Improving Robustness in Multilingual Machine Translation via Data Augmentation

Anonymous ACL submission

Abstract

Multilingual humans can and do seamlessly switch back and forth between languages when communicating. However, multilingual (machine) translation models are not robust to such sudden changes. In this work, we explore the robustness of multilingual MT models to language switching and propose checks to measure switching capability. We also investigate simple and effective data augmentation methods that can enhance robustness. A glass-box analysis of attention modules demonstrates the effectiveness of these methods in improving robustness.

1 Introduction

Neural machine translation (NMT) (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017) has made significant progress, from supporting only a pair of languages per model to now simultaneously supporting up to hundreds of languages (Johnson et al., 2017; Zhang et al., 2020; Tiedemann, 2020; Gowda et al., 2021b). Multilingual NMT models have been deployed in production systems and are actively used to translate across languages in day-to-day settings (Wu et al., 2016; Turovsky, 2017; Mohan and Skotdal, 2021). A great many metrics for evaluation of machine translation have been proposed (Doddington, 2002; Banerjee and Lavie, 2005; Snover et al., 2006; Gowda et al., 2021a; Popović, 2015); simply citing a more comprehensive list would exceed space limitations, and inevitably the BLEU metric (Papineni et al., 2002) remains the most popular choice, however nearly all approaches consider translation in the context of a *single sentence*. Even approaches that generalize to support translation of multiple languages (Zhang et al., 2020; Tiedemann, 2020; Gowda et al., 2021b) continue to use the single-sentence paradigm. In reality, however, multilingual environments involve switching between languages and scripts. For instance, the European Parliament[1] and Parliament of India[2] hold debates in multilingual environments where speakers seamlessly switch languages. Figure 1 shows an example of language switching between two Indian languages.

Original: *"bandaaginda bari bageeche ke bahar-e iddivi. kahaani ke andhar bandu bidona. kaam bolo saab."*

English Translation: "From the time I've reached here, we've stayed outside of the topic. Let's come into the matter. Tell me the work, sir."

Figure 1: Demonstration of language switching between Kannada and Hindi. The original dialogue is taken from an Indian movie. Such seamless language switching is common among multilingual speakers.

In this work, we show that, as commonly built, multilingual NMT models are *not robust* to multi-sentence translation, especially when language switching is involved. The contributions of this work are outlined as follows: Firstly, inspired by CHECKLIST (Ribeiro et al., 2020), a few simple but effective checks for improving the test coverage in multilingual NMT evaluation are described (Section 2). Secondly, we explore training data augmentation techniques such as concatenation and noise addition in the context of multilingual NMT (Section 3). Third, using a many-to-one multilingual translation task setup (Section 4), we investigate the relationship between training data augmentation methods and their impact on multilingual test cases. Fourth, we conduct a glass-box analysis of cross-attention in the Transformer architecture and show visually as well as quantitatively that the models trained with concatenated training sentences learn a more sharply focused attention mechanism than others. Finally, we examine how our data aug-

[1] https://www.europarl.europa.eu/doceo/document/CRE-9-2021-11-10_EN.pdf

[2] https://web.archive.org/web/20220105061052/http://loksabhadocs.nic.in/debatestextmk/17/VII/01.12.2021.pdf

mentation strategies generalize to multi-sentence translation for a variable number of sentences, and determine that two-sentence concatenation in training is sufficient model many-sentence concatenation in inference (Section 5.2).

2 Multilingual Translation Evaluation: Additional Checks

Inspired by the behavior testing paradigm in software engineering, Ribeiro et al. (2020) propose a CHECKLIST to test beyond the accuracy of NLP models. The central idea of CHECKLIST is that given any held-out set, one can improve the coverage of testing by modifying the set in a systematic way designed to test linguistic capabilities of natural language processing (NLP) modeling. Some of the modifications CHECKLIST employs are: synonym replacement, named entity replacement, negation, etc. Although these checks are straightforward in tasks such as sentiment classification, they are non-trivial in machine translation between varieties of languages. Nevertheless, the principles of behavior testing and their application to improve test coverage in machine translation are intriguing. We, therefore, explore suitable checks in the context of multilingual NMT.

Definitions: Translation tasks are categorized as *bilingual* if a single source language is translated to a single target language, and *multilingual* if two or more languages are on either of the source or target side. Multilingual tasks are further sub-classified based on the number of languages and the side they are on as many-to-one, one-to-many, and many-to-many. In this work, we focus on many-to-one (i.e. many source, one target) multilingual translation.

Notation: For simplicity, consider a many-to-one model that translates sentences from K source languages, $\{L_k | k = 1, 2, ...K\}$, to a target language, T. Let $x_i^{(L_k)}$ be a sentence in the source language L_k, and let its translation in the target language be $y_i^{(T)}$; where unambiguous we omit the superscripts.

We propose the following checks to be used for multilingual NMT:

C-SL: Concatenate consecutive sentences in the same language. It is not always trivial to determine sentence segmentation in continuous language. This check thus tests if the model is invariant to a missed segmentation. This check is possible if and only if held-out set sentence order preserves the coherency of the original document. Formally,

$$x_i^{(L_k)} + x_{i+1}^{(L_k)} \rightarrow y_i + y_{i+1}$$

In practice, we use a space character to join sentences, indicated by the concatenation operator '+'.[3]

C-TL: Consecutive sentences in the source and target languages. This check tests if the translator can preserve phrases that are already in the target language, and if the translator can translate in the presence of code and language switching settings. For completeness, we can test both source-to-target and target-to-source language switching, as follows:

$$x_i^{(L_k)} + y_{i+1} \rightarrow y_i + y_{i+1} \quad (121)$$
$$y_i + x_{i+1}^{(L_k)} \rightarrow y_i + y_{i+1} \quad (122)$$

Similar to C-SL, this check also requires the held-out set sentence order to preserve the coherency of the original document.

