Synthetic Pre-Training Tasks for Neural Machine Translation

Zexue He[1]*, **Graeme Blackwood**[2]*, **Rameswar Panda**[2],
Julian McAuley[1], **Rogerio Feris**[2]
[1]University of California, San Diego
[2]MIT-IBM Watson AI Lab, IBM Research
[1]{zehe,jmcauley}@ucsd.edu
[2]{blackwood,rpanda,rsferis}@us.ibm.com

Abstract

Pre-training models with large crawled corpora can lead to issues such as toxicity and bias, as well as copyright and privacy concerns. A promising way of alleviating such concerns is to conduct pre-training with synthetic tasks and data, since no real-world information is ingested by the model. Our goal in this paper is to understand the factors that contribute to the effectiveness of pre-training models when using synthetic resources, particularly in the context of neural machine translation. We propose several novel approaches to pre-training translation models that involve different levels of lexical and structural knowledge, including: 1) generating obfuscated data from a large parallel corpus 2) concatenating phrase pairs extracted from a small word-aligned corpus, and 3) generating synthetic parallel data without real human language corpora. Our experiments on multiple language pairs reveal that pre-training benefits can be realized even with high levels of obfuscation or purely synthetic parallel data. We hope the findings from our comprehensive empirical analysis will shed light on understanding what matters for NMT pre-training, as well as pave the way for the development of more efficient and less toxic models.

1 Introduction and Motivation

Neural Machine Translation (NMT) models depend on large quantities of aligned training data (Aharoni et al., 2019; Fan et al., 2021; NLLB Team et al., 2022). For many language pairs of interest, however, high quality parallel data is either unavailable or exists only in limited quantities. Training robust NMT systems with such limited data can be a significant challenge.

Even for high-resource language pairs, parallel data can be noisy and frequently contains toxic speech or biased language. Such problems are particularly acute for comparable corpora crawled automatically from the web (Kreutzer et al., 2022) since it can cause catastrophic mistranslations (Costa-jussà et al., 2022) or hallucinated toxicity. It is preferable to avoid exposing the model to such data in order to prevent accidental generation of offensive content or egregiously embarrassing translations. Crawled data can also present problematic copyright, attribution, and privacy issues. As an example, the JW300 corpus of Jehovah's Witnesses publications (Agić and Vulić, 2019) was recently withdrawn due to a copyright infringement claim.

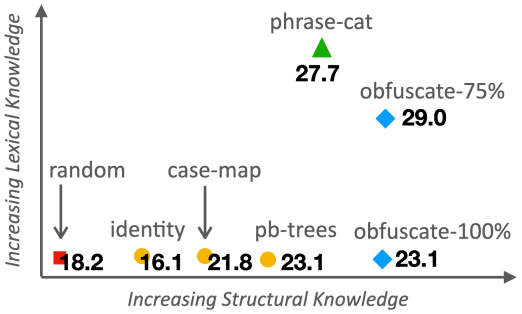


Figure 1: A comparison of the extent to which the synthetic data generation methods described in Section 3 encode lexical and/or structural translation knowledge. The vertical axis compares methods with respect to lexical knowledge. The horizontal axis compares structural knowledge. BLEU scores correspond to the Indonesian-to-English translation task described in Section 4.

Our primary motivation is to investigate how knowledge transfer from NMT pre-training can help to avoid or minimize the data issues described above. We study the impact of pre-training and transfer learning on translation tasks by comparing various procedural approaches to synthetic data generation. Each approach has varying degrees of inherited or artificially constructed lexical and structural translation knowledge. The degree to which each method encodes lexical and/or structural translation knowledge is plotted in abstract form in Figure 1. We describe each of our synthetic data generation methods in Section 3.

*Equal contribution

Findings of the Association for Computational Linguistics: ACL 2023, pages 8080–8098

Our first approach (§3.1) studies the extent to which the transfer benefits of regular pre-training can be realized when using obfuscated or encrypted data. Our obfuscated corpus is derived from real parallel data by mapping the original words to a vocabulary of 'nonsense' tokens. Experiments on six different language pairs show that obfuscated pre-training is able to capture much of the transferable knowledge: pre-training with an obfuscation ratio as high as 75% is still able to achieve BLEU scores close to those obtained by a model pre-trained on the original un-obfuscated parallel data.

Our second approach (§3.2) seeks to maximize the benefit that can be derived from a specific limited quantity of fine-tuning data. We do this by pre-training on newly constructed artificial sentence pairs synthesized directly from the fine-tuning corpus. The synthetic sentence pairs are created by concatenating randomly sampled aligned phrase pairs extracted from the fine-tuning corpus. Although the sentence-level fluency and grammaticality of sentences constructed using this technique are both quite poor, they do retain word- and phrase-level correspondences and local reordering information that can greatly improve translation quality and robustness compared to models trained using only the original fine-tuning data.

Our third approach (§3.3) explores the pre-training impact of important translation phenomena such as alignments and reordering. We pre-train models on procedurally generated synthetic parallel data that does not derive from any real human language corpus. We design three simple synthetic sequence-to-sequence translation tasks and associated data sets. Since our data is procedurally generated, problems of toxicity, attribution and copyright can be avoided. We evaluate the effectiveness of pre-training and transfer for our synthetic tasks in the context of low-resource NMT. Our results show that – to a surprising degree – the transfer benefits of pre-training can be realized even with purely synthetic tasks and data. Our analysis shows that structure, in the form of aligned sub-trees, matters in synthetic pre-training for NMT.

We empirically evaluate the impact of each of our proposed synthetic pre-training methods in low-resource MT settings (§4), followed by a discussion and analysis explaining our insights into what makes for a good pre-trained model (§5). We also consider the question of model toxicity. We measure the extent of hallucinated toxicity in each synthetic data generation method, showing that synthetic methods can result in substantially reduced toxicity compared to models pre-trained on real parallel corpora.

The primary **contributions** of our paper are as follows: (i) we propose several novel synthetic pre-training tasks, that encode varying degrees of structural and lexical knowledge, in order to gain insights into what makes for a good pre-trained NMT model; (ii) we conduct a comprehensive empirical evaluation of knowledge transfer in NMT from synthetic data pre-training, considering metrics of both translation quality and toxicity; and (iii) we demonstrate that synthetic data is a promising stepping stone towards relieving the data burden in low-resource translation and building more accurate and trustworthy NMT systems.

2 Related Work

Transferring knowledge from pre-trained language models (Devlin et al., 2018; Raffel et al., 2019; Brown et al., 2020) is a common technique for ensuring robust NLP downstream task performance. Early work by Zoph et al. (2016) explored transfer learning for NMT from a model pre-trained on a single language pair. More recently, methods that transfer from large-scale multilingual pre-trained models (Conneau et al., 2019; Liu et al., 2020; Goyal et al., 2022; NLLB Team et al., 2022) have achieved improved translation performance across a wide range of language pairs. Aji et al. (2020) conducted a study on pre-training and transfer for low-resource NMT. These works depend on real human language for pre-training and therefore inherit data issues such as toxicity and bias. In contrast, our work studies NMT pre-training and transfer from synthetic data based on 'nonsense' words.

