 2020), BART (Lewis et al., 2020a), and ALBERT (Lan et al., 2020). Meanwhile, MLM-GC pre-training is substantially better than MLM pre-training beyond multilingual tasks. We provide further experiments on monolingual tasks including SQuAD v1&v2 (Rajpurkar et al., 2016) in Appendix §B, using ALBERT as the MLM instance.

5 Result

5.1 Cross-lingual Embedding and Understanding Co-occurrence

Setup We configure an identical MLM instance to XLM with a 12-layer Transformer encoder. However, instead of 80K BPE and 15 languages in the original work, we learn 60K BPE and pre-train the model on Wikipedia dumps⋄ of the 2 languages. After 400K pre-training steps, we extract the embeddings required by the test set from the embedding space of the model. For words split into 2+ sub-tokens, we average all the sub-token embeddings. See details in Appendix A.2.1. As mentioned early, this is our *dev experiment*.

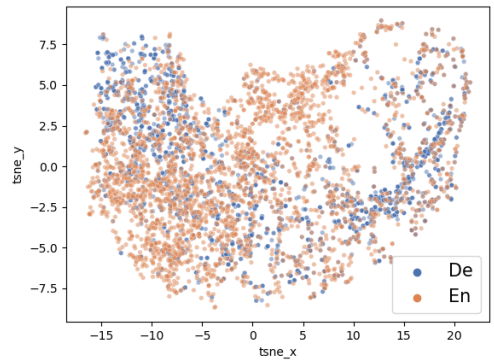
Performance We follow the instruction to compute the cosine similarity for the MUSE task, reporting the result in Table 2 for $En \leftrightarrow De$ test sets. MLM-GC pre-training outperforms the baseline model with different n. A large n does not consistently improve performance. We suspect that a large n may impact the capacity of the contextualized representation, which makes the model hard to be trained. Furthermore, $n = 2$ shows the best performance, and we may explain that in our comparison of co-occurrence counts (Figure 1), $n = 2$ has slightly smaller absolute difference avg and the ratio avg and narrower confidence (95%) intervals in large-count bins ($> log(1e7)$) contributing to over 45% co-occurrence counts on the multilingual corpora. Since we do not inject any cross-lingual supervision into the embedding space, this test can quantitatively report how MLM-GC refines the language spaces from co-occurrence counts for the isomorphic space and multilinguality.

Visualization and Multilingual Word Analogy We visualize all the words from the MUSE test sets. Since the task is originally designed for word translation including nouns, verbs, and other meaningful words, analogical words should be clustered and aligned in isomorphic spaces. As reported in Google's NMT (Johnson et al., 2017), the t-SNE can visualize isomorphic spaces across languages. Then, we employ the t-SNE visualization [3] to observe the isomorphic space. Figure 2 shows that MLM-GC pre-training help the model learn to form a better isomorphic space than MLM pre-training. Another insight is from the classic analogy test: "English: *King - Man + Woman = Queen* and German: *König-Mann+Frau = Königin*", and we show results in Table 3. MLM-GC pre-training consistently improves the performance on monolingual tests (only English or German) and multilingual tests (mixing English with German). Then, we can further observe the effectiveness of our method in improving the quality of isomorphic spaces across languages.

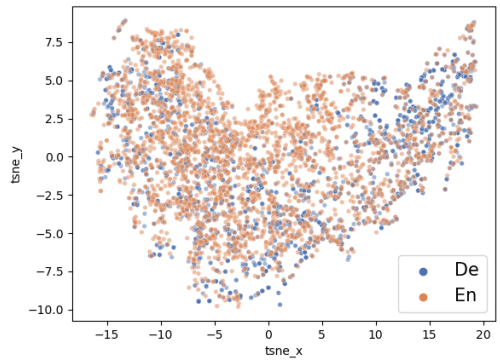
5.2 UNMT

Setup&Training We consider two similar language pairs $\{De, Ro\} \leftrightarrow En$ from WMT⋄ (Bojar et al., 2018) and a dissimilar language pair

[3] We reduce the dimension of embeddings to 3 by using PCA and then configure the t-SNE visualization.



(a) XLM



(b) XLM + OURS

Figure 2: T-SNE visualization for words of the MUSE task. Each point is a word instance.

$En \leftrightarrow Ne$ (Nepali) from FLoRes⋄ (Guzmán et al., 2019). Transformer, configurations, corpora, and BLEU scripts are identical to XLM and MASS. We pre-train the model around 400K iterations on only monolingual corpora of the two languages. After MLM-GC pre-training, we follow XLM and MASS to train the model for translation from pre-trained weights. In the training phase, we use Adam optimizer with parameters $\beta_1 = 0.9$, $\beta_2 = 0.997$ and $\epsilon = 10^{-9}$, and a dynamic learning rate with $warm\_up = 8000$ ($learning\_rate \in (0, 7e^{-4}]$) is employed. We set dropout with $rate = 0.1$ and label smoothing with $gamma = 0.1$. After around

