Harnessing Multilinguality in Unsupervised Machine Translation for Rare Languages

Xavier Garcia and **Aditya Siddhant** and **Orhan Firat** and **Ankur P. Parikh**

Google Research
Mountain View
California
{xgarcia,adisid,orhanf,aparikh}@google.com

Abstract

Unsupervised translation has reached impressive performance on resource-rich language pairs such as English-French and English-German. However, early studies have shown that in more realistic settings involving low-resource, rare languages, unsupervised translation performs poorly, achieving less than 3.0 BLEU. In this work, we show that *multilinguality* is critical to making unsupervised systems practical for low-resource settings. In particular, we present a single model for 5 low-resource languages (Gujarati, Kazakh, Nepali, Sinhala, and Turkish) to and from English directions, which leverages monolingual and auxiliary parallel data from other high-resource language pairs via a three-stage training scheme. We outperform all current state-of-the-art unsupervised baselines for these languages, achieving gains of up to 14.4 BLEU. Additionally, we outperform strong supervised baselines for various language pairs as well as match the performance of the current state-of-the-art supervised model for Ne→En. We conduct a series of ablation studies to establish the robustness of our model under different degrees of data quality, as well as to analyze the factors which led to the superior performance of the proposed approach over traditional unsupervised models.

1 Introduction

Neural machine translation systems (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Bahdanau et al., 2015; Wu et al., 2016) have demonstrated state-of-the-art results for a diverse set of language pairs when given large amounts of relevant parallel data. However, given the prohibitive nature of such a requirement for low-resource language pairs, there has been a growing interest in unsupervised machine translation (Ravi and Knight, 2011) and its neural counterpart, unsupervised neural machine translation (UNMT) (Lample et al., 2018a; Artetxe et al., 2018), which leverage only monolingual source and target corpora for learning. Bilingual unsupervised systems (Lample and Conneau, 2019; Artetxe et al., 2019; Ren et al., 2019; Li et al., 2020a) have achieved surprisingly strong results on high-resource language pairs such as English-French and English-German.

However, these works only evaluate on high-resource language pairs with high-quality data, which are not realistic scenarios where UNMT would be utilized. Rather, the practical potential of UNMT is in low-resource, rare languages that may not only lack parallel data but also have a shortage of high-quality monolingual data. For instance, Romanian (a typical evaluation language for unsupervised methods) has 21 million lines of high-quality in-domain monolingual data provided by WMT. In contrast, for an actual low-resource language, Gujarati, WMT only provides 500 thousand lines of monolingual data (in news domain) and an additional 3.7 million lines of monolingual data from Common Crawl (noisy, general-domain).

Given the comparably sterile setups UNMT has been studied in, recent works have questioned the usefulness of UNMT when applied to more realistic low-resource settings. Kim et al. (2020) report BLEU scores of less than 3.0 on low-resource pairs and Marchisio et al. (2020) also report dramatic degradation under domain shift.

However, the negative results shown by the work above only study bilingual unsupervised systems and do not consider *multilinguality*, which has been well explored in supervised, zero-resource and zero-shot settings (Johnson et al., 2017; Firat et al., 2016a,b; Chen et al., 2017; Neubig and Hu, 2018; Gu et al., 2018; Liu et al., 2020; Ren et al., 2018; Zoph et al., 2016) to improve performance for low-resource languages. The goal of this work is to study if multilinguality can help UNMT be more robust in the low-resource, rare language setting.

In our setup (Figure 1), we have a single model

1126

for 5 target low-resource unsupervised directions (that are not associated with any parallel data): Gujarati, Kazakh, Nepali, Sinhala, and Turkish. These languages are chosen to be studied for a variety of reasons (discussed in §3) and have been of particular challenge to unsupervised systems. In our approach, as shown in Figure 1, we also leverage auxiliary data from a set of higher resource languages: Russian, Chinese, Hindi, Arabic, Tamil, and Telugu. These higher resource languages not only possess significant amounts of monolingual data but also auxiliary parallel data with English that we leverage to improve the performance of the target unsupervised directions[1].

Existing work on multilingual unsupervised translation (Liu et al., 2020; Garcia et al., 2020; Li et al., 2020b; Bai et al., 2020), which also uses auxiliary parallel data, employs a two-stage training scheme consisting of pre-training with noisy reconstruction objectives and fine-tuning with on-the-fly (iterative) back-translation and cross-translation terms (§4). We show this leads to sub-optimal performance for low-resource pairs and propose an additional intermediate training stage in our approach. Our key insight is that pre-training typically results in high X→En (to English) performance but poor En→X (from English) results, which makes fine-tuning unstable. Thus, after pre-training, we propose an intermediate training stage that leverages offline back-translation (Sennrich et al., 2016) to generate synthetic data from the X→En direction to boost En→X accuracy.

Our final results show that our approach outperforms a variety of supervised and unsupervised baselines, including the current state-of-the-art supervised model for the Ne→En language pair. Additionally, we perform a series of experimental studies to analyze the factors that affect the performance of the proposed approach, as well as the performance in data-starved settings and settings where we only have access to noisy, multi-domain monolingual data.

