

# Gated Adapters for Multi-Domain Neural Machine Translation

Mateusz Klimaszewski<sup>a,\*</sup>, Zeno Belligoli<sup>b</sup>, Satendra Kumar<sup>b</sup> and Emmanouil Stergiadis<sup>b</sup>

<sup>a</sup>Warsaw University of Technology

<sup>b</sup>Booking.com

**Abstract.** The Adapter framework introduces lightweight modules that reduce the complexity of Multi-Domain Machine Translation systems. Compared to fine-tuned models, Adapters train faster, do not overfit, have smaller memory requirements, and maintain the base model intact. However, just like fine-tuned models, they need prior information about the domain of the sentence. Otherwise, their performance decreases for out-of-domain and unknown-domain samples. In this work, we propose a solution that does not require the information and can decide on the sample’s origin on-the-fly without compromising quality or latency. We introduce a built-in gating mechanism utilising a knowledge distillation framework to activate a subset of softly-gated, domain-specific Adapters that are relevant to the sentence. The effectiveness of the proposed solution is demonstrated through our experiments on two language pairs, using both in-domain and out-of-domain datasets. Our analysis reveals that Gated Adapters provide significant benefits, particularly in the case of ambiguous, misclassified samples, resulting in an improvement of over +5 COMET points.

## 1 Introduction

Neural Machine Translation (NMT) emerged as a go-to solution for Machine Translation, providing state-of-the-art results, especially in high-resource scenarios [2, 43, 32]. NMT models are usually trained using large, general-purpose parallel corpora. Therefore, to limit one of the known shortcomings of NMT – out-of-domain translation [19], there is a need to perform domain adaptation and improve the quality in the unknown domain, which might not be well represented in the parallel corpora.

Multi-Domain Machine Translation (MDMT) is a technique aimed at addressing the shortcomings of a general-purpose NMT model in translating text that falls outside its scope from various domains. According to Koehn and Knowles [19], a domain is characterized by a corpus from a particular source and may differ in terms of topic, genre, style, level of formality, among other things. This complexity underscores the challenge of MDMT. While fine-tuning one model for each domain is a straightforward approach that has been proven to be effective [12], it becomes challenging to implement in real-world scenarios where the number of domains and language pairs is substantial.

Recently, the Adapter framework [15] has been introduced as

an alternative to regular fine-tuning. Adapters are lightweight modules injected into a pre-trained model and fine-tuned to a specific task. This method requires training only newly introduced parameters, keeping the base model frozen. In a multi-domain setting, one Adapter per domain must be trained. However, unlike fine-tuned models, Adapters can be deployed together when they share the same base model. On the downside, the domain of each sentence must be known at inference time to activate the right Adapter. When the origin of the sentence is unknown or out-of-domain (we refer to both cases as an unannotated domain), a classifier is typically used to predict a likely domain [17]. This solution has two drawbacks: (i) it comes with a latency cost, as a pipeline approach increases the overall complexity, and (ii) it requires extra computation resources (i.e. additional GPU unit) to perform on-the-fly classification.

In this work, we propose a built-in gating mechanism, named Gated Adapters (GAD), to handle unannotated domains without compromising quality or latency. Gated Adapters extend the Adapter framework with the gates learnt via knowledge distillation [14]. The gates perform a fusion between sample-relevant Adapter modules. In contrast to the Adapters, GAD performs a soft-gating, i.e. multiple Adapters might be triggered, rather than a hard-gating when only one Adapter is used. Soft-gating in Adapter modules allows them to share relevant, cross-domain knowledge with each other (i.e. enhancing positive transfer learning). This is unlike the standard Adapters, which isolate a medical Adapter from a law one, for example. Additionally, the proposed method does not require an external classifier during inference and performs the domain prediction on-the-fly.

We evaluate the Gated Adapters on in- and out-of-domain translation, showing that the performance is on-par or better than the previous work. Moreover, our analysis reveals that in the case of ambiguous, misclassified examples (i.e. samples where the external classifier would assign an incorrect label), GAD outperforms other MDMT systems. To summarise, our contributions are as follows:

- We propose Gated Adapters as an extension to Adapters in the MDMT setting that does not require an external classifier at inference when the origin of the sentence to translate is unknown.
- We present an extensive evaluation of two language pairs: English to Polish and English to Greek, with six domains per pair.

## 2 Method

### 2.1 Adapters

Adapters [15] are lightweight modules injected into a pre-trained model and trained on new data while keeping the pre-trained model

\* Corresponding Author. Email: mateusz.klimaszewski.dokt@pw.edu.pl. Work done during an internship at Booking.com. Code and Appendix at <https://github.com/mklimasz/gated-adapters>

frozen. This means that Adapters train only a fraction of the parameters of the initial model. Furthermore, because Adapters do not alter the base model, unlike conventional fine-tuning, there is no need to maintain a separate model for each task (e.g. domain).

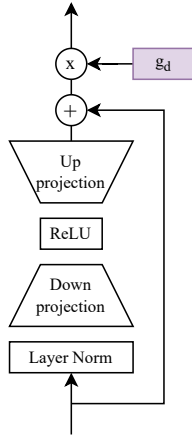
In the standard NMT setup [4], an Adapter (AD) processes a transformer hidden state  $x$  at a layer  $i$  and consists of a residual connection [13], a layer norm LN [1] and two linear layers: down-project  $D$  and up-project  $U$ , creating a bottleneck with an activation function ReLU [24].

$$AD_i(x_i) = U(\text{ReLU}(D(\text{LN}(x_i)))) + x_i \quad (1)$$

## 2.2 Gated Adapters

This work extends the Adapter framework by introducing a gating mechanism that allows the system to handle sentences from any domain and decide on its domain on the fly. The module provides probability-based soft gating that, given a set of domain-specific Adapters, multiplies each Adapter’s output by a factor proportional to the probability of the sentence belonging to the Adapter’s domain. This approach follows the mixture-of-experts (MoE) technique [38]; however, in contrast to regular MoEs, the experts in our proposed model have a pre-defined role – they are domain-specific modules.