X	cos (X , Queen)		cos(X , Königin)	
	XLM	XLM+OURS	XLM	XLM+OURS
mono: King-Man+Woman	0.44	0.46	0.35	0.39
mono: König-Mann+Frau	0.33	0.42	0.45	0.52
multi: King-Man+Frau	0.34	0.41	0.33	0.37
multi: King-Mann+Woman	0.45	0.48	0.33	0.38
multi: King-Mann+Frau	0.42	0.49	0.35	0.40
multi: König-Man+Woman	0.35	0.39	0.44	0.49
multi: König-Man+Frau	0.25	0.34	0.40	0.46
multi: König-Mann+Woman	0.38	0.42	0.43	0.49

Table 3: Word analogy: King - Man + Woman = Queen (German: König-Mann+Frau = Königin).

6-layer Transformer encoder-decoder, 60K BPE for each bilingual UNMT.						
Language pair	$De \leftrightarrow En$		$Ro \leftrightarrow En$		$Ne \leftrightarrow En$	
Test set	*newstest2016*				FLoRes⋄	
Corpora	News Crawl ⋄ from 2007 to 2017				FLoRes⋄	
	default *multi-BLEU.perl*⋄					
XLM	34.3	26.4	31.8	33.3	0.5	0.1
XLM + PMI-Masking ⋆	35.2	27.1	33.4	34.2	3.1	2.0
XLM + OURS	35.8	27.8	34.1	35.0	4.8	2.9
MASS	35.2	28.3	33.1	35.2		
MASS + OURS	36.5	28.7	34.6	36.4	5.5	3.0
default *sacreBleu*⋄:nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.0.0						
mBART + CC25 corpora	34.0	29.8	30.5	35.0	10.0	4.4
XLM + OURS	35.6	27.7	33.8	34.8	4.7	2.8
MASS +OURS	36.3	28.5	34.4	36.2	5.3	2.9

Table 4: Results of UNMT. ⋆ is reimplemented.

400K iterations, we report results. See details in Appendix A.2.2.

Performance In Table 4, we report *multi-BLEU.perl*⋄ to compare with XLM and MASS and *sacreBleu*⋄ to compare with mBART (Liu et al., 2020) so that the evaluation is based on the same BLEU script. MLG-GC pre-training consistently improves the performance of baseline models on all the similar language pairs by $3\% \sim 7\%$ and on the dissimilar pair by $2.5 \sim 5.0$ BLEU. The performance on the dissimilar pair is competitive to SOTA: mBART and is better than mBART on similar language pairs. However, mBART uses CC25 (Wenzek et al., 2020) for pre-training and obtains benefits from more languages (25 languages) and samples. The global co-occurrence information across languages is general and abstract for isomorphic spaces, which allows for cross-lingual representations. It eventually helps the model understand translation knowledge. Meanwhile, we observe substantial gains on MASS + OURS (and ALBERT (Lan et al., 2020) in Appendix B), where MASS (ALBERT) is based on span masking. As discussed in §Related Work and Introduction, span-based masking (also including Whole Word Masking (Devlin et al., 2019) and PMI-Masking (Levine et al., 2021)) implicitly leverages co-occurrence information for improving context understanding. In addition to the empirical study in Figure 1, the gain further confirms that global co-occurrence information significantly injects some signals for cross-lingual transfer beyond improving context understanding.

5.3 Cross-lingual Classification

Setup&Fine-tuning The model configuration, preprocessing, and corpora are identical to XLM[4]. For the classification objective, we deploy a linear classification layer on top of the encoder. Af-

[4]In the literature, this setup also refers to XLM-15.

Model	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	Avg
baseline	73.7	67.7	68.7	67.7	68.9	67.9	65.4	64.2	64.8	66.4	64.1	65.8	64.1	55.7	58.4	65.6
mBERT	82.1	73.8	74.3	71.1	66.4	68.9	69.0	61.6	64.9	69.5	55.8	69.3	60.0	50.4	58.0	66.3
12-layer Transformer encoder, 80K BPE, and 15 XNLI languages from Wikipedia dumps downloaded by WikiExtractor◇.																
XLM	83.2	76.5	76.3	74.2	73.1	74.0	73.1	67.8	68.5	71.2	69.2	71.9	65.7	64.6	63.4	71.5
XLM + PMI-Masking ⋆	84.1	78.4	77.8	76.6	75.1	75.5	74.9	69.7	70.8	73.0	70.7	73.4	68.1	66.1	65.3	73.3
XLM + OURS	84.9	78.6	78.7	77.5	76.2	77.1	74.8	71.5	72.6	75.7	72.6	76.2	68.2	67.5	66.5	74.6
+ Parallel Sentences from OPUS◇																
XLM + TLM	85.0	78.7	78.9	77.8	76.6	77.4	75.3	72.5	73.1	76.1	73.2	76.5	69.6	68.4	67.3	75.1
XLM + TLM + OURS	85.0	79.5	79.4	78.5	77.3	78.0	76.2	73.1	74.0	76.8	74.0	77.1	70.5	70.0	68.5	75.9

Table 5: Results of cross-lingual classification on XNLI. ⋆ is reimplemented.