2   Related work

Multilinguality has been extensively studied in the supervised literature and has been applied to the related problem of zero-shot translation (Johnson et al., 2017; Firat et al., 2016a; Arivazhagan et al.,
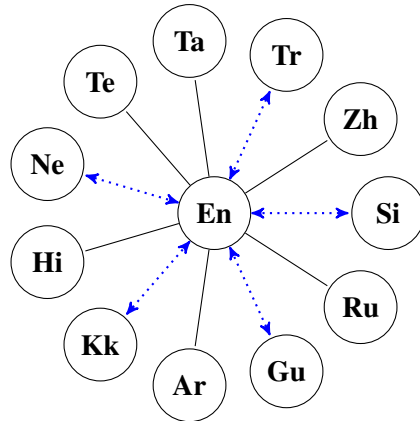

Figure 1: A pictorial depiction of our setup. The dashed edge indicates the target unsupervised language pairs that lack parallel training data. Full edges indicate the existence of parallel training data.

2019a; Al-Shedivat and Parikh, 2019). Zero-shot translation concerns the case where direct (source, target) parallel data is lacking but there is parallel data via a common pivot language to both the source and the target. For example, in Figure 1, Ru↔Zh and Hi↔Te would be zero-shot directions.

In contrast, a defining characteristic of the multilingual UNMT setup is that the source and target are disconnected in the graph and one of the languages is not associated with any parallel data with English or otherwise. En↔Gu or En↔Kk are such example pairs as shown in Figure 1.

Recently Guzmán et al. (2019); Liu et al. (2020) showed some initial results on multilingual unsupervised translation in the low-resource setting. They tune language-specific models and employ a standard two-stage training scheme (Lample and Conneau, 2019), or in the case of Liu et al. (2020) directly fine-tuning on a related language pair (e.g. Hi→En) and then test on the target X→En pair (e.g. Gu→En). In contrast our approach trains one model for all the language pairs targetted and employs a three stage training scheme that leverages synthetic parallel data via offline back-translation.

Offline backtranslation (Sennrich et al., 2016) was originally used for unsupervised translation (Lample et al., 2018b; Artetxe et al., 2019), especially with phrase-based systems.

2.1   Terminology

There is some disagreement on the definition of multilingual unsupervised machine translation, which we believe arises from extrapolating unsu-

---

[1]This makes our setting considerably more challenging than the zero-shot/zero resource setting. See §2 and §2.1 for a discussion.

	Domain	En	Tr News	Kk News	Gu News	Ne Wiki	Si Wiki	Te Wiki	Ta Wiki	Hi IITB	Ru UN	Ar UN	Zh UN
Monolingual	News	233M	17M	1.8M	530K	-	-	2.5M	2.3M	32.6M	93.8M	9.2M	4.7M
	Wikipedia	-	-	-	384K	92k	155k	-	-	-	-	-	22.7M
	Crawled	-	-	7.1M	3.7M	3.5M	5.1M	-	-	-	-	-	-
Auxiliary parallel (w/ English)	Mixed	-	205k	225k	10k	564k	647k	290K	350K	1.5M	23.2M	9.2M	15.8M
In-domain (%)		-	100%	20.2%	11.4%	2.0%	2.9%	-	-	-	-	-	-

Table 1: The amount and domain of the data used in these experiments. For the unsupervised language pairs, we additionally included the domain of the development and test sets. For Arabic, we took the 18.4M samples from the UN Corpus and divided it in two, treating one half of it as unpaired monolingual data. We include the amount of parallel data for the unsupervised language pairs, which is only utilized for our in-house supervised baseline.

pervised translation to multiple languages. In the case of only two languages, the definition is clear: unsupervised machine translation consists of the case where there is no parallel data between the source and target languages. However, in a setting with multiple languages, there are multiple scenarios which satisfy this condition. More explicitly, suppose that we want to translate between languages $\mathcal{X}$ and $\mathcal{Y}$ and we have access to data from another language $\mathcal{Z}$. Then, we have three possible scenarios:

- We possess parallel data for $(\mathcal{X}, \mathcal{Z})$ and $(\mathcal{Z}, \mathcal{Y})$ which would permit a 2-step supervised baseline via the pivot. Existing literature (Johnson et al., 2017; Firat et al., 2016b) has used the term "zero-shot" and "zero-resource" to refer specifically to this setup.

- We have parallel data for $(\mathcal{X}, \mathcal{Z})$ but only monolingual data in $\mathcal{Y}$, as considered in (Li et al., 2020b; Liu et al., 2020; Garcia et al., 2020; Bai et al., 2020; Guzmán et al., 2019; Artetxe et al., 2020). Note that the pivot-based baseline above is not possible in this setup.

- We do not have any parallel data among any of the language pairs, as considered in (Liu et al., 2020; Sun et al., 2020).

We believe the first setting is not particularly suited for the case where either $\mathcal{X}$ or $\mathcal{Y}$ are true low-resource languages (or extremely low-resource languages), since it is unlikely that these languages possess any parallel data with any other language. On the other hand, we usually assume that one of these languages is English and we can commonly find large amounts of parallel data for English with other high-resource auxiliary languages. For these reasons, we focus on the second setting for the rest of this work.

Arguably, the existence of the auxiliary parallel data provides some notion of indirect supervision that is not present when only utilizing monolingual data. However, this signal is weaker than the one encountered in the zero-shot setting, since it precludes the 2-step supervised baseline. As a result, recent work (Artetxe et al., 2020; Guzmán et al., 2019; Garcia et al., 2020; Liu et al., 2020) has also opted to use the term "unsupervised". We too follow this convention and use this terminology, but we emphasize that independent of notation, our goal is to study the setting where *only* the (extremely) low-resource languages of interest possess *no* parallel data, whether with English or otherwise.