In the following subsections, we describe (i) the gating mechanism and (ii) the knowledge distillation framework used to train the gates. The overview of our method is presented in Figures 1 and 2.



**Figure 1.** The schema illustrates a single Gated Adapter module, where the Adapter’s output is multiplied by the probability value provided by the external gating module ( $g_d$ ). This probability value indicates the degree to which a sentence belongs to the domain represented by the Adapter.

### 2.2.1 Gating mechanism

The gating mechanism is injected at each transformer layer  $i$  and acts as a weighted average over the output of each Adapter at that layer:

$$x_{\text{out}} = \sum_d^D g_{\text{norm}_d} AD_d(x_{\text{in}}) \quad (2)$$

where  $x_{\text{in}} \in R^{\text{hidden\_dim}}$  is the Adapter’s input, and  $g_{\text{norm}}$  is computed as:

$$g_{\text{norm}} = \text{norm}(W_g \times \text{agg}(x_{\text{in}_T})) \quad (3)$$

Here  $W_g \in R^{D \times \text{hidden\_dim}}$  is a matrix of learnable weights, norm is a general normalisation function, and agg is a general aggregation function over all the time steps  $T$   $x_{\text{in}_T} = x_{1:T}$  for the encoder layers, and over all the steps up to the current one  $x_{\text{in}_T} = x_{1:t}$  for the decoder layers ( $x_{\text{in}_T} \in R^{\text{hidden\_dim} \times |T|}$ ). In this work, we set norm to a softmax with a temperature parameter  $\beta$  and agg to a standard average operation.

### 2.2.2 Knowledge distillation

A standard NMT model is trained using a cross-entropy loss ( $\mathcal{L}_{\text{CE}}$ ) with label smoothing [41]. We extend this setup and apply the knowledge distillation framework [14] to learn the values of the gates. Given a source sentence  $s$ , we estimate a probability distribution over domains conditioned on the sentence. As the actual distribution is unknown, we provide it as an estimation from an external classifier ( $P_{\text{clf}} = P_\theta(d|s)$ , implementation details in Section 3.2.2). The additional objective, Kullback–Leibler divergence ( $\mathcal{L}_{\text{KL}}$ ), teaches the gates to mimic the teacher model.

We train the model jointly, in the same manner as Adapters, freezing everything but the parameters of Adapters and gates. The hyperparameter  $\alpha$  weights the impact of the additional loss function, and  $\tau$  is a softmax temperature used to estimate the probabilities  $P_g$  (obtained as a softmax function over the gates values  $g$  from Equation 3).

$$\mathcal{L}_{\text{KL}} = \tau^2 D_{\text{KL}}(P_{\text{clf}_\tau} || P_{g_\tau}) \quad (4)$$

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \alpha \mathcal{L}_{\text{KL}} \quad (5)$$

## 3 Experiments

### 3.1 Data

Our experimental setup involves two language pairs: English to Polish and English to Greek. We initiated the experiments by training a general-purpose machine translation model using ParaCrawl [3] as a baseline (BASE). To ensure the effectiveness of our approach, we selected a diverse set of domains from OPUS [42], including medical, legal, and IT domains, which vary significantly in terms of style, level of formality, and domain-specific terminology. By incorporating these domains, we aimed to demonstrate the robustness of our approach in handling various domain adaptation scenarios. The chosen six domains are listed below:

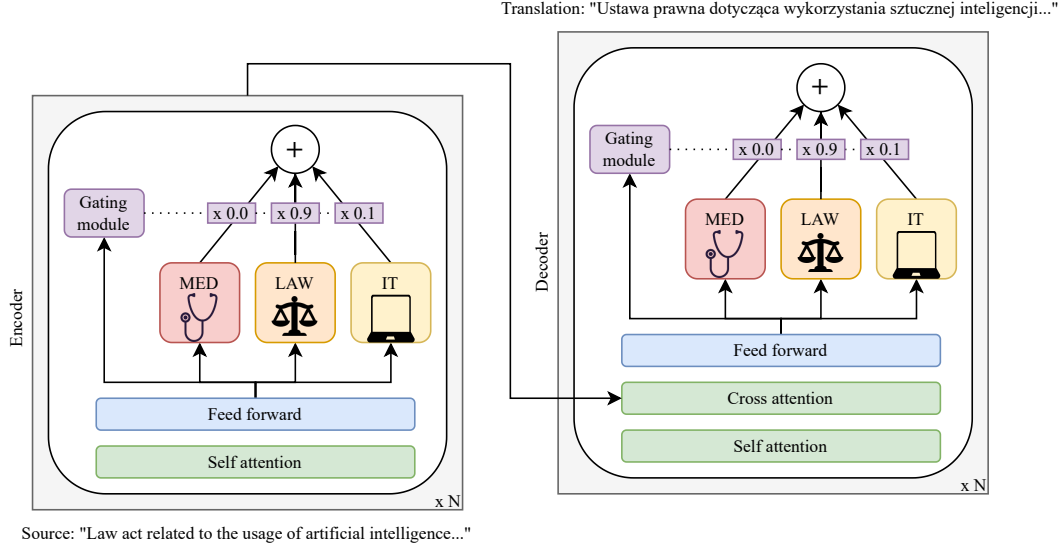
- LAW: legal documents from JRC Acquis
- IT: combination of KDE4 (only EN→PL), PHP, GNOME and Ubuntu localisation files
- SUB: a subset of OpenSubtitles 2018<sup>1</sup>
- TALK: TED Talks transcripts [36]
- MED: medical documents from European Medicines Agency (EMA)
- REL: Bible (EN→EL) [8] and Koran (EN→PL)

The statistics of the training data after pre-processing (including punctuation normalisation, ratio, language [23], length and dictionary-based filtering) are presented in Table 1. We held 2000 examples per domain for evaluation purposes (1000 for validation and 1000 as a test set).