C-XL: This check tests if a multilingual translator is agnostic to language switching. This check is created by concatenating consecutive sentences across source languages. This is possible iff the held-out sets are multi-parallel across languages, and, similar to the previous two, each preserves the coherency of the original documents. Given two languages L_k and L_m, we obtain a test sentence as follows:

$$x_i^{(L_k)} + x_{i+1}^{(L_m)} \rightarrow y_i + y_{i+1}$$

R-XL: This check tests if a multilingual translator can function in light of a topic switch among its supported source languages. For any two languages L_k and L_m and random positions i and j in their original corpus, we obtain a test segment by concatenating them as:

$$x_i^{(L_k)} + x_j^{(L_m)} \rightarrow y_i + y_j$$

This method makes the fewest assumptions about the nature of held-out datasets, i.e., unlike previous methods, neither multi-parallelism nor coherency in sentence order is necessary.

[3] We focus on orthographies that use space as a word-breaker. In orthographies without a word-breaker, joining may be performed without any glue character.

3 Achieving Robustness via Data Augmentation Methods

In the previous section, we described several ways of improving *test* coverage for multilingual translation models. In this section, we explore *training* data augmentation techniques to improve robustness to language switching settings.

3.1 Concatenation

Concatenation of training sentences has been proven to be a useful data augmentation technique; Nguyen et al. (2021) investigate key factors behind the usefulness of training segment concatenations in *bilingual* settings. Their experiments reveal that concatenating random sentences performs as well as consecutive sentence concatenation, which suggests that discourse coherence is unlikely the driving factor behind the gains. They attribute the gains to three factors: context diversity, length diversity, and position shifting.

In this work, we investigate training data concatenation under *multilingual* settings, hypothesizing that concatenation helps achieve the robustness checks that are described in the prior section. Our training concatenation approaches are similar to our check sets, with the notable exception that we do not consider consecutive sentence training specifically, both because of Nguyen et al. (2021)'s finding and because training data gathering techniques can often restrict the availability of consecutive data (Bañón et al., 2020). We investigate the following sub-settings for concatenations:

CatSL: Concatenate a pair of source sentences in the same language, using space whenever appropriate (e.g. languages with space separated tokens).

$$x_i^{(L_k)} + x_j^{(L_k)} \rightarrow y_i + y_j$$

CatXL: Concatenate a pair of source sentences, without constraint on language.

$$x_i^{(L_k)} + x_j^{(L_m)} \rightarrow y_i + y_j$$

CatRepeat: The same sentence is repeated and then concatenated. Although this seems uninteresting, it serves a key role in ruling out gains possibly due to data repetition and modification of sentence lengths.

$$x_i^{(L_k)} + x_i^{(L_k)} \rightarrow y_i + y_i$$

3.2 Adding Noise

We hypothesize that introducing noise during training might help achieve robustness and investigate two approaches that rely on noise addition:

DenoiseTgt: Form the source side of a target segment by adding noise to it. Formally, $noise(y;r) \rightarrow y$, where hyperparameter r controls the noise ratio. Denoising is an important technique in unsupervised NMT (Artetxe et al., 2018; Lample et al., 2018).

NoisySrc: Add noise to the source side of a translation pair. Formally, $noise(x;r) \rightarrow y$. This resembles back-translation (Sennrich et al., 2016a) where augmented data is formed by pairing noisy source sentences with clean target sentences.

The function $noise(...;r)$ is implemented as follows: (i) $r\%$ of random tokens are dropped, (ii) $r\%$ of random tokens are replaced with random types uniformly sampled from vocabulary, and (iii) $r\%$ of random tokens' positions are displaced within a sequence. We use $r = 10\%$ in this work.

Language	In-domain	All-data
Bengali (BN)	23.3k/0.4M/0.4M	1.3M/19.5M/21.3M
Gujarati (GU)	41.6k/0.7M/0.8M	0.5M/07.2M/09.5M
Hindi (HI)	50.3k/1.1M/1.0M	3.1M/54.7M/51.8M
Kannada (KN)	28.9k/0.4M/0.6M	0.4M/04.6M/08.7M
Malayalam(ML)	26.9k/0.3M/0.5M	1.1M/11.6M/19.0M
Marathi (MR)	29.0k/0.4M/0.5M	0.6M/09.2M/13.1M
Oriya (OR)	32.0k/0.5M/0.6M	0.3M/04.4M/05.1M
Punjabi (PA)	28.3k/0.6M/0.5M	0.5M/10.1M/10.9M
Tamil (TA)	32.6k/0.4M/0.6M	1.4M/16.0M/27.0M
Telugu (TE)	33.4k/0.5M/0.6M	0.5M/05.7M/09.1M
All	326k/5.3M/6.1M	9.6M/143M/175M

Table 1: Training dataset statistics: *segments / source / target tokens*, before tokenization.