3 Choice of languages

The vast majority of works in UNMT (multilingual or otherwise) have focused on traditionally high-resource languages, such as French and German. While certain works simulate this setting by using only a smaller subset of the available monolingual data, such settings neglect common properties of true low-resource, rare languages: little-to-no lexical overlap with English and noisy data sources coming from multiple domains. Given the multifaceted nature of what it means to be a low-resource language, we have chosen a set of languages with many of these characteristics. We give a detailed account of the available data in Table 1.

Target unsupervised directions: We select Turkish (Tr), Gujarati (Gu), and Kazakh (Kk) from WMT . The latter two possess much smaller amounts of data than most language pairs considered for UNMT e.g. French or German. In order to vary the domain of our test sets, we additionally include Nepali (Ne) and Sinhala (Si) from the recently-introduced FLoRes dataset (Guzmán et al., 2019), as the test sets for these languages are drawn from Wikipedia instead of news. Not only do these languages possess monolingual data in amounts comparable to the low-resource languages from WMT, the subset of in-domain monolingual data for both languages make up less than 5% of

the available monolingual data of each language.

Auxiliary languages: To choose our auxiliary languages that contain both monolingual data and parallel data with English, we took into account linguistic diversity, size, and relatedness to the target directions. Russian shares the same alphabet with Kazakh, and Hindi, Telugu, and Tamil are related to Gujarai, Nepali and Sinhala. Chinese, while not specifically related to any of the target language, is high resource and considerably different in structure from the other languages.

4   Background

For a given language pair $(\texttt{X}, \texttt{Y})$ of languages $\texttt{X}$ and $\texttt{Y}$, we possess *monolingual* datasets $\mathcal{D}_{\texttt{X}}$ and $\mathcal{D}_{\texttt{Y}}$, consisting of unpaired sentences of each language.

Neural machine translation In supervised neural machine translation, we have access to a *parallel* dataset $\mathcal{D}_{\texttt{X} \times \texttt{Z}}$ consisting of translation pairs (x, z). We then train a model by utilizing the *cross-entropy* objective:

$$\mathcal{L}_{\text{cross-entropy}}(x, y) = -\log p_\theta(y|x)$$

where p_θ is our translation model. We further assume p_θ follows the encoder-decoder paradigm, where there exists an encoder Enc_θ which converts x into a variable-length representation which is passed to a decoder $p_\theta(y|x) := p_\theta(y|\text{Enc}_\theta(x))$.

Unsupervised machine translation In this setup, we no longer possess $\mathcal{D}_{\texttt{X} \times \texttt{Y}}$. Nevertheless, we may possess auxiliary parallel datasets such as $\mathcal{D}_{\texttt{X} \times \texttt{Z}}$ for some language $\texttt{Z}$, but we enforce the constraint that we do not have access to analogous dataset $\mathcal{D}_{\texttt{Y} \times \texttt{Z}}$. Current state-of-the-art UNMT models divide their training procedure into two phases: *i)* the *pre-training* phase, in which an initial translation model is learned through a combination of language modeling or noisy reconstruction objectives (Song et al., 2019; Lewis et al., 2019; Lample and Conneau, 2019) applied to the monolingual data; *ii)* the *fine-tuning* phase, which resumes training the translation model built from the pre-training phase with a new set of objectives, typically centered around iterative back-translation i.e. penalizing a model's error in round-trip translations. We outline the objectives below:

Pre-training objectives We use the *MASS* objective (Song et al., 2019), which consists of masking[2]

a contiguous segment of the input and penalizing errors in the reconstruction of the masked segment. If we denote the masking operation by MASK, then we write the objective as follows:

$$\mathcal{L}_{\text{MASS}}(x) = -\log p_\theta(x|\texttt{MASK}(x), l_x)$$

where l_x denotes the language indicator of example x. We also use cross-entropy on the available auxiliary parallel data.

Fine-tuning objectives We use *on-the-fly back-translation*, which we write explicitly as:

$$\mathcal{L}_{\text{back-translation}}(x, l_y) = -\log p_\theta(x|\tilde{y}(x), l_x)$$

where $\tilde{y}(x) = \text{argmax}_y p_\theta(y|x, l_y)$ and we apply a stop-gradient to $\tilde{y}(x)$. Computing the mode $\tilde{y}(x)$ of $p_\theta(\cdot|x, l_y)$ is intractable, so we approximate this quantity with a greedy decoding procedure. We also utilize cross-entropy, coupled with *cross-translation* (Garcia et al., 2020; Li et al., 2020b; Xu et al., 2019; Bai et al., 2020), which ensures cross-lingual consistency:

$$\mathcal{L}_{\text{cross-translation}}(x, y, l_z) = -\log p_\theta(y|\tilde{z}(x), l_y)$$

where $\tilde{z}(x) = \text{argmax}_z p_\theta(z|x, l_z)$.

5   Method

For the rest of this work, we assume that we want to translate between English ($\texttt{En}$) and some low-resource languages which we denote by $\texttt{X}$. In our early experiments, we found that proceeding to the fine-tuning stage immediately after pre-training with MASS provided sub-optimal results (see §7.2), so we introduced an intermediate stage which leverages synthetic data to improve performance. This yields a total of three stages, which we describe below.

