The Source-Target Domain Mismatch Problem in Machine Translation

Jiajun Shen[◇†] **Peng-Jen Chen**[◇†] **Matt Le**[†] **Junxian He**[•*] **Jiatao Gu**[†]
Myle Ott[†] **Michael Auli**[†] **Marc'Aurelio Ranzato**[†]
[†]Facebook AI Research
[•]Carnergie Mellon University
{jiajunshen,pipibjc,mattle,jgu}@fb.com
{myleott,michaelauli,ranzato}@fb.com junxianh@cs.cmu.edu

Abstract

While we live in an increasingly interconnected world, different places still exhibit strikingly different cultures and many events we experience in our every day life pertain only to the specific place we live in. As a result, people often talk about different things in different parts of the world. In this work we study the effect of local context in machine translation and postulate that this causes the domains of the source and target language to greatly mismatch. We first formalize the concept of source-target domain mismatch, propose a metric to quantify it, and provide empirical evidence for its existence. We conclude with an empirical study of how source-target domain mismatch affects training of machine translation systems on low resource languages. While this may severely affect back-translation, the degradation can be alleviated by combining back-translation with self-training and by increasing the amount of target side monolingual data.

1 Introduction

The use of language greatly varies with the geographic location (Firth, 1935; Johnstone, 2010). Even within places where people speak the same language (Britain, 2013), there is a lot of lexical variability due to change of style and topic distribution, particularly when considering content posted on social media, blogs and news outlets. For instance, while a primary topic of discussion between British sport fans is cricket, American sport fans are more likely to discuss other sports such as baseball (Leech and Fallon, 1992).

The effect of local context in the use of language is even more extreme when considering regions where different languages are spoken. Despite the increasingly interconnected world we live in, people in different places tend to talk about different things. There are several reasons for this, from cultural differences due to geographic separation and history, to the local nature of many events we experience in our every day life.

This phenomenon has not only interesting sociolinguistic aspects but it has also strong implications in machine translation (MT) (Bernardini and Zanettin, 2004). In particular, machine translation aims at automatically translating content in two languages that are often spoken in very distant geographic locations by people with rather different cultures.

As of today, most MT research has been based on the often implicit assumption that content in the two languages is *comparable*. Sentences comprising the parallel dataset used for training are assumed to cover the same topic distribution, regardless of the originating language. Even when there exist a mismatch between the source and the target domain, the dataset creator is assumed to have made the effort to equalize the two distributions.

The major contribution of this work is to raise awareness in the MT community that this assumption may not hold in many real world settings of interest, which often involve distant and low-resource language pairs and for content produced every day on the Internet by means of blogs, social platforms and news outlets. The goal of an MT system is to translate source sentences sampled from the source domain to the target language. Training an MT system on a comparable corpus may lead to poor generalization unless the test domain matches the domain of the training data. Likewise, training on an uncurated corpus exhibiting mismatch between the source and the target domain may also work poorly when naïvely applying popular methods like back-translation (Sennrich et al., 2015).

[*]Work done during an internship at Facebook AI Research.
[◇]Equal contribution.

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 1519–1533
April 19 - 23, 2021. ©2021 Association for Computational Linguistics

Notice that there may very well be several factors contributing to such mismatch, such as change of register, different functional use of the text in the two languages, and difference in the quality of the data generation process of the two languages. Regardless of what is contributing to the observed mismatch, it is important to be aware of its existance and effect on MT algorithms.

The source-target domain mismatch (STDM) can be understood as an instance of *multi-domain* MT (see Fig. 1 for an illustration and §3 for a formal definition), whereby part of the parallel dataset and the source monolingual dataset are "in-domain" because they originate from the source domain, and the remaining part of the parallel dataset as well as the target monolingual data are "out-of-domain". There already exist several techniques for domain adaptation, like domain tagging (Caswell et al., 2019) and dataset weighting (Edunov et al., 2018; Wang et al., 2017; van der Wees et al., 2017), which are applicable and which we also employ in this work. However, these may not be enough to improve generalization on low resource languages because STDM effectively decreases the already scarce amount of useful (in-domain) parallel data, hindering good generalization. It is therefore important to quantify STDM (§4) and consider how STDM affects methods that leverage monolingual data (§5).

For instance, STDM may negatively impact the effectiveness of back-translation because, even if the backward model was perfect, the back-translated data is out-of-domain relative to the source domain from which we aim to translate. Empirically we found that this is the case both in a controlled setting (§6.1) as well as in realistic datasets (§6.2). However, this issue can be compensated by adding more target-side monolingual data and by combining back-translation with self-training (Yarowski, 1995).

2 Related Work

The observation that topic distributions and various kinds of lexical variabilities depend on the local context has been known and studied for a long time (Firth, 1935). For instance, Firth (1935) says "*Most of the give-and-take of conversation in our everyday life is stereotyped and very narrowly conditioned by our particular type of culture*". In her seminal work, Johnstone (2010) analyzed the role of place in language, focusing on lexical variations within the same language, a subject further explored by Britain (2013). Some of these works were the basis for later studies that introduced computational models for how language changes with geographic location (Mei et al., 2006; Eisenstein et al., 2010).

In the field of topic modeling, there has been a new sub-field emerging over the past 10 years focusing on modeling multi-lingual corpora (Mimno et al., 2009; Boyd-Graber and Blei, 2009; Gutierrez et al., 2016). However, only recently had researchers dropped assumptions on the use of parallel and comparable corpora (Hao and Paul, 2018; Yang et al., 2019). While some works do investigate issues related to STDM (Gutierrez et al., 2016), like how named entities receive a different distribution over words in different languages (Lin et al., 2018), none of these works have analyzed how the overall topic distribution of data originating in the source and target language differ.

In MT, researchers have often made an explicit assumption on the use of *comparable* corpora (Fung and Yee, 1998; Munteanu et al., 2004; Irvine and Callison-Burch, 2013), i.e. corpora in the two languages that roughly cover the same set of topics. Unfortunately, monolingual corpora are seldom comparable in practice. Leech and Fallon (1992) analyzes two comparable corpora, one in American English and the other in British English, and demonstrate differences that reflect the cultures of origin. Similarly, Bernardini and Zanettin (2004) observes that parallel datasets built for MT exhibit strong biases in the selection of the original documents, making the text collection not quite comparable.

The non-comparable nature of MT datasets is even more striking when considering low resource language pairs, for which differences in local context and cultures are often more pronounced. Recent studies (Søgaard et al., 2018; Neubig and Hu, 2018) have warned that removing the assumption on comparable corpora strongly deteriorates performance of lexicon induction techniques which are at the foundation of MT.