Only a few methods have addressed the problem of pre-training from synthetic data in NLP. Krishna et al. (2021) proposed pre-training for summarization using synthetic article and summary pairs derived from manually curated tasks and a vocabulary of nonsense symbols. Sinha et al. (2021) have shown that masked language model pre-training with limited word-order information can be almost as effective as regular pre-training. Chiang and Lee (2020, 2021) show that non-human language data and artificial datasets (e.g. nested sequences of parentheses), can still demonstrate knowledge transfer to downstream NLP tasks. Wu et al. (2022) compare the effect of pre-training on many sim-

ple synthetic tasks. Our work in this paper represents the first empirical evaluation of synthetic pre-training for neural machine translation. To the best of our knowledge, our proposed synthetic tasks have not been explored in previous work.

The quality of a pre-trained model should not be measured purely by performance. We should also consider trustworthiness (He et al., 2022; Xu et al., 2022; He et al., 2021). Recent works have noted that translation systems pre-trained on web-scale corpora are prone to produce toxic (Costa-jussà et al., 2022) or biased outputs (Prates et al., 2020; Cho et al., 2021; Costa-jussà et al., 2020), and/or present privacy issues (Prates et al., 2020; Kamocki and O'Regan, 2016), which reduces user trustworthiness. Bias mitigation for NMT has been well-investigated while privacy and toxicity issues for translation are still not extensively explored (Costa-jussà et al., 2022). Wang et al. (2021) propose federated neural machine translation to protect privacy such as commercial leakage or copyright. (Costa-jussà et al., 2022) mitigate toxicity by filtering training data that matches pre-defined multilingual toxic word lists.

3 Synthetic Pre-Training for NMT

Pre-training followed by fine-tuning is a common approach to training robust NMT models (Conneau et al., 2019; Liu et al., 2020). Our motivation is to understand the extent to which the transfer benefits of pre-training can be replicated using synthetic tasks and data. In this section, we describe three approaches to the programmatic generation of synthetic data: (i) pre-training with obfuscated parallel data that implicitly preserves certain language properties such as distributional frequencies, (ii) pre-training with synthetic data created by concatenating aligned phrases, and (iii) pre-training with synthetic tasks designed to encourage transfer learning of important translation properties such as long-distance reordering.

3.1 Pre-Training on Obfuscated Parallel Data

In order to gain insight into what makes a good pre-trained model, we design an obfuscated pre-training experiment in which the model learns to translate obfuscated source sequences to obfuscated target sequences. The synthetic training data for this experiment is created by obfuscating words in the original parallel data. We define separate 1-to-1 nonsense token vocabulary mappings for the set of all words that occur in the source and target sides of the data: each source word s_i and target word t_j has a corresponding obfuscated nonsense source token $\mathcal{O}_{s_i}$ and target token $\mathcal{O}_{t_j}$. The synthetic pre-training corpus is created by replacing, with probability R, each source and target word with its corresponding obfuscated nonsense token. R thus determines the proportion of obfuscated tokens, allowing us to evaluate the extent to which pre-training knowledge transfer occurs with different obfuscation ratios. This method of obfuscation can be viewed as a trivial form of encrypted training. Although the original word identities are obscured, a great deal of useful information such as distributional frequencies, word order, dependency relations, alignments, and grammatical structure remain implicit in the obfuscated data. An example German-English parallel sentence pair and obfuscations at $R = 0.25$ and $R = 1.00$ (i.e. all tokens obfuscated) are shown below:

$R = 0.00$	src	Meine zweite Bemerkung ist etwas ernsthafter.
	trg	My second comment is rather more serious.
$R = 0.25$	src	wfnzc zweite Bemerkung ist etwas ernsthafter .
	trg	My IJODB comment is AHBNB more serious .
$R = 1.00$	src	wfnzc kqknd gmlfd tlieb ghzwa jdfnd engwd
	trg	UKVFB IJODB XRWOB SZEIA AHBNB LATAA MCSDA ETFJA

3.2 Pre-Training on Concatenated Phrases

In this section, we propose pre-training an NMT model with synthetic parallel data formed by concatenating aligned phrases. The main advantage of aligned phrases is that they are extracted from real parallel data and thus encode both lexical and structural translation knowledge. Lexical knowledge is defined by the word- and phrase-level correspondences between the source and target language. Structural knowledge, encoded by local reordering within aligned phrases, can also be leveraged.

We first extract a collection of aligned phrases $\mathcal{P}$ using the standard recipe implemented in the Moses SMT Toolkit (Koehn et al., 2007). The accuracy of the aligned phrases depends on the size and quality of the parallel data: we target low-resource MT and assume there is only a limited quantity of parallel data available. We generate synthetic parallel sentence pairs by first sampling a normally distributed phrase length P. We sample each phrase position $p = 1 \ldots P$ uniformly at random from $\mathcal{P}$. The source and target sentences thus consist of concatenated source and target phrases. The word order within each sampled phrase is fluent and local reordering may also be captured. The boundaries

between phrases, however, typically do not respect natural word order or grammar. In spite of these limitations, we show in Section 4.3 that this simple method of data augmentation can significantly improve the quality of an NMT model when training data is limited. An example Indonesian-to-English synthetic sentence pair, with phrase boundaries indicated by parentheses, is shown below:

src	[sejak Wright] [sambil seringkali] [kami] [50 juta mengingat]
trg	[from Wright] [in most times] [we] [50 millions as]

3.3 Pre-Training on Synthetic Tasks and Data

In this section, we define three completely synthetic task variants that can be used for NMT pre-training: (1) the identity operation, (2) case-mapping, and (3) permuted binary trees. All three tasks are based on a procedural data generation model and can thus be used to generate arbitrary quantities of synthetic data. Procedural generation of synthetic parallel sentence pairs allows for complete control over the alignments, length distribution, token frequency distribution, and level of noise in the data.

All three synthetic tasks are based on a 1-to-1 paired dictionary of source and target synthetic tokens: $\mathcal{S}$ for source and $\mathcal{T}$ for target. We define a pairwise mapping between the two vocabularies such that each synthetic source token $\mathcal{S}_i$ is paired with a corresponding synthetic target token $\mathcal{T}_i$ for each $i \in 1 \ldots N$, where N is the size of the paired vocabulary. In the examples below, the source vocabulary consists of all $26^3 = 17576$ three-character synthetic tokens that can be created using the lowercase English letters $\{a, \ldots, z\}$.