5.1   First stage of training

In the first stage, we leverage monolingual and auxiliary parallel data, using the MASS and cross-entropy objectives on each type of dataset respectively. We describe the full procedure in Algorithm 1.

---

[2]We choose a starting index of less than half the length l of the input and replace the next $l/2$ tokens with a [MASK]

token. The starting index is randomly chosen to be 0 or $l/2$ with 20% chance for either scenario otherwise it is sampled uniformly at random.

Algorithm 1 STAGE 1 & 2

Input: Datasets $\mathfrak{D}$, number of steps N, parameterized family of translation models p_θ

1: Initialize $\theta \leftarrow \theta_0$.
2: **for** step in 1, 2, 3, ..., N **do**
3:     Choose dataset D at random from $\mathfrak{D}$.
4:     **if** D consists of monolingual data **then**
5:         Sample batch x from D.
6:         MASS Loss: ml $\leftarrow \mathcal{L}_{\text{MASS}}(x)$.
7:         Update: $\theta \leftarrow$ optimizer_update(ml, θ).
8:     **else if** D consists of auxiliary parallel data **then**
9:         Sample batch (x, z) from D.
10:        tl $\leftarrow \mathcal{L}_{\text{cross-entropy}}(x, z) + \mathcal{L}_{\text{cross-entropy}}(z, x)$.
11:        Update: $\theta \leftarrow$ optimizer_update(tl, θ).
12:     **end if**
13: **end for**

5.2 Second stage of training

Once we have completed the first stage, we will have produced an initial model capable of generating high-quality X→En (to English) translations for all of the low-resource pairs we consider, also known as many-to-one setup in multilingual NMT (Johnson et al., 2017). Unfortunately, the model does not reach that level of performance for the En→X translation directions, generating very low-quality translations into these low-resource languages. Note that, this phenomenon is ubiquitously observed in multilingual models (Firat et al., 2016a; Johnson et al., 2017; Aharoni et al., 2019). This abysmal performance could have dire consequences in the fine-tuning stage, since both on-the-fly back-translation and cross-translation rely heavily on intermediate translations. We verify that this is in fact the case in §7.2.

Instead, we exploit the strong X→En performance by translating subsets[3] of the monolingual data of the low-resource languages using our initial model and treat the result as pseudo-parallel datasets for the language pairs En→X. More explicitly, given a sentence x from a low-resource language, we generate an English translation $\tilde{y}_{\text{En}}$ with our initial model and create a synthetic translation-pair $(\tilde{y}_{\text{En}}, x)$. We refer to this procedure as *offline back-translation* (Sennrich et al., 2015). We add these datasets to our collection of auxiliary parallel corpora and repeat the training procedure from the first stage (Algorithm 1), starting from the last checkpoint. Note that, while offline back-translated (synthetic) data is commonly used for zero-resource translation (Firat et al., 2016b; Chen et al., 2017), it is

worth emphasizing the difference here again, that in the configuration studied in this paper, we do not assume the existence of any parallel data between En↔X, which is exploited by such methods.

Upon completion, we run the procedure a second time, with a new subset of synthetic data of twice the size for the En→X pairs. Furthermore, since the translations from English have improved, we take disjoint subsets[4] of the English monolingual data and generate corpora of synthetic X→En translation pairs that we also include in the second run of our procedure.

5.3 Third stage of training

For the third and final stage of training, we use back-translation of the monolingual data and cross-translation[5] on the auxiliary parallel data. We also leverage the synthetic data through the cross-entropy objective. We present the procedure in detail under Algorithm 2.

6 Main experiment

In this section, we describe the details of our main experiment. As indicated in Figure 1, we consider five languages (Nepali, Sinhala, Gujarati, Kazakh, Turkish) as the target unsupervised language pairs with English. We leverage auxiliary parallel data from six higher-resource languages (Chinese, Russian, Arabic, Hindi, Telugu, Tamil) with English. The domains and counts for the datasets considered can be found in Table 1 and a more detailed discussion on the source of the data and the preprocessing steps can be found in the Appendix. In the following subsections, we provide detailed descriptions of the model configurations, training parameters, evaluation and discuss results of our main experiment.

6.1 Datasets and preprocessing

We draw most of our data from WMT. The monolingual data comes from News Crawl[6] when available. For all the unsupervised pairs except Turkish, we supplement the News Crawl datasets with monolingual data from Common Crawl and Wikipedia[7].

---

[3]We utilize 10% of the monolingual data for each low-resource language.

[4]1 million lines of English per low-resource language.

[5]For Nepali, Sinhala and Gujarati, we use Hindi as the pivot language. For Turkish, we use Arabic and for Kazakh, we use Russian.

[6]http://data.statmt.org/news-crawl/

[7]We used the monolingual data available from https://github.com/facebookresearch/flores for Nepali and Sinhala in order to avoid any data leakage from the test sets.

