Non-fluent synthetic target-language data improve neural machine translation

Víctor M. Sánchez-Cartagena, Miquel Esplà-Gomis, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez

Abstract—When the amount of parallel sentences available to train a neural machine translation is scarce, a common practice is to generate new synthetic training samples from them. A number of approaches have been proposed to produce synthetic parallel sentences that are similar to those in the parallel data available. These approaches work under the assumption that non-fluent target-side synthetic training samples can be harmful and may deteriorate translation performance. Even so, in this paper we demonstrate that synthetic training samples with non-fluent target sentences can improve translation performance if they are used in a multilingual machine translation framework as if they were sentences in another language. We conducted experiments on ten low-resource and four high-resource translation tasks and found out that this simple approach consistently improves translation performance as compared to state-of-the-art methods for generating synthetic training samples similar to those found in corpora. Furthermore, this improvement is independent of the size of the original training corpus, the resulting systems are much more robust against domain shift and produce less hallucinations.

Index Terms—machine translation, low-resource languages, data augmentation, multi-task learning

1 Introduction

Machine translation (MT) —the application of computers to the task of translating a text in one natural language into another without human intervention— is one of the key technologies enabling communication in our globalized world. Millions of users rely on MT on a daily basis for either *assimilation*, the use of the raw MT output to get an idea of the meaning of texts in languages they do not understand, or for *dissemination*, the use of the MT output to create a draft translation which is then manually corrected and published.

The uptake of MT technology has gradually increased over the last ten years [1], mainly motivated by the recent advances in the state-of-the-art approach to MT, namely *neural machine translation* (NMT). NMT models are data intensive and require large amounts of *parallel corpora* in the form of several hundreds of thousands or millions of human-translated sentence pairs used for their training. Besides that, monolingual data has also proven to be a valuable resource to train NMT systems.

Although there exist language pairs, such as English–German or English–Spanish, that have large, freely available parallel corpora, most language pairs may be considered *low-resource* because there is little or no translated text available to train NMT models for them. This problem has been addressed in NMT through different approaches, such as transfer-learning from high-resource language pairs [2], using linguistic annotations [3], training multilingual systems [4] and applying data augmentation strategies [5], [6], i.e., artificially generating additional parallel sentences.

Data augmentation (DA) is formalized by many authors as a solution to a data distribution mismatch problem [7], [8] in which the empirical data distribution of the sentence pairs in the training corpus differs from the true data distribution. DA is therefore used to build unseen synthetic training samples that are plausible under the true data distribution. When these synthetic training samples are used during training, the resulting augmented data distribution is expected to be closer to the true data distribution. The final objective is to obtain models that are able to properly translate sentences plausible under the true data distribution, even though they may be unlikely under the empirical data distribution in the training corpus.

For this reason, most DA approaches pay special attention to avoiding non-fluent target-language sentences in the synthetic training samples [7], [9], as their use may lead to worse NMT models with lower translation performance. In this paper, however, and in contrast to previous works, we describe a DA approach that consists of building synthetic training samples that are deliberately implausible under the true data distribution. The objective is to strengthen the encoder of the NMT model through DA. Voita et al. [10] claim that the influence of encoder representations in the output predictions of an NMT system is higher when the NMT system is trained on large corpora than when it is trained on small corpora. By producing *unlikely* synthetic training samples, especially as regards their target side, we aim at exposing the network to new situations where the target-language prefix does not provide sufficient context to predict the next token, therefore forcing the decoder to rely more on the encoder representations for its predictions; this should be viewed as a desirable trait since MT systems should build upon the source sentence to produce accurate

- *V.M. Sánchez-Cartagena, M. Esplà-Gomis, J.A. Pérez-Ortiz and F. Sánchez-Martínez are with the Dep. de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, Spain.*
  E-mail: {vmsanchez,mespla,japerez,fsanchez}@dlsi.ua.es

V.M. Sánchez-Cartagena, M. Esplà-Gomis, J.A. Pérez-Ortiz and F. Sánchez-Martínez (2023), "Non-fluent synthetic target-language data improve neural machine translation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 46, No. 2, pp. 837–850, doi: 10.1109/TPAMI.2023.3333949.
Accepted version. Published online: November 2023.
https://doi.org/10.1109/TPAMI.2023.3333949

translations. In this way, it is possible to build NMT models for low-resource language pairs that, even though they have been trained on small parallel corpora, are able to behave as if they had been trained on larger training corpora.

Obviously, synthetic training samples with non-fluent target-language sentences cannot be used as if they were original training samples, since their use would harm the target-language model learned by the NMT system. To avoid this, we propose the use of a multi-task learning framework during training. This is easily achieved, without changing the model architecture, by prepending a task-specific artificial token to the source sentence to constrain the kind of output to be produced [4], [11], similarly to what is done in one-to-many multilingual NMT [12] to specify the target language. We term this DA approach as *multi-task learning data augmentation* (MaTiLDA).

Our framework —which extends preliminary work[1] reported in a conference paper by the same authors [13]— does not require elaborate preprocessing steps, training additional systems, or data besides the available training parallel corpora. Experiments with ten low-resource translation tasks show that it systematically outperforms state-of-the art methods aimed at extending the support of the empirical data distribution. Additional experiments on four high-resource translation tasks show that this approach is also able to improve translation performance even in high-resource conditions in which NMT systems are trained on large parallel corpora. Furthermore, we show that this new approach and the standard DA approach, back-translation [9], complement each other and allow further performance improvements when they are used together.

In addition to these experiments, we perform an analysis of the relevance of the encoder and decoder representations in the NMT system output, which shows that, thanks to the added synthetic training samples, MaTiLDA increases the contribution of the source representations generated by the encoder to the decisions made by the NMT decoder during inference. Moreover, systems trained with MaTiLDA are much more robust against domain shift, and produce less hallucinations [14].

The remainder of the paper is organized as follows. The next section briefly describes the neural approach to MT. After that, Sec. 3 describes the DA strategies we follow and evaluate in our experiments and the modifications introduced to the training process of the NMT system. Sec. 4 then describes the experimental settings, whereas Sec. 5 reports and discusses the results obtained on low-resource and high-resource translation tasks. Sec. 6 presents an analysis of the changes in the use of the encoder representations induced by our DA strategies, and an analysis of the tendency to hallucinate of the systems trained with MaTiLDA. The paper ends with a review of the most relevant works in the area of DA for NMT in Sec. 7, followed by some concluding remarks in Sec. 8.

2 Neural machine translation

Given a source sentence, $\mathbf{x} = \langle x_1, ..., x_n \rangle$, and its translation, $\mathbf{y} = \langle y_1, ..., y_m \rangle$, NMT systems factorize the translation probability $p(\mathbf{y}|\mathbf{x})$ as:

$$p(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^{m} p(y_j|\mathbf{y}_{<\mathbf{j}}, \mathbf{x}), \tag{1}$$

where $\mathbf{y}_{<\mathbf{j}}$ stands for the target prefix produced before predicting the j-th token, y_j, in the target language.

Different families of neural networks have been proposed for producing this probability distribution: convolutional [15], recurrent [16] and transformer [17], the latter being the current state of the art. These neural networks are optimized by seeking the model parameters θ^* that maximize the likelihood of the training data $\mathcal{D}$:

$$\theta^* = \arg\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y} \rangle \in \mathcal{D}} p(\mathbf{y}|\mathbf{x}; \theta). \tag{2}$$

Stochastic gradient-based optimization methods are commonly applied to find a local maximum of that likelihood. These optimization methods involve iteratively updating the model parameters using only a mini-batch of sentences $\mathcal{B}$ from the training corpus. The parameters θ are updated in terms of the gradient of a loss $\mathcal{L}(\mathcal{B}, \theta)$ computed over the mini-batch. For instance, parameters in the batched stochastic gradient descent algorithm [18] are updated as follows, where i is the iteration index, n is the number of target words in the mini-batch $\mathcal{B}$ and η is the learning rate:

$$\theta_{i+1} = \theta_i - \eta \frac{1}{n} \nabla \mathcal{L}(\mathcal{B}, \theta). \tag{3}$$

NMT systems are usually optimized using a loss based on cross-entropy. The basic formulation of the cross-entropy loss function is shown below, although some enhancements such as label smoothing have become increasingly popular [19]:

$$\mathcal{L}(\mathcal{B}, \theta) = \sum_{\langle \mathbf{x}, \mathbf{y} \rangle \in \mathcal{B}} \log p(\mathbf{y}|\mathbf{x}; \theta). \tag{4}$$

3 Data augmentation strategies

The DA strategies we follow can be formalised as transformations that are applied to the original training samples to produce synthetic training samples. These synthetic samples are expected to force the NMT system to rely more on the source-language representations generated by the encoder during translation. Most of the transformations described next produce synthetic samples with non-fluent target-language sentences. Some transformations are controlled by a hyperparameter α that determines the fraction of target words affected by the transformation. In what follows, m denotes the amount of words in the original target sentence. Table 1 provides an example of the effect of the different transformations on a single sentence pair.

swap: Pairs of random target words are swapped until only $(1-\alpha) \cdot m$ words remain in their original position. This transformation [20] tries to force the system to trust less the target prefix when generating a new token.

[1] The additional contributions of this paper with respect to the conference paper [13] are as follows: (i) more sophisticated approach for generating the synthetic samples during training; (ii) additional experiments on both low-resource and high-resource translation tasks; (iii) more exhaustive comparison to other DA methods; (iv) improved evaluation of the contribution of the source representations generated by the encoder during inference; and (v) better grounded analysis of the tendency to hallucinate of the models evaluated.

Task	Lang.	Synthetic training sample
original training sample	source	Es gibt andere Möglichkeiten , die Pyramide zu durchbrechen .
	target	There ’s other ways of breaking the pyramid .
swap	target	There . other ways of breaking pyramid ’s the
unk	target	There ’s other UNK of UNK UNK UNK .
source	target	Es gibt andere Möglichkeiten , die Pyramide zu durchbrechen .
reverse	target	. pyramid the breaking of ways other ’s There
mono	target	’s There other ways the pyramid of breaking .
replace	source	Es gibt aufzurüsten kalt , Schach Spezialwissen zu durchbrechen .
	target	There ’s arming cold of breaking chess specialties .