Our analysis of the data involved utilising an SVM classifier<sup>2</sup> with averaged BERT [9] embeddings as features to measure the A-Proxy [6] distance between the domains. The A-Proxy distance is

<sup>1</sup> <http://www.opensubtitles.org>

<sup>2</sup> As implemented in scikit-learn [28]



**Figure 2.** Overview of Gated Adapters. Given a sentence, the gating module predicts the probability of the sentence belonging to each domain. The probabilities behave as a weighting factor for the corresponding domain-specific Adapters. In the example sentence, the gates lean towards law and IT Adapters and discard the medical one, as the text concerns an AI-related law act.

**Table 1.** Statistics of the training corpora as the number of parallel sentences. The table does not include 2000 parallel sentences per domain for validation and test purposes.

	English→Polish	English→Greek
BASE	33M	20,1M
LAW	838k	1244k
IT	96k	89k
SUB	1854k	1780k
TALK	206k	257k
MED	229k	257k
REL	108k	59k
SUM	3331k	3686k

a measure that falls within the range of 0 to 2, where 0 indicates a perfect domain match and 2 represents complete separability. As shown in Figure 3, the domains were correctly separated, with SUB and TALK demonstrating the closest relationship and LAW exhibiting the greatest distance from the others.

### 3.2 Systems

We employed the Transformer Base [43] architecture implemented in fairseq<sup>3</sup> [26] for all our models. It consists of six encoder and six decoder layers, with an embedding dimension of 512, an FFN of 2048, and eight attention heads. The source and target embeddings are shared and tied with the output layer. We tokenised the data using a unigram SentencePiece model [20, 21] with a size of 32k. Table 2 presents the parameters of all the systems described in the following sections.

#### 3.2.1 Baselines

The experiments begin with training a general-purpose model, labelled as BASE, using large-scale data from ParaCrawl (refer to Ta-

LAW	1.91	1.97	1.94	1.87	1.99
IT		1.91	1.91	1.85	1.96
SUB			1.19	1.96	1.83
TALK				1.94	1.82
MED					1.98
	IT	SUB	TALK	MED	REL

**Figure 3.** The A-Prox distance between the domains.

**Table 2.** The rounded number of overall and trainable parameters of the evaluated models. In square brackets, we denoted the relative difference to the BASE model.

Model	Parameters	Trainable
BASE	79M	79M
FT	6 × 79M [+5 × 79M]	6 × 79M
MIX	79M	79M
TAG	79M [+3k]	79M
AD	98M [+19M]	19M
GAD	98M [+19M]	19M

<sup>3</sup> fairseq architecture: transformer\_wmt\_en\_de

ble 1). This model is evaluated on all domains to establish a lower bound for all MDMT systems and is used as a pre-trained Machine Translation model (i.e. MDMT systems build upon the model rather than starting from scratch). Additionally, we employ this model for fine-tuning (FT) to create a set of domain-specific models, each for a different domain. This strategy is an upper bound for MDMT systems; however, it has limitations when scaling the solution across various language pairs and domains as it produces a separate model per domain. The training details of these and the following models are described in the Appendix.

We employ two MDMT, non-adapter baselines: (i) *MIX*, which is straightforward training the model on a concatenation of domain corpora, (ii) *TAG*, which adds a domain-control mechanism in the form of domain-specific tag included into each source sentence and enables the model to differentiate between the domains [17, 39]. Both methods use *BASE* as a starting checkpoint.

### 3.2.2 Adapter-based systems

To assess the effectiveness of the Adapter-based systems, we examine the standard Adapters (AD) and compare their performance with the newly proposed Gated Adapters (GAD). Compared to other MDMT systems, Adapter-based systems train a fraction of parameters (refer to Table 2) as these methods freeze the NMT model and train only the Adapter modules. For AD and GAD, we rely on *MIX* as a starting checkpoint [31] and use Adapter modules with a bottleneck of 2 (i.e. reducing the dimensionality via the down-project layer  $D$  by 2). The rest of the training procedure is consistent with the other MDMT systems.

**Gated Adapters** use RoBERTa<sup>4</sup> [22] as a base model for an external classifier required for knowledge distillation (see Eq. 4). We train two classifiers, one per language pair, using the English side of the parallel corpora as the datasets are not equivalent, e.g. EN→EL uses Bible and EN→PL Koran. To prevent data leakage, only the training parallel corpora are used to train and validate the models. The evaluation of the classifiers is presented in Table 3. The classifiers serve not only as a teacher model for GAD (i.e. required only during training) but also as a means of predicting the domain for the TAG and AD baselines during inference. In the results section, we denote the systems that rely on the classifier during inference with an index CLF. Those baselines are constructed as a pipeline solution, i.e. first, the classifier predicts a domain, and then the MDMT model translates a sentence. For clarity, we present the ORACLE version as an upper bound of those systems, which always utilises the ground truth domain.

Furthermore, we up-sampled all the domains to the one with the highest sentence count. This step prevents high-resource domains from overshadowing other domains’ weights. Otherwise, we noticed in preliminary experiments that a high-resource domain could harm a similar (in terms of domain closeness) lower-resource domain (i.e. TALK in SUB-TALK pair).

### 3.3 Metrics

Following the study and recommendation of Kocmi et al. [18], we use COMET<sup>5</sup> [35] as main evaluation metric. In addition, we provide

**Table 3.** Quality of the RoBERTa-based classifiers in terms micro-averaged F1 score.

Model	F1
EN → PL	95.65
EN → EL	94.83

chrF [33] and BLEU [27] scores using SacreBLEU<sup>6,7</sup> [34]. Due to computational and time constraints, we compute three independent runs exclusively for Adapter-based systems (AD and GAD) and report an average score with standard deviation for them.

### 3.4 Results

Table 4 presents the evaluation results. We report both per-domain scores and aggregated metrics - unweighted and weighted averages *AVG*, *wAVG*. The *AVG* metric should be treated as the primary metric determining the quality of an MDMT system in the case of balanced test distribution; the *wAVG* in the case of the test distribution matching the training one. The weights for the latter metric are derived from the ratio of domain-specific data based on the number of sentences (see Table 1).