Back-translation (Sennrich et al., 2015) has been the workhorse of modern neural MT, enabling very effective use of target side monolingual data. Back-translation is beneficial because it helps regularizing the model and adapting to new domains (Burlot and Yvon, 2018). However, the typical setting of current MT benchmarks as popularized by re-

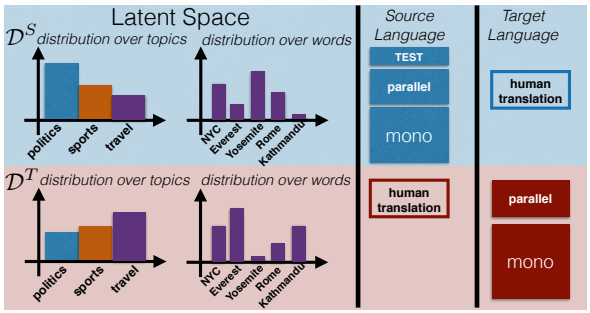

Figure 1: Toy illustration of STDM in MT. There are two domains, the source domain $\mathcal{D}^S$ (top) and the target domain $\mathcal{D}^T$ (bottom). We postulate that in a latent concept space these two domains differ because the topic distributions are different (e.g., in the source domain politics is more popular than travel) and because for the same topic the word distributions are different (e.g., the word "Everest" is more common than "Yosemite" in the travel topic of the target domain). On the right hand side, we show how STDM manifests in machine translation datasets. All data originating in the source language belongs to the source domain, this includes a portion of the parallel dataset, the source side monolingual dataset and the test set we eventually would like to translate. Empty boxes represent human translated data in the parallel training dataset.

cent WMT competitions (Bojar et al., 2019) is a mismatch between *training and test* sets, as opposed to a mismatch between *source and target* domains as in this work. Self-training (Ueffing, 2006; Yarowski, 1995; He et al., 2020) has then been employed to make better use of source side monolingual data as this is in-domain with the text we would like to translate at test time. Finally, there is a vast literature on domain adaptation which has so far mostly focused on domain shift between training and test distribution, and presence of multiple domains. Finetuning (Freitag and Al-Onaizan, 2016), domain tagging (Caswell et al., 2019) and various kinds of dataset weighting (Wang et al., 2017; van der Wees et al., 2017) are among the most popular methods to cope with domain issues, and this is also what we use in this work.

3 The STDM Problem

In this section we formalize the definition of Source-Target Domain Mismatch (STDM); this is an intrinsic property of the data which is independent of the particular MT system under consideration. In practice, there might be several factors contributing to STDM. Here, we are going to consider only the difference in topic distributions, since this is what we can easily quantify.

We assume there exists a latent concept space shared across all languages. The process to generate a sentence follows the standard data generation process used in topic modeling, whereby we first sample a distribution over topics, $\pi_i \sim \Pi$ where i is an index over topics, and then a distribution over words for each topic, $w_{ij} \sim \pi_i$, where j indexes the words in the dictionary. Next, we assume there are two distinct domains, the source domain $\mathcal{D}^S$ and the target domain $\mathcal{D}^T$. These two domains differ in both the distribution over topics Π, and the distribution over words given a certain topic π_i, as depicted in Fig. 1. For the sake of conciseness, we will refer to z^s and z^t as sentences in the concept space generated from domain $\mathcal{D}^S$ and $\mathcal{D}^T$, respectively.

Let's imagine now that we have generated two sets of sentences in each domain. What we observe in practice is their realization in each language, $\text{src}(z^s)$ and $\text{tgt}(z^t)$, where src and tgt map sentences from the concept space to the source and target language, respectively. Finally, let's denote with $h_{s\to t}$ and $h_{t\to s}$ the functions representing human translations of source sentences in the target language and vice versa.

In the simplest setting, a machine translation dataset is composed of parallel and monolingual datasets. Using the notation introduced above, the parallel dataset is denoted by $\mathcal{P} = \{(\text{src}(z^s), h_{s\to t}(\text{src}(z^s))\}_{z^s\sim\mathcal{D}^S} \cup \{(h_{t\to s}(\text{tgt}(z^t)), \text{tgt}(z^t)\}_{z^t\sim\mathcal{D}^T}$. The first set originates in the source language and belongs to the source domain, while the second set originates in the target language and belongs to the target domain. We then have a source side monolingual dataset, $\mathcal{M}^S = \{\text{src}(z^s)\}_{z^s\sim\mathcal{D}^S}$, and a target side monolingual dataset, $\mathcal{M}^T = \{\text{tgt}(z^t)\}_{z^t\sim\mathcal{D}^T}$, belonging to the source and target domains, respectively. Most importantly, the *test set* which we would like to eventually translate contains sentences in the source language, all belonging to the *source* domain. The existence of distinct source and target domains and datasets derived from these two domains as described above define the STDM problem.

While in previous MT studies, there is a mismatch between the training and the test distribution, STDM is a particular case of multi-domain training, with the test set matching one of the training domains. We would like to a) understand the effects of such mismatch and b) understand how to best leverage the out-of-domain data originating from the target language (target monolingual dataset and

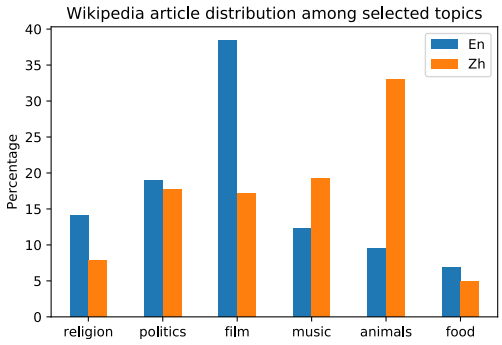

Figure 2: Topic distribution of Wikipedia pages written in English and Chinese.

portion of the parallel dataset originating in the target language).

3.1 Empirical Evidence

In this section we first provide anecdotal evidence that documents originating in different languages possess different distributions over topics. We train two topic classifiers (see Appendix A for details), one for Chinese and the other for English, using the Wikipedia annotated data from Yuan et al. (2018). We apply this classifier to 20,000 documents randomly sampled from English and Chinese Wikipedia. Fig. 2 shows that according to this classifier, English Wikipedia has more pages about entertainment and religion than Chinese Wikipedia, for instance.

Second, we refer the readers to Leech and Fallon (1992)'s study to find evidence that corpora originating in different places may have different word distributions for the same set of topics. In that study, Leech and Fallon (1992) analyzed a British and an American corpus constructed using exactly the same distribution of topics, and yet the word distribution was different, reflecting the different cultural biases of the two countries.

4 Metric: The STDM Score

Given the framework and assumptions introduced in §3, in this section we are going to discuss a practical way to measure STDM. Ideally, we would like to measure a distance between two sample distributions, $z^s \sim \mathcal{D}^S$ and $z^t \sim \mathcal{D}^T$. Unfortunately, we have no access to such latent space. What we observe are realizations in the source and target language. However, it is also an open research question (Hao and Paul, 2018; Yang et al., 2019) how to compare the distribution of $\{\text{src}(z^s)\}$ against $\{\text{tgt}(z^t)\}$, since these are two possibly incomparable corpora in different languages.