TABLE 1: A German–English, word-aligned training sample (first row) and the result of applying the transformations described in Sec. 3 using $\alpha = 0.5$ for those transformations controlled by this hyperparameter. Words modified by each transformation are coloured; for *swap* and *replace*, a different colour identifies the pair of words that are either swapped or replaced together, respectively.

unk: $\alpha \cdot m$ random target words are replaced by a special UNK token [21] when they are fed to the neural network as previous context ($\mathbf{y_{<j}}$; see Eq. 1) for the prediction of the next target token y_j. Note that, when computing the training loss, the original token, rather than the UNK token, is used as the expected output. This strategy is similar to the word dropout used when preventing posterior collapse in variational autoencoders [22] and makes the target prefix less informative for the prediction of the next target token.

source: The target sentence becomes a copy of the source sentence. Thus, the most efficient way of predicting the right output is checking the encoder representation to copy from the source. Although such training instances have been identified as harmful [23], [24], we empirically found the opposite (see Sec. 5) thanks to the multi-task learning framework defined later in this section.

reverse: The order of the words in the target sentence is reversed. Voita et al. [10] suggest that the influence of the encoder decreases along the target sentence; therefore, by reversing the order we expect the system to learn to use more information from the encoder when generating tokens that usually appear near the end of the sentence.

mono: Target words are reordered to make the alignment between source and target words monotonous by using one-to-many word alignments as in the compression of parallel corpora [25]. By making the alignment between source and target words monotonous, the target sentences become less fluent, so we expect the system to pay more attention to the encoder.

replace: $\alpha \cdot m$ source–target aligned pairs of words are selected at random and replaced by random entries in a bilingual lexicon obtained from the training corpus; to this end, one-to-one word alignments are used.[2] This transformation is likely to introduce words that are difficult to produce by relying only on the target language prefix, thus forcing the system to pay attention to the source words. Fadaee et al. [26] followed a similar approach; however, they constrained the replacements to produce only fluent target sentences.

Using the original training samples together with the synthetic ones, without distinction between them, would degrade the overall translation performance. On the one hand, the system would not be able to learn the kind of output to be produced (e.g. well-formed, fluent target sentences); on the other hand, the system could learn spurious correlations from the synthetic training samples. In order to minimize the negative impact of having non-fluent target sentences in the synthetic training samples while keeping their ability to force reliance on the encoder, we applied the multi-task learning strategy described next.

We organize the original training data in mini-batches as if a vanilla NMT system were trained. Then, for each original sentence pair $\langle \mathbf{x}, \mathbf{y} \rangle$ in a mini-batch, we apply one or more of the transformations aforementioned, depending on the experimental setting. For each transformation i, a new synthetic sentence pair $\langle \hat{\mathbf{x}}^{\mathbf{i}}, \hat{\mathbf{y}}^{\mathbf{i}} \rangle$ is generated, and a new term is added to the loss function to account for its cross-entropy.[3] As mini-batches are created at the beginning of each epoch, the result of applying the transformations that involve random decisions is different for each epoch, thus preventing the system from learning spurious correlations from synthetic data.[4] In addition, we add a token to each source sentence to indicate whether it is part of an original training sample or of synthetic one, and, in the latter case, which transformation was used for its generation. The latter resembles what is done in multilingual NMT [4], [11] to indicate the language of the output to be produced.

Hence, within our MaTiLDA framework, and with r transformations, the cross-entropy loss function described in Eq. (4) turns into the following expression:

$$\mathcal{L}(\mathcal{B},\theta) = \sum_{\langle \mathbf{x},\mathbf{y}\rangle \in \mathcal{B}} \frac{1}{r+1} \sum_{i=0}^{r} \log p(\hat{\mathbf{y}}^{\mathbf{i}} | \langle t_i, \hat{x}^i_1, ..., \hat{x}^i_{\hat{n}_i} \rangle ; \theta) \quad (5)$$

[2] If the number of aligned words is below $\alpha \cdot m$, all available alignments are used.

[3] Note that in all transformations but *replace*, $\hat{\mathbf{x}}^{\mathbf{i}} = \mathbf{x}$.

[4] This differs from previous work [13] in which transformations were applied to the original training samples during the pre-processing of the corpus, and therefore before training.

where $\langle \hat{\mathbf{x}}^{\mathbf{0}}, \hat{\mathbf{y}}^{\mathbf{0}} \rangle$ corresponds to the original training sample $\langle \mathbf{x}, \mathbf{y} \rangle$, $\hat{n_i}$ stands for the length of $\hat{\mathbf{x}}^{\mathbf{i}}$, and t_i denotes the artificial token introduced to indicate whether the training sample is the original one ($i = 0$) or the synthetic one corresponding to the i-th transformation ($i \geq 1$).

4 Experimental settings

We have conducted experiments with five language pairs in a low-resource scenario: English (en) into German (de), Hebrew (he) and Vietnamese (vi), and German into Upper Sorbian (hsb) and Romansh (rm). For this purpose, a collection of corpora commonly considered for evaluating DA techniques in low-resource conditions were used (see Sec. 4.1). In addition, we have conducted experiments with two high-resource language pairs: English into Romanian (ro) and English into German. Both translation directions were evaluated for each language pair, which makes a total of ten low-resource translation tasks and four high-resource translation tasks.

We evaluated the effect of using each of the MaTiLDA transformations, as well as the combination of the best performing ones. We explored the combination of MaTiLDA and back-translation, and we also evaluated four strong DA methods that aim at extending the support of the empirical data distribution: SwitchOut [7], RAML [27], the combination of both (SwOut+RAML) and SeqMix [28]. SwitchOut and RAML replace some words by random samples from the vocabulary: SwitchOut works on the source language, and RAML on the target language. SeqMix creates synthetic training samples by randomly combining parts of two sentences.

4.1 Datasets

For the experiments in low-resource conditions on five different language pairs we used the following corpora. For English–German and English–Hebrew, we followed Guo et al. [29] and Gao et al. [28], and used the training data (speeches of TED and TEDx talks) of the IWSLT 2014 text translation track [30];[5] for development and testing we used the *tst2013* and *tst2014* datasets, respectively. For English–Vietnamese we used, like Wang et al. [7], the training data (also TED talks) of the IWSLT 2015 text translation track [31];[6] datasets *tst2012* and *tst2013* were used, respectively, for development and testing. For German–Upper Sorbian we used the corpora released as part of the WMT 2021 task on very low resource supervised machine translation;[7] for training we used all the corpora allowed for the task; for development and testing we used the *devel* and *devel_test* sets, respectively, provided by the organizers of the task. For German–Romansh we used the law corpus released by Müller et al. [32] as well as the same split into training, development and testing they used.

Pair	# sent.	# left tok.	# right tok.
Low-resource conditions			
Parallel data			
en–de	174,443	3,575,407	3,353,855
en–he	187,817	3,862,985	2,958,136
en–vi	133,317	2,965,962	3,361,789
de–hsb	147,521	2,240,126	1,998,047
de–rm	102,192	1,773,683	2,414,749
Parallel data + back-translated data			
de–en	269,213	5,537,986	5,843,264
he–en	282,587	4,728,840	6,130,842
vi–en	228,087	6,232,006	5,413,428
hsb–de	247,521	4,597,777	4.644.705
rm–de	202,192	5,412,126	4,178,262
High-resource conditions			
Parallel data			
en–ro	612,422	15,919,293	16,149,695
en–de	4,468,840	126,720,053	119,907,183

TABLE 2: Number of sentences and tokens in the training corpora used in our experiments. The first ten rows correspond to the experiments in low-resource conditions, only with parallel data and with parallel and back-translated data; the last two rows correspond to the experiment in high-resource conditions (using only parallel data).

For the combination of MaTiLDA and back-translation [9], we used additional English and German monolingual corpora. In particular, for English we used all the available monolingual English sentences in the IWSLT 2018 shared task on low-resource MT of TED talks after removing those sentences present in the parallel training data described above. For German, we used a corpus with 100,000 sentences randomly sampled from the News Commentary v16 German monolingual corpus.[8]

Finally, we used the following corpora for the experiments in high-resource conditions. For English–German we used the training corpora available for the WMT 2014 shared task on machine translation;[9] for development we used the concatenation of *newstest2012* and *newstest2013*, and for testing *newstest2014*. For English–Romanian, we used the corpora available for the WMT 2016 shared task on machine translation of news; for training we used the concatenation of Europarl v8 and SETIMES2; for development and testing we used *newsdev2016* and *newstest2016*, respectively. Table 2 provides the amount of sentences and tokens in the training corpora used in our experiments.

In order to study the domain robustness of MaTiLDA, we evaluated the systems trained for German–English and German–Romansh on out-of-domain test sets commonly used for this task [32].[10] In line with Wang & Sennrich [14], for German–English we chose IT, law and medical test sets, and for German–Romansh we chose a blog test set.

All corpora were tokenized and truecased with the Moses scripts;[11] then, sentences longer than 100 tokens or with less than 5 tokens were removed from the training corpora. Afterwards, byte-pair encoding [33] (BPE) with 10,000 merge operations was applied on the concatenation of the source and target sides of the training corpora to obtain the vocabulary. Finally, those sentence pairs in the training corpora with more than 100 BPE tokens were removed.

[5] https://sites.google.com/site/iwsltevaluation2014/data-provided

[6] https://wit3.fbk.eu/2015-01

[7] https://www.statmt.org/wmt21/unsup_and_very_low_res.html

[8] https://data.statmt.org/news-commentary/v16/training-monolingual/

[9] https://nlp.stanford.edu/projects/nmt/

[10] https://github.com/ZurichNLP/domain-robustness

[11] https://github.com/moses-smt/mosesdecoder/tree/master/scripts

One-to-many word alignments in both translation directions were obtained using `mgiza++` [34].[12] Source-to-target word alignments were used for the *mono* transformations (see Sec. 3); the one-to-one word alignments required by the *replace* transformation were obtained by computing the intersection between the one-to-many word alignments in both translation directions. The bilingual lexicon for the *replace* transformation was built by annotating each source word with the target word it is most frequently aligned with in the one-to-one word alignments.

4.2 Training

Our neural model is a transformer *base* model as defined by Vaswani et al. [17], with the exception of the amount of warm-up steps, which was set to 8,000. All the experiments were carried out on a single GPU with mini-batches made of 4,000 tokens. Validation was done every 1,000 updates in the low-resource scenario and every 5,000 updates in the high-resource one, and the patience, based on the BLEU score on the development set, was set to 6 validation cycles; we then kept the intermediate model performing best on the development set. We trained the systems with the `fairseq` toolkit [35]. For RAML and SwitchOut, we integrated into `fairseq` the sampling function released by their authors [7]. For SeqMix, we also integrated the modifications in data processing and training loss published by their authors as a `fairseq` task.[13]

Systems trained with MaTiLDA were fine-tuned on the original training samples after being trained on the combination of original and synthetic training samples. The results reported are those obtained with the model that maximizes BLEU on the development set.