Model	en	es	de	ar	hi	vi	zh	Avg
mBERT-102	77.7 / 65.2	64.3 / 46.6	57.9 / 44.3	45.7 / 29.8	43.8 / 29.7	57.1 / 38.6	57.5 / 37.3	57.7 / 41.6
12-layer Transformer encoder, 80K BPE, and and 15 XNLI languages from Wikipedia dumps downloaded by WikiExtractor◇.								
XLM	74.9 / 62.4	68.0 / 49.8	62.2 / 47.6	54.8 / 36.3	48.8 / 27.3	61.4 / 41.8	61.1 / 39.6	61.6 / 43.5
XLM + PMI-Masking ⋆	76.0 / 63.9	69.2 / 50.2	64.1 / 48.0	55.8 / 38.0	49.8 / 28.5	62.9 / 42.2	63.3 / 40.5	63.1 / 44.4
XLM + OURS	77.7 / 65.9	71.5 / 51.1	65. 7 / 48.9	57.4 / 40.0	51.5 / 30.3	64.5 / 43.2	64.7 / 41.9	64.7 / 45.9

Table 6: Results of cross-lingual question answering on MLQA. We report the F1 and EM (exact match) scores for zero-shot prediction. ⋆ is reimplemented.

ter pre-training, we deploy the randomly initialized linear classifier and fine-tune the encoder and the linear classifier on the En NLI dataset with mini-batch size 16. We use Adam optimizer with $lr = 5 \times 10^{-4}$ and linear decay of lr. After fine-tuning, we make zero-shot prediction for the other 14 languages. See details in Appendix A.2.3.

Performance We report the result in Table 5. Our method consistently improves baseline models by 3.5% (Avg). As discussed in previous models (Conneau et al., 2020b; Karthikeyan et al., 2020; Wu and Dredze, 2019; Pires et al., 2019; Dufter and Schütze, 2020), multilinguality is essential for this task. Then, we confirm the effectiveness of MLM-GC pre-training. Furthermore, our method outperforms XLM + PMI-Masking (span-based). Similar to the comparison in UNMT, MLM-GC pre-training uses co-occurrence information for better context understanding and cross-lingual transfer, whereas XLM + PMI-Masking leverages co-occurrence information for context understanding but performs worse for cross-lingual transfer because of the lack of a mechanism to help cross-lingual transfer. We also include XLM + TLM (Lample and Conneau, 2019) for comparison. In this experiment, XLM + TLM using parallel sentences in pre-training slightly outperforms MLM-GC, which indicates the knowledge gap between co-occurrence information and parallel sentences for cross-lingual supervision. Besides, when applying MLM-GC pre-training for XLM + TLM, we still observe gains. We attribute the additional gains to the contextualized representations that are further refined by co-occurrence information to represent similar abstractions for cross-lingual transfer. Intuitively, the co-occurrence information gives extra cross-lingual supervision beyond a limited amount of parallel sentences.

5.4 Cross-lingual Question Answering

Setup&Fine-tuning The setup is similar to §Cross-lingual Classification. We follow the instruction of SQuAD from BERT, fine-tuning the model with a span extraction loss on the English dataset. We use Adam optimizer with $lr = 5 \times 10^{-5}$ and linear decay of lr. As suggested, we fine-tune the model on SQuAD v1.1 (Rajpurkar et al., 2016) and then make zero-shot prediction for the 7 languages of MLQA. See details in Appendix A.2.4.

Performance In Table 6, MLM-GC pre-training substantially improves the performance (Avg) in both F1 and EM metrics by 4.8 % and 5.0 % respectively. Meanwhile, MLM-GC pre-training yields more improvements for low-resource languages. We attribute all the improvements to the global co-occurrence objective the model learns in MLM-GC pre-training. Intuitively, spans (groups of words) of answers across languages are most likely to consist of nouns and terms and can be easily represented, clustered, and aligned in the improved isomorphic space because they are analogical and might have similar co-occurrence counts as discussed in the empirical study (Figure 1).

6 Conclusion

In this work, we leverage the global co-occurrence information from multilingual corpora. The result is MLM-GC pre-training with a combined objective of MLM and global co-occurrence modeling.

