SYNTHETIC PRE-TRAINING TASKS FOR NEURAL MACHINE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Pre-training is an effective technique for ensuring robust performance on a variety of machine learning tasks. It typically depends on large-scale crawled corpora that can result in toxic or biased models. Such data can also be problematic with respect to copyright, attribution, and privacy. Pre-training with synthetic tasks and data is a promising way of alleviating such concerns since no real-world information is ingested by the model. Our goal in this paper is to understand what makes for a good pre-trained model when using synthetic resources. We answer this question in the context of neural machine translation by considering two novel approaches to translation model pre-training. Our first approach studies the effect of pre-training on obfuscated data derived from a parallel corpus by mapping words to a vocabulary of 'nonsense' tokens. Our second approach explores the effect of pre-training on procedurally generated synthetic parallel data that does not depend on any real human language corpus. Our empirical evaluation on multiple language pairs shows that, to a surprising degree, the benefits of pre-training can be realized even with obfuscated or purely synthetic parallel data. In our analysis, we consider the extent to which obfuscated and synthetic pre-training techniques can be used to mitigate the issue of hallucinated model toxicity.

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), typically hundreds of thousands, tens of millions, or – more recently – even larger parallel corpora (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 & Vulić, 2019) was recently withdrawn due to a copyright infringement claim.

Our primary motivation is to understand how knowledge transfer from NMT pre-training can help to avoid or minimize the data issues described above. We seek to understand how pre-training and transfer learning impact the performance of downstream fine-tuning tasks by designing two procedural approaches to synthetic data generation. We use our synthetic data sets and tasks to conduct a comprehensive empirical evaluation of transfer learning from NMT pre-training.

Our first approach studies the extent to which the transfer benefits of pre-training can be realized with obfuscated or encrypted data. Our obfuscated corpus is derived from a real parallel corpus 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 93% of the BLEU score obtained by a model pre-trained on the original un-obfuscated parallel data. Additionally, we show that synthetic pre-training produces models that are considerably less toxic than publicly available large-scale multilingual translation models.

Our second approach explores the pre-training impact of important translation phenomena such as word alignment 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 additionally observe a reduction in model toxicity.

Our primary contributions are a comprehensive empirical evaluation and analysis of the use of synthetic pre-training tasks and data in NMT, as well as showing that synthetic data can be a promising stepping stone towards relieving the data burden in NMT as well as building 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 rely on real human languages for pre-training and therefore inherit 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 & 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 simple synthetic tasks. They find that much of the benefit of real masked language model pre-training can still be observed, even for very simple synthetic tasks. Our work in this paper represents the first empirical evaluation of synthetic pre-training for neural machine translation.

The quality of a pre-trained model should not be measured purely by performance. We should also consider trustworthiness. 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 & 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 while mitigating model toxicity. In this section, we describe two approaches to the programmatic generation of synthetic data: (1) pre-training with obfuscated parallel data that implicitly preserves language properties such as distributional frequencies, and (2) pre-training with synthetic tasks designed to encourage transfer learning of important translation properties such as long-distance reordering.

3.1 Pre-Training with 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 (e.g. German-to-English). 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 $\mathcal{O}_{t_j}$. We create our synthetic pre-training corpus by replacing each source and target word with their corresponding obfuscated nonsense tokens. This method of obfuscation can be viewed as a trivial form of encrypted training. We note that 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-to-English parallel sentence pair is shown below:

Original src	Meine zweite Bemerkung ist etwas ernsthafter.
Original trg	My second comment is rather more serious.

Obfuscating the sentence pair results in the following pair of nonsense token sequences:

Obfuscated src	wfnzc kqknd gmlfd tlieb ghzwa jdfnd engwd
Obfuscated trg	UKVFB IJODB XRWOB SZEIA AHBNB LATAA MCSDA ETFJA

It is also possible to randomly obfuscate words. We define R as the desired proportion of obfuscated tokens. Varying R allows us to explore the extent to which knowledge transfer from pre-training can be preserved with different degrees of obfuscation. With $R = 25\%$ we have:

Obfuscated src	wfnzc zweite Bemerkung ist etwas ernsthafter .
Obfuscated trg	My IJODB comment is AHBNB more serious .