Name	Dev	Test
Orig	10k/140.5k/163.2k	23.9k/331.1k/385.1k
C-SL	10k/281.0k/326.4k	23.9k/662.1k/770.1k
C-TL	10k/303.7k/326.4k	23.9k/716.1k/770.1k
C-XL	10k/283.9k/326.4k	23.9k/670.7k/770.1k
R-XL	10k/216.0k/251.2k	23.9k/514.5k/600.5k

Table 2: Development and test set statistics: *segments / source / target tokens*, before tokenization. The row named 'Orig' is the union of all ten individual languages' datasets, and the rest are created as per definitions in Section 2. Dev-Orig set is used for validation and early stopping in all our multilingual models.

C-SL	BN-1 + BN-2	আগামী ২০২২ সালের মধ্যে এই কাজ সম্পূর্ণ করার লক্ষ্যমাত্রা স্থির হয়েছে। প্রধানমন্ত্রী বলেন, সরকার সুনির্দিষ্ট লক্ষ্যমাত্রা এবং সময়সীমার মধ্যেবিভিন্ন ধরনের প্রকল্প রূপায়ণের কাজ করে যাচ্ছে।
	EN-1 + EN-2	He said the aim is to complete this task by 2022. The Prime Minister said that the Government is working on various schemes with clear targets and timelines.
C-XL	BN-1 + GU-2	আগামী ২০২২ সালের মধ্যে এই কাজ সম্পূর্ণ করার লক্ষ্যমাত্রা স্থির হয়েছে। પ્રધાનમંત્રીએ જણાવ્યું કે સરકાર સ્પષ્ટ લક્ષ્યો અને સમયસૂચકતા સાથે અનેક યોજનાઓ પર કામ કરી રહી છે.
	EN-1 + EN-2	He said the aim is to complete this task by 2022. The Prime Minister said that the Government is working on various schemes with clear targets and timelines.
C-TL	BN-1 + EN-2	আগামী ২০২২ সালের মধ্যে এই কাজ সম্পূর্ণ করার লক্ষ্যমাত্রা স্থির হয়েছে। The Prime Minister said that the Government is working on various schemes with clear targets and timelines.
	EN-1 + EN-2	He said the aim is to complete this task by 2022. The Prime Minister said that the Government is working on various schemes with clear targets and timelines.
R-XL	KN-m + HI-n	ನಾನು ಸಾರ್ವಜನಿಕರನ್ನು ಉದ್ದೇಶಿಸಿ ಭಾಷಣ ಮಾಡಲಿದ್ದೇನೆ. राज्यांना सुप्रशासनाच्या आधारावर मानांकन देण्यात येते.
	EN-m + EN-n	I will also address a public meeting. States are being rated on parameters of Good Governance.

Table 3: Concatenated sentence examples from the development set. Bengali (BN), Gujarati (GU), Kannada (KN), and Hindi (HI) are chosen for illustrations; similar augmentations are performed for all other languages in the corpus. Indices 1 and 2 indicate consecutive positions, and m and n indicate random positions.

4 Setup

4.1 Dataset

We use publicly available datasets from The Workshop on Asian Translation 2021 (WAT21)'s *MultiIndicMT* (Nakazawa et al., 2021)[4] shared task. This task involves translation between English(EN) and 10 Indic Languages, namely: Bengali(BN), Gujarati(GU), Hindi(HI), Kannada(KN), Malayalam(ML), Marathi(MR), Oriya(OR), Punjabi(PA), Tamil(TA) and Telugu(TE). The development and held-out test sets are multi-parallel and contain 1,000 and 2,390 sentences, respectively. The training set contains a small portion of data from the same domain as the held-out sets, as well as additional datasets from other domains. All the training data statistics are given in Table 1. We focus on the Indic→English (many-to-one) translation direction in this work.

Following the definitions in Section 2, we create C-SL, C-TL, C-XL, and R-XL versions of development and test sets; statistics are given in Table 2. An example demonstrating the nuances in all these four methods is shown in Table 3. Following the definitions in Section 3, we create CatSL, CatXL, CatRpeat, DenoiseTgt, and NoisySrc augmented training segments. For each of these training corpus augmentation methods, we restrict the total augmented sentences to be roughly the same number of segments as the original corpus, i.e., 326k and 9.6M segments in the in-domain and the all-data setup, respectively.

4.2 Model and Training Process

We use a Transformer base model (Vaswani et al., 2017) which has 512 hidden dimensions, 6 encoder and decoder layers, 8 attention heads, and intermediate feedforward layers of 2048 dimensions. We use a Pytorch based NMT toolkit.[5] Tuning the vocabulary size and batch size are important to achieve competitive performance. We use byte-pair-encoding (BPE) (Sennrich et al., 2016b), with vocabulary size adjusted as per the recommendations from Gowda and May (2020). Since the source side has many languages and the target side has only a single language, we use a larger source vocabulary than that of the target. The source side vocabulary contains BPE types from all 11 languages (i.e., ten source languages and English), whereas to improve the efficiency in the decoder's softmax layer, the target vocabulary is restricted to contain English only. Our in-domain limited-data setup learns BPE vocabularies of 30.4k and 4.8k types for source and target languages. Similarly, the all-data setup learns 230.4k and 63.4k types. The training batch size used for all our multilingual models is 10k tokens for the in-domain limited-data setup, and 25k tokens for the larger all-data setup. The batch size for the baseline bilingual models is adjusted as per data sizes using *'a thousand per million tokens'* rule of thumb that we have come to devise with a maximum of 25k tokens. The median sequence lengths in training after subword segmentation but before sentence concatenation are 15 on the Indic side and 17 on the English side. We model sequence lengths up to 512 time steps during training. We use the same learning rate schedule as Vaswani et al. (2017). We train our models until a maximum of 200k optimizer steps, and use early stopping with a patience of 10 validations. Validations are performed after every 1000 optimizer steps. All our models are trained using one Nvidia A40 GPU per setting. The smaller in-domain setup takes less than 24 hours per run, whereas the larger all-data setup takes at most 48 hours per run (or less