As regards the DA hyperparameters, the proportion of words affected by the *swap*, *unk* and *replace* transformations is controlled by a hyperparameter α for which we explored, for each translation task, values in $[0.1, 0.9]$ at intervals of 0.1. RAML and SwitchOut are governed by a temperature τ. To set its value we tried, for each translation task, a set of values around the best ones reported by Wang et al. [7].[14] For the combination of SwitchOut and RAML, firstly the best τ_x for SwitchOut was determined and, afterwards, the best τ_y for RAML was sought by fixing τ_x, as done by Wang et al. SeqMix is influenced by a hyperparameter that controls the sampling of the sub-parts of the training samples that are mixed up. We experimented with values in the interval $[0.1, 1.5]$, similarly to Guo et al. [28],[15] and selected the systems performing best on the development set.

5 Results and discussion

In this section we report the results achieved by MaTiLDA when it is used to train NMT systems for low-resource translation tasks (Sec. 5.1), when it is applied in combination with back-translation (Sec. 5.2), and when it is used in high-resource conditions (Sec. 5.3). We also report the results when the NMT systems are evaluated on out-of-domain test sets (Sec. 5.4).

5.1 Low-resource conditions

Table 3 reports the mean and standard deviation of the translation performance, measured in terms of BLEU [36],[16] of three different executions (with different random seeds) for each of the systems built in low-resource conditions. We employ the *almost stochastic order* (ASO) method [38], following Ulmer et al.'s implementation [39], to assess statistical significance at a p-value threshold of 0.05. For every language pair, we highlight the scores achieved by the top-performing model in bold, as well as those whose difference with the top-performing model is not statistically significant.[17] These notation is used for the rest of results reported from this point onward. COMET and chrF++ scores, which were also computed, show a similar trend in all the experiments in this paper and will not be reported.

The results show that MaTiLDA consistently outperforms the baseline system in all language pairs and translation directions, regardless of the transformations applied, except for *source* (copying the source sentence; see results for vi–en and hsb–de) and *mono* (reordering target words for monotonous alignment; see results for en–de and vi–en). In general, the transformations *replace* (random replacement of target words and the source words they are aligned with) and *reverse* (translation into the target language but in the reverse order) are the best-performing ones; although for some language pairs *swap* (random swapping of target words) behaves better than *reverse*.

Interestingly, training on synthetic training samples generated with the three best transformations (*reverse+replace+swap*) further improves the performance, achieving the best results in all translation tasks, except for en–he and he–en, for which the best results are obtained by combining only the two best transformations (*reverse+replace*), and for hsb–de for which the best results are obtained with a single transformation (*swap*). This suggests that different transformations affect the NMT system in different ways and are somehow complementary. The improvement over the baseline in terms of BLEU varies from 1.1 (hsb–de) to 4.1 (de–rm) BLEU points, being 2.1 BLEU points the average improvement.

A comparison of MaTiLDA with RAML, SwitchOut, their combination (SwOut+RAML) and SeqMix, shows that our approach outperforms all of them. In general, of these four reference systems, the best performing one is SeqMix, which outperforms the others in six of the ten translation tasks. In any case, for all language pairs, MaTiLDA outperforms the best of these four reference systems in all cases, with improvements that range from 0.4 to 2.5 BLEU points; 1.1 BLEU points on average.

5.2 Combination with back-translation

Next we explore the combination of the standard DA method, back-translation [9], and MaTiLDA when the best performing transformations are used together (*reverse+replace+swap*) under the aforementioned low-resource

[12] https://github.com/moses-smt/mgiza

[13] Code available at https://github.com/transducens/MaTiLDA.

[14] $\tau^{-1} \in \{0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.95, 1.0, 1.1, 1.2, 1.3\}$

[15] $\{0.1, 0.5, 1.0, 1.5\}$.

[16] `sacrebleu` [37] version string: `BLEU+case.mixed+lang.vi-en+numrefs.1+smooth.exp+tok.13a+version.1.5.0`

[17] Following Ulmer et al. [39], we set an ASO decision threshold of $\tau = 0.2$.

Task	en–de	de–en	en–he	he–en	en–vi	vi–en	de–hsb	hsb–de	de–rm	rm–de
baseline	25.4 ± .2	29.4 ± .1	21.4 ± .3	31.7 ± .6	28.3 ± .4	27.8 ± .5	49.7 ± .5	49.1 ± .4	49.3 ± .5	50.6 ± .3
SwOut	25.0 ± .4	30.4 ± .3	22.0 ± .0	32.4 ± .6	28.9 ± .4	27.8 ± .6	50.1 ± .8	**49.6 ± 1.1**	50.9 ± .2	51.7 ± .3
RAML	25.4 ± .3	30.4 ± .2	21.8 ± .6	33.0 ± .4	29.0 ± .3	28.0 ± .1	50.6 ± .3	49.1 ± 1.0	49.7 ± 1.9	51.0 ± 1.7
SwOut+RAML	25.6 ± .4	30.2 ± .4	22.2 ± .2	32.5 ± .6	28.9 ± .5	28.0 ± .8	50.6 ± .9	49.0 ± 1.3	50.9 ± 1.1	50.8 ± 1.5
SeqMix	25.9 ± .3	30.6 ± .1	**22.5 ± .4**	32.9 ± .6	29.7 ± .3	**28.6 ± .3**	50.1 ± .1	48.7 ± 1.1	50.8 ± 1.0	49.6 ± .3
swap	25.4 ± .5	30.5 ± .3	**22.2 ± .5**	32.7 ± .5	29.3 ± .2	27.9 ± .8	50.6 ± .3	**50.2 ± .2**	52.0 ± .7	52.4 ± .5
unk	25.5 ± .3	30.4 ± .3	21.6 ± .7	32.7 ± .5	29.4 ± .3	28.3 ± .6	50.5 ± .3	49.9 ± .2	52.4 ± .6	52.2 ± .4
source	25.3 ± .0	30.0 ± .3	21.8 ± .3	31.8 ± .5	28.8 ± .4	27.6 ± .5	50.1 ± .2	48.4 ± .8	51.9 ± .4	51.3 ± .9
reverse	**25.9 ± .5**	30.3 ± .1	21.8 ± .3	33.5 ± .3	28.8 ± .4	28.4 ± .4	50.9 ± .5	49.5 ± .2	51.8 ± .3	51.7 ± .9
mono	25.2 ± .2	30.0 ± .3	21.7 ± .5	32.5 ± .6	29.0 ± .2	27.7 ± .1	50.5 ± .7	49.5 ± .3	51.9 ± .6	51.4 ± .7
replace	**26.1 ± .6**	**31.5 ± .2**	**22.6 ± .5**	33.5 ± .3	**30.1 ± .2**	**28.9 ± .6**	**51.1 ± .4**	**49.7 ± .5**	**53.3 ± .8**	52.7 ± 1.4
rev+repl	**26.3 ± .2**	**31.7 ± .6**	**22.9 ± .6**	**34.1 ± .5**	**30.1 ± .5**	**28.8 ± .4**	51.1 ± .2	**49.9 ± .4**	**53.0 ± .7**	52.8 ± 1.0
rev+sw+repl	**26.5 ± .3**	**31.8 ± .4**	**22.8 ± .1**	**33.9 ± .6**	**30.4 ± .5**	**29.1 ± .6**	**51.6 ± .4**	**49.7 ± .8**	**53.4 ± .6**	**54.1 ± .2**

TABLE 3: For low-resource conditions, mean and standard deviation of the BLEU scores obtained when translating in-domain test sets with the baseline system, four other DA reference systems, and MaTiLDA, using different transformations and combinations of them.

Task	de-en	he-en	vi-en	hsb-de	rm-de
baseline	31.4 ± .2	34.0 ± .5	29.4 ± .4	49.4 ± .5	49.1 ± .9
MaTiLDA	**32.8 ± .1**	**34.8 ± .3**	**30.6 ± .4**	**50.3 ± .3**	**51.3 ± .5**

TABLE 4: For low-resource conditions and back-translated data, mean and standard deviation of the BLEU scores obtained when translating in-domain test sets with the baseline system, and MaTiLDA using the combination of the best transformations according to the experiments without back-translated data (*reverse+swap+replace*).

conditions. We do this only for the translation into the high resource languages, for which large monolingual corpora are available. The back-translated data was obtained by translating, with the baseline system used in the experiments reported in Table 3, the corpora described in Sec. 4.1 from English into German, Hebrew and Vietnamese, and from German into Upper Sorbian and Romansh. The systems were then trained as usual on a corpus made of the original training corpus plus the synthetic corpus made of back-translated data (see Table 2) on the source and the original English or German sentence (depending on the language pair) on the target. Note that MaTiLDA was applied to the back-translated data as well, just as if it were the original training corpus.[18]

Table 4 shows the results of the aforementioned experiment using back-translated data. A comparison of the performance, reported in tables 3 and 4 for the baseline systems, shows that when the target language is English, the use of back-translated data improves performance in around 2.0 BLEU points; when the target language is German the use of back-translation makes the resulting MT system to perform comparably (see results for hsb–de) or even worse (see results for rm–de). This difference between the systems translating into English and into German may be explained by the fact that, for hsb–de and rm–de, the original training corpus and the test set come from the same source (hsb–de; Witaj Language Center) or belong to the same domain (rm–de; law) —note the high BLEU scores for these language pairs as compared to the others— whereas the German monolingual corpus used to generate the back-translated data comes from a different source (News Commentary) and belongs to a different domain (news).

Task	en–ro	ro–en	en–de	de–en
baseline	23.3 ± .1	30.5 ± .2	24.3 ± .5	30.0 ± .4
SwitchOut	23.3 ± .3	30.9 ± .2	24.8 ± .2	30.1 ± .8
RAML	23.6 ± .2	30.7 ± .1	24.9 ± .1	30.6 ± .6
SwOut+RAML	23.5 ± .2	31.2 ± .1	24.7 ± .2	30.0 ± .3
SeqMix	23.5 ± .2	31.3 ± .3	24.6 ± .2	30.0 ± .8
swap	23.5 ± .2	30.9 ± .4	24.9 ± .4	**30.7 ± .7**
unk	23.6 ± .1	30.7 ± .0	25.3 ± .2	**31.1 ± .2**
source	23.6 ± .2	31.1 ± .3	23.9 ± .6	29.7 ± .5
reverse	23.8 ± .1	31.2 ± .2	24.3 ± .3	30.2 ± .7
mono	23.3 ± .4	30.4 ± .2	24.6 ± .2	30.5 ± .2
replace	23.9 ± .2	31.6 ± .3	**25.6 ± .2**	**30.9 ± .1**
sw+unk+repl	**24.3 ± .2**	**32.1 ± .1**	**25.8 ± .2**	**31.5 ± .8**

TABLE 5: For high-resource conditions, mean and standard deviation of the BLEU scores obtained when translating in-domain test sets with the baseline system, four other DA reference systems, and MaTiLDA, using different transformations and combinations of them.