3.2 Pre-Training on Synthetic Tasks and Data

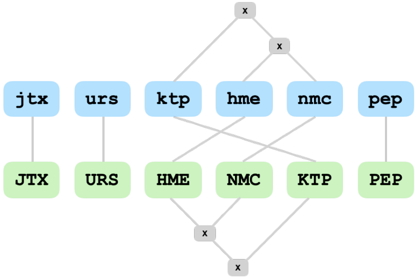
In this section, we define three completely synthetic task variants that can be used for NMT pre-training: (1) the identity operation, (2) case-mapping, and (3) permuted binary trees. All three tasks are based on a procedural data generation model and can thus be used to generate arbitrary quantities of synthetic aligned parallel data. Procedural generation of synthetic parallel sentence pairs allows for complete control over the alignments, corpus 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 tokens and $\mathcal{T}$ for target tokens. 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 $\{a, \ldots, z\}$.

3.2.1 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. Example:

source	cea qne bwr jda rnu jkq ozf dke kzl hpo
target	cea qne bwr jda rnu jkq ozf dke kzl hpo



$$
\begin{aligned}
X &\rightarrow \langle X_1\,X_2,\ X_1\,X_2\rangle\\
X &\rightarrow \langle X_1\,X_2,\ X_2\,X_1\rangle\\
X &\rightarrow \langle \texttt{jtx}\,X_1,\ \texttt{JTX}\,X_1\rangle\\
X &\rightarrow \langle \texttt{urs}\,X_1,\ \texttt{URS}\,X_1\rangle\\
X &\rightarrow \langle \texttt{ktp}\,X_1,\ X_1\,\texttt{KTP}\rangle\\
X &\rightarrow \langle \texttt{hme}\,\texttt{nmc},\ \texttt{HME}\,\texttt{NMC}\rangle\\
X &\rightarrow \langle X_1\texttt{pep},\ X_1\,\texttt{PEP}\rangle
\end{aligned}
$$

Figure 1: Example synthetic parallel sentence pair and partial derivation for the aligned permuted binary trees task. In this example, a single non-terminal node was reordered. This parallel sentence pair implies the existence of the synchronous context free grammar rules shown on the right.

3.2.2 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 $d_s = d_t = 0.15$:

source	`qdo dzz zwj iub uxj pls nsn igk mrz ojw`
target	`QDO DZZ ZWJ IUB KWP UXJ PLS NSN IGK MRZ OJW`

3.2.3 Synthetic Task 3: Aligned Permuted Binary 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:

source	`[ jtx [ [ urs [ ktp [ hme nmc ] ] ] pep ] ]`
target	`[ 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 1 shows the token-level alignment and reordering operations encoded in this parallel sentence pair, together with the implied SCFG rules that could be used to derive it.

4 Experimental Framework

4.1 Dataset Preparation

For English-centric translation directions, we use fine-tuning data sets similar to those described in Aji et al. (2020). For Myanmar-English, our 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 divide the original corpus into distinct train (90%), dev (5%), and test (5%) sets. For Turkish-English, we use the WMT 2017 News Translation Task (Yepes et al., 2017) data. The training set includes 207.7k parallel sentence pairs. We use the WMT `newsdev2016` set for validation, and report results on `newstest2017` set.

For non-English-centric translation directions, we simulate low-resource translation conditions by sampling data from OPUS NLP (Tiedemann, 2012)[1]. 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 of these language pairs, 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 languages.

All pre-training and fine-tuning parallel data is tokenized with the `sentencepiece` (Kudo & Richardson, 2018) model from XLMR (Conneau et al., 2019). The dictionary contains 250k word pieces and covers 100 languages. We score our translations using `sentencepiece` BLEU (Papineni et al., 2002) (spBLEU) in order to facilitate comparison with large-scale multilingual models such as FLORES-101 (Goyal et al., 2022), and to avoid the need for custom pre-processing and post-processing for individual languages with unusual scripts and/or complex morphology.