	Model	newstest2019 Gu ↔ En		newstest2019 Kk ↔ En		newstest2017 Tr ↔ En	
No parallel data	Kim et al. (2020)	0.6	0.6	0.8	2.0	-	-
No parallel data for{Gu,Kk,Tr}	*Stage 1* (Ours)	4.4	19.3	3.9	14.8	8.4	15.9
	Stage 2 (Ours)	**16.4**	20.4	9.9	15.6	**20.0**	**20.5**
	Stage 3 (Ours)	**16.4**	**22.2**	**10.4**	**16.4**	19.8	19.9
With parallel data for{Gu,Kk,Tr}	*Mult. MT Baseline* (Ours)	<u>15.5</u>	<u>19.3</u>	<u>9.5</u>	<u>15.1</u>	<u>18.1</u>	22.0
	mBART (Liu et al., 2020)	0.1	0.3	2.5	7.4	17.8	<u>22.5</u>

Table 2: BLEU scores of various supervised and unsupervised models on the WMT *newstest* sets. The bolded numbers are the best unsupervised scores and the underlined numbers represent the best supervised scores. For any X↔Y language pair, the X→Y translation results are listed under each Y column, and vice-versa.

	Model	FLoRes devtest Ne ↔ En		FLoRes devtest Si ↔ En	
No parallel data	Guzmán et al. (2019)	0.1	0.5	0.1	0.1
No parallel data with{Ne,Si}	Liu et al. (2020)	-	17.9	-	9.0
	Guzmán et al. (2019)	8.3	18.3	0.1	0.1
	Stage 1 (Ours)	3.3	18.3	1.4	11.5
	Stage 2 (Ours)	8.6	20.8	7.7	15.7
	Stage 3 (Ours)	**8.9**	**21.7**	**7.9**	**16.2**
With parallel data for{Ne,Si}	*Mult. MT Baseline* (Ours)	8.6	20.1	7.6	15.3
	Liu et al. (2020)	<u>9.6</u>	21.3	<u>9.3</u>	<u>20.2</u>
	Guzmán et al. (2019)	8.8	<u>21.5</u>	6.5	15.1

Table 3: BLEU scores of various supervised and unsupervised models on the FLoRes *devtest* sets. The bolded numbers are the best unsupervised scores and the underlined numbers represent the best supervised scores. For any X↔Y language pair, the X→Y translation results are listed under each Y column, and vice-versa.

The parallel data we use came from a variety of sources, all available through WMT. We drew our English-Hindi parallel data from IITB (Kunchukuttan et al., 2017); English-Russian, English-Arabic, and English-Chinese parallel data from the UN Corpus (Ziemski et al., 2016); English-Tamil and English-Telugu from Wikimatrix (Schwenk et al., 2019). We used the scripts from Moses (Koehn, 2009) to normalize punctuation, remove non-printing characters, and replace the unicode characters with their non-unicode equivalent. We additionally use the normalizing script from Indic NLP (Kunchukuttan, 2020) for Gujarati, Nepali, Telugu, and Sinhala.

We concatenate two million lines of monolingual data for each language and use it to build a vocabulary with SentencePiece[8] (Kudo and Richardson, 2018) of 64,000 pieces. We then separate our data into SentencePiece pieces and remove all training samples that are over 88 pieces long.

6.2 Model architecture

All of our models were coded and tested in Tensorflow (Abadi et al., 2016). We use the Transformer architecture (Vaswani et al., 2017) as the basis of our translation models. We use 6-layer encoder and decoder architecture with a hidden size of 1024 and an 8192 feedforward filter size. We share the same encoder for all languages. To differentiate between the different possible output languages, we add (learned) language embeddings to each token's embedding before passing them to the decoder. We follow the same modification as done in Song et al. (2019) and modify the output transformation of each attention head in each transformer block in the decoder to be distinct for each language. Besides these modifications, we share decoder parameters for every language.

6.3 Training parameters

We use three different settings, corresponding to each stage of training. For the first stage, we use the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.0002, weight decay of 0.2 and batch size of 2048 examples. We use a learning rate schedule consisting of a linear warmup of 4000 steps to a value 0.0002 followed by a linear decay for 1.2 million steps. At every step, we choose a single dataset from which to draw a whole batch using the following process: with equal probability, choose either monolingual or parallel. If the choice

---
[8]We build the SentencePiece model with the following settings: vocab_size=64000, model_type=bpe, user_defined_symbols=[MASK], character_coverage=1.0, split_by_whitespace=true.

Algorithm 2 STAGE 3

Input: Datasets $\mathfrak{D}$, languages $\mathfrak{L}$, parameterized family of translation models p_θ, initial parameters from pre-training θ_0