Gated Adapters perform the best out of all MDMT systems based on aggregated metrics in both language pairs. Overall, Gated Adapters are on-par or better than not only methods that require a classifier but also their oracle version (e.g. Adapters with ground truth domain tag) while simultaneously providing the possibility of handling unannotated domains. Especially in the case of the *AVG* metric, the GAD outperforms AD<sub>CLF</sub> with +1.5 and +3.5 COMET point gain in English to Polish and English to Greek language pairs correspondingly. We report other automatic evaluation metrics: chrF and BLEU, in the Appendix.

## 4 Method analysis

This section dissects the Gated Adapters to examine the method’s advantages and explain its performance beyond the main, in-domain results. We analyse the cross-domain and out-of-domain capabilities in Sections 4.1 and 4.3, measure the efficiency in Section 4.2 and perform an ablation study in 4.4.

### 4.1 Knowledge sharing

The preliminary analysis revealed that the SUB and TALK domains are the most related in terms of A-Proxy distance. This observation is consistent with the achieved results. In Table 4, the CLF versions of TAG and AD models have the most decrease in quality compared to the ORACLE counterpart in these two domains. Additionally, the confusion matrix of the classifier presented in Figure 4 demonstrates that those two domains were the most difficult to distinguish in the EN→PL dataset. While the other domains are classified with high accuracy, rarely making any mistakes, the pair of SUB and TALK is the most troublesome to both classifiers (the same phenomenon appears in EN→EL, see Appendix).

The GAD model can handle ambiguous, cross-domain examples (i.e. examples for which two or more domains are probable according to the classifier) because it has learnt a soft gating mechanism that allows knowledge sharing among outputs of different Adapters (see

<sup>4</sup> roberta-large [45]

<sup>5</sup> We use wmt20-comet-da COMET model and multiply results by 100

<sup>6</sup> chrF2|#:1|c:mixed|e:yes|nc:6|nw:0|s:no|v:2.2.0

<sup>7</sup> BLEU|#:1|c:mixed|e:no|tok:13a|s:exp|v:2.2.0

**Table 4.** Translation performance measured using COMET. For each system, we aggregate scores using an unweighted and weighted average, where the weights come from the ratio of domain-specific training data based on the number of sentences. We report average scores over three runs with a standard deviation for AD and GAD.

	LAW	IT	SUB	TALK	MED	REL	AVG	wAVG
English – Polish								
BASE	84.58	39.18	30.74	46.24	54.79	11.21	44.46	46.50
FT	97.30	66.73	48.93	53.26	86.29	105.36	76.31	66.28
TAGORACLE	95.76	61.20	47.22	53.03	82.78	89.12	71.52	64.00
ADORACLE	96.11±0.27	63.47±1.90	46.98±0.53	53.36±0.80	83.26±0.38	98.83±1.25	73.67±0.36	64.38±0.31
MIX	95.79	61.93	<b>47.50</b>	52.42	82.56	86.88	71.18	64.05
TAGCLF	95.73	61.30	45.90	52.41	82.12	88.86	71.05	63.16
ADCLF	<b>95.81</b> ±0.21	63.37±1.73	46.31±0.36	52.81±0.66	82.71±0.40	98.04±1.21	73.17±0.31	63.84±0.23
GAD	95.47±0.13	<b>64.78</b> ±0.87	46.97±0.35	<b>53.57</b> ±0.22	<b>83.55</b> ±0.67	<b>103.67</b> ±0.23	<b>74.67</b> ±0.30	<b>64.44</b> ±0.18
English – Greek								
BASE	80.74	24.31	38.53	66.89	35.38	10.67	42.75	53.74
FT	87.55	73.09	53.28	77.02	74.68	78.31	73.99	68.87
TAGORACLE	88.34	62.70	51.30	75.19	72.36	46.03	65.99	67.13
ADORACLE	88.07±0.10	68.75±2.65	51.91±0.47	75.71±0.54	74.21±0.61	51.23±0.46	68.31±0.51	67.72±0.31
MIX	87.79	66.51	51.51	73.74	73.30	45.64	66.42	67.09
TAGCLF	<b>88.19</b>	61.97	50.05	73.58	72.08	45.85	65.29	66.32
ADCLF	88.04±0.08	68.60±2.54	50.91±0.41	73.97±0.26	73.79±0.62	50.91±0.41	67.70±0.45	67.07±0.26
GAD	87.69±0.10	<b>69.34</b> ±0.35	<b>52.29</b> ±0.25	<b>74.67</b> ±0.59	<b>73.83</b> ±0.29	<b>70.23</b> ±1.22	<b>71.34</b> ±0.40	<b>68.00</b> ±0.14

Equation 2). Considering just misclassified (i.e. with a predicted non-ground truth domain label) examples from the test dataset, the GAD outperforms its counterpart in both language pairs by over 5 and 8 COMET points. Table 5 presents the results of the evaluation. The quality of the methods that require a classifier during inference (TAG, AD) drops significantly compared to GAD. While the Gated Adapters use the same classifier during training (the classifier makes the same mistakes), GAD is aware of the uncertainty (i.e. soft-gating instead of hard-gating) and learns to handle such cases during knowledge distillation. Table 6 presents the translation examples with the impact of misclassification, showing that a wrong domain label may lead to a meaningless translation in extreme cases.

**Table 5.** Translation evaluation of the misclassified sentences from the test dataset using COMET. Gated Adapters outperform both methods that require a classifier at inference whenever the classifier fails to predict a correct domain label.

	EN→PL	EN→EL
TAGCLF	31.98	45.46
ADCLF	42.59	55.33
GAD	<b>48.33</b>	<b>64.01</b>

Predicted class	LAW	982	4	0	5	7	0
	IT	7	985	2	1	5	0
	SUB	1	3	906	93	1	8
	TALK	3	2	78	893	1	5
	MED	7	6	5	2	986	0
	REL	0	0	9	6	0	987
		LAW	IT	SUB	TALK	MED	REL
		True class					

## 4.2 Efficiency

In order to evaluate the efficiency of our proposed model, we conducted experiments to compare inference time. We calculated the number of generated tokens and translation duration per domain and aggregated the values to report the number of processed sentences and tokens per second. We performed the inference per domain because each domain differs in its characteristics, such as average sentence length. All experiments were run on a single NVIDIA V100 GPU, with a batch size of 64 and with greedy decoding.