Our experiments consist of a pre-training stage followed by a fine-tuning stage. We use `fairseq` (Ott et al., 2019) to train transformer base (Vaswani et al., 2017) sequence-to-sequence translation models with the Adam Kingma & 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

We use data from the WMT 2014 News Translation Task for our obfuscated pre-training experiments. We randomly sample 1 million sentence pairs for use as fine-tuning data. The remaining 3.5 million pairs are used for pre-training. We further sub-sample the fine-tuning data to create additional fine-tuning sets of size 20k, 50k, 100k and 500k. We vary the obfuscation ratio R from 0% to 100% in 25% increments. We test our models using the test sets described in Section 4.1.

Matched Language Condition 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 2 show that the highest BLEU scores are obtained by this baseline, confirming the transfer benefit of pre-training. We also evaluate the effect of training from scratch on only the fine-tuning data. For small fine-tuning sets, the BLEU scores are low which further emphasizes the importance of pre-training.

Models pre-trained on obfuscated data have higher BLEU scores than their corresponding from-scratch counterparts. This implies that obfuscated pre-training can still be useful, even when 100% of the tokens are encrypted. However, for a given fine-tuning set size, increasing the pre-training obfuscation ratio R makes the downstream task more challenging. For example, increasing the obfuscation ratio R from 75% to 100% greatly reduces BLEU when the fine-tuning data size is small. For larger fine-tuning set sizes, however, the effect of varying the obfuscation ratio is reduced. We note the surprising observation that high BLEU scores can still be obtained even at an obfuscation ratio of 75%. This suggests that even a relatively small proportion of the original real-world data can still provide the majority of the benefit of large-scale regular pre-training.

Unmatched Language Condition In this section, we investigate whether the transfer benefit of obfuscated pre-training depends on matching the pre-training and fine-tuning languages. We measure the effectiveness of obfuscated pre-training in simulated low-resource conditions using the language pairs and data described in Section 4.1. We evaluate the extent to which the benefits of pre-training on regular parallel data can be replicated using obfuscation.

[1] `http://opus.nlpl.eu`

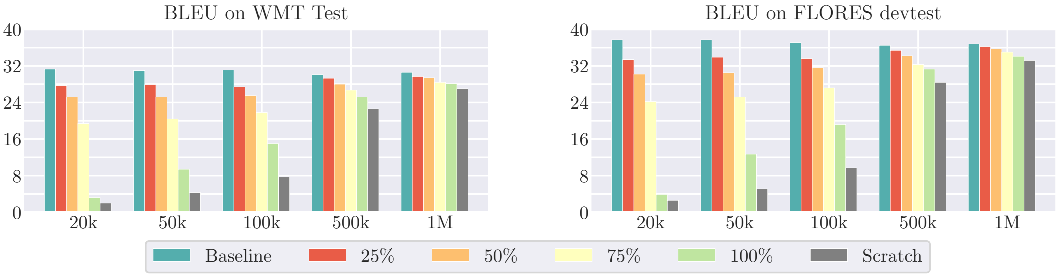


Figure 2: Translation decoding results for regular parallel corpus baseline vs. obfuscated pre-training as a function of fine-tuning set size and obfuscation ratio R.

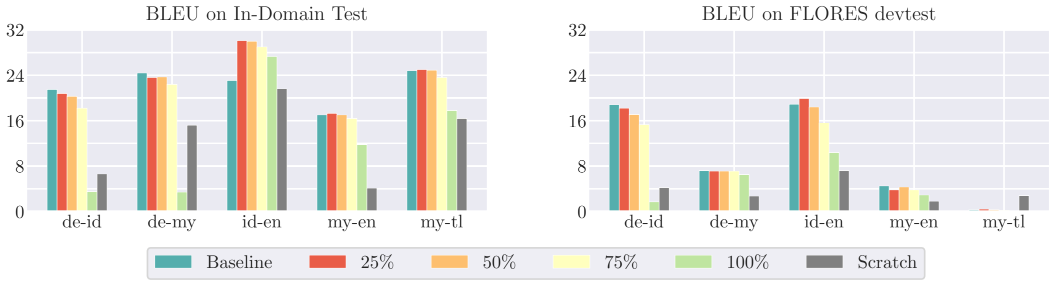


Figure 3: Translation decoding results for in-domain test sets (described in Section 4.1) and FLORES-devtest as a function of the obfuscation ratio R. The fine-tuning data size is 20k for `de-en` and 25k for all other languages.