[4] http://lotus.kuee.kyoto-u.ac.jp/WAT/indic-multilingual/

[5] Additional details are withheld at the moment to preserve the anonymity of authors. All code, data, and models will be publicly released.

when early stopping criteria are reached). We run each experiment two times and report the average. During inference, we average the last 5 checkpoints and use a beam decoder of size 4 and length penalty of $\alpha = 0.6$ (Vaswani et al., 2017; Wu et al., 2016).

5 Results and Analysis

First, to test our setup with its various hyperparameters such as vocabulary and batch size, we train bilingual models using in-domain data, similar to WAT21 organizer baselines. As shown in Table 4, our baselines achieve competitive BLEU scores (Papineni et al., 2002).[6] Next, we train multilingual many-to-one models for both in-domain and all data.

Table 5 presents our results from a limited quantity in-domain dataset. The baseline model (#I1) has strong performance on individual sentences, but degrades on held-out sets involving missed sentence segmentation and language switching. Experiments with concatenated data, namely CatXL (#I3) and CatSL (#I4), while they appear to make no improvements on regular held-out sets, make a significant improvement in BLEU scores on C-SL, C-XL, and R-XL. Furthermore, both CatSL and CatXL show a similar trend. While they also make a small gain on the C-TL setting, DenoiseTgt method is clearly an out-performer on C-TL. The model that includes both concatenation and denoising (#I7) achieves consistent gains across all the robustness check columns. In contrast, the CatRepeat (#I2) and NoisySrc (#I5) methods do not show any gains.

Our results from the all-data setup are provided in Table 6. While none of the augmentation methods appear to surpass baseline BLEU on the regular held-out sets (i.e., Avg column), their improvements to robustness can be witnessed similar to the in-domain setup. We show a qualitative example in Table 8.

5.1 Attention Bleed

Figures 2 and 3 visualize cross-attention[7] from our baseline model without augmentation as well as models trained with augmentation. Generally, the NMT decoder is run autoregressively; however, to facilitate the analysis described in this section, we force-decode reference translations and extract cross-attention tensors from all models. The cross-attention visualization between a pair of concatenated sentences, say $(x_{i1}+x_{i2} \rightarrow y_{i1}+y_{i2})$, shows that models trained on augmented datasets appear to have less cross-attention mass across sentences, i.e. in the attention grid regions representing $x_{i2} \leftarrow y_{i1}$, and $x_{i1} \leftarrow y_{i2}$. We call attention mass in such regions *attention bleed*. This observation confirms some of the findings suggested by Nguyen et al. (2021). We quantify attention bleed as follows: consider a Transformer NMT model with L layers, each having H attention heads and a held-out dataset of $\{(x_i\ y_i)|i = 1, 2, ...N\}$ segments. Further more, let each segment (x_i, y_i) be a concatenation of two sentences i.e. $(x_{i1} + x_{i2},\ y_{i1} + y_{i2})$, with known sentence boundaries. Let $|x_i|$ and $|y_i|$ be the sequence lengths after BPE segmentation, and $|x_{i1}|$ and $|y_{i1}|$ be the indices of the end of the first sentence (i.e., the sentence boundary) on the source and target sides, respectively. The average attention bleed across all the segments, layers, and heads is defined as:

$$\bar{B} = \frac{1}{N \times L \times H} \sum_{i=1}^{N} \sum_{l=1}^{L} \sum_{h=1}^{H} b_{i,l,h}$$

where $b_{i,l,h}$ is the attention bleed rate in an attention head $h \in [1, H]$, in layer $l \in [1, L]$, for a single record at $i \in [1, N]$. To compute $b_{i,l,h}$, consider that an attention grid $A^{(i,l,h)}$ is of size $|y_i| \times |x_i|$. Then

$$b_{i,l,h} = \frac{1}{|y_i|} \Big[\sum_{t=1}^{|y_{i1}|} \sum_{s=|x_{i1}|+1}^{|x_i|} A_{t,s}^{(i,l,h)} + \sum_{t=|y_{i1}|+1}^{|y_i|} \sum_{s=1}^{|x_{i1}|} A_{t,s}^{(i,l,h)} \Big]$$

where $A_{t,s}^{(i,l,h)}$ is the percent of attention paid to source position s by target position t at decoder layer l and head h in record i. Intuitively, a lower value of $\bar{B}$ is better, as it indicates that the model has learned to pay attention to appropriate regions. As shown in Table 7, the models trained on augmented sentences achieve lower attention bleed.

[6] WAT21 baseline scores are obtained from http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/, which reports BLEU using an external tokenizer script (`moses-tokenizer.perl`). Apart from the row tagged ‡ in Table 4, which is intended to provide direct comparison to baselines, all other BLEU scores are obtained using SacreBLEU with signature: `BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.4.13`.