In this work, we therefore leverage the existence of a parallel corpus and compare the distribution of $\mathcal{A}^T = \{\text{tgt}(z^t)\}_{z^t \sim \mathcal{D}^T}$ with $\mathcal{A}^S = \{h_{s\to t}(\text{src}(z^s))\}_{z^s \sim \mathcal{D}^S}$. The underlying assumptions are a) we know the originating language of each training example, b) the effect of the change of the word distribution is negligible compared to the shift in topic distribution, and c) the effect of translationese (Baker, 1993; Zhang and Toral, 2019; Toury, 2012) is negligible compared to the actual STDM, and therefore, we can ignore changes to the distribution brought by the mapping $h_{s\to t}$ (we validate this assumption in §C).

Under these assumptions, we define the score as a measure of the topic discrepancy between $\mathcal{A}^S$ and $\mathcal{A}^T$. Let $\mathcal{A} = \mathcal{A}^S \cup \mathcal{A}^T$ be the concatenation of the corpus originating in the source and target language. We first extract topics using LSA. Let $A \in \mathbb{R}^{(n^S+n^T)\times k}$ be the TF-IDF matrix derived from $\mathcal{A}$ where the first n^S rows are representations taken from $\mathcal{A}^S$, the bottom n^T rows are representations of $\mathcal{A}^T$, and k is the number of words in the dictionary. The SVD decomposition of A yields: $A = USV = (U\sqrt{(S)})(\sqrt{(S)}V) = \bar{U}\bar{V}$. Matrix $\bar{U}$ collects topic representations of the original documents; let's denote by $\bar{U}^S$ the first n^S rows corresponding to $\mathcal{A}^S$ and $\bar{U}^T$ the remaining n^T rows corresponding to $\mathcal{A}^T$. Let $C = \bar{U}\bar{U}' = \begin{bmatrix} C^{SS} & C^{ST} \\ C^{ST\prime} & C^{TT} \end{bmatrix}$, where $C^{SS} = \bar{U}^S\bar{U}^{S\prime}$, $C^{ST} = \bar{U}^S\bar{U}^{T\prime}$ and $C^{TT} = \bar{U}^T\bar{U}^{T\prime}$. The STDM score is defined as:

$$\text{score} = \frac{s^{\text{ST}} + s^{\text{TS}}}{s^{\text{SS}} + s^{\text{TT}}}, \text{with } s^{\text{AB}} = \frac{1}{n^A n^B}\sum_{i=1}^{n^A}\sum_{j=1}^{n^B} C^{AB}_{i,j} \quad (1)$$

where s^{AB} measures the average similarity between documents of set A to documents of set B. The score measures the cross-corpus similarity normalized by the within corpus similarity. In the extreme setting where $\mathcal{D}^S$ and $\mathcal{D}^T$ are fully disjoint, then we would have that the off-diagonal block C^{ST} is going to be a zero matrix and therefore the score is equal to 0. When the two domains perfectly match instead, $s^{\text{SS}} = s^{\text{TT}} = s^{\text{ST}} = s^{\text{TS}}$, and therefore, the score is equal to 1. In practice, we expect a score in the range $[0, 1]$. Note that we opted for this metric because of its simplicity, but other methods to extract topics and measure domain mismatch could have been used.

4.1 A Controlled Setting

Similarly to Kilgarriff and Rose (1998), we introduce a *synthetic* benchmark to finely control the

α	0	0.25	0.5	0.75	1.0
STDM score	0.29	0.55	0.78	0.93	0.99

Table 1: STDM score as a function of the parameter α controlling the STDM in the synthetic setting.

domain of the target originating data, and therefore the amount of STDM. The objective is to assess whether the STDM score defined in Eq. 1 captures well the expected amount of mismatch, since we have full control over how the data was generated and its domain.

The key idea of this controlled setting is to use data from two very different domains, assign the source domain to one of them and the target domain to a convex combination of the two. In this work we use EuroParl (Koehn, 2005) as our source originating data, while our target originating data contains a mix of data from EuroParl and OpenSubtitles (Lison and Tiedemann, 2016). Specifically, we consider a French to English translation task with a parallel dataset composed of 10,000 sentences from EuroParl (which "originates" in French) and 10,000 sentences from the target domain (which is "originates" in English)[1]. Let $\alpha \in [0, 1]$, the domain of the target is set to: α EuroParl + $(1 - \alpha)$ OpenSubtitles. α controls how similar the target domain is to the source domain.

For instance, when $\alpha = 0$ then the target domain (OpenSubtitles) is totally out-of-domain with respect to the source domain (EuroParl). When $\alpha = 1$ instead, the target domain matches perfectly the source domain. For intermediate values of α, the match is only partial. Notice that even when $\alpha = 0$, we assume that the parallel dataset is comprised of two halves, one originating from the EuroParl domain (the "French originating" data) and one from OpenSubtitles (the "English originating" data).

Next, we evaluate the STDM score as a function of α. As we can see from Table 1 and as desired, the STDM score increases fairly linearly as we increase the value of α. We refer the reader to Appendix B for experimental details.

4.2 STDM Score of Various Datasets

We now evaluate the STDM score on real data. We consider six language pairs, German-English, Finnish-English, Russian-English, Nepali-English, Chinese-English and Japanese-English. We analyze datasets from WMT, MTNT (Michel and Neubig, 2018) and from a social media platform (SMD). For each language, we sample 5000 sentences from WMT newstest sets and MTNT dataset, and 20000 sentences from SMD. We then merge all these datasets and their English translations to compute a common set of topics, making STDM scores comparable across language pairs and datasets.

	De-En	Fi-En	Ru-En	Ne-En	Zh-En	Ja-En
WMT	0.79	0.79	0.76	-	0.65	-
MTNT	-	-	-	-	-	0.69
SMD	0.81	0.71	0.71	0.64	0.71	0.61

Table 2: STDM score on several language pairs using parallel data from WMT, MTNT and from a social media platform (SMD) test sets.

The results in Table 2 are striking. First, WMT datasets, except for Chinese, show relatively mild signs of STDM and negligible difference across language pairs, suggesting that the data curation process of WMT datasets have made source and target originating corpora rather comparable. The distribution of WMT Chinese originating data instead is rather different because it contains much more local news, while the other languages are mostly about international news which are largely language independent. Interestingly, En-De data derived from social media data has even milder STDM, Fi-En and Ru-En have more substantial STDM. Instead, *MTNT and SMD exhibit strong signs of STDM for distant languages* like Nepali, Chinese and Japanese. This agrees well with our intuition that STDM is more severe for more distant languages associated to more diverse cultures.

5 Machine Translation Baselines

In this section, we turn our attention to how STDM affects training of MT systems. We consider state-of-the-art neural machine translation (NMT) systems based on the transformer architecture (Vaswani et al., 2017) with subword vocabularies learned via byte-pair encoding (BPE) (Sennrich et al., 2015). In order to adapt to the different domains, we employ domain tagging (Zheng et al., 2019) by adding a domain token to the input source sentence, and we cross validate the weights between in-domain and out-domain data[2]. We also

[1] Clearly, EuroParl does not originate all in French. However, the chosen domains are so distinct that difference in topic distribution between EuroParl and OpenSubtitles will dominate discrepancies caused by other factors such as the actual origin of the data.