Our method introduces two additional drawbacks that affect efficiency: (i) the gating module and (ii) the requirement of using all the Adapters to perform the aggregation. To limit the impact of the latter

**Figure 4.** Confusion matrix for the classifier in the EN→PL language pair. Albeit its overall high quality, the model makes almost exclusively mistakes between the SUB–TALK domain pair.

**Table 6.** Misclassified examples from the EN→PL test dataset. AD generates higher-quality translation when we manually provide the right domain during inference (i.e. by changing from CLF to ORACLE). At the same time, GAD does not rely on an external classifier and therefore does not suffer from the aforementioned issue. Misclassification can lead to meaningless translation, as in the second example, where the model produces a relevant translation only after providing the correct label, i.e. changing from TALK to REL (“trunk” given the context is incorrectly translated to 🌳 “bagażnik” instead of 🌴 “pień”).

Source	These events are often transitory.
Reference	Zaburzenia te są często przemijające.
AD <sub>CLF</sub> =TALK	Zdarzenia te są często przejściowe.
AD <sub>ORACLE</sub> =MED	Zdarzenia te są często przemijające.
GAD	Zdarzenia te są często przemijające.
Source	The birth pangs brought her to the trunk of a date palm.
Reference	I doprowadziły ją bóle porodowe do pnia drzewa palmowego.
AD <sub>CLF</sub> =TALK	Pangi narodzin przywiozły ją do bagażnika palmy randkowej.
AD <sub>ORACLE</sub> =REL	I przyniosły ją bóle porodowe do pnia drzewa palmowego.
GAD	I doprowadziły ją bóle porodowe do pnia palmy daktylowej.

drawback, we implemented a parallel approach instead of a sequential one. In the sequential approach, domain-specific Adapters are processed one at a time, whereas in the parallel approach, all steps are processed simultaneously, except for layer norms, via multi-channel linear layers (i.e. the down-project  $D$  and up-project  $U$  layer with the non-linear function ReLU) instead of iterating over domains.

We present the comparison between the Adapters + classifier (AD<sub>CLF</sub>) pipeline versus Gated Adapters in Table 7. For reference, we also include the raw Adapters, which assume a scenario where the right domain is known. The Gated Adapters outperform the pipeline scenario of Adapters preceded by a classifier. While GAD adds an overhead over the Adapters setup, it does not require an additional classifier. The Gated Adapters can use just one device (i.e., GPU) at a time, whereas the pipeline requires two devices to avoid the overhead of checkpoint loading for online translation. Additionally, compared to the Adapters without a classifier, GAD does not need information about the origin of a sample.

**Table 7.** Efficiency comparison in terms of processed sentences per second and tokens per second between the classifier and Adapters pipeline (AD<sub>CLF</sub>) and Gated Adapters (GAD). For reference, we include the standalone Adapters (AD<sub>ORACLE</sub>) values, that assumes prior domain knowledge for each sentence.

	sentences/s	tokens/s
AD <sub>ORACLE</sub>	88.11	2091.77
AD <sub>CLF</sub>	51.64	1225.90
GAD	<b>58.65</b>	<b>1392.12</b>

### 4.3 Out-of-domain evaluation

In Sections 3.4 and 4.1, we demonstrated the in-domain and cross-domain capabilities of the Gated Adapters. However, as the gates are merely a distilled version of an external classifier, the out-of-domain capabilities remain in question. Therefore, we performed an additional, out-of-domain evaluation to verify the gating mechanism’s robustness. This step checks whether the GAD’s quality does not decrease for out-of-domain samples and persists quality of the classifier as in AD<sub>CLF</sub>. Both MDMT systems attempt to use an external classifier (RoBERTa in AD<sub>CLF</sub>) or an internal one (gates in GAD) to map out-of-domain samples into one of the pre-defined domains.

We evaluate the models on two out-of-domain datasets: Flores-200 devtest [25] and WMT’20 News test [5] dataset (the latter available only for EN→PL). We report the results in Table 8. Although the gates match around 0.03% size of the classifier in terms of the

number of parameters (the gates introduce less than 40k new parameters), they retain similar performance and generalizability. On both datasets, GAD presents on-par results with AD<sub>CLF</sub> while using a distilled version of the classifier embedded into the model and making domain prediction on-the-fly, verifying the gating mechanism’s robustness.

**Table 8.** COMET scores for an out-of-domain evaluation on the Flores-200 devtest and News WMT’20 test dataset. We report average of three runs with the standard deviation.

	EN → PL	EN → EL
Flores		
AD <sub>CLF</sub>	57.61±0.24	67.00±0.46
GAD	57.63±0.24	66.90±0.32
News		
AD <sub>CLF</sub>	46.44±0.71	–
GAD	46.65±0.46	–

### 4.4 Knowledge distillation ablation

We conducted an ablation study to examine the effect of treating gates as a regular classifier and using cross-entropy loss instead of knowledge distillation, which is in line with the method used in previous works by Britz et al. [7] and Pham et al. [30]. The validation dataset was used to present the ablation results in Table 9. The outcomes demonstrate the advantages of the proposed approach, as it enables Gated Adapters to match and even surpass the quality of Adapters.

**Table 9.** Ablation on the EN→PL validation dataset comparing training the gates as a classifier (CE) against the knowledge distillation (KD) framework. We report AVG and wAVG for COMET score

	AVG	wAVG
CE	72.02	61.51
KD	<b>73.62</b>	<b>62.73</b>

## 5 Related work

The mixture-of-experts models are gaining more traction in the Machine Translation field [38]. Recently, Dua et al. [10] propose a temperature heating mechanism and dense pre-training for easing the

convergence of MoE MT models. The NLLB Team [25] presented a multilingual MoE model on a larger scale, breaking the 200 languages barrier.