3.3.1 Synthetic Task 1: Identity Operation

The simplest of the pre-training tasks we consider is the identity operation, which has been previously proposed by Wu et al. (2022) as a synthetic task for language model pre-training. For this task, the source and target sentences are identical. We include it not because we believe it to be in any way a proxy for the true translation task, but instead to serve as the simplest possible baseline sequence-to-sequence synthetic task. We generate parallel sentence pairs by first sampling a sentence length L from the normal distribution. Each source token s_i for $i = 1 \ldots L$ is sampled uniformly from the source vocabulary $\mathcal{S}$. The target sentence is simply a copy of the source:

src	cea qne jda rnu jkq ozf dke kzl hpo
trg	cea qne jda rnu jkq ozf dke kzl hpo

3.3.2 Synthetic Task 2: Case-Mapping

Our second pre-training task defines a case-mapping operation. Each synthetic parallel sentence pair consists of the same sequence of tokens but the source sentence is lowercase and the target sentence is uppercase. We also design an extension of this task that includes insertions and deletions. Source and target tokens can be deleted with fixed probability d_s (for source) and d_t (for target). Random insertions and deletions are added to avoid having identical source and target lengths for every sentence pair, which might entrench the tendency of the model to mimic such behavior even at the fine-tuning stage where it is likely inappropriate. From the perspective of the translation task, a sentence pair with a missing target token corresponds to a deletion, while a missing source token corresponds to an insertion. The following example shows a parallel sentence pair for the case-mapping task with fixed source and target deletion probabilities $d_s = d_t = 0.15$:

src	qdo zwj iub uxj pls nsn igk mrz ojw
trg	QDO ZWJ IUB KWP UXJ PLS NSN IGK MRZ OJW

3.3.3 Synthetic Task 3: Permuted Trees

The third of our synthetic pre-training tasks is designed to reflect some aspects of the reordering process that occurs during natural language translation. We first generate random sentences with normally distributed lengths and uniformly distributed synthetic tokens, as for tasks 1 and 2. We then induce an artificial binary tree over the source sentence by picking a random point at which to split the sentence, and recursively repeat this process for the left and right sub-strings. The resulting binary tree structure allows us to generate synthetic parallel data with reordering that preserves the alignment of contiguous source-to-target token spans. The target tree is generated as a permutation of the source tree: we randomly swap left and right sub-trees with some fixed probability r. Generating synthetic sentence pairs in this way implies the existence of lexicalised synchronous context free grammar (SCFG) rules (Chiang, 2007) that could be used to generate the sentence pair as a parallel derivation. The example below shows a synthetic sentence pair generated using this method:

src	[jtx [[urs [ktp [hme nmc]]] pep]]
trg	[JTX [[URS [[HME NMC] KTP]] PEP]]

Parentheses indicating the tree structure are shown for clarity. During pre-training, however, only the source and target synthetic token sequences are actually seen by the model. In this example, the source token 'ktp' was reordered with respect to the sub-tree containing the tokens 'hme nmc'. Figure 2 shows the token-level alignment and reordering operations encoded by this parallel sentence pair.

4 Experimental Framework

We evalute our synthetic pre-training data generation methods for NMT using using both English-centric and non-English-centric language pairs.

4.1 Experiment Setup

English-Centric Language Pairs For English-centric translation directions, we use fine-tuning data sets similar to Aji et al. (2020). For German-English, we use the official data from the WMT 2014 News Translation Task. For Myanmar-English, the fine-tuning data consists of 18.0k parallel sentence pairs in the news domain collected for the Asian Language Treebank (ALT) project (Ding et al., 2018). We use the original train, dev and test split. For Indonesian-English, we use a filtered set of 24.6k parallel sentence pairs from the IDENTIC v1.0 corpus (Larasati, 2012) which covers various genres. We randomly divide the original corpus into distinct train (90%), dev (5%), and test (5%) sets. For Turkish-English, we use data from the WMT 2017 News Translation Task (Yepes et al., 2017). The training set includes 207.7k parallel sentence pairs. We use the WMT `newsdev2016` set for validation, and report results on `newstest2017`.

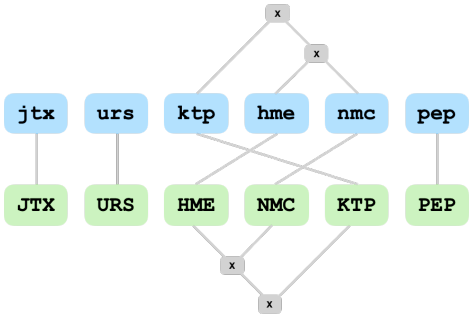


Figure 2: Example synthetic sentence pair and partial derivation for the aligned permuted binary trees task. In this example, a single non-terminal node was reordered.

Non-English-Centric Language Pairs For non-English-centric directions, we simulate low-resource translation conditions by sampling data from OPUS NLP (Tiedemann, 2012). The non-English-centric language pairs we evaluate are as follows: Indonesian-Myanmar, Indonesian-Turkish, Indonesian-Tagalog, Myanmar-Turkish, Myanmar-Tagalog, Tagalog-Turkish, German-Indonesian, and German-Myanmar. For each pair, we simulate low-resource conditions by creating fine-tuning sets of size 10k, 25k, 50k, and 100k via sampling from the set of all parallel corpora for that language pair on OPUS NLP. Minimal filtering is applied to our parallel data sets: we remove duplicates, discard sentences with extreme length ratios, and keep only sentence pairs for which the `fasttext` (Joulin et al., 2016) language ID matches the stated source and target.

Evaluation Following the evaluation setting of large-scale multilingual models such as FLORES-101 (Goyal et al., 2022), we score our translation hypotheses using `sentencepiece` BLEU (Papineni et al., 2002) (spBLEU). This avoids the need for custom post-processing for individual languages with unusual scripts and/or complex morphology such as Burmese.

Model Training Strategy Our experiments consist of a pre-training stage followed by a fine-tuning stage. We use the transformer sequence-to-sequence 'base' model architecture (Vaswani et al., 2017) for all translation experiments. Since our goal is to gain insight into the relative importance of various aspects of synthetic pre-training, our baseline models are created by fine-tuning randomly initialized models using only the downstream task parallel data.

We use `fairseq` (Ott et al., 2019) to train our models with the Adam (Kingma and Ba, 2014) optimizer. We reset the learning rate scheduler and optimizer before starting the fine-tuning stage. Pre-training and fine-tuning continue until the BLEU score on the validation set converges. Further implementation details can be found in Appendix B.

4.2 Pre-training with Obfuscated Data

Following previous work that showed German-to-English to be a good pre-training direction for several language pairs (Aji et al., 2020), we also use German-to-English (`de-en`) for pre-training and randomly sample two million pairs from its training corpus to use as obfuscated parallel data. We

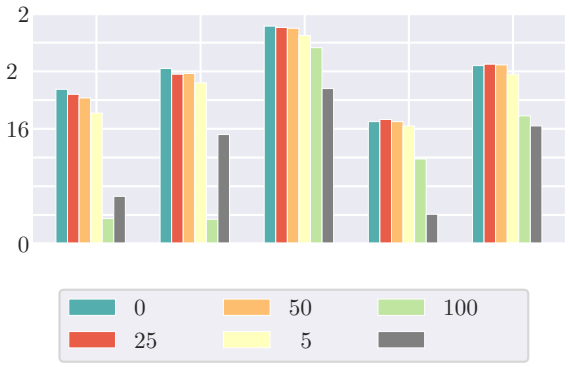


Figure 3: Translation spBLEU scores after pre-training with different levels of obfuscation and real-world fine-tuning on downstream language pairs. *Scratch* refers to training from scratch using only fine-tuning data. Similar results on FLORES can be found in Appendix A.2.

vary the obfuscation ratio R from 0% to 100% in 25% increments. After pre-training, we fine-tune the models on the real-world parallel training corpus (described in Section 4.1) for each downstream language pair. We also investigate the scaling effect of different fine-tuning set sizes and show the results in Appendix A.1.