1: Initialize $\theta \leftarrow \theta_0$.
2: Target Languages: $\mathfrak{L}_T \leftarrow \{\text{Gu, Kk, Ne, Si, Tr}\}$.
3: **while** not converged **do**
4:   **for** D in $\mathfrak{D}$ **do**
5:     **if** D consists of monolingual data **then**
6:       $l_D \leftarrow$ Language of D.
7:       Sample batch x from D.
8:       **if** l_D is English **then**
9:         **for** l in $\mathfrak{L}_T, l \neq l_D$ **do**
10:           Translation: $\hat{y}_l \leftarrow$ Decode $p_\theta(\hat{y}_l|x)$.
11:           bt $\leftarrow \mathcal{L}_{\text{back-translation}}(x, l)$.
12:           Update: $\theta \leftarrow$ optimizer_update(bt, θ).
13:         **end for**
14:       **else**
15:         $\mathfrak{R}_D \leftarrow$ Auxiliary languages for l_D.
16:         **for** l in $\mathfrak{R}_D \cup$ English **do**
17:           Translation: $\hat{y}_l \leftarrow$ Decode $p_\theta(\hat{y}_l|x)$.
18:           bt $\leftarrow \mathcal{L}_{\text{back-translation}}(x, l)$.
19:           Update: $\theta \leftarrow$ optimizer_update(bt, θ).
20:         **end for**
21:       **end if**
22:     **else if** D consists of parallel data **then**
23:       Sample batch (x, z) from D.
24:       Source language: $l_x \leftarrow$ Language of x.
25:       Target language: $l_z \leftarrow$ Language of z.
26:       **if** D is not synthetic **then**
27:         **for** l in $\mathfrak{L}, l \neq l_x, l_z$ **do**
28:           ct $\leftarrow \mathcal{L}_{\text{cross-translation}}(x, z, l)$.
29:           Update: $\theta \leftarrow$ optimizer_update(ct, θ).
30:         **end for**
31:       **else**
32:         Cross-entropy: ce $\leftarrow \mathcal{L}_{\text{cross-entropy}}(x, z)$.
33:         Update: $\theta \leftarrow$ optimizer_update(ce, θ).
34:       **end if**
35:     **end if**
36:   **end for**
37: **end while**

is monolingual, then we select one of the monolingual datasets uniformly at random. If the choice is parallel, we use a temperature-based sampling scheme based on the numbers of samples with a temperature of 5 (Arivazhagan et al., 2019b). In the second stage, we retain the same settings for both rounds of leveraging synthetic data except for the learning rate and number of steps. In the first round, we use the same number of steps, while in the second round we only use 240 thousand steps, a 1/5th of the original.

For the final phase, we bucket sequences by their sequence length and group them up into batches of at most 2000 tokens. We train the model with 8 NVIDIA V100 GPUs, assigning a batch to each one of them and training synchronously. We also use the Adamax optimizer instead, and cut the learning rate by four once more.

6.4 Baselines

We compare with the state-of-the-art unsupervised and supervised baselines from the literature. Note all the baselines build language-specific models, whereas we have a single model for all the target unsupervised directions.

Unsupervised baselines: For the bilingual unsupervised baselines, we include the results of Kim et al. (2020)[9] for En↔Gu and En↔Kk and of Guzmán et al. (2019) for En↔Si. We also report other multilingual unsupervised baselines. mBART (Liu et al., 2020) leverages auxiliary parallel data (e.g. En↔Hi parallel data for Gu→En) after pre-training on a large dataset consisting of 25 languages and the FLoRes dataset benchmark (Guzmán et al., 2019) leverages Hi↔En data for the En↔Ne language pair. All the unsupervised baselines that use auxiliary parallel data perform considerably better than the ones that don't.

Supervised baselines: In addition to the unsupervised numbers above, mBART and the FLoRes dataset benchmarks report supervised results that we compare with. We additionally include one more baseline where we followed the training scheme proposed in stage 1, but also included the missing parallel data. We labeled this model "Mult. MT Baseline", though we emphasize that we also leverage the monolingual data in this baseline, as in recent work (Siddhant et al., 2020a; Garcia et al., 2020).

6.5 Evaluation

We evaluate the performance of our models using BLEU scores (Papineni et al., 2002). BLEU scores are known to be dependent on the data pre-processing (Post, 2018) and thus proper care is required to ensure the scores between our models and the baselines are comparable. We thus only considered baselines which report detokenized BLEU scores with sacreBLEU (Post, 2018) or report explicit pre-processing steps. In the case of the Indic languages (Gujarati, Nepali, and Sinhala), both the baselines we consider (Guzmán et al., 2019; Liu et al., 2020) report tokenized BLEU using the tokenizer provided by the Indic-NLP library (Kunchukuttan, 2020). For these languages, we follow this convention as well so that the BLEU scores

---

[9]Due to the limited literature on unsupervised machine translation on low-resource languages, this was the best bilingual unsupervised system we could find.

| Data configuration | | newsdev2019 | |
Monolingual	Parallel	Kk↔En	
Ru	Ru	6.8	9.5
Ru, Ar, Zh	Ru	7.3	14.8
Ru	Ru, Ar, Zh	9.6	18.4
Ru, Ar, Zh	Ru, Ar, Zh	9.8	18.6

Table 4: BLEU scores for a model trained with various configurations for the auxiliary data.

remain comparable. Otherwise, we follow suit with the rest of the literature and report detoknized BLEU scores through sacreBLEU[10].

6.6 Results & discussion

We list the results of our experiments for the WMT datasets in Table 2 and for the FLoRes datasets in Table 3. After the first stage of training, we obtain competitive BLEU scores for X→En translation directions, outperforming all unsupervised models as well as mBART for the language pairs Kk→En and Gu→En. Upon completion of the second stage of training, we see that the En→X language pairs observe large gains, while the X→En directions also improve. The final round of training further improves results in some language pairs, yielding an increase of +0.44 BLEU on average.

Note that in addition to considerably outperforming all the unsupervised baselines, our approach outperforms the supervised baselines on many of the language pairs, even matching the state-of-the-art on Ne→En. Specifically, it outperforms the supervised mBART on six out of ten translation directions despite being a smaller model and Guzmán et al. (2019) on all pairs. Critically, we outperform our own multilingual MT baseline, trained in the same fashion and data as Stage 1, which further reinforces our assertion that unsupervised MT can provide competitive results with supervised MT in low-resource settings.