[2] In the controlled setting of §6.1 we found that tagging yields a small but consistent improvement by up to 1 BLEU point, and dataset weighting yields an improvement of up to 0.3 BLEU. ST and BT which are the focus of this study,

use label smoothing (Szegedy et al., 2016) and dropout (Srivastava et al., 2014) to improve generalization, as we focus on low resource language pairs where models tend to severely overfit. Finally, we explore ways to leverage both target and source side monolingual data via back-translation and self-training which we review next.

We simplify our notation and denote with $x^s = \text{src}(z^s)$ and $y^t = \text{tgt}(z^t)$ the source and target originating sentences, $y^s = h_{s\to t}(x^s)$ and $x^t = h_{t\to s}(y^t)$ the corresponding human translations, and $\hat{y}^s$ and $\hat{x}^t$ the corresponding machine translations. The superscript always specifies the domain. We assume access to a parallel dataset $\mathcal{P} = \{(x^s, y^s)\} \cup \{(x^t, y^t)\}$, a source side monolingual dataset $\mathcal{M}^s = \{x^s\}$ and a target side monolingual dataset $\mathcal{M}^t = \{y^t\}$. The test side consists of sentences from the source domain. We study STDM by training with the following algorithms:

Back-translation (BT) (Sennrich et al., 2015) is a very effective data augmentation technique that leverages $\mathcal{M}^t$. The algorithm proceeds in three steps. First, a reverse MT system is trained from target to source using the provided parallel data: $\overleftarrow{\theta} = \arg\max_\theta \mathbb{E}_{(x,y)\sim\mathcal{P}} \log p(x|y;\theta)$. Then, the reverse model is used to translate the target monolingual data: $\hat{x}^t \approx \arg\max_z p(z|y^t; \overleftarrow{\theta})$, for $y^t \sim \mathcal{M}^t$. The maximization is typically approximated by beam search. Finally, the forward model is trained over the concatenation of the original parallel and back-translated data: $\overrightarrow{\theta} = \arg\max_\theta \mathbb{E}_{(x,y)\sim\mathcal{Q}} \log p(y|x;\theta)$ with $\mathcal{Q} = \mathcal{P} \cup \{\hat{x}^t, y^t\}_{y^t\sim\mathcal{M}^t}$. In practice, the parallel data is weighted more in the loss, with a weight selected via hyper-parameter search on the validation set.

Self-Training (ST) (He et al., 2020) is another method for data augmentation that instead leverages $\mathcal{M}^s$; see Alg. 1 in Appendix D.

Also this algorithm proceeds in three steps. First, a forward MT system is trained from source to target using the provided parallel data: $\overrightarrow{\theta} = \arg\max_\theta \mathbb{E}_{(x,y)\sim\mathcal{P}} \log p(y|x;\theta)$. Then, this model is used to translate the source monolingual data: $\hat{y}^s \approx \arg\max_z p(z|x^s; \overrightarrow{\theta})$, for $x^s \sim \mathcal{M}^s$. Finally, the forward model is retrained over the concatenation of the original parallel and forward-translated data: $\overrightarrow{\theta} = \arg\max_\theta \mathbb{E}_{(x,y)\sim\mathcal{Q}} \log p(y|x;\theta)$ with $\mathcal{Q} = \mathcal{P} \cup \{x^s, \hat{y}^s\}_{y^t\sim\mathcal{M}^s}$. As with BT, the parallel data is weighted more in the loss.

improve by more than 2 BLEU points instead.

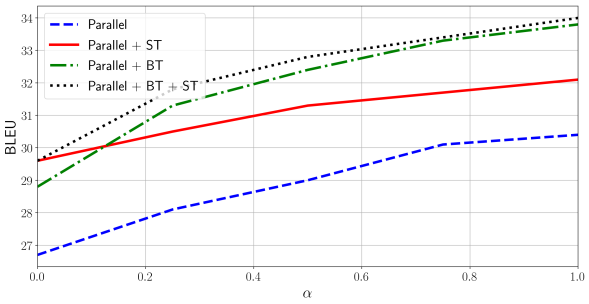

Figure 3: BLEU score in Fr-En as a function of the amount of STDM. The target domain is fully out-of-domain when $\alpha = 0$, and fully in-domain when $\alpha = 1$.

ST + BT also proceeds in three steps. First, we train an initial forward and reverse model using the parallel dataset. Second, we back-translate target side monolingual data using the reverse model and iteratively forward translate source side monolingual data using the forward model. We then retrain the forward model from random initialization using the union of the original parallel dataset, the synthetic back-translated data, and the synthetic forward translated data at the last iteration of the ST algorithm. This combined algorithm aims at leveraging the strengths of both ST and BT: the use of in-domain source monolingual data and the use of synthetic data with correct targets, respectively.

6 Machine Translation Results

In this section, we first study the effect of STDM on NMT using the controlled setting introduced in §4.1 which enables us to assess the influence of various factors, such as the extent to which target originating data is out-of-domain, and the effect of monolingual data size. We then report experiments on genuine low resource language pairs, namely Nepali-English and English-Myanmar. We report SacreBLEU (Post, 2018).

6.1 Controlled Setting

In the default setting, we have a parallel dataset with 20,000 parallel sentences. 10,000 are in-domain source originating data (EuroParl) and the remaining 10,000 are target originating data from a mix of domains, controlled by $\alpha \in [0, 1]$: α EuroParl + $(1-\alpha)$ OpenSubtitles. The source side monolingual dataset has 100,000 French sentences from EuroParl. The target side monolingual dataset has 100,000 English sentences from: α EuroParl + $(1-\alpha)$ OpenSubtitles. Finally, the test set consists of novel French sentences from EuroParl which we wish to translate to English.

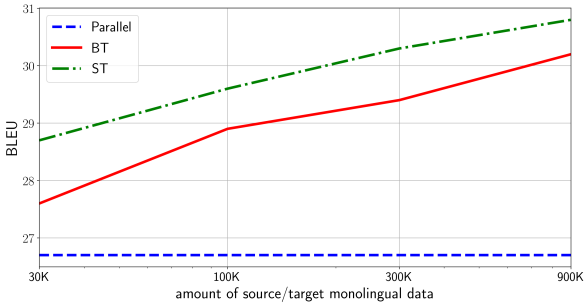

Figure 4: BLEU as a function of the amount of monolingual data when $\alpha = 0$.

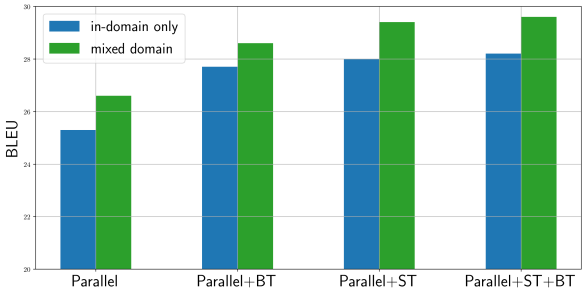

Figure 5: BLEU when using only source originating in-domain data (blue bars) or also out-of-domain target originating data (green bars) for $\alpha = 0$.