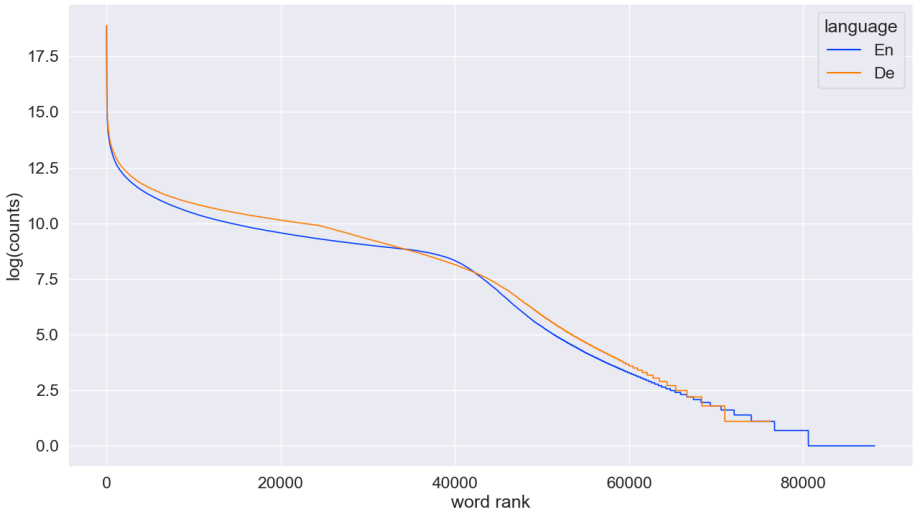


Figure 3: Word distributions of De and En on Wikipedia after applying BPE.

Our experiments show that MLM-GC pre-training can substantially improve the performance of naive MLM pre-training for 4 multilingual tasks, and additional experiments show that it can work for monolingual tasks. The isomorphic space across languages benefits from co-occurrence information, which allows for cross-lingual transfer. Meanwhile, the model is encouraged to distinguish relevant information from irrelevant information and to discriminate between the two relevant information across languages from co-occurrence counts (normalized) and refine contextualized representations accordingly. We believe that leveraging co-occurrence information for cross-lingual transfer is an interesting avenue in multilingual pre-training.

7 Limitation

Theoretically, our method might benefit from comparable corpora across languages, where words and compound words might have similar distribution because Zipf's law might be satisfied only for similar domains. For instance, as presented in Figure 3, word distributions of De and En on Wikipedia are similar after applying BPE. In our experiments, we only confirm the effectiveness of our methods on Wikipedia corpora in different languages, which are comparable across languages. This might limit the scope of our method. However, multilingual models are commonly pre-trained on comparable corpora, e.g., Wikipedia and CC.

Another limitation is about the combined objective in Eq. 4. In our experiments, we try to eliminate the MLM objective, only considering global regression modeling $\mathcal{L}_{GC}$. The result is not promising, and it seems that $\mathcal{L}_{GC}$ can not work well without the help of the MLM objective. However, our experiment is very simple. This might be further confirmed or designed in future work.

References

Martin Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283.

Xi Ai and Bin Fang. 2021. Empirical regularization for synthetic sentence pairs in unsupervised neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12471–12479.

Xi Ai and Bin Fang. 2022. Leveraging relaxed equilibrium by lazy transition for sequence modeling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2904–2924, Dublin, Ireland. Association for Computational Linguistics.

Hanan Aldarmaki and Mona Diab. 2019. Context-aware cross-lingual mapping. 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 3906–3911, Minneapolis, Minnesota. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Ond rej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Ond rej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, and Christof Monz. 2018. Findings of the 2018 conference on machine translation (wmt18). In *Proceedings of the Third Conference on Machine Translation*, pages 272–307, Belgium, Brussels. Association for Computational Linguistics.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual Alignment of Contextual Word Representations. In *8th International Conference on Learning Representations, ICLR 2020 - Conference Track Proceedings*.

Pi Chuan Chang, Michel Galley, and Christopher D Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *3rd Workshop on Statistical Machine Translation, WMT 2008 at the Annual Meeting of the Association for Computational Linguistics, ACL 2008*, pages 224–232.

Aditi Chaudhary, Karthik Raman, Krishna Srinivasan, and Jiecao Chen. 2020. Dict-mlm: Improved multilingual pre-training using bilingual dictionaries. *arXiv preprint arXiv:2010.12566*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Holger Schwenk, Veselin Stoyanov, Adina Williams, and Samuel R. Bowman. 2020b. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485. Association for Computational Linguistics.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020c. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

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*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Philipp Dufter and Hinrich Schütze. 2020. Identifying elements essential for BERT's multilinguality. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 4423–4437, Online. Association for Computational Linguistics.

Francisco Guzmán, Peng Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The Flores evaluation datasets for low-resource machine translation: Nepali-English and Sinhala-English. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 6098–6111.

Le Quan Ha, E. I. Sicilia-Garcia, Ji Ming, and F. J. Smith. 2002. Extension of Zipf's law to words and phrases. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

K Karthikeyan, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-lingual ability of multilingual bert: An empirical study. In *8th International Conference on Learning Representations, ICLR 2020 - Conference Track Proceedings*.

Diederik P Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. In *Advances in neural information processing systems*.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018a. Word translation without parallel data. In *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *8th International Conference on Learning Representations, ICLR 2020 - Conference Track Proceedings*.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. 2021. PMI-Masking: Principled masking of correlated spans. In *9th International Conference on Learning Representations, ICLR 20201- Conference Track Proceedings*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020b. MLQA: Evaluating Cross-lingual Extractive Question Answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Mauro Mezzini. 2018. Empirical study on label smoothing in neural networks. In *WSCG 2018 - Short papers proceedings*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? pages 4996–5001.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 784–789.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Shuo Ren, Yu Wu, Shujie Liu, Ming Zhou, and Shuai Ma. 2019. Explicit cross-lingual pre-training for unsupervised machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 770–779, Hong Kong, China. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2104.06644*.