We report spBLEU scores on the test set for each language pair in Figure 3. We find that, surprisingly, even when as much as 75% of the pre-training data is obfuscated, the models are still able to achieve high or even comparable spBLEU scores to real-world pre-trained models (i.e., those with 0% obfuscation). Additionally, most of the models pre-trained on obfuscated data performed better than those trained from scratch on real-world fine-tuning data, even when the pre-training data was 100% obfuscated (e.g., 100% in `id-en`, `my-en`, and `my-tl`). This suggests that a small proportion of real-world data can provide the majority of the benefits of large-scale regular pre-training, implying a promising research direction for efficient pre-training or improving low-resource NMT.

4.3 Pre-training with Phrase Concatenation

The translation decoding results in Table 1 show substantial transfer learning benefits from pre-training with 2m sentence pairs of synthetic data generated by concatenating uniformly sampled aligned phrase pairs (phrase-cat). Compared to a model with no pre-training, i.e. one that trains from random initialization using only the fine-tuning data (random-init), we observe large gains of up to +9.9 spBLEU for language pairs with less than 25k of fine-tuning data (`my`↔`en` and `id`↔`en`). The gains of +1.4 to +2.1 for `tr`↔`en` are smaller: this pair has more fine-tuning data (>200k pairs) so the improved coverage and robustness of synthetic pre-training is less critical for good performance. It is important to note that this method does not utilize any additional real parallel or monolingual data, but instead derives new data directly from the existing fine-tuning corpus. Our synthetic pre-training corpus, although unnatural at the sentence-level, contains many phrase-level alignments and reordering information which reinforces the translation knowledge captured by the model. Any destructive effect from presenting to the model during pre-training sentence pairs with unnatural word order or bad grammar, can be rectified in the fine-tuning stage by showing the model the original fluent source and target sentences.

4.4 Pre-Training with Synthetic Data

We pre-train transformer (Vaswani et al., 2017) models using 2m sentence pairs of synthetic parallel data to match the data size used in our obfuscation experiments. We further explore the effect of scaling the synthetic pre-training data size in Appendix A.4. Separate synthetic training sets were generated for each of the three task variants described in Section 3.3. Additional sets of 4000 synthetic pairs were generated as validation data. Each pre-trained model is subsequently fine-tuned with real parallel data for a specific language pair: `my`↔`en`, `id`↔`en`, and `tr`↔`en`. In Table 1, we report `sentencepiece` BLEU (spBLEU) (Goyal et al., 2022) scores for our three synthetic pre-training task variants. For comparison purposes, we also show the scores obtained without pre-training – i.e. a randomly initialized model trained on only the fine-tuning data.

Our first observation is that synthetic pre-training with the identity operation task (§3.3.1) does not perform well. For all three language pairs it is slightly worse than simply fine-tuning from a randomly initialized model. This is to be expected since the pre-training task is too crude: a simple copy operation from source to target with identical lengths. Pre-training with the case-mapping synthetic task (§3.3.2) and deletion probability $d_s = d_t = 0$ improves the scores, with gains of +1.0 to +5.0 spBLEU over the identity operation on our test set. Although the case-mapping pre-training task is still quite crude, it is able to beat fine-tuning from a randomly initialized model

	my-en		id-en		tr-en		en-my		en-id		en-tr	
	Test	Flores	Test	Flores	Test	Flores	Test	Flores	Test	Flores	Test	Flores
scratch	4.1	1.8	18.2	7.2	14.7	17.7	16.2	6.3	19.1	8.3	17.0	16.4
identity	3.2	1.1	16.8	7.6	12.4	13.8	12.7	4.5	18.1	9.7	13.8	13.5
case-map	6.7	1.6	21.8	12.1	13.4	15.1	16.4	6.0	22.9	13.8	15.6	15.2
pb-trees	11.4	2.5	23.1	12.2	14.4	16.9	18.9	7.0	23.8	14.4	16.6	16.3
phrase-cat	14.0	3.9	27.3	14.4	16.5	19.1	23.0	8.6	28.1	17.0	18.4	18.5

Table 1: Translation decoding results (spBLEU) for three purely synthetic pre-training variants and concatenation of aligned phrases vs. fine-tuning from a randomly initialized baseline (scratch) (English-centric language pairs).

for both Myanmar-to-English and Indonesian-to-English. Our best performing synthetic task is pb-trees (§3.3.3) with a node reordering probability $r = 0.15$. This model shows that transfer learning from synthetic pre-training to real-world tasks can be substantial, with scores as high as +7.3 spBLEU over the baseline for Myanmar-to-English and +4.9 for Indonesian-to-English. We do not see gains for Turkish-to-English for any of our purely synthetic pre-training tasks. The fine-tuning data for this language pair is much larger than that of the other language pairs. As the fine-tuning data size increases, the benefits of transfer learning from pre-training diminish.

We also evaluate the strongest of our three purely synthetic pre-training tasks, pb-trees, on additional non-English-centric language pairs. Table 8 in Appendix A.7 shows spBLEU decoding results for these additional pairs. We compare performance over a range of different fine-tuning set sizes. On both OPUS-Test and FLORES-devtest, and for the majority of fine-tuning set sizes, synthetic pre-training with the pb-trees task typically outperforms fine-tuning from a randomly initialized baseline.

5 Analysis and Discussion

5.1 Synthetic Knowledge Transfer

In this section, we discuss what kind of useful representations are actually learned by the model when pre-training with purely synthetic tasks and data. Our empirical study has shown that pre-training on synthetic data can result in improved translation quality after fine-tuning for a specific language pair. Even though the pre-training data is entirely synthetic, the model must have successfully learned representations and structures relevant for translation that can be leveraged via transfer learning to the downstream task.

In Table 2, we show the word piece overlap between our tokenized synthetic pre-training corpus and the real human language corpus for three fine-tuning language pairs. Our vocabulary consists of 26^3 paired lowercase-uppercase synthetic tokens, but after tokenization the number of unique word pieces is much lower. For example, there are only 3,541 unique source and 2,405 unique target word pieces after tokenizing a corpus of 2M synthetic parallel sentence pairs. The fine-tuning data, although much smaller, has a far greater token diversity for English, Indonesian, and Turkish. Myanmar is the exception: it is aggressively segmented by the XLMR `sentencepiece` model which results in far fewer unique word pieces.

We compute the intersection between the set of word pieces in the synthetic pre-training data and those in the fine-tuning data in the last column of Table 2. We observe low word piece overlap. For example, only 35 of the 3541 word pieces that occur in the source side of the synthetic corpus also occur in the source side of the `my-en` corpus. This number is low because the Myanmar script is so different from English. But overlap remains low even for languages such as Indonesian and Turkish which have similar alphabets to English. Low levels of overlap were also observed in our obfuscated pre-training experiments (Table 6). The low word piece overlap means that most of the word embeddings learned during pre-training have little relevance to the fine-tuning or inference stages. We conclude that any transfer learning benefit exhibited by the model on the downstream task must be captured in the inner layers of the transformer.