A comparison of the results obtained when back-translation and MaTiLDA are used together shows an improvement over the system using only back-translation of around 1.3 BLEU points, which accounts for the complementarity of both DA approaches. Compared to the application of MaTiLDA alone (last row of Table 3) the improvement of the systems translating into English is around 1.1 BLEU points. When translating into German, translation performance is improved by 0.6 BLEU points in the case of hsb–de, and worsened by 2.8 BLEU points in the case of rm–de. Note that the rm–de baseline system trained on back-translated data performs worse than the baseline system trained solely on parallel corpora (1.5 BLEU points worse).

5.3 High-resource conditions

Even thought MaTiLDA is aimed at improving the translation quality of NMT systems trained on scarce parallel corpora, we have also studied the performance of MaTiLDA when it is used to train systems in high-resource conditions. Table 5 reports the results of these experiments for English–Romanian and English–German in both translation directions. For English–Romanian the training corpus is around

[18] We tried different combinations of using the back-translated data: with and without applying MaTiLDA on back-translated sentence pairs, using a tag (as with the transformations) to flag back-translated sentence pairs, and using a tag to flag back-translated sentence pairs on which MaTiLDA was applied. All of them performed similarly.

4 times larger than for the low-resource languages with which we have experimented so far; for English–German the training corpus is around 30 times larger (see Table 2).

As the results in Table 5 show, MaTiLDA outperforms the baseline in the four translation tasks —the improvement ranges from 1.0 to 1.6 BLEU points; 1.4 BLEU points on average— as well as the other DA approaches. It is worth noting that the improvement it brings seems not to be conditioned by the amount of parallel data used, as it is around 1.3 BLEU point for English–Romanian (about 600,000 training parallel sentences) and 1.5 BLEU points for English–German (more than 4.4 million parallel sentences). As happened with the low-resource translation tasks, almost all transformations improve over the baseline, and the combination of the best performing ones (*swap+unk+replace*) further improve the baseline results.

5.4 Domain robustness

Finally, we evaluate the performance of the English–German and German–Romansh NMT systems when translating out-of-domain test sets. For English–German we used test sets in the IT, law, and medical domains; for German–Romansh we used a test set whose sentence pairs were extracted from blogs (see Sec. 4.1 for more details).

Tables 6 and 7 report the translation performance attained by the NMT systems trained in low-resource and high-resource conditions, respectively, when they are evaluated on out-of-domain test sets. The systems being evaluated are: the baseline system, MaTiLDA using the best three transformations, the combination of SwitchOut and RAML (SwOut+RAML) and SeqMix. These tables show that MaTiLDA outperforms the baseline, SwOut+RAML and SeqMix systems both in low-resource and high-resource conditions. The best performing reference system in low-resource conditions is SeqMix and MaTiLDA outperforms SeqMix by 2.7 BLEU points on average. In high-resource conditions the best performing reference system is the combination of SwitchOut and RAML (SwOut+RAML) and MaTiLDA outperforms SwOut+RAML by 3.8 BLEU points on average. Note that there are cases, such as the IT domain for German-English, in which the improvement is above 8 BLEU points.

6 Explainability

In the previous section we have exhaustively evaluated the performance of MaTiLDA and have shown that it systematically outperforms state-of-the-art DA methods both when translating in-domain and out-of-domain test sets. In this section we study if the use of MaTiLDA increases the contribution of the source representations produced by the encoder to the generation decisions made by the decoder (Sec. 6.1). In addition, we also study if MaTiLDA leads NMT systems to produce less hallucinations [40], i.e., completely inadequate output translations which are strongly unrelated to the input text (Sec. 6.2).

6.1 Relative source and target contributions

To compute the relative contribution of source and target tokens to each prediction made by the system we used an embedding perturbation method [41]. Given a source sentence $\mathbf{x}$ and its translation $\mathbf{y}$, the absolute source contribution $C_S(y_j)$ when producing the probability of the j-th token y_j is defined as the variance of y_j's output probability across N random perturbations of the word embeddings of $\mathbf{x}$. Specifically, $C_S(y_j)$ is computed according to the following equation:

$$C_S(y_j) = \frac{1}{N}\sum_{n=1}^{N}\left(p(y_j|\mathbf{y_{<j}}, \mathbf{\tilde{x}^n}) - \frac{1}{N}\sum_{m=1}^{N} p(y_j|\mathbf{y_{<j}}, \mathbf{\tilde{x}^m})\right)^2,$$

where $\mathbf{\tilde{x}^k}$ stands for the k-th perturbation of the word embeddings of $\mathbf{x}$.

In order to perturb the word embedding of a source token x, Gaussian noise with a standard deviation proportional to the Euclidean norm of the embedding is added to it:[19]

$$\tilde{x} = x + \mathcal{N}(0, \sigma_x^2); \sigma_x = \lambda \cdot ||x||$$

The absolute target contribution $C_T(y_j)$ is computed analogously by perturbing $\mathbf{y_{<j}}$ instead of $\mathbf{x}$. The relative source contribution $C_{\text{SR}}(y_j)$, which we use in our analysis, is then obtained after normalizing $C_S(y_j)$ as follows:

$$C_{\text{SR}}(y_j) = \frac{C_S(y_j)}{C_S(y_j) + C_T(y_j)}.$$

We analysed the values of $C_{\text{SR}}(y_j)$ in two different ways to shed light on the way MaTiLDA affects reliance on the source language information. On the one hand, we averaged $C_{\text{SR}}(y_j)$ for all the tokens of the translation of the test set produced by a system to obtain an estimation of its general degree of reliance on the source language. On the other hand, we studied how $C_{\text{SR}}(y_j)$ changes throughout the different tokens of the target sentence for each of the DA approaches. In both cases, and following Voita et al. [10], we teacher force the reference translations so as to perform comparisons between systems when producing the same output.

Tables 8 and 9 show the general source influence for systems trained, respectively, in low-resource and high-resource conditions, when translating the in-domain test sets. As before, we used the *almost stochastic order* (ASO) method [38], [39] to determine if the variations in source influence among the assessed systems are statistically significant, using a p-value threshold of 0.5. We highlight in bold the system with the highest source influence, along with those that do not exhibit statistically significant differences from it; we do this in all the tables reporting source influences.

As tables 8 and 9 show, in low-resource conditions, MaTiLDA systematically increases reliance on the source language, thus, behaving similarly to NMT systems trained on larger original parallel corpora [10]. *Reverse* and *unk* are the transformations that bring the largest increase, and *replace* the one that brings the smallest one. The combination of the best transformations also brings a large increase in source-language reliance. Concerning the other DA approaches, only RAML increases source reliance. Note that this approach involves replacing some words in target-language sentences with other words randomly chosen from

[19] We set $N = 50$ and $\lambda = 0.01$ [41].

Domain	IT		Law		Medical		Blogs	
Direction	**en–de**	**de–en**	**en–de**	**de–en**	**en–de**	**de–en**	**de–rm**	**rm–de**
baseline	6.9 ± .4	4.5 ± 1.8	7.7 ± .5	7.6 ± 1.8	11.0 ± .9	10.0 ± 2.0	15.8 ± .2	15.2 ± .3
SwOut+RAML	5.9 ± 1.4	8.1 ± 1.4	7.0 ± 1.1	7.8 ± .8	12.0 ± .4	11.3 ± 1.2	16.1 ± .7	15.4 ± .4
SeqMix	8.3 ± 1.8	9.4 ± 1.3	8.7 ± .4	9.3 ± .2	12.7 ± .9	12.0 ± .4	17.5 ± .3	16.3 ± .4
MaTiLDA	**15.3 ± .3**	**13.3 ± .7**	**10.2 ± .5**	**10.7 ± .1**	**16.9 ± .6**	**15.1 ± 1.0**	**20.8 ± .1**	**19.8 ± .5**

TABLE 6: For low-resource conditions, mean and standard deviation of the BLEU scores obtained when translating out-of-domain test sets. The MaTiLDA results were computed using the combination of the best transformations on low-resource conditions (*reverse+swap+replace*).

Domain	IT		Law		Medical	
Direction	**en–de**	**de–en**	**en–de**	**de–en**	**en–de**	**de–en**
baseline	**10.6 ± 2.1**	20.8 ± 1.0	28.3 ± 2.4	32.0 ± .1	17.4 ± 1.0	23.3 ± .9
SwOut+RAML	**11.5 ± 3.2**	20.8 ± 1.8	28.9 ± 1.4	33.7 ± .2	18.3 ± 1.4	25.0 ± .4
SeqMix	**11.2 ± 2.0**	18.5 ± 2.6	28.1 ± 1.1	30.1 ± 1.4	17.7 ± 1.0	23.3 ± .7
MaTiLDA	**13.7 ± 2.7**	**29.0 ± 1.1**	**31.0 ± 1.9**	**35.7 ± .8**	**22.2 ± 1.4**	**29.1 ± .8**

TABLE 7: For high-resource conditions, mean and standard deviation of the BLEU scores obtained when translating out-of-domain test sets. The MaTiLDA results were computed using the combination of the best transformations on high-resource conditions (*swap+unk+replace*).