We show the effect of fine-tuning for specific language pairs in Figure 3. In these experiments, German-to-English data with various obfuscation ratios R was used for pre-training. We observe that the BLEU scores for obfuscated pre-training are close to or exceed (i.e. `id-en`) those of the baseline, even with an obfuscation ratio as high as 75%. When the fine-tuning language matches the pre-training language, obfuscation reduces the BLEU score on the downstream task. But when the language pairs are unmatched, real-world German-to-English parallel data has less relevance to the downstream task so obfuscated pre-training is closer to the baseline performance. We note that `id-en` shows the clearest evidence of transfer learning. We believe the reasons are (i) both Indonesian and English share the same alphabet with the letters used to construct nonsense tokens, while Burmese does not, and (ii) the pre-training and fine-tuning target languages are both English which allows for powerful transfer to the downstream task. It also implies that training a strong decoder is most important for good downstream task performance. In the end, we conclude that word identity may not be such an important component in a good pre-trained model, since even with an obfuscation ratio of 75% we still see much of the transfer benefit.

4.3 Pre-Training with Synthetic Data

We pre-train sequence-to-sequence transformer (Vaswani et al., 2017) models using two million sentence pairs of synthetic parallel data. Separate synthetic training sets were generated for each of the three task variants described in Section 3.2. Additional sets of 4000 synthetic pairs were generated as validation data and for model selection. Our pre-trained models are entirely synthetic since no human language parallel corpus was used. Each pre-trained model is subsequently fine-tuned with real parallel data for a specific language pair: Myanmar-to-English (`my-en`), Indonesian-to-English (`id-en`), and Turkish-to-English (`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 any pre-training – i.e. a randomly initialized model trained using only the fine-tuning data.

Our first observation is that synthetic pre-training with the identity operation task (Section 3.2.1) 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:

Table 1: Translation decoding results (spBLEU) for three synthetic pre-training variants and a fine-tuning from random initialization baseline (English-centric language pairs).

Pair	Model	Test	FLORES	Pair	Model	Test	FLORES
my-en	random-init	4.1	1.8	en-my	random-init	16.2	6.3
	identity	3.2	1.1		identity	12.7	4.5
	case-map	6.7	1.6		case-map	16.4	6.0
	pb-trees	11.4	2.5		pb-trees	18.9	7.0
id-en	random-init	18.2	7.2	en-id	random-init	19.1	8.3
	identity	16.8	7.6		identity	18.1	9.7
	case-map	21.8	12.1		case-map	22.9	13.8
	pb-trees	23.1	12.2		pb-trees	23.8	14.4
tr-en	random-init	14.7	17.7	en-tr	random-init	17.0	16.4
	identity	12.4	13.8		identity	13.8	13.5
	case-map	13.4	15.1		case-map	15.6	15.2
	pb-trees	14.4	16.9		pb-trees	16.6	16.3

Table 2: German-to-Indonesian and German-to-Myanmar spBLEU scores for synthetic pre-training with permuted binary trees vs. fine-tuning from random initialization by fine-tuning set size.

		OPUS-Test				FLORES-devtest			
Language Pair	Model	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

a simple copy operation from source to target with identical lengths. Pre-training with the case-mapping synthetic task (Section 3.2.2) and deletion probability $d_s = d_t = 0$ improves the scores substantially, 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 for both Myanmar-to-English and Indonesian-to-English. Our best performing synthetic task is the one that generates aligned synthetic parallel data using permuted binary trees (Section 3.2.3) and a node reordering probability $r = 0.15$. The pb-trees 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 synthetic pre-training tasks. The fine-tuning data for this language pair is an order of magnitude larger than that of the other language pairs. As the fine-tuning data size increases, the necessity of transfer learning from pre-training diminishes.

We evaluate the strongest of our synthetic pre-training tasks, pb-trees, on additional non-English-centric language pairs. Table 2 shows spBLEU decoding results for German-to-Indonesian and German-to-Myanmar. We compare fine-tuning from a randomly initialized model vs. fine-tuning on top of a synthetically pre-trained model using the pb-trees task. We compare performance over a range of different fine-tuning set sizes. Table 5 in Appendix A.3 shows fine-tuning results for six additional non-English-centric language pairs. 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 What Knowledge is Transferred from Synthetic Pre-Training?