Adapters, as a specific version of a MoE, were lately also used for the task of domain adaptation. The work of Vu et al. [44] focuses on the domain generalisation task via Adapter leave one-out strategy. In the similar, regularisation focused way, (and additionally improving overall complexity), Rücklé et al. [37] proposed AdapterDrop technique to drop out Adapter layers, similarly to removing Transformer layers [11]. The presented works can be applied to any Adapter-based MDMT system and could be applied with the GAD model.

Pfeiffer et al. [29] introduce the AdapterFusion technique, which, as our work, shares the knowledge between multiple Adapter modules. However, their method requires additional, separate training as they extend the regular Adapter setup with a fusion layer on top of the multiple Adapters and train the new parameters with the base model and Adapter modules frozen. Moreover, they focus on the multi-task setup rather than the multi-domain one. Pham et al. [30] propose to extend a highway version of residual Adapters with domain classifiers on top of an encoder and decoder and decide on a domain on a word-per-word basis. They evaluate the solution in the MDMT setting. As in the work of Pfeiffer et al. [29], they use an additional training procedure that requires separate training of the classifiers.

## 6 Conclusions

In this work, we present an extension to the Adapters framework in the MDMT setting called Gated Adapters, which perform soft-gating over multiple domain-specific Adapters. We evaluate the validity of the proposed solution on two language pairs and across six domains.

We show that GAD not only improves upon regular Adapters but also demonstrates resistance to domain misclassification and provides high-quality translation, even when the sentences are ambiguous in terms of their domain. Moreover, the proposed solution does not require an external classifier at the inference time, making the use more efficient – it requires less computational resources than a pipeline solution of a classifier with an MDMT model (e.g. Adapters AD).

## 7 Limitations

The main limitation of our technique is the data requirements. We test our method on high-resource language pairs and domains that fall within the mid-to-high-resource range. There is not enough evidence that the technique would work for (extremely) low-resource domains, considering the up-sampling required by Gated Adapters. Future work could investigate if this is a shortcoming of the proposed method. Furthermore, we rely on a classifier that is built upon a pre-trained language model [22], which may not be sufficiently robust to attain the desired level of accuracy in low-resource languages or may not be accessible at all.

## Acknowledgements

We thank Alex Umnov and Fengjun Wang for their reviews and comments during this project. Part of the computations was performed at Poznań Supercomputing and Networking Center.

## References

[1] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton, ‘Layer normalization’, *CoRR*, **abs/1607.06450**, (2016).

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, ‘Neural machine translation by jointly learning to align and translate’, *CoRR*, **abs/1409.0473**, (2015).

[3] 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 Sarrias, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza, ‘ParaCrawl: Web-scale acquisition of parallel corpora’, in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4555–4567, Online, (July 2020). Association for Computational Linguistics.

[4] Ankur Bapna and Orhan Firat, ‘Simple, scalable adaptation for neural machine translation’, in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1538–1548, Hong Kong, China, (November 2019). Association for Computational Linguistics.

[5] Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri, ‘Findings of the 2020 conference on machine translation (WMT20)’, in *Proceedings of the Fifth Conference on Machine Translation*, pp. 1–55, Online, (November 2020). Association for Computational Linguistics.

[6] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira, ‘Analysis of representations for domain adaptation’, in *Advances in Neural Information Processing Systems*, volume 19. MIT Press, (2006).

[7] Denny Britz, Quoc Le, and Reid Pryzant, ‘Effective domain mixing for neural machine translation’, in *Proceedings of the Second Conference on Machine Translation*, pp. 118–126, Copenhagen, Denmark, (September 2017). Association for Computational Linguistics.

[8] Christos Christodouloupoulos and Mark Steedman, ‘A massively parallel corpus: the bible in 100 languages’, *Language Resources and Evaluation*, **49**(2), 375–395, (Jun 2015).

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, ‘BERT: Pre-training of deep bidirectional transformers for language understanding’, in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, (June 2019). Association for Computational Linguistics.

[10] Dheeru Dua, Shruti Bhosale, Vedanuj Goswami, James Cross, Mike Lewis, and Angela Fan, ‘Tricks for training sparse translation models’, in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3340–3345, Seattle, United States, (July 2022). Association for Computational Linguistics.

[11] Angela Fan, Edouard Grave, and Armand Joulin, ‘Reducing transformer depth on demand with structured dropout’, in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, (2020).

[12] Markus Freitag and Yaser Al-Onaizan, ‘Fast domain adaptation for neural machine translation’, *ArXiv*, **abs/1612.06897**, (2016).

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, ‘Deep residual learning for image recognition’, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, (2016).

[14] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean, ‘Distilling the knowledge in a neural network’, *ArXiv*, **abs/1503.02531**, (2015).

[15] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morroni, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly, ‘Parameter-efficient transfer learning for NLP’, in *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, (09–15 Jun 2019).

[16] Diederik P. Kingma and Jimmy Ba, ‘Adam: A method for stochastic optimization’, in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, (2015).

[17] Catherine Kobus, Josep Crego, and Jean Senellart, ‘Domain control for neural machine translation’, in *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*,