5.2 Lexical and Structural Knowledge

The results in Table 1 show phrase-cat to be an effective pre-training strategy for low-resource NMT. Both lexical and structural knowledge is captured in the aligned phrases. However, since the phrases are sampled from the uniform distribution, long-

Pair	PT/FT	$\|V_{PT}\|$	$\|V_{FT}\|$	Overlap
my-en	src: lc/my	3,541	1,598	35
	trg: uc/en	2,405	18,514	740
id-en	src: lc/id	3,541	18,095	1,377
	trg: uc/en	2,405	18,167	740
tr-en	src: lc/tr	3,541	24,616	1,938
	trg: uc/en	2,405	26,236	1,358

Table 2: Tokenized pre-training (PT) and fine-tuning (FT) word piece counts and overlap statistics: 'lc' and 'uc' denote lowercase and uppercase synthetic tokens.

distance structure is ignored and only local reordering information is captured. The pb-trees method also allows us to encode structural knowledge into our synthetic data since it is possible to generate sentence pairs that reorder sub-trees over long distances. Comparing the effectiveness of both methods shows that surprising gains in translation quality are possible even for synthetic data generation methods such as phrase-cat that encode only very local structural knowledge. This insight, that it is mainly collocations (especially, for NMT, parallel collocations) agrees with the conclusions about the relative lack of importance of word order to LM pre-training in Sinha et al. (2021).

5.3 Translation Quality vs. Toxicity

To evaluate model toxicity, we consider catastrophic mistranslations (Costa-jussà et al., 2022). These errors occur when a model hallucinates toxic terms in the translated text, even though no such terms occur in the source text. Following the toxicity measurement setup of Goyal et al. (2022), we use the FLORES Toxicity-200[1] word lists to calculate the toxicity rate of translations produced by a model. The lists cover 200 languages and contain frequently used profanities, insults, and hate speech terms. We consider a sentence toxic if it contains words that match entries in these lists. The toxicity rate for each model is defined as the proportion of sentences with hallucinated toxicity in translations of the test set and a larger set of 100k monolingual sentences randomly sampled from CC-100 (Wenzek et al., 2020; Conneau et al., 2019). We compare BLEU scores and toxicity rates for various models including current state-of-the-art large pre-trained multilingual translation models in Table 3.

Results and Analysis We first observe that models pre-trained on synthetic data obtain significantly higher BLEU scores than baselines trained from scratch using only the fine-tuning data. This confirms that our proposed synthetic tasks indeed capture useful knowledge that can be applied through transfer learning to low-resource NMT tasks. When compared to the multilingual translation models FLORES-101 (615M parameters) and M2M-100 (1.2B parameters), we note that models pre-trained on synthetic data obtain comparable performance for languages my-en and even outperform multilingual models by a large margin on de-my, id-en, and my-tl, though with inferior translation quality on de-id. It should be noted that some of these language pairs represent zero-shot directions for M2M-100. We compare our synthetic methods with the standard NMT data augmentation technique of back-translation in Appendix A.3.

While these results are quite promising, we note that our goal in this paper is not to surpass the state-of-the-art in translation quality achieved by large-scale massively multilingual models on low-resource NMT. Instead, we seek to further understand which properties of pre-training based on synthetic tasks and data - along the structural and lexical knowledge axes of Figure 1 - enhance transfer learning performance, while minimizing toxicity and other data issues inherent in models that rely on large-scale pre-training using real data.

Analyzing toxicity, we observe the presence of catastrophic mistranslations in all models, but less frequently when training from scratch in most cases. This is because the low-resource fine-tuning data contains very little toxic content. On the other hand, as noted above, the BLEU scores when training models from scratch are very low. We see that the FLORES-101 and M2M-100 models both exhibit toxicity, since they were pre-trained on real-world corpora that can include toxic content. Our results show that synthetic pre-training can produce models with comparable BLEU scores while significantly reducing catastrophic mistranslations. We observe that parallel data generated from permuted binary trees has the lowest toxicity among the three synthetic pre-training methods, since it relies on purely synthetic data. This may indicate that patterns in the data can still trigger toxic terms, even after the words have been obfuscated or phrases have been shuffled. We include additional toxicity results and analysis in Appendix A.5.

[1] http://github.com/facebookresearch/flores/tree/main/toxicity

	Model	de-id		de-my		id-en		my-en		my-tl	
		BLEU	Toxicity	BLEU	Toxicity	BLEU	Toxicity	BLEU	Toxicity	BLEU	Toxicity
Baseline	scratch	6.6	0.68	15.2	0.01	18.2	0.05	4.1	0.02	16.4	0.04
Large Pretrained Multilingual Model	M2M-100	32.9	0.68	9.1	0.03	30.2	0.28	1.8	0.15	14.2	0.06
	FLORES-101	30.0	0.63	12.3	0.03	26.0	0.23	4.6	0.18	12.8	0.08
Synthetic Pre-training	obfuscation	18.2	0.34	22.4	0.01	29.0	0.11	16.4	0.08	23.6	0.04
	phrase-cat	14.7	0.50	19.6	0.02	27.3	0.10	14.0	0.02	22.5	0.03
	pb-trees	11.7	0.45	12.3	0.01	23.1	0.10	11.4	0.01	20.7	0.02

Table 3: BLEU scores and toxicity rates for various models on low-resource language pairs. Baseline is training on fine-tune real-world data as lower bound of performance. Large pre-trained models are upper bound of performance.

6 Conclusion

Our study of synthetic pre-training tasks for NMT showed that pre-training benefits can still be achieved even when using synthetic or obfuscated data. Additionally, we have shown that synthetic data has the potential to reduce model toxicity compared to models trained on web-scale crawled corpora. Our research provides insights into what types of knowledge transfer make for a good pre-trained model. We believe that synthetic data augmentation techniques based on synthetic tasks and procedurally generated data are a promising solution for addressing pre-training data concerns, and can lead to efficient, accurate, and trustworthy NMT. In future work, we plan to further investigate synthetic pre-training by exploring more advanced data generation models and directly optimizing the parameters for specific downstream fine-tuning tasks. Increasing the effectiveness of synthetic data at different data scales is also worthy of further exploration.

7 Limitations

Our work seeks to gain insight into what pre-training knowledge is transferred and useful for downstream fine-tuning in NMT using synthetic tasks and data. We note that changes in the data generation methods do require re-running the pre-training stage, which is computationally expensive compared to the fine-tuning stage.

Our current synthetic data generation methods are somewhat crude. Although they are designed to encode varying degrees of lexical and structural translation knowledge, they do so in a rather simplistic way. For example, sampling phrases from the normal distribution ignores distributional frequencies which represent information that is likely useful for the synthetic data generation task. In this paper we have presented some interesting initial findings regarding the suitability of synthetic pre-training for NMT. We plan to explore more sophisticated data generation models in future work.

We acknowledge that synthetic pre-training is unlikely to surpass the quality of real-world massively multilingual pre-trained models in performance, especially if synthetic data is the only data used for pre-training. However, good performance can probably be achieved by combining synthetic pre-training and real-data pre-training. Of course, this risks exposing the model to toxic and sensitive or private content. Therefore, concerns of both model quality and data quality should be considered when evaluating the impact and benefits of synthetic pre-training. We view synthetic pre-training as a complimentary approach to finding an optimal balance rather than as a replacement for previous state-of-the-art NMT pre-training methods.