Task	**en–de**	**de–en**	**en–he**	**he–en**	**en–vi**	**vi–en**	**de–hsb**	**hsb–de**	**de–rm**	**rm–de**
baseline	63.2 ± .8	67.1 ± 2.0	71.8 ± 1.7	67.2 ± 2.6	68.1 ± 1.1	58.7 ± 1.2	77.6 ± 1.1	76.2 ± 1.4	72.9 ± .9	63.8 ± 1.7
SwOut	60.4 ± .2	63.1 ± 1.1	68.5 ± .7	64.2 ± 3.5	64.2 ± 1.1	56.7 ± .2	72.2 ± .8	67.1 ± 1.1	69.0 ± .7	58.8 ± 1.9
RAML	**68.7 ± 1.8**	70.3 ± 1.7	74.3 ± 2.2	72.2 ± 2.2	73.2 ± 3.0	60.9 ± 1.4	79.6 ± 1.5	79.9 ± 1.2	75.8 ± 1.4	69.3 ± 1.9
SwOut+RAML	65.4 ± 2.5	68.5 ± 1.9	71.0 ± .8	67.0 ± 1.5	67.5 ± 2.3	59.1 ± .6	76.0 ± 1.8	72.5 ± 3.1	73.6 ± 2.2	64.8 ± 1.8
SeqMix	63.5 ± .6	64.2 ± 2.1	67.8 ± 1.5	62.8 ± 1.2	58.0 ± 1.6	54.7 ± .9	75.0 ± 2.8	75.2 ± 1.5	57.6 ± 1.6	63.5 ± 1.4
swap	68.3 ± .7	**74.9 ± 2.3**	76.9 ± .8	74.0 ± 1.6	74.0 ± 1.3	63.4 ± 1.9	79.7 ± 1.4	78.9 ± 1.2	**79.6 ± .3**	68.4 ± 1.8
unk	**71.2 ± .8**	74.3 ± .7	75.8 ± 1.2	71.2 ± 1.6	71.5 ± 2.0	**70.3 ± 1.2**	**84.1 ± 1.1**	**84.6 ± 1.4**	**80.2 ± 1.2**	66.1 ± .7
source	68.6 ± 1.5	71.8 ± 1.8	74.6 ± 1.6	67.8 ± 2.9	71.1 ± 1.3	64.3 ± .4	82.0 ± .6	81.6 ± 2.2	76.8 ± .0	68.3 ± 1.8
reverse	**71.3 ± 2.2**	**75.1 ± .5**	**80.1 ± 1.0**	**76.3 ± 1.5**	**77.2 ± .9**	**70.6 ± 2.1**	**84.0 ± 1.6**	**83.0 ± .9**	**80.0 ± 1.5**	**75.1 ± 1.2**
mono	64.7 ± 1.7	70.0 ± .3	72.4 ± .1	66.9 ± 1.1	69.6 ± 1.5	61.7 ± 2.0	76.4 ± .7	76.6 ± .6	75.5 ± .9	66.7 ± 1.7
replace	62.1 ± 1.9	67.1 ± .2	71.6 ± 1.6	72.5 ± 1.4	68.3 ± 1.0	60.5 ± 1.6	76.0 ± 1.8	75.9 ± 2.3	72.0 ± 1.0	65.2 ± 1.0
rev+repl	**67.9 ± 3.0**	73.5 ± 1.2	78.9 ± 1.2	**77.7 ± .9**	71.9 ± 2.2	65.7 ± 1.4	81.0 ± 1.9	**83.6 ± 1.5**	76.1 ± .5	71.7 ± .4
rev+sw+repl	**69.3 ± 2.5**	**74.7 ± 1.2**	**81.1 ± .1**	**77.0 ± .7**	**75.8 ± 2.3**	65.6 ± 1.3	81.5 ± 1.7	82.1 ± 1.7	**79.4 ± 1.1**	71.9 ± 2.8

TABLE 8: For low-resource conditions, mean and standard deviation of the source influence obtained when translating in-domain test sets with the baseline system, four other DA reference systems, and MaTiLDA using different transformations and combinations of them.

Task	**en–ro**	**ro–en**	**en–de**	**de–en**
baseline	72.0 ± .9	84.3 ± .6	**62.5 ± 5.3**	68.4 ± 1.0
SwitchOut	68.6 ± 1.1	80.8 ± .9	56.9 ± 4.1	67.7 ± .9
RAML	**78.5 ± .6**	**87.7 ± 1.6**	61.9 ± .6	**76.9 ± 2.8**
SwOut+RAML	76.2 ± 1.0	86.1 ± .5	**65.1 ± 5.0**	**77.7 ± .7**
SeqMix	71.6 ± 1.2	80.6 ± 1.6	56.5 ± 2.1	69.7 ± 4.5
swap	76.1 ± .4	**86.9 ± .7**	**61.8 ± 4.3**	72.4 ± .2
unk	**76.8 ± 2.4**	85.9 ± .3	59.3 ± 3.9	72.7 ± 1.2
source	70.4 ± 2.5	82.0 ± .5	**65.9 ± 3.5**	71.4 ± .8
reverse	71.0 ± 1.2	**86.5 ± 1.5**	**60.4 ± 7.0**	69.1 ± 1.6
mono	74.3 ± 2.1	**87.3 ± 1.1**	**66.5 ± 1.9**	72.7 ± 1.7
replace	70.7 ± .8	80.3 ± .8	**66.0 ± 4.6**	**75.7 ± 2.2**
sw+unk+repl	75.9 ± .9	79.7 ± 1.1	62.4 ± 3.2	74.6 ± 1.4

TABLE 9: For high-resource conditions, mean and standard deviation of the source influence obtained when translating in-domain test sets with the baseline system, four other DA reference systems, and MaTiLDA using different transformations and combinations of them.

the vocabulary. Thus, it makes the target prefix less predictive, similarly to the MaTiLDA transformations. Nevertheless, both BLEU scores and source reliance are lower than those obtained with MaTiLDA.

In high-resource conditions, the increase in source reliance brought by MaTiLDA is smaller than in low-resource conditions, and it is focused mainly on the systems translating into English. When English is the source language, the target language (German and Romanian) is highly inflected and may require more reliance on the target context to produce a grammatical output.

The general source influence when translating the different out-of-domain test sets, depicted in tables 10 and 11, follows similar trends to those identified for the in-domain scenario. The largest source influence is achieved by MaTiLDA for low-resource translation tasks, while the gap between it and SwitchOut+RAML vanishes when larger training data is available.

Figure 1 depicts the value of $C_{\text{SR}}(y_j)$ for each target-language token of the low-resource in-domain English–German test and for the different DA methods evaluated; the rest of language pairs show the same behaviour. As sentences have different lengths, the position of a token in the sentence (x-axis) is represented as the proportion of the words already predicted, hence the values in the x-axis are in the interval $[0, 1]$, being 0 the first token of the sentence and 1 the last one. The polynomial that best fits the data (obtained via least squares) is also depicted in the plot.[20] It can be observed that the source influence

[20] The degree of the polynomial was empirically obtained by incrementally exploring different values. We found that the lowest polynomial degree that best fits the data was 6.

Domain	IT		Law		Medical		Blogs	
Direction	**en–de**	**de–en**	**en–de**	**de–en**	**en–de**	**de–en**	**de–rm**	**rm–de**
baseline	62.4 ± 1.4	66.2 ± 1.9	58.8 ± .7	58.7 ± 1.6	**62.0 ± 1.0**	62.6 ± 2.2	64.8 ± .9	61.4 ± 1.3
SwOut+RAML	**66.1 ± 2.1**	68.1 ± 2.4	**62.4 ± 2.6**	61.0 ± 1.9	**65.3 ± 2.1**	64.6 ± 2.2	67.2 ± 3.2	62.4 ± 1.9
SeqMix	61.6 ± 1.7	66.1 ± 2.1	60.2 ± 1.4	61.8 ± 2.1	**62.4 ± 1.1**	63.4 ± 2.2	55.9 ± 1.2	54.8 ± 1.2
MaTiLDA	**66.8 ± 2.4**	**76.4 ± 1.1**	**66.3 ± 3.2**	**70.7 ± .6**	**66.2 ± 3.2**	**72.6 ± .7**	**74.1 ± 1.7**	**68.8 ± 2.8**

TABLE 10: For low-resource conditions, mean and standard deviation of the source influence obtained when translating out-of-domain texts in the IT, law, medical and blogs domains. The MaTiLDA results were computed using the combination of the best transformations on low-resource conditions (*reverse+swap+replace*)

Domain	IT		Law		Medical	
Direction	**en–de**	**de–en**	**en–de**	**de–en**	**en–de**	**de–en**
baseline	**61.3 ± 4.8**	75.4 ± 1.3	**58.1 ± 6.1**	71.2 ± 1.3	**59.9 ± 4.4**	65.8 ± 1.4
SwOut+RAML	**61.7 ± 4.8**	**81.2 ± 1.0**	**63.1 ± 6.4**	**78.8 ± 1.3**	**60.0 ± 5.2**	**70.8 ± 1.3**
SeqMix	**54.8 ± 3.7**	**74.3 ± 4.7**	52.2 ± 1.2	71.8 ± 5.3	**53.6 ± 3.5**	65.2 ± 4.4
MaTiLDA	**60.7 ± 1.1**	**81.6 ± 2.8**	**57.9 ± 3.3**	**79.4 ± 2.3**	**59.2 ± 1.2**	**72.4 ± 1.9**

TABLE 11: For high-resource conditions, mean and standard deviation of the source influence obtained when translating out-of-domain test sets. The MaTiLDA results were computed using the combination of the best transformations on high-resource conditions (*swap+unk+replace*).

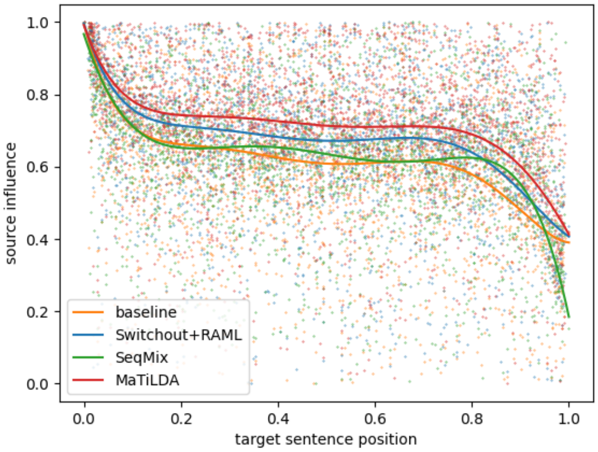


Fig. 1: Source influence throughout relative target sentence positions for the English–German low-resource in-domain test set.

decreases as decoding progresses, in line with the findings of Voita et al. [10]. The difference between MaTiLDA and the baseline remains relatively constant through the sentence except for the first tokens, where the target prefix is too short to make a difference between the DA methods. The relative position between the DA methods evaluated matches those shown in Table 8.

6.2 Hallucinations

In spite of the significant improvements in translation performance brought by the recent advancements in NMT, the phenomenon of hallucinations in NMT still remains a concern. Hallucinations usually appear in out-of-domain translations [42], and may undermine user trust.