We tune model hyper-parameters (e.g., number of layers and hidden state size) and BPE size on the validation set. Based on cross-validation, when training on datasets with less than 300k parallel sentences (including those from ST or BT), we use a 5-layer transformer with 8M parameters. The number of attention heads, embedding dimension and inner-layer dimension are 2, 256, 512, respectively. When training on bigger datasets, we use a bigger transformer with 5 layers, 8 attention heads, 1024 embedding dimension, 2048 inner-layer dimension and a total of 110M parameters. We find that using bigger model can better utilize the monolingual data, but we do not find the bigger model benefits when training with less than 300k parallel sentences. The full list of hyper-parameters can be found in Appendix E.

Varying amount of STDM. In Fig. 3, we benchmark our baseline approaches while varying α (see §4.1 and Tab. 1), which controls the overlap between source and target domain.

First, we observe improved BLEU (Papineni et al., 2002) scores for all methods as we increase α. Second, there is a big gap between the baseline trained on parallel data only and methods which leverage monolingual data. Third, combining ST and BT works better than each individual method, confirming that these approaches are complementary. Finally, BT works better than ST but the gap reduces as the target domain becomes increasingly different from the source domain (small values of α). In the extreme case of STDM ($\alpha = 0$), ST outperforms BT. In fact, we observe that the gain of BT over the baseline decreases as α decreases, despite that the amount of monolingual data and parallel data remains constant across these experiments, thus showing that *BT is less effective in the presence of STDM*.

Varying amount of monolingual data. We next explore how the quantity of monolingual data affects performance and if the relative gain of ST over BT when $\alpha = 0$ disappears as we provide BT with more monolingual data. The experiment in Fig. 4 shows that a) the gain in BLEU tapers off exponentially with the amount of data (notice the log-scale in the x-axis), b) for the same amount of monolingual data ST is always better than BT and by roughly the same amount, and c) BT would require about 3 times more target monolingual data (which is out-of-domain) to yield the performance of ST. Therefore, *increasing the amount of data can compensate for domain mismatch*.

Varying amount of in-domain data. Now we explore whether, in the presence of extreme STDM ($\alpha = 0$), it may be worth restricting the training data to only contain in-domain source originating sentences. In this case, the parallel set is reduced to 10,000 EuroParl sentences, the target side monolingual data is removed and back-translation is performed on the target side of the parallel dataset. Fig. 5 demonstrates that in all cases it is better to include the out-of-domain data originating on the target side (green bars). Particularly in the low resource settings considered here, *neural models benefit from all available examples even if these are out-of-domain*.

Finally, we investigate how to construct a parallel dataset when STDM is significant ($\alpha = 0$), i.e. the target domain is OpenSubtitles. If we have a translation budget of 20,000 sentences, is it best to translate 20,000 sentences from EuroParl or to also include sentences from OpenSubtitles? This is not obvious when training with BT, since the backward model may benefit from in-domain OpenSubtitles data. In order to answer this question, we consider a *parallel* dataset with 20,000 sentences defined as: β EuroParl + $(1-\beta)$ OpenSubtitles, with $\beta \in [0, 1]$.

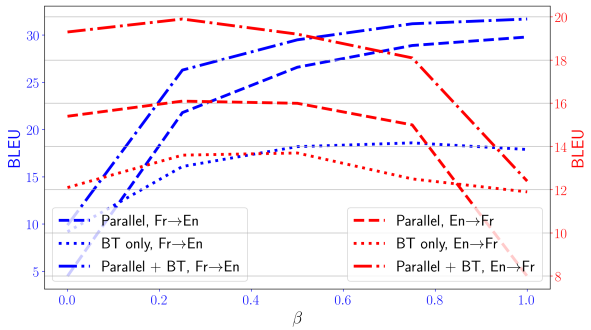

Figure 6: BLEU score as a function of the proportion of parallel data originating in the source and target domain domain. When $\beta = 0$ all parallel data originates from OpenSubtitles, when $\beta = 1$ all parallel data originates from EuroParl. Source and target monolingual corpora have 900,000 sentences from EuroParl and OpenSubtitles, respectively. The blue curves show BLEU in the forward direction (Fr-En translation of EuroParl data). The red curves show BLEU in the reverse direction (En-Fr translation of OpenSubtitles sentences).

When $\beta = 0$, the parallel dataset is out-of-domain; when $\beta = 1$ the parallel data is all in-domain. The target side monolingual dataset is fixed and contains 900,000 sentences from OpenSubtitles.

Fig. 6 shows that taking *all* sentences from EuroParl ($\beta = 1$) is optimal when translating from French (EuroParl) to English (blue curves). At high values of β, we observe a slight decrease in accuracy for models trained only on back-translated data (dotted line), confirming that BT loses its effectiveness when the reverse model is trained on out-of-domain data. However, this is compensated by the gains brought by the additional in-domain parallel sentences (dashed line). In the more natural setting in which the model is trained on both parallel and back-translated data (dash-dotted line), we see monotonic improvement in accuracy with β. A similar trend is observed in the other direction (English to French, red lines). Therefore, if the goal is to maximize translation accuracy in *both* directions, an intermediate value of β (≈ 0.5) is more desirable.

6.2 Low-Resource MT

We now measure STDM on two low-resource language pairs and verify whether in practice BT's performance deteriorates as expected when the STDM score is low, while the combination of ST+BT offers better generalization. We consider two low-resource language pairs, Nepali-English (Ne-En) and English-Myanmar (En-My). Nepali and Myanmar are spoken in regions with unique local context that is very distinct from English-speaking regions, and thus these make good language pairs for studying the STDM setting in real life.

Model	Ne → En STDM score=0.64	En → My STDM score=0.27
baseline	20.4	28.1
BT	22.3	30.0
ST	22.1	31.9
ST + BT	22.9	32.4

Table 3: BLEU scores for the Nepali to English and English to Myanmar translation task.

Data. The Ne-En parallel dataset is composed of 40,000 sentences originating in Nepali and only 7,500 sentences originating in English. There are 5,000 sentences in the validation and test sets all originating in Nepali. We also have 1.8M monolingual sentences in Nepali and English, collected from public posts from a social media platform. This dataset closely resembles our idealized setting of Fig. 1. The STDM score of this dataset is 0.64 (see Tab. 2) and is analogous to our synthetic setting (§6.1) where α is low but β is large.

The En-My parallel data is taken from the Asian Language Treebank (ALT) corpus (Thu et al., 2016; Ding et al., 2018, 2019) with 18,088 training sentences all originating from English news. The validation and test sets have 1,000 sentences each, all originating from English. Following Chen et al. (2019), we use 5M English sentences from NewsCrawl as source side monolingual data and 100K Myanmar sentences from Common Crawl as target side monolingual data. The STDM Score is even lower on this dataset, only 0.27. Comparing to our controlled setting this dataset would have β equal to 1 and presumably a small value of α, an ideal setting for ST.