Anders Søgaard. 2020. Some languages seem easier to parse because their treebanks leak. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 2765–2770.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936. PMLR.

Ashish Vaswani, Google Brain, 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.

Pascal Vincent. 2010. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of Machine Learning Research*, 11:3371–3408.

Zirui Wang, Jiateng Xie, Ruochen Xu, Yiming Yang, Graham Neubig, and Jaime Carbonell. 2020. Cross-lingual Alignment vs Joint Training: A Comparative Study and A Simple Unified Framework. In *8th International Conference on Learning Representations, ICLR 2020 - Conference Track Proceedings*.

Guillaume Wenzek, Marie Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012.

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*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho Jui Hsieh. 2020. Large batch optimization for deep learning: Training bert in 76 minutes. In *8th International Conference on Learning Representations, ICLR 2020 - Conference Track Proceedings*.

Yukun Zhu, Ryan Kiros, Richard 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 19–27.

George Kingsley Zipf. 1949. Human behavior and the principle of least effort: an introd. to human ecology.

George Kingsley Zipf. 2013. *The psycho-biology of language: An introduction to dynamic philology*. Routledge.

A Experiment

A.1 MLM Instance

We adapt our method to two MLM instances: XLM (Lample and Conneau, 2019) and MASS (Song et al., 2019). We follow the instructions of BERT (Devlin et al., 2019) that each selected token is replaced with the probabilities $(p[unchanged], p[random], p[mask]) = (0.1, 0.1, 0.8)$.

XLM XLM is similar to BERT (Devlin et al., 2019) but uses text streams of an arbitrary number of sentences. Following the instruction, we randomly select 15% of the tokens from the input sentence for replacing.

MASS MASS is different from XLM and BERT but similar to SpanBERT (Joshi et al., 2020), using spans to replace consecutive tokens. Given an input sentence with length N, we randomly select consecutive tokens with length $N/2$ for replacing.

A.2 Multilingual Task

A.2.1 Cross-lingual Embedding

We are interested in the isomorphism of languages' embedding spaces. To investigate, we attempt MUSE⋄ tasks (Lample et al., 2018a) that measure similarities between two paired words. This test can generally evaluate the degree of the isomorphism of languages' embedding spaces. Meanwhile, as discussed in Lample and Conneau (2019); Wang et al. (2020) and our preliminary experiment, the performance of the isomorphism is potentially proportional to the performance of cross-lingual transfer learning tasks. Therefore, we treat this experiment as our *dev experiment* to search for n.

Setup We configure a 12-layer Transformer encoder and use Moses tokenizer⋄ with default rules for tokenization, identical to XLM (Lample and Conneau, 2019). For fast *dev experiment*, we employ fastBPE⋄ to learn 60K BPE (Sennrich et al., 2016b) from concatenated corpora with a sampling criterion from (Lample and Conneau, 2019) and pre-train the model on 2 languages instead of 80K BPE and 15 languages in the reported work.

Training In the pre-training phase, we pre-train the model on Wikipedia dumps⋄ of the two languages for 400K steps. After pre-training, we extract the words' embeddings required by the test set from the embedding space of the model. For

words split into 2+ sub-tokens, we average all the extracted embeddings of sub-tokens. We then evaluate paired embeddings in cosine similarity.

A.2.2 UNMT

UNMT (unsupervised neural machine translation) (Lample and Conneau, 2019; Lample et al., 2018b; Song et al., 2019; Liu et al., 2020) tackles bilingual translation (Bahdanau et al., 2015; Vaswani et al., 2017) on non-parallel bilingual corpora without having access to any parallel sentence. In other words, there is no supervision for translation. The model requires pre-training to obtain some initial multilingual knowledge for decent performance.

Setup We configure an identical Transformer model to XLM (Lample and Conneau, 2019) and MASS (Song et al., 2019), which has 6 layers in both the encoder and decoder using default configurations. We consider multiple families of languages. Specifically, we consider similar language pairs $\{De, Ro\} \leftrightarrow En$, using the same dataset as previous works (Lample and Conneau, 2019). The dataset consists of monolingual corpora $\{De, En\}$ from WMT 2018◇ (Bojar et al., 2018) including all available $NewsCrawl$ datasets from 2007 through 2017 and monolingual corpora Ro from WMT 2016◇ (Bojar et al., 2016) including $NewsCrawl$ 2016. We report the performance for $\{De, Ro\} \leftrightarrow En$ on *newstest2016*. Meanwhile, we share the FLoRes◇ (Guzmán et al., 2019) task to evaluate a dissimilar language pair $Ne \leftrightarrow English$ (Nepali). For tokenization, we use the Moses tokenizer◇ developed by Koehn et al. (2007) with default rules except for Ne that is tokenized by Indic-NLP Library◇. We employ fastBPE◇ to learn 60K BPE (Sennrich et al., 2016b) from concatenated corpora of paired languages, using the same sampling criteria in Lample and Conneau (2019). We use learnable language embeddings and position embeddings.