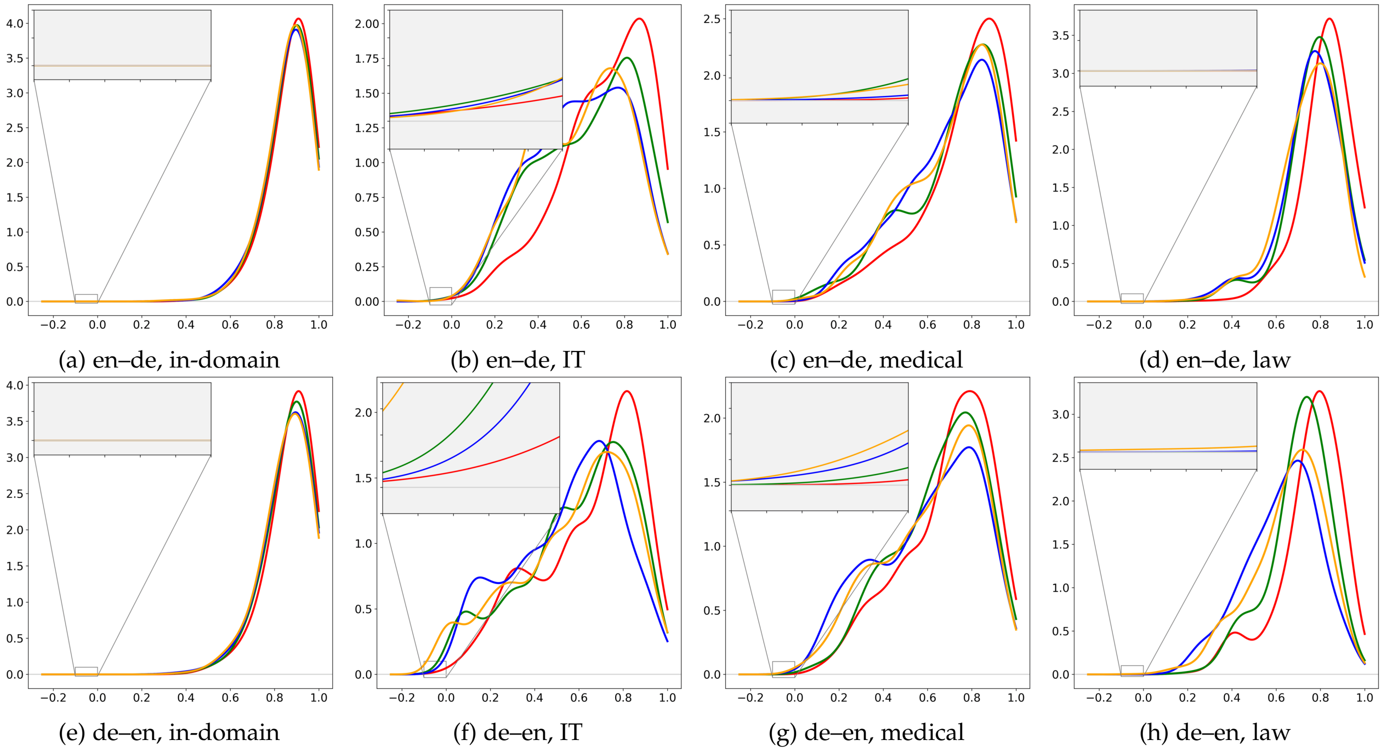
Hallucinations have been associated with systems failing to use source information properly [10], [43], thus showing abnormal patterns in the cross-attention to the encoder. As we have already proved that MaTiLDA consistently improves source relevance, in this section we analyse whether this additionally results in the mitigation of hallucinations in both in-domain and out-of-domain translation tasks.

In order to estimate the number of hallucinations produced by systems trained with different DA techniques we used LaBSE [44] cross-lingual sentence embeddings. A number of studies [43], [45], [46] have shown that the use of cross-lingual sentence embeddings to compute the similarity of MT outputs and their reference translations outperforms previous methods for detecting hallucinations, such as COMET [47], [48] or lexical-based metrics [40], [42]. Noticeably, LaBSE, as a discriminator of hallucinations, unlike COMET, has been shown to differentiate hallucinations from poor translations [43].

We evaluate the tendency to hallucinate of four systems: the baseline, MaTiLDA including the best performing transformations, the combination of SwitchOut and RAML, and SeqMix. We compute the sentence-level LaBSE embeddings for the system outputs and the references, and then represent the cosine similarities between them. Computing the cosine similarity between the systems' output and the source sentences in the test sets results in similar plots.

Figures 2 and 3 show the kernel density estimations of the distributions of cosine similarities for the systems trained in low-resource and high-resource conditions, respectively, for the English–German and German–English translation tasks. To obtain these plots we used the same test sets and domains (in-domain, IT, legal and medical) used in Sec. 5.4. A larger view of the area close to zero where the strongest hallucinations are supposed to live [43] is also included with each plot.

As the plots show, MaTiLDA cosine distribution curves are shifted to the right when compared to the baseline and the other DA methods; even when the systems are trained in high-resource conditions (see Fig. 3). This can be interpreted as a clear sign of reduction of hallucinations in the systems trained with MaTiLDA, especially under domain shift. In this regard, it is worth noting how the in-domain plots are prominently shifted to the right when compared to the out-of-domain ones because the systems hallucinate less and produce more semantically accurate



(a) en–de, in-domain (b) en–de, IT (c) en–de, medical (d) en–de, law

(e) de–en, in-domain (f) de–en, IT (g) de–en, medical (h) de–en, law

Fig. 2: Kernel density estimations (bandwidth=0.06) for LaBSE-based cosine similarities between the output produced by NMT models trained in low-resource conditions and the reference translations in test sets belonging to different domains. DA methods: 🟧 baseline, 🟦 SwitchOut+RAML, 🟩 SeqMix, 🟥 MaTiLDA.

Type	Cosine	Sentence
Source		hinweise fÜr die richtige anwendung
Reference		advice on correct administration
Baseline	0.167	you know, evidence for die’s application.
SwOut+RAML	0.098	it’s called ”die.”
SeqMix	0.280	clues to die’s real application.
MaTiLDA	0.515	clues to the right use.
Source		artikel 16
Reference		article 16
Baseline	-0.049	they said, ”wwhwhwhwhwhwhwhwhwhwhwhw [...] were were were were were were were were were [...] were.”
SwOut+RAML	0.529	I was 16 years old.
SeqMix	0.634	it’s 16 articles of 16.
MaTiLDA	0.893	articles 16.

TABLE 12: Examples of output translations in which MaTiLDA attains the highest cosine similarity with the reference translation.

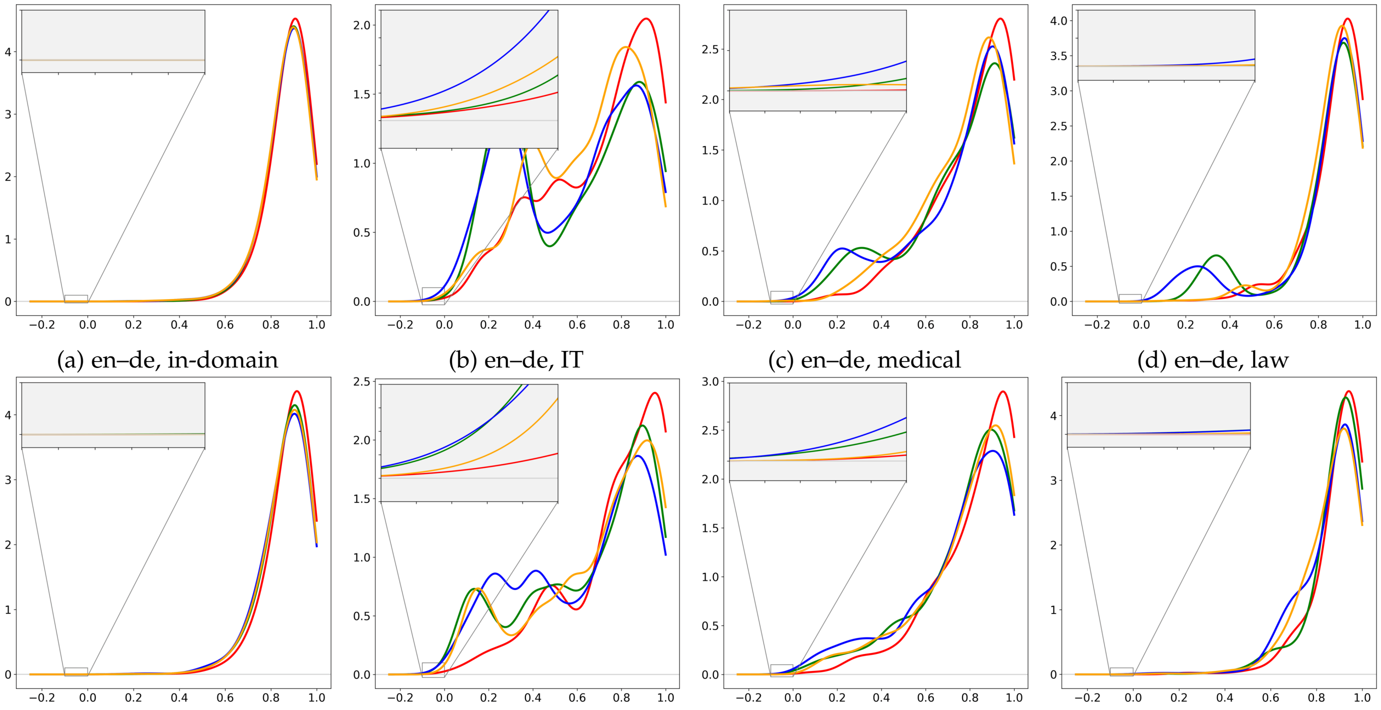
outputs when translating in-domain texts. Table 12 shows the output translations of each system under study in a couple of representative cases in which MaTiLDA attains the highest cosine similarity with the reference translation. Note, however, that a relatively high cosine value may still correspond to a hallucination (see cosine similarity values for the second example in Table 12), which supports the idea that the consistent shift to the right of the cosine distribution curves for MaTiLDA in figures 2 and 3 clearly indicate a reduction in the tendency to hallucinate of the systems trained with MaTiLDA.

7 Related work

The back-translation [9] approach for leveraging additional target monolingual data, is, probably, the most popular DA approach for NMT. The set of related approaches covered in this section, however, mainly focus on methods that, as MaTiLDA, do not require additional resources besides the training parallel corpus.

Li et al. [5] evaluate back- and forward-translation in such a setting. They train forward and backward NMT systems on the available parallel data and use them to produce new synthetic samples by translating either the target side [49] or the source side [50] of the original training corpus. Other approaches simply select two training samples at random and concatenate, on one side, the source sentences and, on the other side, the target sentences to generate larger training samples [51], [52] in order to improve the translation quality of long source sentences.