Models. We run model hyper-parameter sweep on the validation set and pick the best-performing model architecture (e.g., number of layers and hidden layer sizes). On both datasets, the parallel data baseline is a 5-layer transformer with 8 attention heads, 512 embedding dimensions and 2048 inner-layer dimensions, which consists of 42M parameters. When training with BT and ST, we use a 6-layer transformer with 8 attention heads, 1024 embedding dimensions, 2048 inner-layer dimensions, resulting in 186M parameters. The detailed hyper-parameters search range can be found in Appendix §E.

Results. In Table 3, we observe that on the Ne-En task augmenting the parallel dataset with either forward- or back-translated monolingual data achieves almost 2 BLEU points improvement over the supervised baseline. On the En-My task where STDM is more severe (with a value of 0.27 which is similar to $\alpha = 0$ in Tab. 1), BT outperforms the baseline by 1.9 BLEU, while ST improves by twice as much. This is perhaps not surprising since source side monolingual data is in-domain and abundant, while target side monolingual data is scarce and out-of-domain. On both tasks, combining ST and BT outperforms each individual method.

7 Conclusions

While the commonly used WMT datasets exhibit mild STDM, we find that less curated datasets, often in more distant and lower resource language pairs (§4.2), exhibit much stronger STDM. How can these findings inform us on how to better set up a MT system in practice? Our first recommendation is to be aware of possible STDM, and *(i)* check whether origin language information is available. If this is available, then it may be possible to *(ii)* quantitatively measure STDM as described in §4. Next, *(iii)* be aware that when STDM is severe (STDM score is low), BT performance suffers (Fig. 3). However, *(iv)* we may be able to combat this by increasing the amount of target side (out-of-domain) monolingual data (Fig. 4) and *(v)* by combining BT with ST (Fig. 3).

Of course, the relative ratio of monolingual data in the source and target side and the actual degradation brought by STDM depend on the particular language pair. The more distant two languages, the more difficult the learning task and the more data is needed to learn it. And finally, the less parallel data there is, the more monolingual data will be needed to compensate. The intricate dependency between all these factors merits future investigation.

References

Mona Baker. 1993. Corpus linguistics and translation studies: Implications and applications. *Text and technology: In honour of John Sinclair*, 233:250.

Silvia Bernardini and Federico Zanettin. 2004. When is a universal not a universal. *Translation universals: do they exist? John Benjamin publisher Edited by Anna Mauranen and Pekka Kujammak*, pages 51–62.

Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, and Christof Monz. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proc. of WMT*.

Jordan Boyd-Graber and David M. Blei. 2009. Multilingual topic models for unaligned text. In *Conference on Uncertainty in Artificial Intelligence*.

David Britain. 2013. *Space, Diffusion and Mobility*. Wiley publishers; Book Editor(s): J.K. Chambers Natalie Schilling First. Chapter 22.

Franck Burlot and François Yvon. 2018. Using monolingual data in neural machine translation: a systematic study. In *Empirical Methods in Natural Language Processing*.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63, Florence, Italy. Association for Computational Linguistics.

Peng-Jen Chen, Jiajun Shen, Matt Le, Vishrav Chaudhary, Ahmed El-Kishky, Guillaume Wenzek, Myle Ott, and Marc'Aurelio Ranzato. 2019. Facebook ai's wat19 myanmar-english translation task submission. In *Workshop on Asian Translation*.

Chenchen Ding, Hnin Thu Zar Aye, Win Pa Pa, Khin Thandar Nwet, Khin Mar Soe, Masao Utiyama, and Eiichiro Sumita. 2019. Towards Burmese (Myanmar) morphological analysis: Syllable-based tokenization and part-of-speech tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(1):5.

Chenchen Ding, Masao Utiyama, and Eiichiro Sumita. 2018. NOVA: A feasible and flexible annotation system for joint tokenization and part-of-speech tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(2):17.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Conference on Empirical Methods in Natural Language Processing*.

Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. 2010. A latent variable model for geographic lexical variation. In *Annual Meeting of the Association for Computational Linguistics*.

John Rupert Firth. 1935. On sociological linguistics. *Transactions of the Royal Society*, pages 67–69.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast domain adaptation for neural machine translation.

Pascale Fung and Lo Yuen Yee. 1998. An ir approach for translating new words from nonparallel, comparable texts. In *The 17th International Conference on Computational Linguistics*.

E.D. Gutierrez, Ekaterina Shutova annd Patricia Lichtenstein, Gerard de Melo, and Luca Gilardi. 2016. Detecting cross-cultural differences using a multilingual topic model. In *Conference of the Association for Computational Linguistics*.

Shudong Hao and Michael J. Paul. 2018. Learning multilingual topics from incomparable corpora. In *International Conference on Computational Linguistics (COLING)*.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2020. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations*.

Ann Irvine and Chris Callison-Burch. 2013. Combining bilingual and comparable corpora for low resource machine translation. In *Proceedings of the eighth workshop on statistical machine translation*, pages 262–270.

Barbara Johnstone. 2010. *Language and place*. R. Mesthrie and W. Wolfram, editors, Cambridge Handbook of Sociolinguistics. Cambridge University Press.

Adam Kilgarriff and Tony Rose. 1998. Measures for corpus similarity and homogeneity. In *Conference on Empirical Methods in Natural Language Processing*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. *MT Summit*.

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, Brussels, Belgium. Association for Computational Linguistics.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Geoffrey Leech and Roger Fallon. 1992. Computer corpora: What do they tell us about culture? *ICAME Journal Computers in English Linguistics*, (16).

Bill Yuchen Lin, Frank F. Xu, Kenny Q. Zhu, and Seung won Hwang. 2018. Mining cross-cultural differences and similarities in social media. In *Conference of the Association for Computational Linguistics*.

P. Lison and J. Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *10th International Conference on Language Resources and Evaluation (LREC)*.

Qiaozhu Mei, Chao Liu, and Hang Su. 2006. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *WWW*.

Paul Michel and Graham Neubig. 2018. Mtnt: A testbed for machine translation of noisy text. In *Conference on Empirical Methods in Natural Language Processing*.

David Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. 2009. Polylingual topic models. In *Conference on Empirical Methods in Natural Language Processing*.

D.S. Munteanu, A. Fraser, and D. Marcu. 2004. Improved machine translation performance via parallel sentence extraction from comparable corpora. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Brussels, Belgium.

Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Analyzing uncertainty in neural machine translation. In *International Conference on Machine Learning*.

K. Papineni, S. Roukos, T. Ward, and W.J. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

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, Belgium, Brussels. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 86–96.

Anders Søgaard, Sebastian Ruder, and Ivan Vulic. 2018. On the limitations of unsupervised bilingual dictionary induction. In *Conference of the Association for Computational Linguistics (ACL)*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *IEEE conference on computer vision and pattern recognition*.

Ye Kyaw Thu, Win Pa Pa, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Introducing the asian language treebank (alt). In *LREC*.

Gideo Toury. 2012. *Descriptive translation studies and beyond: Revised edition*. John Benjamins Publishing.