[7] Also known as encoder-decoder attention.

	Dev	Test	BN	GU	HI	KN	ML	MR	OR	PA	TA	TE
WAT21 biling indomain ‡		18.6	11.3	26.2	28.2	20.3	13.6	15.1	16.4	23.7	16.1	14.7
Biling; indomain ‡	24.1	21.6	13.2	29.3	32.9	22.7	17.9	16.9	16.4	27.4	18.1	21.0
Biling; indomain	23.9	21.5	13.1	29.2	32.6	22.5	17.7	16.8	16.4	27.3	18.0	20.9
Many-to-one; indomain	26.5	22.7	18.7	25.7	27.8	23.1	21.2	20.8	21.1	25.8	20.6	22.4
Many-to-one; all data	35.0	32.4	26.2	36.8	40.1	31.7	30.0	29.8	30.5	38.8	29.1	30.8

Table 4: Indic→English BLEU scores. Rows indicated by ‡ match the evaluation settings used by WAT21 shared task (i.e., tokenized BLEU). The rows without ‡ are detokenized BLEU obtained from SacreBLEU (Post, 2018). Dev and Test are average across 10 languages.

		Dev					Test				
ID	In-domain	Avg	C-TL	C-SL	C-XL	R-XL	Avg	C-TL	C-SL	C-XL	R-XL
#I1	Baseline (B)	26.5	10.8	17.0	16.9	15.9	22.7	9.4	14.9	14.7	13.6
#I2	B+CatRepeat	25.3	9.9	14.5	14.7	13.3	21.6	8.6	13	13	11.4
#I3	B+CatXL	26.2	12.6	26.1	25.9	26.5	22.6	11.1	22.7	22.5	22.3
#I4	B+CatSL	26.1	13.2	26.1	25.9	26.5	22.6	11.4	22.9	22.6	22.3
#I5	B+NoisySrc	25.2	10.5	16.2	16.0	15.2	21.2	9.1	14.3	14.1	12.9
#I6	B+DenoiseTgt	26.7	40.4	17.9	17.7	16.6	23.2	39.7	15.7	15.4	14.1
#I7	B+CatXL+DenoiseTgt	26.1	55.2	26.3	26.0	26.4	22.6	53.4	23.0	22.6	22.4

Table 5: Indic→English BLEU scores for models trained on in-domain training data only.

		Dev					Test				
ID	All-data	Avg	C-TL	C-SL	C-XL	R-XL	Avg	C-TL	C-SL	C-XL	R-XL
#A1	Baseline (B)	35.0	43.1	30.0	29.5	28.2	32.4	42.2	27.8	27.3	26.1
#A2	B+CatRepeat	34.5	43.7	30.3	29.9	28.8	32.0	42.9	28.0	27.6	26.3
#A3	B+CatXL	34.1	53.3	31.9	33.7	34.4	31.6	52.4	29.7	31.0	31.2
#A4	B+CatSL	33.6	54.0	32.5	32.2	34.3	31.3	53.3	30.4	29.9	31.1
#A5	B+NoisySrc	34.9	42.1	29.8	29.2	27.8	32.3	41.7	27.6	27.1	25.8
#A6	B+DenoiseTgt	33.3	60.0	28.9	28.4	27.3	31.3	59.4	27.1	26.5	25.4
#A7	B+CatXL+DenoiseTgt	33.3	65.8	31.1	33.0	33.6	31.0	64.7	28.9	30.4	30.3

Table 6: Indic→English BLEU scores for models trained on all data. *Abbreviations:* Avg: average across ten languages, C-: consecutive sentences, R-: random sentences, TL: target-language (i.e, English), SL: same-language, XL: cross-language.

		Dev				Test			
ID		C-TL	C-SL	C-XL	R-XL	C-TL	C-SL	C-XL	R-XL
#A1	Baseline (B)	14.3	10.4	10.3	10.1	14.3	10.6	10.5	10.3
#A2	B+CatRepeat	12.3	8.9	8.9	8.6	12.5	9.0	9.0	8.7
#A3	B+CatXL	5.8	7.2	4.3	4.3	5.8	7.2	4.4	4.3
#A4	B+CatSL	5.3	6.2	6.1	5.2	5.4	6.2	6.2	5.2
#A5	B+NoisySrc	17.4	16.1	16.1	15.8	17.5	16.2	16.2	15.9
#A6	B+DenoiseTgt	7.9	8.3	8.4	8.0	8.1	8.5	8.5	8.1
#A7	B+CatXL+DenoiseTgt	4.3	6.8	3.9	4.1	4.4	7.0	4.0	4.1

Table 7: Cross-attention bleed rate (lower is better); all numbers have been scaled from $[0, 1]$ to $[0, 100]$ range for easier interpretation. Models trained on concatenated sentences have lower attention bleed rate. Denoising is better than baseline, but not as much as concatenation. The lowest bleed rate is achieved by using both concatenation and denoising methods.