The approaches we have evaluated in our experiments, RAML [27], SwitchOut [7] and SeqMix [28], aim at extending the support of the empirical data distribution, which is expected to prevent the model to memorize long segments and improve the model generalization capabilities. To that end, RAML and SwitchOut replace words with other words sampled from a uniform distribution over the vocabulary, which, in practice, results in infrequent words being overrepresented; RAML works on the target side, whereas SwitchOut works on the source side. SeqMix approaches the problem in a different way and generates synthetic training samples by randomly combining parts of two sentences in order to encourage compositional behaviour.



(a) en–de, in-domain (b) en–de, IT (c) en–de, medical (d) en–de, law

(e) de–en large, in-domain (f) de–en, IT (g) de–en, medical (h) de–en, law

Fig. 3: Kernel density estimations (bandwidth=0.06) for LaBSE-based cosine similarities between the output produced by NMT models trained in high-resource conditions and the reference translations in test sets belonging to different domains. DA methods: 🟧 baseline, 🟦 SwitchOut+RAML, 🟩 SeqMix, 🟥 MaTiLDA.

Applying simple transformations, such as swapping words, to existing sentences is an idea that has been widely applied in the context of NMT with multiple purposes. For instance, several simple transformations (word deletion, replacement, swapping) have been applied to back-translated data [53] with the aim of allowing the system to better distinguish between original and back-translated sentences. In the context of unsupervised NMT, the noisy input sentences to denoising autoencoders are generated by random token swaps [20].

Focusing on approaches that modify word order in the context of DA, it is worth highlighting a self-translation approach using a right-to-left decoder [54], which is similar to inverting the order of the target words. However, unlike MaTiLDA, this last approach needs to generate translations from the model during training.

Replacing tokens with placeholders (as we do in *unk*) has already been applied to the source language [55] in combination with two self-supervised learning objectives for detecting replaced and dropped tokens. Xie et al. [21] also evaluate the impact of random replacements of words in the source and target sides of the training samples by either a random word from the vocabulary, or by a blank.

Gao et al. [29] replace source-side words selected at random with *soft words* whose representations are obtained from the probability distribution provided by a language model. Fadaee et al. [26] replace some words in their training samples by infrequent words in order to improve the performance of the NMT model when dealing with them at translation time. Words to be replaced are identified using a large source language model. Once the source words to be replaced are identified, a word-alignment model and a probabilistic dictionary are used to also replace the corresponding counterpart by the most probable translation of the new source word. In MaTiLDA, the *replace* transformation, which is similar, does not require any language model.

As regards the special token we use to prevent negative transfer between tasks, a similar strategy [56] has been applied to identify synthetic samples when combining actual parallel data and back-translated data for training. Yang et al. [57] extends this last work by including forward-translated data for training using two different special tokens to distinguish the two types of synthetic data. Another strategy that has been reported to be effective to combine synthetic and original training instances is the AugMix method [58], initially defined in the context of image processing. This method involves creating training samples through linear interpolation of the embeddings of both the original and the synthetic samples and adding an auxiliary loss that enforces model probabilities of both types of samples to be similar. It has been subsequentially applied to NMT [59] using the simple transformations mentioned above (word deletion, replacements as those defined in SwitchOut/RAML, swapping), which are not the best performing ones according to our analyses.

The problem of the NMT system relying too much on the target-language context has been addressed in ways other than DA. Miao et al. [60] define a metric to measure the prevalence of the decoder's language model over the encoder representations and use it to define specific auxiliary loss functions to reduce this prevalence; Weng et al. [61] use a similar auxiliary loss that is only optimized on mistranslated fragments selected from the training data.

Finally, a number of approaches that mitigate the amount of hallucinations produced by NMT systems have been proposed [14], [32], [40], [42], [48]. However most of them either do not evaluate the impact of the techniques proposed in the general performance of the NMT models built [40], [48], or report mixed results with a drop in performance for some language pairs or in some translation scenarios [14], [32], [42]. In contrast, the approach described in this paper not only reduces the degree of hallucinations as measured by LaBSE, but it also improves the general quality of the translations produced by the NMT models.

8 Concluding remarks

We have presented a novel method for data augmentation (DA) for neural machine translation (NMT) that we have termed as multi-task learning DA (MaTiLDA). In contrast to state-of-the-art DA approaches, MaTiLDA aims at generating new synthetic training samples with non-fluent target-language sentences by means of aggressive transformations, such as reversing the order of the target sentence or swapping random target words. The new synthetic training samples, which are considered as data for additional learning tasks, provide new contexts during training where the target prefix is not sufficiently informative to predict the next token, thus strengthening the relevance of the encoder and increasing at inference time the reliance on the source-language representations it generates. MaTiLDA is agnostic to the NMT model architecture and does not require elaborate preprocessing steps, additional training systems, or data besides the available training parallel corpora.

We have extensively evaluated this new approach on ten low-resource and four high-resource translation tasks. The results show consistent improvements over a baseline without DA, and over three strong state-of-the-art DA methods that aim at extending the support of the empirical data distribution by generating synthetic training samples with fluent target sentences. This improvement shows up both when training NMT systems in low- and high-resource conditions. Furthermore, NMT systems trained with MaTiLDA are much more robust under domain shift and generate fewer hallucinations than the baseline or any of the state-of-the-art DA methods we have compared with when translating out-of-domain texts. In addition, our evaluation also demonstrates that MaTiLDA can be easily combined with the standard DA method, namely back-translation, and that both methods complement each other as their combination results in further translation performance improvements.

An analysis of the influence of the encoder and decoder representations in the NMT system output shows that, thanks to the transformations used for building synthetic training samples, MaTiLDA increases the contribution of the source representations from the encoder to the decisions made by the NMT decoder during inference. Hence, MaTiLDA makes it possible to build NMT models for low-resource language pairs that, even though they have been trained on small parallel corpora, are able to behave as if they had been trained on larger training corpora.

All in all, the method we have presented offers promising implications for the field of NMT. By utilizing a robust and contrasted approach, we have demonstrated the potential to enhance the accuracy of virtually any existing NMT system by seamlessly integrating MaTiLDA in their training pipelines to make the most of the existing corpora. We leave for future work the study of the potential effects of integrating MaTilDA in scenarios with even more resources than those used in our study, as it is the case of the utilization of large pre-trained multilingual NMT models. This pre-trained models make use of extensive amounts of monolingual and parallel data from various language pairs and significantly outperform systems trained exclusively on parallel data [62].[21]

Acknowledgments

This paper is part of the R+D+i project PID2021-127999NB-I00 funded by the Spanish Ministry of Science and Innovation (MCIN), the Spanish Research Agency (AEI/10.13039/501100011033) and the European Regional Development Fund A way to make Europe. The computational resources used were funded by the European Regional Development Fund through project IDIFEDER/2020/003. We thank Mikel L. Forcada for his comments on a draft version of this paper.

References

[1] H. Hassan, A. Aue, C. Chen, V. Chowdhary, J. Clark, C. Federmann, X. Huang, M. Junczys-Dowmunt, W. Lewis, M. Li, S. Liu, T. Liu, R. Luo, A. Menezes, T. Qin, F. Seide, X. Tan, F. Tian, L. Wu, S. Wu, Y. Xia, D. Zhang, Z. Zhang, and M. Zhou, "Achieving human parity on automatic chinese to english news translation," *CoRR*, vol. abs/1803.05567, 2018.

[2] T. Kocmi and O. Bojar, "Trivial transfer learning for low-resource neural machine translation," in *Proceedings of the 3rd Conference on Machine Translation: Research Papers*, Brussels, Belgium, Oct. 2018, pp. 244–252.

[3] R. Sennrich and B. Haddow, "Linguistic input features improve neural machine translation," in *Proceedings of the 1st Conference on Machine Translation: Volume 1, Research Papers*, Berlin, Germany, Aug. 2016, pp. 83–91.

[4] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, M. Hughes, and J. Dean, "Google's multilingual neural machine translation system: Enabling zero-shot translation," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 339–351, 2017.

[5] G. Li, L. Liu, G. Huang, C. Zhu, and T. Zhao, "Understanding data augmentation in neural machine translation: Two perspectives towards generalization," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, Nov. 2019, pp. 5689–5695.

[6] S. Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy, "A survey of data augmentation approaches for NLP," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Aug. 2021, pp. 968–988.

[7] X. Wang, H. Pham, Z. Dai, and G. Neubig, "SwitchOut: an efficient data augmentation algorithm for neural machine translation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Oct.-Nov. 2018, pp. 856–861.

[8] X. Wei, H. Yu, Y. Hu, R. Weng, L. Xing, and W. Luo, "Uncertainty-aware semantic augmentation for neural machine translation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Nov. 2020, pp. 2724–2735.

[21] Indeed, the performance gap between our baseline systems and the version with 1.3B parameters of the pre-trained model NLLB [63] when it is fine-tuned on the parallel corpora used in our low-resource experiments is, on average, 7.4 BLEU points when translating into the low-resource languages, and 10.4 BLEU points when translating into the high-resource languages (English or German).

[9] R. Sennrich, B. Haddow, and A. Birch, “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)*, Berlin, Germany, Aug. 2016, pp. 86–96.

[10] E. Voita, R. Sennrich, and I. Titov, “Analyzing the source and target contributions to predictions in neural machine translation,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Aug. 2021, pp. 1126–1140.

[11] R. Sennrich, B. Haddow, and A. Birch, “Controlling politeness in neural machine translation via side constraints,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California, Jun. 2016, pp. 35–40.

[12] D. Dong, H. Wu, W. He, D. Yu, and H. Wang, “Multi-task learning for multiple language translation,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Beijing, China, Jul. 2015, pp. 1723–1732.

[13] V. M. Sánchez-Cartagena, M. Esplà-Gomis, J. A. Pérez-Ortiz, and F. Sánchez-Martínez, “Rethinking data augmentation for low-resource neural machine translation: A multi-task learning approach,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic, Nov. 2021, pp. 8502–8516.

[14] C. Wang and R. Sennrich, “On exposure bias, hallucination and domain shift in neural machine translation,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Jul. 2020, pp. 3544–3552.

[15] J. Gehring, M. Auli, D. Grangier, and Y. Dauphin, “A convolutional encoder model for neural machine translation,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, Jul. 2017, pp. 123–135.

[16] D. Bahdanau, K. H. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” in *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[18] H. Robbins and S. Monro, “A stochastic approximation method,” *The Annals of Mathematical Statistics*, pp. 400–407, 1951.