8 Ethics Statement

All of the training data used in our experiments are official releases of publicly available benchmarks. In addition, the toxic word lists used to measure toxicity are obtained from the public FLORES repository which requires a password to access, thus reducing the risk of hacking by a malicious user or adversarial bot. In addition, as for the issue of hallucinated toxicity discussed previously, we note that our work also has the potential to address other problematic translation behaviors, such as hallucinated bias.

9 Acknowledgements

This material is based upon work supported by the Defense Advanced Research Projects Agency under Contract No. FA8750-19-C-1001. Disclaimer: Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency. Zexue He is supported by an IBM Ph.D.

Fellowship and is independent of the Defense Advanced Research Projects Agency.

References

Željko Agić and Ivan Vulić. 2019. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy. Association for Computational Linguistics.

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. 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 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Alham Fikri Aji, Nikolay Bogoychev, Kenneth Heafield, and Rico Sennrich. 2020. In neural machine translation, what does transfer learning transfer? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7701–7710, Online. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Cheng-Han Chiang and Hung-yi Lee. 2020. Pre-training a language model without human language. *arXiv preprint arXiv:2012.11995*.

David Chiang. 2007. Hierarchical phrase-based translation. *computational linguistics*, 33(2):201–228.

David Cheng-Han Chiang and Hung-yi Lee. 2021. On the transferability of pre-trained language models: A study from artificial datasets. *CoRR*, abs/2109.03537.

Won Ik Cho, Jiwon Kim, Jaeyeong Yang, and Nam Soo Kim. 2021. Towards cross-lingual generalization of translation gender bias. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 449–457.

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. *CoRR*, abs/1911.02116.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Marta R Costa-jussà, Carlos Escolano, Christine Basta, Javier Ferrando, Roser Batlle, and Ksenia Kharitonova. 2020. Gender bias in multilingual neural machine translation: The architecture matters. *arXiv preprint arXiv:2012.13176*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Chenchen Ding, Masao Utiyama, and Eiichiro Sumita. 2018. Nova: A feasible and flexible annotation system for joint tokenization and part-of-speech tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(2):1–18.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *J. Mach. Learn. Res.*, 22(107):1–48.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzman, and Angela Fan. 2022. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Zexue He, Bodhisattwa Prasad Majumder, and Julian McAuley. 2021. Detect and perturb: Neutral rewriting of biased and sensitive text via gradient-based decoding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4173–4181, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zexue He, Yu Wang, Julian McAuley, and Bodhisattwa Prasad Majumder. 2022. Controlling bias exposure for fair interpretable predictions. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5854–5866, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Paweł Kamocki and Jim O'Regan. 2016. Privacy issues in online machine translation services-european perspective. In *Proceedings of the Tenth International*

Conference on Language Resources and Evaluation (LREC'16), pages 4458–4462.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 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.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Kundan Krishna, Jeffrey Bigham, and Zachary C Lipton. 2021. Does pretraining for summarization require knowledge transfer? *arXiv preprint arXiv:2109.04953*.

Septina Dian Larasati. 2012. Identic corpus: Morphologically enriched indonesian-english parallel corpus. In *LREC*, pages 902–906.

Klas Leino, Emily Black, Matt Fredrikson, Shayak Sen, and Anupam Datta. 2018. Feature-wise bias amplification. In *International Conference on Learning Representations*.

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. *CoRR*, abs/2001.08210.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *CoRR*, abs/1904.01038.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Marcelo OR Prates, Pedro H Avelar, and Luís C Lamb. 2020. Assessing gender bias in machine translation: a case study with google translate. *Neural Computing and Applications*, 32(10):6363–6381.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96.

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*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Lrec*, volume 2012, pages 2214–2218. Citeseer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Jianzong Wang, Zhangcheng Huang, Lingwei Kong, Denghao Li, and Jing Xiao. 2021. Modeling without sharing privacy: Federated neural machine translation. In *International Conference on Web Information Systems Engineering*, pages 216–223. Springer.

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 Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Yuhuai Wu, Felix Li, and Percy Liang. 2022. Insights into pre-training via simpler synthetic tasks. *arXiv preprint arXiv:2206.10139*.

Canwen Xu, Zexue He, Zhankui He, and Julian McAuley. 2022. Leashing the inner demons: Self-detoxification for language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11530–11537.

Antonio Jimeno Yepes, Aurélie Névéol, Mariana Neves, Karin Verspoor, Ondřej Bojar, Arthur Boyer, Cristian Grozea, Barry Haddow, Madeleine Kittner, Yvonne Lichtblau, et al. 2017. Findings of the wmt 2017 biomedical translation shared task. In *Proceedings of the Second Conference on Machine Translation*, pages 234–247.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

A Supplementary Results

A.1 Scaling Effect of Obfuscated Pre-training

We first evaluate the performance of regular pre-training and fine-tuning with various quantities of real-world German-to-English data. The results in Figure 4 show that the highest BLEU scores are obtained by using regular real-world parallel data (i.e. 0% obfuscation). We compare vs. models trained solely on the fine-tuning data ('Scratch'): the resulting BLEU scores are quite low when the training data size is small, confirming the importance and benefits of pre-training for NMT.

A.2 FLORES Obfuscated Pre-training Results

We show additional decoding results for the matched (with source and target fine-tuning languages that are the same as the pre-training languages: `de-en`) vs. unmatched (with source or target fine-tuning languages that differ from the pre-training languages: `de-id`, `de-my`, `id-en`, `my-en`, `my-tl`) conditions of obfuscated pre-training on the FLORES devtest set in Figure 5.

A.3 Back-Translation Comparison

Back-translation (Sennrich et al., 2016) is an effective technique for improving the quality of machine translation. It works by creating new parallel sentence pairs by translating target-side monolingual data into the source language using an inverse direction MT system. The new sentence pairs consist of a (possibly noisy) back-translated source sentence paired with a high-quality target sentence. We compare our synthetic training methods to an NMT system that has been trained on an augmented data set that includes back-translated parallel data. We use our baseline models for `my-en` and `en-my` to produce the back-translated sentences. For each direction `my-en` and `en-my`, we generate an additional set of 2m back-translated sentences. The results are shown in Table 4. We note that back-translation provides only limited improvements vs. the baseline model trained from scratch for `my-en` and actually hurts for `en-my`. This is because back-translation requires a good quality model in the target-to-source direction in order to produce accurate and relevant translations. The `my-en` baseline model is not of sufficiently high quality to produce useful back-translations. Our synthetic methods significantly outperform back-translation for both translation directions, confirming our expectation about the limitations of back-translation in low-resource conditions, and further illustrating the potential of our proposed synthetic approaches.