7 Further analysis

Given the substantial quality gains delivered by our proposed method, we set out to investigate what design choices can improve the performance of unsupervised models. To ease the computational burden, we further filter the training data to remove any sample which are longer than 64 SentencePiece[11] pieces long and cut the batch size in

half for the first two stages. Additionally, we only do one additional round of training with synthetic data as opposed to the two rounds performed for the benchmark models. While these choices negatively impact performance, the resulting models still provide competitive results with our baselines and hence are more than sufficient for the purposes of experimental studies.

7.1 Increasing multilinguality of the auxiliary parallel data improves performance

It was shown in Garcia et al. (2020); Bai et al. (2020) that adding more multilingual data improved performance, and that the inclusion of auxiliary parallel data further improved the BLEU scores (Siddhant et al., 2020b). In this experiment, we examine whether further increasing multilinguality under a fixed data budget improves performance. For all configurations in this subsection, we utilize all the available English and Kazakh monolingual data. We fix the amount of auxiliary monolingual data to 40 million, the auxiliary parallel data to 12 million, and vary the number of languages which manifest in this auxiliary data.

We report the results on Table 4. It is observed that increasing the multilinguality of the parallel data is crucial, but the matter is less clear for the monolingual data. Using more languages for the monolingual data can potentially harm performance, but in the presence of multiple auxiliary language pairs with supervised data this degradation vanishes.

7.2 Synthetic data is critical for both stage 2 and stage 3 of training

In the following experiments, we evaluate the role of synthetic parallel data in the improved performance found at the end of stage 2 and stage 3 of our training procedure. We first evaluate whether the improved performance at the end of stage 2 comes from the synthetic data or the continued training. We consider the alternative where we repeat the same training steps as in stage 2 but without the synthetic data. We then additionally fine-tune these models with the same procedure as stage 3, but without any of the terms involving synthetic data. We report the BLEU scores for all these configurations in Table 5. The results suggest: the baseline without synthetic parallel data shows inferior performance across all language pairs compared to our approach leveraging synthetic parallel data.

---

[10]BLEU + case.mixed + numrefs.1 + smooth.exp + tok.13a + version.1.4.14

[11]For all the experiments in this section, we use the same SentencePiece vocabulary as our benchmark model.

	Stage	newsdev2019 Gu ↔ En		newsdev2019 Kk ↔ En		newsdev2016 Tr ↔ En		FLoRes dev Ne ↔ En		FLoRes dev Si ↔ En	
Baseline	First	5.0	23.4	4.0	16.1	6.3	17.7	2.8	15.1	1.3	12.0
Without synthetic data	Second	6.2	24.8	4.24	17.0	6.3	18.5	3.6	16.0	1.6	12.7
	Third	12.9	26.2	6.1	16.3	12.9	19.5	5.9	16.1	5.2	13.1
With synthetic data	Second	19.6	29.8	10.6	20.0	16.7	23.8	7.3	17.4	8.3	16.6
	Third	18.6	30.3	11.6	21.5	17.9	24.7	8.2	17.6	7.7	17.4

Table 5: BLEU scores of model configurations with or without synthetic data. Otherwise, we report the numbers after stage 2 for both models and use the results after stage 1 as a baseline.

	Objectives	ΔBLEU
With synthetic data	BT	0.6
	+ CT with `Hi`	2.1
	+ CT with `Ru` and `Ar`	2.6
Without synthetic data	BT	0.0
	+ CT with `Hi`	1.3
	+ CT with `Ru` and `Ar`	1.4

Table 6: Total BLEU increase for `X→En` over baseline fine-tuning strategy consisting of on-the-fly back-translation (BT) and no synthetic data. We refer to cross-translation as "CT".

Data Configurations	newstest2019 Gu↔ En		newsdev2019 Gu↔ En	
500k News Crawl	6.8	15.7	9.7	21.7
500k Common Crawl	9.2	16.7	9.4	22.5
100k News Crawl	3.6	10.0	5.4	12.4
mBART	-	13.8	-	-
Kim et al. (2020)	0.6	0.6	-	-

Table 7: BLEU scores for various configurations of Gujarati monolingual data, where we vary amount of data and domain. We include the best results of mBART and (Kim et al., 2020) for comparison.

Finally, we inspect whether the synthetic parallel data is still necessary in stage 3 or if it suffices to only leverage it during the second stage. We consider three fine-tuning strategies, where we either **(1)** only utilize on-the-fly back-translation **(2)** additionally include cross-translation terms for Gujarati, Nepali, and Sinhala using Hindi **(3)** additionally include a cross-translation terms for Turkish and Kazakh involving Arabic and Russian respectively. We compare all of the approaches to the vanilla strategy that only leverages on-the-fly back-translation and report the aggregate improvements in BLEU on the `X→En` directions over this baseline in Table 6. We see two trends: The configurations that do not leverage synthetic data perform worse than those that do, and increasing multilinguality through the inclusion of cross-translation further improves performance.