[19] R. Müller, S. Kornblith, and G. E. Hinton, “When does label smoothing help?” *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[20] M. Artetxe, G. Labaka, E. Agirre, and K. Cho, “Unsupervised neural machine translation,” in *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[21] Z. Xie, S. I. Wang, J. Li, D. Lévy, A. Nie, D. Jurafsky, and A. Y. Ng, “Data noising as smoothing in neural network language models,” in *Proceedings of the 5th International Conference on Learning Representation*, Toulon, France, 2017.

[22] S. R. Bowman, L. Vilnis, O. Vinyals, A. Dai, R. Jozefowicz, and S. Bengio, “Generating sentences from a continuous space,” in *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, Berlin, Germany, Aug. 2016, pp. 10–21.

[23] M. Ott, M. Auli, D. Grangier, and M. Ranzato, “Analyzing uncertainty in neural machine translation,” in *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweden, 10–15 Jul 2018, pp. 3956–3965.

[24] H. Khayrallah and P. Koehn, “On the impact of various types of noise on neural machine translation,” in *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, Melbourne, Australia, Jul. 2018, pp. 74–83.

[25] F. Sánchez-Martínez, R. C. Carrasco, M. A. Martínez-Prieto, and J. Adiego, “Generalized biwords for bitext compression and translation spotting,” *Journal of Artificial Intelligence Research*, vol. 43, pp. 389–418, March 2012.

[26] M. Fadaee, A. Bisazza, and C. Monz, “Data augmentation for low-resource neural machine translation,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vancouver, Canada, Jul. 2017, pp. 567–573.

[27] M. Norouzi, S. Bengio, z. Chen, N. Jaitly, M. Schuster, Y. Wu, and D. Schuurmans, “Reward augmented maximum likelihood for neural structured prediction,” *Advances In Neural Information Processing Systems*, vol. 29, pp. 1723–1731, 2016.

[28] D. Guo, Y. Kim, and A. Rush, “Sequence-level mixed sample data augmentation,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Nov. 2020, pp. 5547–5552.

[29] F. Gao, J. Zhu, L. Wu, Y. Xia, T. Qin, X. Cheng, W. Zhou, and T.-Y. Liu, “Soft contextual data augmentation for neural machine translation,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, Jul. 2019, pp. 5539–5544.

[30] M. Cettolo, J. Niehues, S. Stuker, L. Bentivogli, and M. Federico, “Report on the 11th IWSLT evaluation campaign, IWSLT 2014,” in *Proceedings of the 11th International Workshop on Spoken Language Translation*, Lake Tahoe, USA, 2014, pp. 2–17.

[31] M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, R. Cattoni, and M. Federico, “The IWSLT 2015 evaluation campaign,” in *Proceedings of the 12th International Workshop on Spoken Language Translation: Evaluation Campaign*, Da Nang, Vietnam, Dec. 2015, pp. 2–14.

[32] M. Müller, A. Rios, and R. Sennrich, “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)*, Oct. 2020, pp. 151–164.

[33] R. Sennrich, B. Haddow, and A. Birch, “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)*, Berlin, Germany, Aug. 2016, pp. 1715–1725.

[34] Q. Gao and S. Vogel, “Parallel implementations of word alignment tool,” in *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, Columbus, USA, Jun. 2008, pp. 49–57.

[35] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, “fairseq: A fast, extensible toolkit for sequence modeling,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, Minneapolis, Minnesota, Jun. 2019, pp. 48–53.

[36] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: a method for automatic evaluation of machine translation,” in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, USA, 2002, pp. 311–318.

[37] M. Post, “A call for clarity in reporting BLEU scores,” in *Proceedings of the Third Conference on Machine Translation: Research Papers*, Brussels, Belgium, Oct. 2018, pp. 186–191.

[38] R. Dror, S. Shlomov, and R. Reichart, “Deep dominance - how to properly compare deep neural models,” in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, A. Korhonen, D. R. Traum, and L. Màrquez, Eds. Association for Computational Linguistics, 2019, pp. 2773–2785.

[39] D. Ulmer, C. Hardmeier, and J. Frellsen, “deep-significance: Easy and meaningful signifcance testing in the age of neural networks,” in *ML Evaluation Standards Workshop at the Tenth International Conference on Learning Representations*, 2022.

[40] K. Lee, O. Firat, A. Agarwal, C. Fannjiang, and D. Sussillo, “Hallucinations in neural machine translation,” in *Interpretability and Robustness for Audio, Speech and Language, NIPS 2018 Workshop*, Montréal, Canada, 2018.

[41] J. Ferrando and M. R. Costa-jussà, “Attention weights in transformer NMT fail aligning words between sequences but largely explain model predictions,” in *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, Nov. 2021, pp. 434–443.

[42] M. Müller and R. Sennrich, “Understanding the properties of minimum Bayes risk decoding in neural machine translation,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Aug. 2021, pp. 259–272.

[43] D. Dale, E. Voita, L. Barrault, and M. R. Costa-jussà, “Detecting and mitigating hallucinations in machine translation: Model internal workings alone do well, sentence similarity even better,” 2022.

[44] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, “Language-agnostic BERT sentence embedding,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland, May 2022, pp. 878–891.

[45] N. M. Guerreiro, P. Colombo, P. Piantanida, and A. F. T. Martins, “Optimal transport for unsupervised hallucination detection in neural machine translation,” 2022.

[46] W. Xu, S. Agrawal, E. Briakou, M. J. Martindale, and M. Carpuat, "Understanding and detecting hallucinations in neural machine translation via model introspection," 2023.

[47] R. Rei, C. Stewart, A. C. Farinha, and A. Lavie, "COMET: A neural framework for MT evaluation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Nov. 2020, pp. 2685–2702.

[48] N. M. Guerreiro, E. Voita, and A. F. T. Martins, "Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation," 2022.

[49] R. Sennrich, B. Haddow, and A. Birch, "Edinburgh neural machine translation systems for WMT 16," in *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, Berlin, Germany, Aug. 2016, pp. 371–376.

[50] J. Zhang and C. Zong, "Exploiting source-side monolingual data in neural machine translation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, Nov. 2016, pp. 1535–1545.

[51] S. Kondo, K. Hotate, T. Hirasawa, M. Kaneko, and M. Komachi, "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*, Jun. 2021, pp. 143–149.

[52] X. Wu, Y. Xia, J. Zhu, L. Wu, S. Xie, Y. Fan, and T. Qin, "mixSeq: A simple data augmentation methodfor neural machine translation," in *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, Bangkok, Thailand (online), Aug. 2021, pp. 192–197.

[53] S. Edunov, M. Ott, M. Auli, and D. Grangier, "Understanding back-translation at scale," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Oct.-Nov. 2018, pp. 489–500.

[54] Z. Zhang, S. Wu, S. Liu, M. Li, M. Zhou, and T. Xu, "Regularizing neural machine translation by target-bidirectional agreement," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Honolulu, USA, 2019, pp. 443–450.

[55] H. Zhang, S. Qiu, X. Duan, and M. Zhang, "Token drop mechanism for neural machine translation," in *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, Spain, Dec. 2020, pp. 4298–4303.

[56] I. Caswell, C. Chelba, and D. Grangier, "Tagged back-translation," in *Proceedings of the 4th Conference on Machine Translation (Volume 1: Research Papers)*, Florence, Italy, 2019, pp. 53–63.

[57] Z. Yang, W. Chen, F. Wang, and B. Xu, "Effectively training neural machine translation models with monolingual data," *Neurocomputing*, vol. 333, pp. 240–247, 2019.

[58] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "AugMix: A simple method to improve robustness and uncertainty under data shift," in *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[59] C. Jin, S. Qiu, N. Xiao, and H. Jia, "AdMix: A mixed sample data augmentation method for neural machine translation," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 7 2022, pp. 4171–4177, main Track.

[60] M. Miao, F. Meng, Y. Liu, X.-H. Zhou, and J. Zhou, "Prevent the language model from being overconfident in neural machine translation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Aug. 2021, pp. 3456–3468.

[61] R. Weng, H. Yu, X. Wei, and W. Luo, "Towards enhancing faithfulness for neural machine translation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Nov. 2020, pp. 2675–2684.

[62] Y. Tang, C. Tran, X. Li, P.-J. Chen, N. Goyal, V. Chaudhary, J. Gu, and A. Fan, "Multilingual translation from denoising pretraining," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics, Aug. 2021, pp. 3450–3466.

[63] NLLB Team, M. R. Costa-jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard, A. Sun, S. Wang, G. Wenzek, A. Youngblood, B. Akula, L. Barrault, G. M. Gonzalez, P. Hansanti, J. Hoffman, S. Jarrett, K. R. Sadagopan, D. Rowe, S. Spruit, C. Tran, P. Andrews, N. F. Ayan, S. Bhosale, S. Edunov, A. Fan, C. Gao, V. Goswami, F. Guzmán, P. Koehn, A. Mourachko, C. Ropers, S. Saleem, H. Schwenk, and J. Wang, "No language left behind: Scaling human-centered machine translation," 2022, arXiv:2207.04672.

Víctor M. Sánchez-Cartagena is Assistant Professor at Universitat d'Alacant, Spain. He obtained his PhD in Computer Science in 2015, he also worked as a research engineer at Prompsit Language Engineering. His main fields of research are deep learning and machine translation with an emphasis on low-resource languages and the combination of multiple systems and/or sources of information. He authored more than 10 indexed publications, including top natural language processing conferences.

Miquel Esplà-Gomis is Assistant Professor at Universitat d'Alacant, Spain. He obtained his PhD in Computer Science in 2016. His main research fields are parallel data acquisition and application of translation technologies to computer-aided translation. He has published more than 30 articles in international conferences and journals. He has coordinated the EU-funded project MaCoCu, aimed at harvesting monolingual/parallel corpora for low-resourced European languages from the Internet.

Juan Antonio Pérez-Ortiz is Associate Professor of computer science at Universitat d'Alacant, Spain, director of the Transducens research group, and co-founder of Prompsit Language Engineering. He has worked on machine translation (rule-based, statistical and neural) and computer-aided translation since 1999, especially as a member of the team involved in the development of the Apertium platform. His current research focuses on neural language technologies for low-resource languages.

Felipe Sánchez-Martínez is Associate Professor at Universitat d'Alacant, Spain. His main field of research is on low-resource machine translation and the integration of machine translation in other translation technologies. He has participated in several research projects funded by the Spanish Government and the European Commission. He contributed to the design and development of the Apertium shallow-transfer machine translation platform, and co-founded the company Prompsit Language Engineering.