5.2 Sentence Concatenation Generalization

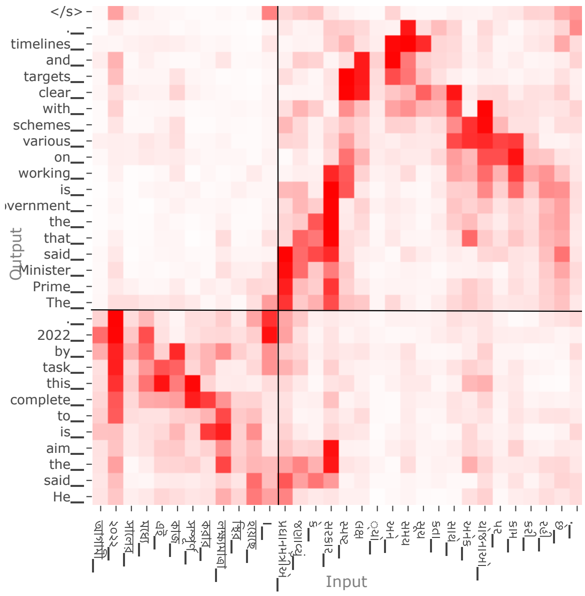
In the previous sections, only two-segment concatenation has been explored; here, we investigate whether more concatenation further improves model performance and whether models trained on two segments generalize to more than two at test time. We prepare a training dataset having up to four sentence concatenations and evaluate on datasets having up to four sentences. As shown in Table 9, the model trained with just two segment concatenation achieves a similar BLEU as model trained with up to four concatenations.

Source	আগামী ২০২২ সালের মধ্যে এই কাজ সম্পূর্ণ করার লক্ষ্যমাত্রা স্থির হয়েছে। પ્રધાનમંત્રીએ જણાવ્યું કે સરકાર સ્પષ્ટ લક્ષ્યો અને સમયસૂચકતા સાથે અનેક યોજનાઓ પર કામ કરી રહી છે.
Reference	He said the aim is to complete this task by 2022. The Prime Minister said that the Government is working on various schemes with clear targets and timelines.
Baseline	He said the Government is working on several schemes with clear objectives and timelines.
B+CatRepeat	The target is to be completed by 2022, the Prime Minister said that the Government is working on several schemes with clear targets and timelines. is the of of of of of of of of of of of of
B+CatXL	The target is to complete it by 2022. The Prime Minister said that the Government is working on a number of schemes with clear targets and timelines.
B+CatSL	We have set a target to complete this task by 2022. The Prime Minister said that the Government is working on a number of schemes with clear objectives and timelines.
B+NoisySrc	The Prime Minister said that the Government is working on several schemes with clear objectives and timelines.
B+DenoiseTgt	He said the Government is working on several schemes with clear objectives and timelines.
B+CatXL +DenoiseTgt	We have set a target of completing it by 2022. The Prime Minister said that the Government is working on a number of schemes with clear targets and timelines.

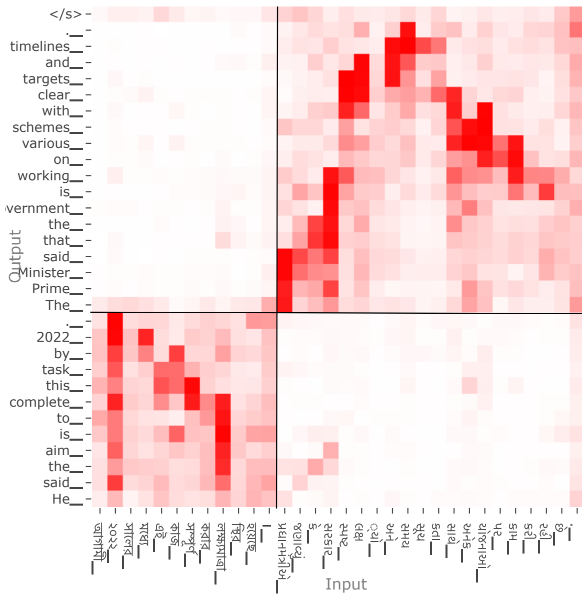
Table 8: Example translations from the models trained on all-data setup. See Table 6 for quantitative scores of these models, and Figures 2 and 3 for a visualization of cross-attention.

	Dev		Test	
	C-SL	C-4SL	C-SL	C-4SL
Baseline / no join	30.0	27.8	27.8	25.7
Up to two joins	31.9	28.9	29.7	26.7
Up to four joins	31.0	28.9	28.8	26.8

Table 9: Indic→English BLEU on held out sets containing up to 4 consecutive sentence concatenations in same language (C-4SL). The two sentences dataset (C-SL) is also given for comparison. The model trained on two concatenated sentences achieves comparable results on C-4SL, indicating that no further gains are obtained from increasing concatenation in training.



(a) Baseline model without sentence concatenation (#A1)



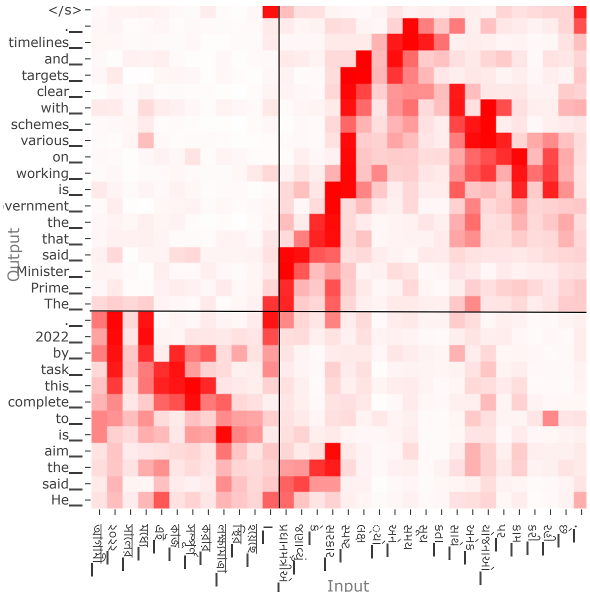
(b) Model trained with concatenated sentences (#A3)

Figure 2: Cross-attention visualization from baseline model and concatenated (cross-language) model. For each position in the grid, only the maximum value across all attention-heads from all the layers is visualized. The darker color implies more attention weight, and the black bars indicate sentence boundaries. The model trained on concatenated sentences has more pronounced cross-attention boundaries than the baseline, indicating less mass is bled across sentences.