Training In MLM-GC pre-training, the model is pre-trained around 400K iterations on only monolingual corpora of different languages. In the training phase, we use Adam optimizer (Kingma and Ba, 2015) with parameters $\beta_1 = 0.9$,$\beta_2 = 0.997$ and $\epsilon = e - 9$, and a dynamic learning rate with $warm\_up = 8000$ and $learning\_rate \in (0, 7e-4]$) (Vaswani et al., 2017) is employed. We set dropout regularization with a drop rate $rate = 0.1$ and label smoothing with $gamma = 0.1$ (Mezzini, 2018). On-the-fly back-translation (Sennrich et al., 2016a) (the inference mode of the model) performs to generate synthetic parallel sentences that can be used for training of translation as NMT (neural machine translation) is trained on genuine parallel sentences in a supervised manner. Meanwhile, UNMT learns an objective of denoising language modeling (Vincent, 2010) to maintain language knowledge in the training phase except for MASS. After around 400K iterations, we report BLEU computed by *multi-BLEU.perl*◇ and *scareBLEU*◇ with default rules, according to baseline models. In conclusion, in pre-training, we only have the objective of MLM-GC, and in training, we have the two objectives: *1)* denoising language modeling for XLM or MASS itself and *2)* translation (i.e., NMT), where the translation objective is finished by using synthetic pairs sentences from on-the-fly back-translation.

A.2.3 Cross-lingual Classification

We experiment with XNLI◇ (Conneau et al., 2020b), a general cross-lingual classification task on 15 languages (including English) under the cross-lingual transfer setting. The model takes in two input sentences and is required to classify into one of the three labels: entailment, contradiction, and neutral. The model is fine-tuned on the English dataset and then attempts zero-shot classification for other languages.

Setup Following the previous work[5] (Lample and Conneau, 2019), we use raw sentences including 15 XNLI languages from Wikipedia dumps downloaded by WikiExtractor◇. Sentences in different languages are sampled with the method of Lample and Conneau (2019). The model configuration and preprocessing are identical to XLM that we use a 12-layer transformer encoder and 80K BPE. For the classification objective, we deploy a linear classification layer on top of the encoder. To tokenize $\{zh, th\}$, we use Stanford Word Segmenter◇ and PyThaiNLP◇ respectively. For the others, we use the Moses tokenizer◇ with default rules. Similar to the Cross-lingual Embedding experiment, we use fastBPE◇ and the sampling strategy to learn BPE.

Fine-tuning After pre-training on the corpora, we deploy a randomly initialized linear classifier and fine-tune the encoder and the linear classifier on the En NLI dataset with mini-batch size 16. We

[5] In the literature, this setup also refers to XLM-15.

use Adam optimizer (Kingma and Ba, 2015) with $lr = 5e-4$ and linear decay of lr. After fine-tuning, we make zero-shot prediction for the other 14 languages. We use categorical cross-entropy with three labels: entailment, contradiction, and neutral.

A.2.4 Cross-lingual Question Answering

We consider the MLQA⋄ (Lewis et al., 2020b) dataset for a cross-lingual question answering task. Given a question and a passage containing the answers, the aim is to predict the answer text span in the passage. This task requires identifying the answer to a question as a span in the corresponding paragraph. The evaluation data for English and 6 other languages are obtained by automatically mining target language sentences that are parallel to sentences in English from Wikipedia, crowdsourcing annotations in English, and translating the question and aligning the answer spans in the target languages. Similar to the cross-lingual classification task, the model is fine-tuned on the English dataset and then attempts zero-shot prediction for other languages.

Setup The setup is similar to the experiment of cross-lingual classification.

Fine-tuning We follow the instruction of SQuAD from BERT (Devlin et al., 2019), fine-tuning the model with a span extraction loss on the English dataset. We use Adam optimizer (Kingma and Ba, 2015) with $lr = 5e-5$ and linear decay of lr. Meanwhile, as suggested, we fine-tune the model on the SQuAD v1.1 (Rajpurkar et al., 2016) dataset and then make zero-shot prediction for the 7 languages of MLQA. Given a sequence T, we only have a start vector $S \in R^{hidden}$ and an end vector $E \in R^{hidden}$ during fine-tuning. The probability of word i being the start of the answer span is computed as a dot product T_i and S d by a $softmax$ over all of the words in the sequence $p_i = \frac{e^{ST_i}}{\sum_{k \in T} e^{ET_k}}$. Similarly, we can compute the end of the span. The score of a candidate span from position i to position j is defined as $ST_i + ET_j$ and the maximum scoring span where $j \geq i$ is used as a prediction.

B Additional and Supportive Result

B.1 Pre-training for Monolingual Task

Although we derive our method from the observation of multilingual models, MLM-GC pre-training is substantially better than MLM pre-training. We provide further experiments on monolingual tasks including SQuAD v1&v2 (Rajpurkar et al., 2016).