Nicola Ueffing. 2006. Using monolingual source-language data to improve mt performance. In *IWSLT*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Proc. of NIPS*.

Rui Wang, Masao Utiyama, Lemao Liu, Kehai Chen, and Eiichiro Sumita. 2017. Instance weighting for neural machine translation domain adaptation. In *Conference on Empirical Methods in Natural Language Processing*.

Marlies van der Wees, Arianna Bisazza, and Christof Monz. 2017. Dynamic data selection for neural machine translation. In *Conference on Empirical Methods in Natural Language Processing*.

Weiwei Yang, Jordan Boyd-Graber, and Philip Resnik. 2019. A multilingual topic model for learning weighted topic links across corpora with low comparability. In *Conference on Empirical Methods in Natural Language Processing*.

David Yarowski. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Annual Meeting of the Association for Computational Linguistics*.

Michelle Yuan, Benjamin Van Durme, and Jordan Boyd-Graber. 2018. Multilingual anchoring: Interactive topic modeling and alignment across languages. In *Neural Information Processing Systems*.

Mike Zhang and Antonio Toral. 2019. The effect of translationese in machine translation test sets. *arXiv*, abs/1906.08069.

Renjie Zheng, Hairong Liu, Mingbo Ma, Baigong Zheng, and Liang Huang. 2019. Robust machine translation with domain sensitive pseudo-sources: Baidu-OSU WMT19 MT robustness shared task system report. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 559–564, Florence, Italy. Association for Computational Linguistics.

A Topic Classifiers

In this section we provide the experimental details of the findings reported in §3.1.

Dataset We use the WIKISHORT dataset provided by Yuan et al. (2018) [3]. The dataset contains thousands of English and Chinese wikipedia articles, and each article is labeled with one of the six categories, film, music, animals, politics, religion, and food. We use the train split in the dataset for training, validate with test split. The train split includes 7730 and 7095 English and Chinese articles respectively. To make the classifier better differentiate articles that is not one of the six categories, we uniformly sample 1500 wikipedia articles from both English and Chinese wikidump [4] and add them into the training data with 'other' label.

Preprocessing WIKISHORT provides shortened text of Wikipedia articles. To generate the shortened text of articles with 'other' label, we use the text from the first paragraph of the articles and follow similar preprocessing as in Yuan et al. (2018). For English articles, we lowercase the paragraph, tokenize by word, filter out stop words, lemmatize the words and remove all the punctuations. For Chinese articles, we apply Stanford CoreNLP Tokenizer [5] to segment the Chinese text.

Training We train two multi-nomial logistic regression models, one for English and one for Chinese. To extract features from the shortened text, we compute tf-idf weights from the training data. We use the LogisticRegression implementation of SCIKIT-LEARN (Pedregosa et al., 2011) with default setting to train the models. We tune the regularization term C on validation set. However, we do not observe significant accuracy difference between different C values, therefore we use default setting $C = 1.0$. The accuracy on the test set of the English and Chinese classifiers are 92.5% and 77.9%, respectively.

Prediction on Wikipedia articles To estimate the category distribution of English and Chinese Wikipedia articles, we uniformly sampled 8500 articles from wikidump for each language, follow the same preprocessing and feature extraction as how we does for the training data, and we use the classifiers to predict the category of each article. After removing articles labeled "other", we re-normalize the distribution of the prediction of the main six categories and report the values in 2.

[3] https://github.com/forest-snow/mtanchor_demo#data

[4] https://dumps.wikimedia.org/

[5] https://stanfordnlp.github.io/stanza/tokenize.html

B Computing STDM score

In this section we provide the experimental details of the STDM evaluations performed in §4.1 and §4.2.

Dataset In §4.1, we have described the details for constructing datasets from EuroParl and OpenSubtitles with different amount of STDM.

For the experiments in §4.2, we evaluate the STDM score on datasets with known language origins. We use three different data sources, including WMT newstest sets, MTNT (Michel and Neubig, 2018) and a social media platform (SMD). For the WMT newstest sets, we combine datasets from year 2014 to 2019 and sample 5000 sentences for each language. For the analysis on MTNT dataset, we use the train split in the dataset and sample 5000 sentences for each language. For SMD, 20000 sentences are sampled for each language.

Preprocessing We use SentencePiece (Kudo and Richardson, 2018) to learn a BPE vocabulary of size 10000 over the combined English text corpus of all datasets. We preprocess sentences from all datasets with BPE and remove sentences with less than 10 BPE tokens.

Topic Learning and STDM Score Computing For each dataset, we derive the TF-IDF matrix from the preprocessed sentences and perform an SVD decomposition of the matrix. We retain the top 400 eigenvalues and collect the corresponding topic representations of the original sentences. The topic representations of the sentences are used to calculate the STDM score as described in §4.

C The Effect of Translationese

In §3 we have made the assumption that the effect of translationese is negligible when estimating STDM. However, there are previous studies showing clear artifacts in (human) translations (Baker, 1993; Zhang and Toral, 2019; Toury, 2012). In this section we aim at assessing whether our STDM score is affected by translationese.

We consider the WMT'17 De-En dataset from Ott et al. (2018) which contains double translations of source and target originating sentences.

From this, we construct paired inputs and labels, $\{(h_{s\to t}(h_{t\to s}(\text{tgt}(z^t))), 1)\} \cup \{(\text{tgt}(z^t), 0)\}$, and train two classifiers to predict whether or not the input is translationese. The first classifier takes as input a TF-IDF representation w of the sentence, while the second classifier takes only the corresponding topic distribution: $\bar{V}w$. On this binary task a linear classifier achieves 58% accuracy on the test set with TF-IDF input representations, and only 52% when given just the topic distribution. If we apply the same binary classifier in the topic space to discriminate between sentences originating in the source and target domain ($\text{tgt}(z^t)$ VS. $h_{s\to t}(\text{src}(z^s))$), the accuracy increases to 64%.

We conclude that once we control for domain effect (by discriminating the same set of sentences in their original form versus their double translationese form), the accuracy is much lower than previously reported (Zhang and Toral, 2019), and working in the topic space further removes translationese artifacts. Therefore, the STDM score computed in the topic space is unlikely affected by such artifacts and captures the desired discrepancy between the source and the target domains.

D Self-Training

1 **Data:** Given a parallel dataset $\mathcal{P}$ and a source monolingual dataset $\mathcal{M}^s$ with N^s examples;
2 **Noise:** Let $n(x)$ be a function that adds noise to the input by dropping, swapping and blanking words;
3 **Hyper-params:** Let k be the number of iterations and $A_1 < \cdots < A_k \leq N_S$ be the number of samples to add at each iteration;
4 Train a forward model: $\overrightarrow{\theta} = \arg\max_\theta \mathbb{E}_{(x,y)\sim\mathcal{P}} \log p(y|x;\theta)$;
5 **for** t ***in*** $[1 \dots k]$ **do**
6 forward-translate data: $(\hat{y}^s, v) \approx \arg\max_z p(z|x^s; \overrightarrow{\theta})$, for $x^s \in \mathcal{M}^s$, where v is the model score;
7 Let $\bar{\mathcal{M}}^s \subset \mathcal{M}^s$ containing the top-A_t highest scoring examples according to v;
8 re-train forward model: $\overrightarrow{\theta} = \arg\max_\theta \mathbb{E}_{(x,y)\sim\mathcal{Q}} \log p(y|x;\theta)$ with $\mathcal{Q} = \mathcal{P} \cup \{n(x^s), \hat{y}^s\}_{x^s\sim\bar{\mathcal{M}}^s}$.
end

Algorithm 1: Self-Training algorithm.