- pp. 372–378, Varna, Bulgaria, (September 2017). INCOMA Ltd.
- [18] Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes, ‘To ship or not to ship: An extensive evaluation of automatic metrics for machine translation’, in *Proceedings of the Sixth Conference on Machine Translation*, pp. 478–494, Online, (November 2021). Association for Computational Linguistics.
  - [19] Philipp Koehn and Rebecca Knowles, ‘Six challenges for neural machine translation’, in *Proceedings of the First Workshop on Neural Machine Translation*, pp. 28–39, Vancouver, (August 2017). Association for Computational Linguistics.
  - [20] Taku Kudo, ‘Subword regularization: Improving neural network translation models with multiple subword candidates’, in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 66–75, Melbourne, Australia, (July 2018). Association for Computational Linguistics.
  - [21] Taku Kudo and John Richardson, ‘SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing’, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 66–71, Brussels, Belgium, (November 2018). Association for Computational Linguistics.
  - [22] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, ‘Roberta: A robustly optimized BERT pretraining approach’, *CoRR*, **abs/1907.11692**, (2019).
  - [23] Marco Lui and Timothy Baldwin, ‘langid.py: An off-the-shelf language identification tool’, in *Proceedings of the ACL 2012 System Demonstrations*, pp. 25–30, Jeju Island, Korea, (July 2012). Association for Computational Linguistics.
  - [24] Vinod Nair and Geoffrey E. Hinton, ‘Rectified linear units improve restricted boltzmann machines’, in *Proceedings of the 27th International Conference on Machine Learning*, ICML’10, p. 807–814, Madison, WI, USA, (2010). Omnipress.
  - [25] NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation, 2022.
  - [26] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli, ‘fairseq: A fast, extensible toolkit for sequence modeling’, in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pp. 48–53, Minneapolis, Minnesota, (June 2019). Association for Computational Linguistics.
  - [27] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, ‘Bleu: a method for automatic evaluation of machine translation’, in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA, (July 2002). Association for Computational Linguistics.
  - [28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, ‘Scikit-learn: Machine learning in Python’, *Journal of Machine Learning Research*, **12**, 2825–2830, (2011).
  - [29] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych, ‘AdapterFusion: Non-destructive task composition for transfer learning’, in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503, Online, (April 2021). Association for Computational Linguistics.
  - [30] Minh Quang Pham, Josep Maria Crego, François Yvon, and Jean Senelart, ‘A study of residual adapters for multi-domain neural machine translation’, in *Proceedings of the Fifth Conference on Machine Translation*, pp. 617–628, Online, (November 2020). Association for Computational Linguistics.
  - [31] Minh Quang Pham, Josep Maria Crego, and François Yvon, ‘Revisiting multi-domain machine translation’, *Transactions of the Association for Computational Linguistics*, **9**, 17–35, (2021).
  - [32] Martin Popel, Marketa Tomkova, Jakub Tomek, Łukasz Kaiser, Jakob Uszkoreit, Ondřej Bojar, and Zdeněk Žabokrtský, ‘Transforming machine translation: a deep learning system reaches news translation quality comparable to human professionals’, *Nature Communications*, **11**(1), 4381, (Sep 2020).
  - [33] Maja Popović, ‘chrF: character n-gram F-score for automatic MT evaluation’, in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 392–395, Lisbon, Portugal, (September 2015). Association for Computational Linguistics.
  - [34] Matt Post, ‘A call for clarity in reporting BLEU scores’, in *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, Brussels, Belgium, (October 2018). Association for Computational Linguistics.
  - [35] Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie, ‘COMET: A neural framework for MT evaluation’, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2685–2702, Online, (November 2020). Association for Computational Linguistics.
  - [36] Nils Reimers and Iryna Gurevych, ‘Making monolingual sentence embeddings multilingual using knowledge distillation’, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4512–4525, Online, (November 2020). Association for Computational Linguistics.
  - [37] Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych, ‘AdapterDrop: On the efficiency of adapters in transformers’, in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7930–7946, Online and Punta Cana, Dominican Republic, (November 2021). Association for Computational Linguistics.
  - [38] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarczyk, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean, ‘Outrageously large neural networks: The sparsely-gated mixture-of-experts layer’, in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, (2017).
  - [39] Emmanouil Stergiadis, Satendra Kumar, Fedor Kovalev, and Pavel Levin, ‘Multi-domain adaptation in neural machine translation through multidimensional tagging’, in *Proceedings of Machine Translation Summit XVIII: Users and Providers Track*, pp. 396–420, Virtual, (August 2021). Association for Machine Translation in the Americas.
  - [40] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le, ‘Sequence to sequence learning with neural networks’, in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS’14*, p. 3104–3112, Cambridge, MA, USA, (2014). MIT Press.
  - [41] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna, ‘Rethinking the inception architecture for computer vision’, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2818–2826, (2016).
  - [42] Jörg Tiedemann, ‘Parallel data, tools and interfaces in OPUS’, in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pp. 2214–2218, Istanbul, Turkey, (May 2012). European Language Resources Association (ELRA).
  - [43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, ‘Attention is all you need’, in *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., (2017).
  - [44] Thuy-Trang Vu, Shahram Khadivi, Dinh Phung, and Gholamreza Haffari, ‘Domain generalisation of NMT: Fusing adapters with leave-one-domain-out training’, in *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 582–588, Dublin, Ireland, (May 2022). Association for Computational Linguistics.
  - [45] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush, ‘Transformers: State-of-the-art natural language processing’, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, (October 2020). Association for Computational Linguistics.



## A Experiment Details

In this section, we describe the hyperparameters used during training. We use Adam optimiser [16] with betas set to 0.9 and 0.98 and a learning rate of 0.0005. The training has a warm-up phase set to 4000 steps and an inverse square root scheduling. The dropout rate is set to 0.3. We use beam search with a beam of size five during inference [40]. For Adapter-based models, we use the bottleneck of 2 (i.e. reducing the dimensionality via  $D$  to 256). For Gated Adapters, we introduced three additional hyperparameters. Those values were chosen on the EN→PL validation dataset, and we used the same values for EN→EL. The impact of the  $\mathcal{L}_{KL}$  was set arbitrary to 0.5. For normalisation temperature  $\beta$  and  $\mathcal{L}_{KL}$  temperature  $\tau$ , we used the values of 2.0 and 0.1. The former was picked from  $\{0.1, 0.5, 1.0, 2.0, 2.5\}$ , the latter:  $\{0.1, 0.5, 1.0, 2.0\}$ . We trained all our models using 4 NVIDIA V100 GPUs, except the BASE model, which used 8 NVIDIA V100 GPUs.

## B Evaluation

Tables 10 and 11 present the evaluation results measured using chrF and BLEU.

## C Confusion matrix

Figure 5 presents the classifier quality for the EN→EL dataset.