Model	SQuAD1.1 (F1)	SQuAD2.0 (F1)
12-base-ALBERT (Lan et al., 2020)	89.3	80.0
⋆12-base-ALBERT	89.4	80.0
12-base-ALBERT + OURS	89.8	80.9

Table 7: MLM-GC pre-training for ALBERT. ⋆ *denotes the baseline model that is reimplemented.*

setup For this monolingual task, our configuration is identical to 12-base-ALBERT (Lan et al., 2020). Specifically, We set the model dimension, word embedding dimension, and the maximum number of layers to 768, 128, and 12. As recommended, we generate a masked span for the MLM targets using the random strategy from Joshi et al. (2020), and we use LAMB optimizer⋄ with a learning rate of 0.00176 (You et al., 2020) instead of Adam optimizer. Following the instructions, we pre-train models on BooksCorpus⋄ (Zhu et al., 2015) and English Wikipedia⋄ (Devlin et al., 2019) for 140k steps.

Fine-tuning Similar to the cross-lingual question answering task, we fine-tune the pre-trained model on SQuAD(v1.1 and v2.0)⋄ (Rajpurkar et al., 2016, 2018).

Result Table 7 shows that MLM-GC pre-training is substantially better than MLM pre-training when pre-training 12-base-ALBERT for monolingual tasks. These observations confirm the effectiveness of MLM-GC pre-training on monolingual tasks.

B.2 Impact of Tokenization Method

We are interested in how the tokenization method affects the performance because it potentially affects the token-token co-occurrence counts. For evaluation, we use all the configurations in UNMT and additionally configure a word-level vocabulary for the model. The word-level vocabulary has the same number of tokens as the BPE vocabulary. Table 8 shows that our method can work with different tokenization methods. Our method can generally improve the performance, regardless of the difference between the two baseline models in the same configuration.

C Reimplementation

We compare our reimplementation with reported results in Table 9.

Model	$De \leftrightarrow En$	
baseline (BPE-based) ⋆	33.8	26.3
+ OURS	35.8	27.8
baseline (Word-level) ⋆	33.0	25.8
+ OURS	35.2	27.2

Table 8: Impact of Tokenization Method. ⋆ denotes reimplemented models.

Language pair	$De \leftrightarrow En$	
multi-BLEU.perl◇ with default rules		
XLM(Lample et al., 2018b) *reported*	34.3	26.4
XLM(Lample et al., 2018b) ⋆	33.9	26.3
XLM + OURS	35.8	27.8
multi-BLEU.perl◇ with default rules		
MASS(Song et al., 2019) *reported*	35.2	28.3
MASS(Song et al., 2019)⋆	35.0	28.0
MASS + OURS	36.5	28.7

Table 9: Performance of UNMT. Baseline models (⋆) are reimplemented with our configurations.

D Alternative

In MLM pre-training, when w_t at the position t is replaced by the artificial masking token $[\mathcal{M}]_t$, the output distribution for w_t is obtained by applying a pre-softmax linear transformation $O \in R^{d \times V}$ from *the final hidden state or the contextualized representation* $H_{[\mathcal{M}]_t}$ to the output vocabulary size V, followed by a $softmax$ operation which generates an output matrix normalized over its rows. Specifically, $Q_{[\mathcal{M}]_t w_t} = \frac{exp(H^T_{[\mathcal{M}]_t} O_{w_t})}{\sum_{k=1}^{V} exp(H^T_{[\mathcal{M}]_t} O_{w_k})}$ is the model for the probability of w_t in the context of $H_{[\mathcal{M}]_t}$, where O_{w_t} and O_{w_k} are vectors factorized from O, i.e., self-recognizing. In this way, the probability of w_n in the context $H_{[\mathcal{M}]_t}$ is similar to $Q_{[\mathcal{M}]_t w_t}$ in the global regression model. Specifically, for w_n, $Q_{[\mathcal{M}]_t w_t}$ could be extended to:

$$Q_{[\mathcal{M}]_t w_n} = \frac{exp(H^T_{[\mathcal{M}]_t} O_{w_n})}{\sum_{k=1}^{V} exp(H^T_{[\mathcal{M}]_t} O_{w_k})}. \quad (5)$$

For all the neighboring tokens $w_{t\pm n}$ of the input sentence at position $[t-n, ..., t) \cup (t, ..., t+n]$, i.e., excluding position t, we have the model $Q_{[\mathcal{M}]_t w_{t\pm n}}$. Then, we employ the new global log-bilinear regression model in MLM pre-training. Formally, given the factorized $O_{w_{t\pm n}}$ and $X_{w_t w_{t\pm n}}$ from O and X respectively, we have the model:

$$\mathcal{L}_{GC} = \frac{1}{2n} \sum_{n} f(X_{w_t w_{t\pm n}}) \\ (\frac{H^T_{[\mathcal{M}]_t} O_{w_{t\pm n}}}{\sqrt{d}} - \log X_{w_t w_{t\pm n}})^2, \quad (6)$$

where d is the model dimension.