Alg. 1 describes self-training, a data augmentation method that leverages $\mathcal{M}^s$, using the notation of §5. First, a baseline forward model is trained on the parallel data (line 4). Second, this initial model is applied to the source monolingual data (line 6). Finally, the forward model is re-trained from random initialization by augmenting the original parallel dataset with the forward-translated data. As with BT, the parallel dataset receives more weight in the loss. In order to increase robustness to prediction mistakes, we make the algorithm iterative and add only the examples for which the model was most confident (line 3, loop in line 5 and line 7). In our experiments we iterate three times. We also inject noise to the input sentences, in the form of word swap and drop (Lample et al., 2018), to further improve generalization (line 8).

E Hyper-parameters Used in MT Experiments

In this section, we report the hyper-parameters used in Sec. 6.

In all our experiments we use the standard machine learning methodology of model cross-validation to select hyper-parameters. We train models with several random combination of hyper-parameters and select the best configuration based on the performance on the validation set. We finally report results on the test set.

Data The full list of datasets we have considered in this work with some basic statistics is reported in Tab. 4

We jointly learn BPEs on both source and target languages. First, we train an MT system on the parallel data for each BPE setting, with values in the set {3000, 5000, 10000, 20000}. Then, we select the number of BPE tokens by selecting the setting that yields the best performance on the validation set. We report below the value that worked best in each experiment.

Loss and Optimizer All models are trained using cross-entropy loss with label smoothing (Szegedy et al., 2016) equal to 0.2. The optimizer is Adam (Kingma and Ba, 2015) with beta1 = 0.9, beta2 = 0.98 and warm-up steps 4000. We share embeddings across encoder and decoder, and input and output lookup tables. We use fixed batch size with 4000 tokens per GPU, and we train with 4 GPUs in fp16. Other optimization hyper-parameters are reported below.

Model Architecture and Training To decide the model architecture hyper-parameters for different amount of parallel and monolingual data, we use random search to find the setting that yields the best performance on validation set. The range of values that we used in our random hyper-parameter search are:

MT Experiments	Language Pair	BPE size	Origin	Parallel data		Monolingual data	
				Domain	# sents (train/valid/test)	Domain	# sents
Controlled Setting	Fr->En	5000	source	Europarl	10K / 10K / 10K	Europarl	900K
			target	OpenSubtitle	10K / 10K / 10K	OpenSubtitle	900K
Low-Resource MT	Ne->En	5000	source	Social Media	40K / 5K / 5K	Social Media	1.8M
			target	Social Media	7.5K / 5K / 5K	Social Media	1.8M
	En-My	10000	source	ALT(News)	18K / 1K / 1K	NewsCrawl	5M
			target	-	-	CommonCrawl	100K

Table 4: Datasets used for the machine translation experiments of Sec. 6

- Layers: {4, 5, 6}
- Embedding dimension: {256, 512, 1024}
- Inner-layer dimension: {512, 1024, 2048, 4096}
- Attention Heads: {2, 4, 8, 16}

We report the model architecture of each experiment in §E.1.

For each data point which corresponds to a particular combination of $(\mathcal{P}, \mathcal{M}^s, \mathcal{S}^t, \alpha, \beta,$ training procedure$)$, we use random search to sweep over hyper-parameters. The hyper-parameters are dropout rate, learning rate, source side noise level for ST experiments and upsample ratio between parallel data, back-translated data and self-translated data. For all experiments, the dropout rates are {0.1, 0.2, 0.3, 0.4, 0.5}, and the learning rate takes values in {0.0007, 0.001, 0.003, 0.005}.

E.1 Hyper-parameter for Controlled Setting Experiments

We use the same BPEs for all the experiments in the controlled setting of §6.1. The BPE size is 5000 and it is shared between English and French.

Varying amount of STDM. We use embedding dimension 256, inner-layer dimension 512, 2 attention heads. We sweep the upsampling ratio of parallel data, back-translated data and self-training data in a range between 1 and 8.

We train the ST system with 2 iterations ($k = 2$), where $A_1 = 30K$ and $A_2 = 100K$. In each iteration, we use random search to sweep over different source side noise, and we report the model with the best performance based on validation set. The values of input noise are as follows: Word shuffling {0, 2, 3}, word dropout {0.0, 0.1, 0.2}, word blank {0.0, 0.1, 0.2}.

Varying amount of monolingual data. When the monolingual data has less than 300k sentences, we use embedding dimension 256, inner-layer dimension 512 and 2 attention heads. For bigger monolingual datasets, we use embedding dimension 1024, inner-layer dimension 2048 and 8 attention heads. We sweep the upsampling ratio of parallel data, back-translated data and self-training data in a range between 1 and 8.

We train the ST system with 5 iterations ($k = 5$), where $A_1 = 10K$, $A_2 = 30K$, $A_3 = 100K$, $A_4 = 300K$ and $A_5 = 900K$. The values of input noise are as follows: Word shuffling {0, 2, 3}, word dropout {0.0, 0.1, 0.2}, word blank {0.0, 0.1, 0.2}.

Varying amount of in-domain data. When training only with parallel data, we use embedding dimension 256, inner-layer dimension 512, and 2 attention heads. For models trained with BT-only and parallel + BT instead, we use embedding dimension 1024, inner-layer dimension 2048 and 8 attention heads. We sweep the parallel data upsampling ratio in a range between 1 and 16.

E.2 Hyper-parameter for Low-Resource MT Experiments

We use a joint BPE tokenization with 5000 tokens for En $\rightarrow$ Ne and 10000 tokens for En $\rightarrow$ My.

On both datasets (Ne $\rightarrow$ En and En $\rightarrow$ My), the baseline trained only on the parallel dataset is a 5-layer transformer model with 512 embedding dimensions, 2048 inner-layer dimensions and 8 attention heads. For models trained with BT and ST, we use a 6-layer transformer with 1024 embedding dimensions, 2048 inner-layer dimensions and 8 attention heads. We sweep the parallel data and ST data ratio between 1 and 32.

For both language pairs, we train the ST systems with 3 iterations ($k = 3$). For Ne $\rightarrow$ En, we use $A_1 = 600$K, $A_2 = 1$M, $A_3 = 1.8$M. For En $\rightarrow$ My, we use $A_1 = 1$M, $A_2 = 3$M and $A_3 = 5$M. The values of input noise are as follows: Word

shuffling {0, 2, 3}, word dropout {0.0, 0.1, 0.2},
word blank {0.0, 0.1, 0.2}.