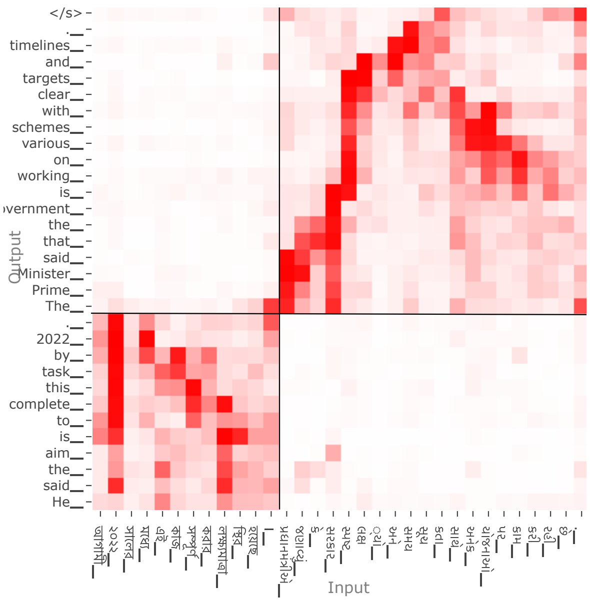
6 Related Work

Machine Translation Robustness: MT robustness has been investigated before within the scope of bilingual translation settings. Some of those efforts include robustness against input perturbations (Cheng et al., 2018), naturally occurring noise (Vaibhav et al., 2019), and domain shift (Müller et al., 2020). However, as we have shown in this work, multilingual translation models can introduce new aspects of robustness to be desired and evaluated. The robustness checklist proposed by Ribeiro et al. (2020) for NLP modeling in general does not cover translation tasks, whereas our work focuses entirely on the multilingual translation task.

Augmentation Through Concatenation: Concatenation has been used before as a simple-to-incorporate augmentation method. Concatenation can be limited to consecutive sentences as a means to provide extended context for translation (Tiedemann and Scherrer, 2017; Agrawal et al., 2018), or additionally include putting random sentences together, which has been shown to result in gains under low resource settings (Nguyen et al., 2021;



(a) Model trained with DenoiseTgt augmentation (#A6)



(b) Model trained with both CatXL and DenoiseTgt augmentations (#A7)

Figure 3: Cross-attention visualization (... continuation from Figure 2) The model trained on both concatenated and denoising sentences has least attention mass across sentences.

Kondo et al., 2021). While in a multilingual setting such as ours, data scarcity is less of a concern as a result of combining multiple corpora, concatenation is still helpful to prepare the model for scenarios where language switching is plausible. Besides data augmentation, concatenation has also been used to train multi-source NMT models. Multi-source models (Och and Ney, 2001) translate multiple semantically-equivalent source sentences into a *single* target sentence. Dabre et al. (2017) show that by concatenating the source sentences (equivalent sentences from different languages), they are able to train a single-encoder NMT model that is competitive with models that use separate encoders for different source languages. Backtranslation (Sennrich et al., 2016a) is another useful method for data augmentation, however it is more expensive when the source side has many languages, and does not focus on language switching.

Attention Weights: Attention mechanism (Bahdanau et al., 2015) enables the NMT decoder to choose which part of the input to *focus* on during its stepped generation. The attention distributions learned while training a machine translation model, as an indicator of the context on which the decoder is focusing, have been used to obtain word alignments (Garg et al., 2019; Zenkel et al., 2019, 2020; Chen et al., 2020). In this work, by visualizing attention weights, we depict how augmenting the training data guides attention to more neatly focus on the sentence of interest while decoding its corresponding target sentence. We are also able to quantify this by the introduction of the attention bleed metric.

7 Conclusion

We have described simple but effective checks for improving test coverage in multilingual NMT (Section 2), and have explored training data augmentation methods such as sentence concatenation and noise addition (Section 3). Using a many-to-one multilingual setup, we have investigated the relationship between these augmentation methods and their impact on robustness in multilingual translation. While the methods are useful in limited training data settings, their impact may not be visible on single-sentence test sets in a high resource setting. However, our proposed checklist evaluation reveals the robustness improvement in both the low resource as well as high resource settings. We have conducted a glass-box analysis of cross-attention in Transformer NMT showing both visually as well as quantitatively that the models trained with augmentations, specifically, sentence concatenation and target sentence denoising, learn a more sharply focused attention mechanism (Section 5.1). Finally, we have determined that two-sentence concatenation in training corpora generalizes sufficiently to many-sentence concatenation inference (Section 5.2).