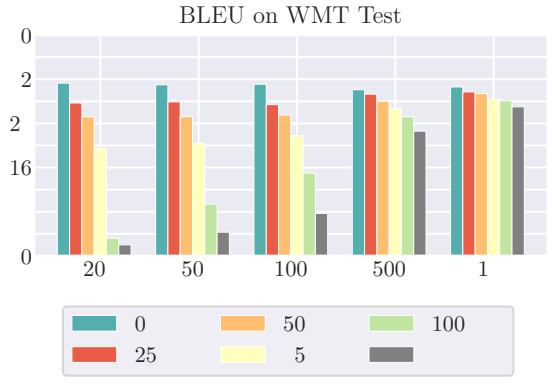


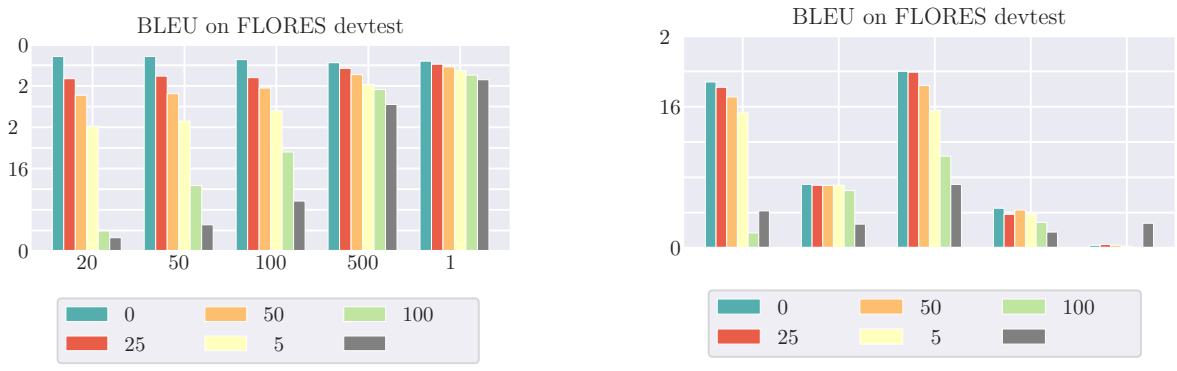
Figure 4: Translation results after pre-training with different levels of obfuscation and real-world fine-tuning on the same language pairs, with various quantities of fine-tuning data in `de-en`. *Scratch* refers to training from scratch using only fine-tuning data.

	my-en		en-my	
Model	Test	Flores	Test	Flores
scratch	4.1	1.8	16.2	6.3
back-translation	10.7	2.0	11.1	4.1
phase-cat	14.0	3.9	23.0	8.6
pb-trees	11.4	2.5	18.9	7.0

Table 4: Synthetic pre-training v.s. back-translation on WMT test set and FLORES devtest set.

A.4 Synthetic Pre-training Data Scaling

Figure 6 shows the data scaling behavior of the pb-trees and phrase-cat synthetic pre-training methods. We pre-train each model with proper subsets of varying sizes sampled from the full 2m pairs used in the rest of our experiments. For pb-trees, the scaling is mostly flat. The BLEU scores, while consistently higher than the baseline (which uses no pre-training at all), increase only gradually with additional synthetic training data. The BLEU gains over the baseline are therefore a result of priming the model for the task of translation, rather than learning any useful real-world lexical relationships between the source and target languages. For phrase-cat, the data scaling curve is much more pronounced. For all three tasks, we observe steadily increasing BLEU scores with larger synthetic training set sizes, reaching a plateau at around 1m pairs. The phrase-cat method benefits from additional samples and combinations of real phrase pairs since



(a) (b)

Figure 5: Translation decoding results on WMT for (a) regular parallel corpus (0%) vs. obfuscated pre-training as a function of fine-tuning set size (x-axis) and obfuscation ratio (in different colors), and (b) unmatch conditions.

the synthetic pairs provide additional coverage of possible word orders and translation relationships that can aid the subsequent fine-tuning and decoding of the testset.

A.5 Further Analysis of Toxicity

We further analyze the toxicity of our models by comparing the toxicity rate of source language sentences and their translations. Firstly, we test `de-en` translation systems with obfuscated pre-training on WMT test, as shown in Table 5. We observe that training with real-world data (i.e. obfuscation ratio $R = 0\%$) generates translations that contain toxic terms more frequently than they occur in the source. This indicates a toxicity amplification effect, a problem highlighted previously for toxicity (Costa-jussà et al., 2022) and bias (Leino et al., 2018). Pre-training with obfuscated data, however, is a promising way of mitigating this phenomenon, as shown by the big reduction in toxicity rate as the obfuscation ratio is increased. We observe a similar pattern for CC-100 data as well. The sentences in the CC-100 corpus are more toxic than those in the WMT testset ($0.57\% > 0.43\%$).

A.6 Word-Piece Overlap Statistics for Obfuscated Pre-Training

Similar to Section 5.1, we also report the token overlap between completely encrypted pre-training data (both source and target corpus) and real-world fine-tuning data, on `de-en` as shown in Table 5 and other language directions `id-en`, `my-tn`, and `tr-en` in Table 7. In `de-en` translation, we notice that the overlap is just 0.08% on the source language and 0.04% on the target language, with the largest size of the fine-tuning set (1M). On low-resource language pairs, we can see there is almost no overlap between pre-training and fine-tuning on both source and target sides, as shown in Table 7. This strong evidence supports the conclusion mentioned in Section 5.1 – most of the representations in the first layers are not touched during pre-training, and the benefits from pre-training may come from the inner layers which capture the transferable high-level knowledge for downstream tasks.

A.7 Synthetic Pre-Training: Additional Language Pairs

Table 8 shows translation decoding results (sp-BLEU) for additional non-English-centric language pairs. We compare synthetic pre-training on permuted binary trees vs. fine-tuning from a randomly initialized model as a function of the fine-tuning set size. Cells marked 'n/a' indicate there was insufficient parallel data to create a fine-tuning set of the specified size.

B Implementation Details

This section describes implementation details for facilitating the reproduction of our work.

B.1 Model Architectures

All translation models described in our experiments are based on the sequence-to-sequence transformer 'base' architecture (Vaswani et al., 2017) as implemented in `fairseq` (Ott et al., 2019). The models have six encoder layers, six decoder layers, and eight attention heads. The word embedding size is 512, and the feed-forward layers have 2048 dimensions. All BLEU scores are computed using SacreBLEU (Post, 2018) with `sentencepiece` tokenization (Goyal et al., 2022). Our SacreBLEU

Fine-Tuning Set Size	Obfuscation Ratio				
	0%	25%	50%	75%	100%
20k	0.57	0.40	0.43	0.37	0.00
50k	0.43	0.53	0.47	0.40	0.03
100k	0.53	0.33	0.40	0.27	0.07
500k	0.50	0.50	0.33	0.33	0.40
1M	0.57	0.47	0.40	0.37	0.37

Fine-Tuning Set Size	Obfuscation Ratio				
	0%	25%	50%	75%	100%
20k	0.37	0.33	0.33	0.21	0.01
50k	0.37	0.35	0.37	0.26	0.05
100k	0.43	0.32	0.30	0.23	0.17
500k	0.36	0.38	0.36	0.32	0.27
1M	0.38	0.45	0.36	0.35	0.33

Table 5: Toxicity rate (%) on WMT Test (left) and sampled CC-100 data (right). Results that increase toxicity compared to the source (0.43% for WMT and 0.57% for CC-100) are colored in red; otherwise they are colored in green. The degree of toxicity is shown by the darkness of the color.