In Table 10 and Table 11, we show the experimental results (also see our previous revision https://openreview.net/forum?id=DswOSXvLfuy). In conclusion, the presented method Eq. 3 slightly outperforms the alternative Eq. 6 on sentence-level tasks. We explain that the alternative involves neighboring embeddings in the objective, which directly improves the quality of cross-lingual embeddings. Compared to that, the presented method of the main paper considers contextualized representations of the masked tokens and their neighboring tokens, which is better for cross-lingual transfer.

E Source

We list all the links of dataset, tools, and other sources in Table 12. Note that for multilingual tasks, datasets can be downloaded from the XTREME link except for UNMT and cross-embeddings.

Model	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	Avg
baseline	73.7	67.7	68.7	67.7	68.9	67.9	65.4	64.2	64.8	66.4	64.1	65.8	64.1	55.7	58.4	65.6
mBERT	82.1	73.8	74.3	71.1	66.4	68.9	69.0	61.6	64.9	69.5	55.8	69.3	60.0	50.4	58.0	66.3
12-layer Transformer encoder, 80K BPE, and 15 XNLI languages from Wikipedia dumps downloaded by WikiExtractor◇.																
XLM	83.2	76.5	76.3	74.2	73.1	74.0	73.1	67.8	68.5	71.2	69.2	71.9	65.7	64.6	63.4	71.5
XLM + PMI-Masking ⋆	84.1	78.4	77.8	76.6	75.1	75.5	74.9	69.7	70.8	73.0	70.7	73.4	68.1	66.1	65.3	73.3
XLM + OURS$_{v2}$	84.9	78.6	78.7	77.1	76.2	77.0	75.2	72.5	72.6	75.1	73.0	74.2	68.2	67.2	67.1	74.5
+ Parallel Sentences from OPUS◇																
XLM + TLM	85.0	78.7	78.9	77.8	76.6	77.4	75.3	72.5	73.1	76.1	73.2	76.5	69.6	68.4	67.3	75.1
XLM + TLM + OURS$_{v2}$	85.0	79.9	79.2	78.5	77.1	78.0	76.4	73.1	74.0	76.7	73.9	76.8	70.2	68.8	67.9	75.5

Table 10: Results of cross-lingual classification on XNLI. ⋆ is reimplemented.

Table 11: Results of cross-lingual question answering on MLQA. We report the F1 and EM (exact match) scores for zero-shot prediction. ⋆ is reimplemented.

Model	en	es	de	ar	hi	vi	zh	Avg
mBERT-102	77.7 / 65.2	64.3 / 46.6	57.9 / 44.3	45.7 / 29.8	43.8 / 29.7	57.1 / 38.6	57.5 / 37.3	57.7 / 41.6
12-layer Transformer encoder, 80K BPE, and and 15 XNLI languages from Wikipedia dumps downloaded by WikiExtractor◇.								
XLM	74.9 / 62.4	68.0 / 49.8	62.2 / 47.6	54.8 / 36.3	48.8 / 27.3	61.4 / 41.8	61.1 / 39.6	61.6 / 43.5
XLM + PMI-Masking ⋆	76.0 / 63.9	69.2 / 50.2	64.1 / 48.0	55.8 / 38.0	49.8 / 28.5	62.9 / 42.2	63.3 / 40.5	63.1 / 44.4
XLM + OURS$_{v2}$	77.5 / 65.6	71.4 / 50.9	65.3 / 48.6	57.1 / 39.6	51.1 / 29.9	64.1 / 43.0	64.5 / 41.7	64.4 / 45.7

Table 12: Links of source.

Item	Links
WMT 2016	http://www.statmt.org/wmt16/translation-task.html
WMT 2018	http://www.statmt.org/wmt18/translation-task.html
FLoRes	https://github.com/facebookresearch/flores
Indic-NLP Library	https://github.com/anoopkunchukuttan/indic_nlp_library
XLM	https://github.com/facebookresearch/XLM
multi-BLEU.perl	https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-BLEU.perl
Moses tokenizer	https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl
Kytea	http://www.phontron.com/kytea/
XTREME	https://github.com/google-research/xtreme
fastBPE	https://github.com/glample/fastBPE
MUSE	https://github.com/facebookresearch/MUSE
Cambridge Dictionary	https://dictionary.cambridge.org/
SemEval'17	https://alt.qcri.org/semeval2017/task2/
WikiExtractor	https://github.com/attardi/wikiextractor
PyThaiNLP	https://github.com/PyThaiNLP/pythainlp
Stanford Word Segmenter Chang et al. (2008)	https://nlp.stanford.edu/software/segmenter.html
Tensor2Tensor	https://github.com/tensorflow
HuggingFace	https://huggingface.co
ORPUS, Wikipedia v1.0	https://opus.nlpl.eu

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1 line 75

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

4 and 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 3.3 line 320 and section 4.1

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 3.3 and Section 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 4

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.