8 Ethical Consideration

Limitations: As mentioned in Section 2, some of the multilingual evaluation checks require the datasets to have multi-parallelism, and coherency in the sentence order. When neither multi-parallelism nor coherency in the held-out set sentence order is available, we recommend R-XL. The data augmentation methods proposed in this paper do not require any specialized hardware or software. Our model and training pipeline can be rerun on a variety of GPU models, including the ones with lesser memory. However, some of the large dataset, large vocabulary models may require multiple distributed training processes, and/or multiple gradient accumulation steps to achieve the described batch size.

Scientific Artifacts: This work uses a dataset from The Workshop on Asian Translation 2021 (WAT21)'s *MultiIndicMT* shared task (Nakazawa et al., 2021), which is available for free download at the public URL: `http://lotus.kuee.kyoto-u.ac.jp/WAT/indic-multilingual/`; we do not redistribute this dataset from our servers.

Our NMT pipeline is already publicly available under a license approved by `https://opensource.org`, and will be referenced in the final copy. Our code and scripts used for data preparation, augmentation, as well as model training and evaluation will be made available via a public GitHub repository with an open source-friendly license after the end of the author anonymity period.

Only a subset of checks on robustness in multilingual settings have been discussed. While they serve as starting points for improving robustness, we do not claim that the proposed checks are exhaustive. We have investigated robustness under Indic-English translation task where all languages use space characters as word-breakers; we have not investigated other languages such as Chinese, Japanese etc. The term *Indic* language to collectively reference 10 Indian languages only, similar to *MultiIndicMT* shared task. While the remaining Indian languages and their dialects are not covered, we believe that the approaches discussed in this work generalize to other languages in the same family.

References

Ruchit Rajeshkumar Agrawal, Marco Turchi, and Matteo Negri. 2018. Contextual handling in neural machine translation: Look behind, ahead and on both sides. In *21st Annual Conference of the European Association for Machine Translation*, pages 11–20.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *International Conference on Learning Representations*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrías, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza. 2020. ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.

Yun Chen, Yang Liu, Guanhua Chen, Xin Jiang, and Qun Liu. 2020. Accurate word alignment induction from neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 566–576, Online. Association for Computational Linguistics.

Yong Cheng, Zhaopeng Tu, Fandong Meng, Junjie Zhai, and Yang Liu. 2018. Towards robust neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1756–1766, Melbourne, Australia. Association for Computational Linguistics.

Raj Dabre, Fabien Cromierès, and Sadao Kurohashi. 2017. Enabling multi-source neural machine translation by concatenating source sentences in multiple languages. *CoRR*, abs/1702.06135.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, page 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. 2019. Jointly learning to align and translate with transformer models. 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 4453–4462, Hong Kong, China. Association for Computational Linguistics.

Thamme Gowda and Jonathan May. 2020. Finding the optimal vocabulary size for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3955–3964, Online. Association for Computational Linguistics.

Thamme Gowda, Weiqiu You, Constantine Lignos, and Jonathan May. 2021a. Macro-average: Rare types are important too. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1138–1157, Online. Association for Computational Linguistics.

Thamme Gowda, Zhao Zhang, Chris Mattmann, and Jonathan May. 2021b. Many-to-English machine translation tools, data, and pretrained models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 306–316, Online. Association for Computational Linguistics.

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

Seiichiro Kondo, Kengo Hotate, Tosho Hirasawa, Masahiro Kaneko, and Mamoru Komachi. 2021. Sentence concatenation approach to data augmentation for neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 143–149, Online. Association for Computational Linguistics.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations*.

Krishna Doss Mohan and Jann Skotdal. 2021. Microsoft translator: Now translating 100 languages and counting! Accessed: 2022-01-14.

Mathias Müller, Annette Rios, and Rico Sennrich. 2020. Domain robustness in neural machine translation. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 151–164, Virtual. Association for Machine Translation in the Americas.

Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, Yusuke Oda, and Sadao Kurohashi. 2021. Overview of the 8th workshop on Asian translation. In *Proceedings of the 8th Workshop on Asian Translation (WAT2021)*, pages 1–45, Online. Association for Computational Linguistics.

Toan Q. Nguyen, Kenton Murray, and David Chiang. 2021. Data augmentation by concatenation for low-resource translation: A mystery and a solution. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 287–293, Bangkok, Thailand (online). Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2001. Statistical multi-source translation. In *Proceedings of Machine Translation Summit VIII*, Santiago de Compostela, Spain.

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

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. 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.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

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

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Jörg Tiedemann. 2020. The tatoeba translation challenge – realistic data sets for low resource and multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.

Jörg Tiedemann and Yves Scherrer. 2017. Neural machine translation with extended context. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 82–92, Copenhagen, Denmark. Association for Computational Linguistics.

Barak Turovsky. 2017. Making the internet more inclusive in India. Accessed: 2022-01-14.

Vaibhav Vaibhav, Sumeet Singh, Craig Stewart, and Graham Neubig. 2019. Improving robustness of machine translation with synthetic noise. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1916–1920, Minneapolis, Minnesota. Association for Computational Linguistics.

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*, volume 30. Curran Associates, Inc.

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*.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2019. Adding interpretable attention to neural translation models improves word alignment. *CoRR*, abs/1901.11359.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2020. End-to-end neural word alignment outperforms GIZA++. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1605–1617, Online. Association for Computational Linguistics.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.