Predicted class	LAW	973	1	19	26	2	2
	IT	0	980	4	1	15	0
	SUB	8	4	885	85	0	1
	TALK	14	2	83	879	4	3
	MED	5	13	3	2	979	0
	REL	0	0	6	7	0	994
		LAW	IT	SUB	TALK	MED	REL
		True class					

**Figure 5.** Confusion matrix for the classifier in the EN→EL language pair. Albeit its overall high quality, the model makes almost exclusively mistakes between the SUB–TALK domain pair.

**Table 10.** Translation performance measured using chrF. For each system, we aggregate scores using an unweighted and weighted average, where the weights come from the ratio of domain-specific training data based on the number of sentences. We report average scores over three runs with a standard deviation for AD and GAD.

	LAW	IT	SUB	TALK	MED	REL	AVG	wAVG
English – Polish								
BASE	63.25	52.07	44.20	48.24	52.27	36.64	49.44	49.78
FT	71.03	61.42	48.42	48.33	72.44	90.79	65.41	57.51
TAG <sub>ORACLE</sub>	69.93	58.66	48.16	48.57	66.40	80.51	62.04	56.27
AD <sub>ORACLE</sub>	70.31±0.06	59.08±0.71	47.97±0.23	48.40±0.14	67.21±0.32	86.52±0.64	63.25±0.14	56.51±0.14
MIX	70.18	58.69	<b>47.91</b>	48.20	67.13	80.02	62.02	56.20
TAG <sub>CLF</sub>	69.92	58.65	<b>47.91</b>	48.44	66.33	80.39	61.94	56.11
AD <sub>CLF</sub>	<b>70.30</b> ±0.06	59.06±0.69	47.83±0.13	48.27±0.19	67.17±0.29	86.24±0.62	63.14±0.12	56.41±0.08
GAD	70.16±0.08	<b>60.59</b> ±0.11	47.78±0.34	<b>48.46</b> ±0.06	<b>67.73</b> ±0.23	<b>89.71</b> ±0.08	<b>64.07</b> ±0.05	<b>56.55</b> ±0.18
English – Greek								
BASE	60.20	54.60	46.00	58.65	55.66	42.90	53.00	52.50
FT	63.02	72.42	50.19	61.26	75.01	85.33	67.87	58.12
TAG <sub>ORACLE</sub>	63.37	68.30	50.24	60.75	69.87	68.43	63.49	57.50
AD <sub>ORACLE</sub>	63.15±0.05	69.51±1.18	50.26±0.22	60.93±0.25	71.20±0.36	71.24±0.28	64.38±0.21	57.62±0.12
MIX	63.16	67.84	<b>50.16</b>	60.27	70.27	68.09	63.30	57.37
TAG <sub>CLF</sub>	<b>63.30</b>	68.11	49.64	60.41	69.81	68.33	63.27	57.16
AD <sub>CLF</sub>	63.09±0.04	69.39±1.09	49.43±0.17	60.59±0.20	<b>71.17</b> ±0.36	71.14±0.29	64.13±0.21	57.17±0.11
GAD	63.05±0.12	<b>71.10</b> ±0.14	49.56±0.08	<b>60.73</b> ±0.20	71.03±0.16	<b>81.45</b> ±0.63	<b>66.15</b> ±0.16	<b>57.43</b> ±0.04

**Table 11.** Translation performance measured using BLEU. For each system, we aggregate scores using an unweighted and weighted average, where the weights come from the ratio of domain-specific training data based on the number of sentences. We report average scores over three runs with a standard deviation for AD and GAD.

	LAW	IT	SUB	TALK	MED	REL	AVG	wAVG
English – Polish								
BASE	41.36	29.36	20.20	19.31	24.91	12.09	24.54	25.79
FT	52.54	41.23	26.87	21.33	55.97	87.38	47.55	37.37
TAG <sub>ORACLE</sub>	50.97	37.43	26.44	21.14	46.14	72.41	42.42	35.45
AD <sub>ORACLE</sub>	51.37±0.03	37.91±0.87	26.29±0.24	21.36±0.19	47.32±0.46	81.06±0.81	44.22±0.23	35.85±0.19
MIX	51.28	37.36	<b>26.42</b>	21.20	47.10	71.65	42.50	35.56
TAG <sub>CLF</sub>	50.96	37.46	26.21	21.03	46.07	72.23	42.33	35.30
AD <sub>CLF</sub>	<b>51.36</b> ±0.04	37.87±0.86	26.21±0.13	<b>21.27</b> ±0.29	47.27±0.42	80.70±0.76	44.11±0.20	35.79±0.11
GAD	51.28±0.12	<b>40.11</b> ±0.11	26.41±0.26	21.24±0.08	<b>48.31</b> ±0.20	<b>85.89</b> ±1.17	<b>45.54</b> ±0.03	<b>36.18</b> ±0.13
English – Greek								
BASE	34.41	31.45	21.37	31.11	28.99	14.88	27.04	27.12
FT	39.01	55.83	28.14	36.16	61.74	75.29	49.36	36.14
TAG <sub>ORACLE</sub>	39.38	49.32	<b>27.98</b>	35.15	52.36	46.58	41.80	34.84
AD <sub>ORACLE</sub>	39.09±0.08	51.37±1.92	27.87±0.07	35.60±0.24	54.34±0.39	50.83±0.57	43.18±0.37	34.98±0.06
MIX	39.12	48.84	27.56	34.82	52.98	46.33	41.61	34.56
TAG <sub>CLF</sub>	<b>39.29</b>	48.95	<b>27.32</b>	34.89	52.30	46.52	41.54	34.46
AD <sub>CLF</sub>	39.01±0.05	51.15±1.77	26.73±0.11	35.36±0.23	<b>54.30</b> ±0.40	50.75±0.57	42.88±0.37	34.38±0.10
GAD	39.05±0.18	<b>54.25</b> ±0.07	26.77±0.24	<b>35.38</b> ±0.18	54.13±0.24	<b>68.17</b> ±1.34	<b>46.29</b> ±0.27	<b>34.76</b> ±0.13