In this section, we discuss what kind of useful representations are actually learned by the model when pre-training with synthetic tasks and data. Our empirical study in Section 4.3 has shown

Table 3: Tokenized Pre-Training (PT) and Fine-Tuning (FT) unique word piece counts and overlap statistics. Language indicators 'lc' and 'uc' denote lowercase and uppercase synthetic tokens.

Language Pair	**PT/FT Languages**	$\lvert V_{PT}\rvert$	$\lvert V_{FT}\rvert$	$\lvert V_{PT} \cap V_{FT}\rvert$
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

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 3, 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 that occur in the synthetic pre-training data and those that occur in the fine-tuning data in the right-most column of Table 3. We observe low word piece overlap for all three pairs. 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 7). 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 Analysis of Translation Quality and Toxicity

Our experiments in Section 4 have shown synthetic pre-training to be a promising approach for NMT. In understanding what makes for a good pre-trained model, we consider not only the translation quality but also whether the model can be trusted.

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. We use the FLORES Toxicity-200[2] word lists to calculate the toxicity rate of translations produced by a model. The lists cover 200 languages and contain frequently used profanities, insults, hate speech terms, pornographic terms, etc. 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 transitions of in-domain test sets and a larger set of 100k monolingual sentences sampled from CC-100 (Wenzek et al., 2020; Conneau et al., 2019). We compare BLEU scores and toxicity for various models in Table 4.

We first compare the BLEU score of synthetically pre-trained models to that of the multilingual translation models FLORES-101 (615M parameters) M2M-100 (1.2B parameters). We note that obfuscated pre-training has higher BLEU than M2M-100 and FLORES-101 for de-my, id-en, my-en, and my-tl. Synthetic pre-training with permuted binary trees also results in higher BLEU

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

Table 4: BLEU scores and toxicity rates for various models and language pairs. Green shading denotes higher BLEU scores and lower toxicity rates, while red denotes the inverse.

Model	de-en		de-id		de-my	
	BLEU	**Toxicity**	**BLEU**	**Toxicity**	**BLEU**	**Toxicity**
Scratch	2.9	0.02	6.6	0.68	15.2	0.01
Baseline	31.3	0.19	21.5	0.62	24.4	0.03
Pb-Tree	6.7	0.03	11.7	0.45	12.3	0.01
Obfuscation	19.4	0.16	18.2	0.34	22.4	0.01
M2M-1.2B	34.6	0.24	32.9	0.68	9.1	0.03
FLORES-615M	29.6	0.18	30	0.63	12.3	0.03
Model	id-en		my-en		my-tl	
	BLEU	**Toxicity**	**BLEU**	**Toxicity**	**BLEU**	**Toxicity**
Scratch	18.2	0.05	4.1	0.02	16.4	0.04
Baseline	23.1	0.21	17	0.07	24.8	0.07
Pb-Tree	23.1	0.10	11.4	0.01	20.7	0.02
Obfuscation	29	0.11	16.4	0.08	23.6	0.04
M2M-1.2B	30.2	0.28	1.8	0.15	14.2	0.06
FLORES-615M	26	0.23	4.6	0.18	12.8	0.08

scores for de-my, my-en, and my-tl than M2M-100 and FLORES-101. It should be noted that some of these language pairs represent zero-shot directions for M2M-100. The BLEU scores show that for some language pairs, large-scale multilingual pre-training offers only limited transfer benefits. Our results show that transfer learning from synthetic pre-training has the potential to help to improve translation robustness for under-represented language pairs in multilingual models.

Analyzing toxicity, we observe catastrophic mistranslations in all models, but less frequently when training from scratch in most cases. This is because the fine-tuning data contains very little toxic content. However, the BLEU score when training from scratch is very low. We see that the baseline, FLORES-101, and M2M-100 models all exhibit toxicity, since they were all pre-trained on real-world corpora that can include toxic content. Our results show that synthetic pre-training can produce models with good BLEU scores while reducing catastrophic mistranslations. Obfuscated pre-training has slightly higher toxicity than synthetic pre-training with permuted binary trees. This may indicate that patterns in the data can still trigger toxic terms, even after the words have been obfuscated. We include additional toxicity results and discussion in Appendix A.1.