7.3 Our approach is robust under multiple domains

We investigate the impact of data quantity and quality on the performance of our models. In this experiment, we focus on `En↔Gu` and use all available monolingual and auxiliary parallel data for all languages except Gujarati. We consider three configurations: **(1)** 500,000 lines from News Crawl (in-domain high-quality data); **(2)** 500,000 lines from Common Crawl (multi-domain data); **(3)** 100,000 lines from News Crawl. We present the results on

both *newstest2019* and *newsdev2019* for `En↔Gu` on Table 7. We see that both Common Crawl and News Crawl configurations produce similar results at this scale, with the Common Crawl configuration having a small edge on average. Notice that even in this data-starved setting, we still outperform the competing unsupervised models. Once we reach only 100,000 lines, performance degrades below mBART but still outperforms the bilingual UNMT approach of Kim et al. (2020), revealing the power of multilinguality in low-resource settings.

8 Conclusion

In this work, we studied how multilinguality can make unsupervised translation viable for low-resource languages in a realistic setting. Our results show that utilizing the auxiliary parallel data in combination with synthetic data through our three-stage training procedure not only yields large gains over unsupervised baselines but also outperforms several modern supervised approaches.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019.

Massively multilingual neural machine translation. In *NAACL*.

Maruan Al-Shedivat and Ankur P Parikh. 2019. Consistency by agreement in zero-shot neural machine translation. In *NAACL*.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Roee Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019a. The missing ingredient in zero-shot neural machine translation. *arXiv preprint arXiv:1903.07091*.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, et al. 2019b. Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019. An effective approach to unsupervised machine translation. In *ACL*.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *ICLR*.

Mikel Artetxe, Sebastian Ruder, Dani Yogatama, Gorka Labaka, and Eneko Agirre. 2020. A call for more rigor in unsupervised cross-lingual learning. *arXiv preprint arXiv:2004.14958*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Hongxiao Bai, Mingxuan Wang, Hai Zhao, and Lei Li. 2020. Unsupervised neural machine translation with indirect supervision. *arXiv preprint arXiv:2004.03137*.

Yun Chen, Yang Liu, Yong Cheng, and Victor OK Li. 2017. A teacher-student framework for zero-resource neural machine translation. *arXiv preprint arXiv:1705.00753*.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016a. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *NAACL*.

Orhan Firat, Baskaran Sankaran, Yaser Al-onaizan, Fatos T. Yarman Vural, and Kyunghyun Cho. 2016b. Zero-resource translation with multi-lingual neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 268–277, Austin, Texas. Association for Computational Linguistics.

Xavier Garcia, Pierre Foret, Thibault Sellam, and Ankur P Parikh. 2020. A multilingual view of unsupervised machine translation. *arXiv preprint arXiv:2002.02955*.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor OK Li. 2018. Universal neural machine translation for extremely low resource languages. *arXiv preprint arXiv:1802.05368*.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The flores evaluation datasets for low-resource machine translation: Nepali–english and sinhala–english. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6100–6113.

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

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709.

Yunsu Kim, Miguel Graça, and Hermann Ney. 2020. When and why is unsupervised neural machine translation useless? *arXiv preprint arXiv:2004.10581*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Philipp Koehn. 2009. *Statistical machine translation*. Cambridge University Press.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.

Anoop Kunchukuttan. 2020. The indicnlp library.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2017. The iit bombay english-hindi parallel corpus. *arXiv preprint arXiv:1710.02855*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only. In *ICLR*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-based & neural unsupervised machine translation.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Zuchao Li, Rui Wang, Kehai Chen, Masso Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. 2020a. Data-dependent gaussian prior objective for language generation. In *ICLR*.

Zuchao Li, Hai Zhao, Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2020b. Reference language based unsupervised neural machine translation. *arXiv preprint arXiv:2004.02127*.

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. *arXiv preprint arXiv:2001.08210*.

Kelly Marchisio, Kevin Duh, and Philipp Koehn. 2020. When does unsupervised machine translation work? *arXiv preprint arXiv:2004.05516*.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. *arXiv preprint arXiv:1808.04189*.

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 on association for computational linguistics*, pages 311–318. 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.

Sujith Ravi and Kevin Knight. 2011. Deciphering foreign language. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 12–21.

Shuo Ren, Wenhu Chen, Shujie Liu, Mu Li, Ming Zhou, and Shuai Ma. 2018. Triangular architecture for rare language translation. *arXiv preprint arXiv:1805.04813*.

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

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia. *arXiv preprint arXiv:1907.05791*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Edinburgh neural machine translation systems for wmt 16. *arXiv preprint arXiv:1606.02891*.

Aditya Siddhant, Ankur Bapna, Yuan Cao, Orhan Firat, Mia Chen, Sneha Kudugunta, Naveen Arivazhagan, and Yonghui Wu. 2020a. Leveraging monolingual data with self-supervision for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2827–2835, Online. Association for Computational Linguistics.

Aditya Siddhant, Ankur Bapna, Yuan Cao, Orhan Firat, Mia Chen, Sneha Kudugunta, Naveen Arivazhagan, and Yonghui Wu. 2020b. Leveraging monolingual data with self-supervision for multilingual neural machine translation. *arXiv preprint arXiv:2005.04816*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning*, pages 5926–5936.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2020. Knowledge distillation for multilingual unsupervised neural machine translation. *arXiv preprint arXiv:2004.10171*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Chang Xu, Tao Qin, Gang Wang, and Tie-Yan Liu. 2019. Polygon-net: a general framework for jointly boosting multiple unsupervised neural machine translation models. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 5320–5326. AAAI Press.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1. 0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534.

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.