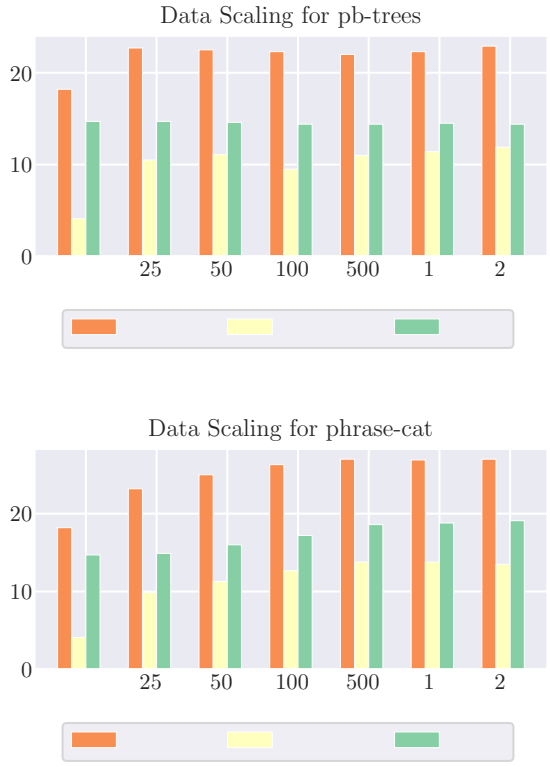


Figure 6: Effect on BLEU score of scaling up the size of the procedurally generated parallel data used during pre-training for two of our synthetic tasks: permuted binary trees 'pb-trees' (top), and concatenated aligned phrases 'phrase-cat' (bottom).

scoring signature[2] indicates that both source and reference are `sentencepiece` tokenized prior to scoring.

B.2 Hyper-Parameters and Training Configuration

Table 9 shows the hyper-parameters and training settings used for our experiments. We found different warm-up schedules were appropriate for the pre-training and fine-tuning stages. We choose the best model during training by maximizing the tokenized BLEU score on the validation set. For both pre-training and fine-tuning, we allow training to continue until the BLEU score has fully converged.

[2] `BLEU+case.mixed+numrefs.1+smooth.exp +tok.spm+version.1.5.1`

Model	FT size	PT/FT Language	$\|V_{PT}\|$	$\|V_{FT}\|$	Overlap
Obfuscated Pre-training	**20k**	src: nonsense-de/de	1,289,374	77,284	119
		trg: nonsense-en/en	680,221	56,339	15
	50k	src: nonsense-de/de	1,289,374	148,282	215
		trg: nonsense-en/en	680,221	102,900	33
	100k	src: nonsense-de/de	1,289,374	241,617	270
		trg: nonsense-en/en	680,221	163,105	50
	500k	src: nonsense-de/de	1,289,374	729,937	651
		trg: nonsense-en/en	680,221	466,678	164
	1m	src: nonsense-de/de	1,289,374	1,170,435	950
		trg: nonsense-en/en	680,221	730,119	271
Regular Pre-training	**20k**	src: de/de	1,861,801	77,284	65,006
		trg: en/en	1137,015	56,339	49,295
	50k	src: de/de	1,861,801	148,282	117,827
		trg: en/en	1,137,015	102,900	85,111
	100k	src: de/de	1,861,801	241,617	180,708
		trg: en/en	1,137,015	163,105	126,278
	500k	src: de/de	1,861,801	729,937	435,333
		trg: en/en	1,137,015	466,678	291,138
	1m	src: de/de	1,861,801	1,170	600,922
		trg: en/en	1,137,015	730,119	394,598

Table 6: Tokenized pre-training (PT) and fine-tuning (FT) word piece counts and overlap statistics comparing obfuscated pre-training (upper part) vs. regular pre-training (lower-part) for German-to-English parallel data with various fine-tuning data set sizes.

Model	Language Pair	PT/FT Language	$\|V_{PT}\|$	$\|V_{FT}\|$	Overlap
Obfuscated Pre-training	id-en	src: nonsense-de/id	1,289,374	18,095	112
		trg: nonsense-en/en	680,221	18,167	0
	my-en	src: nonsense-de/my	1,289,374	1,598	1
		trg: nonsense-en/en	680,221	18,514	0
	tr-en	src: nonsense-de/tr	1,289,374	24,616	270
		trg: nonsense-en/en	680,221	26,236	0
Regular Pre-training	id-en	src: de/id	1,861,801	18,095	3,722
		trg: en/en	1,137,015	26,236	6,483
	my-en	src: de/my	1,861,801	1,598	97
		trg: en/en	1,137,015	18,514	4,407
	tr-en	src: de/tr	1,861,801	24,616	5,569
		trg: en/en	1,137,015	26,236	6,483

Table 7: Tokenized pre-training (PT) and fine-tuning (FT) word piece counts and overlap statistics comparing obfuscated pre-training (upper part) vs. regular pre-training (lower-part) for additional language directions.

Language Pair	Model	OPUS-Test				FLORES-devtest			
		10k	25k	50k	100k	10k	25k	50k	100k
de-id	random-init	5.6	6.6	10.1	16.0	1.8	4.2	7.1	12.5
	pb-trees	6.4	11.7	16.0	19.8	4.1	8.7	12.4	16.3
de-my	random-init	10.7	15.2	19.6	23.6	1.4	2.7	4.2	5.9
	pb-trees	12.3	18.3	24.2	28.3	2.1	4.2	6.2	7.8
id-my	random-init	11.8	16.3	18.9	n/a	1.5	2.5	3.4	n/a
	pb-trees	11.8	17.0	20.2		1.6	3.4	5.0	
id-tl	random-init	15.2	17.6	20.9	23.5	0.2	0.3	0.4	0.6
	pb-trees	16.7	18.5	21.8	24.8	0.5	0.9	1.5	2.9
id-tr	random-init	4.1	6.2	8.0	11.5	0.9	1.7	3.0	5.7
	pb-trees	4.5	8.1	12.3	16.3	1.1	3.5	6.8	10.5
my-tl	random-init	11.9	16.4	21.6	n/a	2.0	2.8	3.7	n/a
	pb-trees	12.8	19.6	27.0		2.4	4.3	5.8	
my-tr	random-init	5.1	6.5	8.0	7.7	0.2	0.4	0.3	0.3
	pb-trees	5.7	8.1	11.4	14.7	0.2	0.5	1.2	1.8
tl-tr	random-init	2.2	3.1	3.8	5.0	0.3	0.7	1.1	1.8
	pb-trees	2.0	3.5	4.9	4.9	0.4	1.0	2.1	2.1

Table 8: Translation decoding results for additional non-English-centric pairs. We report spBLEU for synthetic pre-training with pb-trees vs. fine-tuning from random initialization as a function of the fine-tuning set size.

Training Settings	
Optimizer	Adam
Learning Rate	5e-4
Weight Decay	1e-4
Criterion	label_smoothed_cross_entropy
Label Smoothing	0.1
Learning Rate Scheduler	Inverse sqrt
Warmup Updates (Pre-Training)	4000
Warmup-Updates (Fine-Tuning)	100
Max Token Number	2048
Decoding Strategy	Beam Search
Beam size	5
Max Length a	1.2
Max Length b	10

Table 9: Summary of pre-training and fine-tuning parameters for our experiments.

ACL 2023 Responsible NLP Checklist

A For every submission:

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

☑ A2. Did you discuss any potential risks of your work?
Section 8

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

☒ 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?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix C1

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?
Appendix C2

☐ 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?
No response.

☑ 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 and Appendix C

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.