6 Conclusion

Our empirical evaluation of two different approaches to synthetic pre-training for NMT has led to the surprising conclusion that the transfer benefits of pre-training still apply even when pre-training on obfuscated or entirely synthetic data. That synthetic data can also mitigate model toxicity, especially compared to large-scale multilingual translation models trained on web-scale crawled corpora, is an especially promising feature of synthetic pre-training techniques.

In our analysis, we have shared our insights and understanding of what kinds of knowledge transfer make for a good pre-trained model. We firmly believe that synthetic data augmentation techniques based on synthetic tasks and procedurally generated data represent very promising first steps towards addressing some of the most pressing pre-training data concerns, and can help in satisfying the goal of achieving efficient, accurate, and trustworthy NMT.

In future work we plan to further explore synthetic pre-training by considering more sophisticated parameterizations of our data generation models. For example, we could add an explicit MT fertility model (Brown et al., 1993) or context-sensitivity to the reordering model (Chiang, 2007). There is also potential for directly optimizing the parameters of the data generation model in order to maximize performance on a specific downstream fine-tuning task.

REFERENCES

Željko Agić and Ivan Vulić. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3204–3210, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1310. URL `https://aclanthology.org/P19-1310`.

Roee Aharoni, Melvin Johnson, and Orhan Firat. 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)*, pp. 3874–3884, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1388. URL `https://aclanthology.org/N19-1388`.

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

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19 (2):263–311, 1993. URL `https://aclanthology.org/J93-2003`.

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. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020. URL `https://arxiv.org/abs/2005.14165`.

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

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

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

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

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116, 2019. URL `http://arxiv.org/abs/1911.02116`.

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

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

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

Chenchen Ding, Masao Utiyama, and Eiichiro Sumita. 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, 2018.

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

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

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

Paweł Kamocki and Jim O'Regan. Privacy issues in online machine translation services-european perspective. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 4458–4462, 2016.

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

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. Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72, 01 2022. ISSN 2307-387X. doi: 10.1162/tacl_a_00447. URL `https://doi.org/10.1162/tacl_a_00447`.

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

Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*, 2018.

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

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

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *CoRR*, abs/2001.08210, 2020. URL `https://arxiv.org/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. No language left behind: Scaling human-centered machine translation, 2022. URL `https://arxiv.org/abs/2207.04672`.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-6319.

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

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

Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. *arXiv preprint arXiv:2104.06644*, 2021.

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

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

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

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 4003–4012, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://www.aclweb.org/anthology/2020.lrec-1.494.

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

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

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1568–1575, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1163. URL https://aclanthology.org/D16-1163.

A Supplementary Results

A.1 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 6. 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.2 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 6 and other language directions `id-en`, `my-tn`, and `tr-en` in Table 8. 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 8. 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.3 Synthetic Pre-Training: Additional Language Pairs

Table 5 shows translation decoding results (spBLEU) 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.

Table 5: Translation decoding results for additional non-English-centric language pairs. We show spBLEU scores for synthetic pre-training with permuted binary trees vs. fine-tuning from random initialization as a function of the fine-tuning set size.

		OPUS-Test				**FLORES-devtest**			
Language Pair	**Model**	10k	25k	50k	100k	10k	25k	50k	100k
`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 6: 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.

Fine-tune Size	**Obfuscated Level**				
	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-tune Size	**Obfuscated Level**				
	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

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 scoring signature is shown below:

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

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.

Table 7: Token overlap in obfuscation experiments (de-en): obfuscation pre-training v.s. finetuning (upper part) and normal pre-training v.s. fine-tuning (lower part).

Systems	FT size	PT/FT Language	$\|V_{PT}\|$	$\|V_{FT}\|$	$\|V_{PT} \cap V_{FT}\|$
With Obfuscation 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
With Normal 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 8: Token overlap in obfuscation experiments (other language directions): obfuscation pre-training v.s. finetuning (upper part) and normal pre-training v.s. fine-tuning (lower part).

Systmes	Langauge pair	PT/FT Language	$\|V_{PT}\|$	$\|V_{FT}\|$	$\|V_{PT} \cap V_{FT}\|$
With Obfuscation 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
With Normal 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

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.