Soft Alignment Objectives for Robust Adaptation of Language Generation

Michal Štefánik♣[*] and **Marek Kadlčík**♣ and **Petr Sojka**♣

♣Faculty of Informatics,
Masaryk University, Czech Republic

Abstract

Domain adaptation allows generative language models to address specific flaws caused by the domain shift of their application. However, the traditional adaptation by further training on in-domain data rapidly weakens the model's ability to generalize to other domains, making the open-ended deployments of the adapted models prone to errors. This work introduces novel training objectives built upon a semantic similarity of the predicted tokens to the reference.

Our results show that (1) avoiding the common assumption of a single correct prediction by constructing the training target from tokens' semantic similarity can largely mitigate catastrophic forgetting of adaptation, while (2) preserving the adaptation in-domain quality, (3) with negligible additions to compute costs. In the broader context, the objectives grounded in a continuous token similarity pioneer the exploration of the middle ground between the efficient but naïve exact-match token-level objectives and expressive but computationally- and resource-intensive sequential objectives.

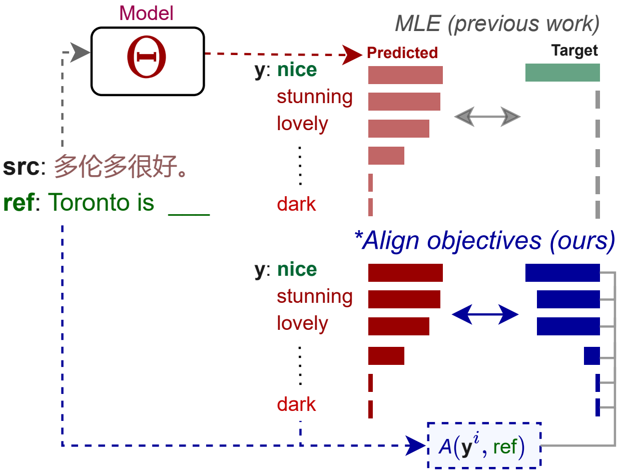

Figure 1: **Soft alignment objectives** (**Align*) replace the single-truth assumption of Maximum Likelihood Estimation (MLE) objective by constructing target distribution using Alignment *A* based on the mutual similarity of token representations. We show that learning to model ambiguity in prediction can largely mitigate the loss of generalization in adaptation.

1 Introduction

Large language models (LLMs) based on instances of encoder-decoder architecture (Neyshabur et al., 2015) provide a strong standard for generative applications of NLP, such as summarization or machine translation, mainly thanks to their outstanding ability to fluently model language. These models might face issues with *adequacy* of the generated text (Ustaszewski, 2019) when applied in data domain(s) different from the training domain, but such errors can be partially mitigated using domain adaptation (Saunders, 2021).

Identically to the pre-training phase, the adaptation is commonly carried out using Maximum Likelihood Estimation (*MLE*) objective with teacher forcing (Bahdanau et al., 2015). The popularity of this approach can be rightfully attributed to its outstanding data and computing efficiency. However, model adaptation using *MLE* notoriously comes for a price of over-specialization to the target domain, also referred to as *catastrophic forgetting* (Goodfellow et al., 2014), characterized by a continuous decay of model performance on the inputs from the *other* domains than the adaptation domain.

We hypothesize that catastrophic forgetting might be related to *MLE*'s naïve single-truth assumption, penalizing models' uncertainty over the possibly valid predictions, such as the synonyms. In domain adaptation, a repeated penalization of possibly valid tokens that are uncommon in the adapted domain might drive the model to unlearn the original features robust to meaning-invariant formulations.

We propose to counteract the single-truth assumption of *MLE* by constructing targets that respect mutual tokens' similarity through the alignment of

[*] Corresponding author: stefanik.m@mail.muni.cz

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8837–8853
July 9-14, 2023 ©2023 Association for Computational Linguistics

output tokens to the reference (Figure 1). Consequentially, the expected target distribution is spread over the tokens that can be accurately aligned to the reference, based on the representations provided by a domain-agnostic embedding model. We find that using such objectives in domain adaptation can eliminate a major portion of model performance loss on out-of-domain (OOD), caused by the adaptation techniques while reaching comparable or higher qualitative gains on the adapted domain.

Our main contributions are the following. (i) We present a framework for training generative language models with an alternative training signal based on token similarity provided by an arbitrary embedding model. A similar methodology can be applied for more robust training and adaptation of any language model. (ii) We introduce efficient and accurate training objectives that alleviate catastrophic forgetting of low-resource domain adaptation in NMT without losing adaptation quality. (iii) We further investigate the covariates that impact the robustness of generative LLM. Among others, we find that a more robust model can be obtained merely by exposing a generative model to its own predictions during the training.

This paper is structured as follows. Section 2 surveys and compares our work to the existing work in training and adapting robust generative LLMs. Section 3 introduces two main objectives that we experiment with: *TokenAlign* and *SeqAlign*. Section 4 describes our experimental methodology and ablation analyses and Section 5 summarizes our findings, highlighting the broader implications.

2 Background

Language generation is the modus operandi for a wide set of problems requiring an open-ended sequence of tokens as the answer. Machine translation is the representative of this group that we focus on, but other tasks such as summarization (Lewis et al., 2020), vision captioning (Wang et al., 2022), question answering (Raffel et al., 2020) or in-context learning (Sanh et al., 2021) are also applications of the described framework.

In the commonly-used auto-regressive generation, for each pair of input and reference texts, i.e. *sequences* of tokens X_j and Y_j, a *language model* $\Theta(Y_{j,i}|X_j, Y_{j,1..i-1})$ is trained to generate output sequence by maximizing the probability of generating the i-th token $y_{ji} = \arg\max(\Theta(X_j, Y_{j,1..i-1}))$ *matching* the reference token Y_{ji} while minimizing the probability of generating other tokens of the vocabulary, conditionally to the input text X_j and *previous* reference tokens $Y_{j,1..i-1}$:

$$\max p(y_{ji} = Y_{ji}|X_j, Y_{j,1..i-1}, \Theta) \tag{1}$$

This goal is implemented in the commonly-used approach that we refer to as the Maximum Likelihood Estimation objective (*MLE*), which minimizes a cross-entropy (CE) of the predicted distribution of $\Theta(X_j, Y_{j,1..i-1})$ to the *expected* distribution, which is a one-hot encoding E_{ji} of the *true* reference token Y_{ji} over the model vocabulary:

$$\mathcal{L}_{MLE}(\Theta) = \min\left(-\log\frac{\exp(\Theta(X_j, Y_{j,1..i-1}))}{\exp(E_{ji})}\right) \tag{2}$$

MLE is commonly used both for training (Bahdanau et al., 2016; Vaswani et al., 2017) and adaptation (Servan et al., 2016; Saunders, 2021) of generative LLMs.

While the adaptation brings benefits in modelling domain-specific terminology (Sato et al., 2020), or in avoiding inadequate generation artefacts such as repetitions or hallucinations (Etchegoyhen et al., 2018), it comes at a price of generalization to other domains; the adapted models improve on the adapted domain but gradually perform worse on other domains.

Previous work in domain adaptation presents methods addressing the mitigation of catastrophic forgetting. Chu et al. (2017) enhance model robustness by mixing the pre-training and adaptation samples in continuous training, assuming that the full pre-training dataset is available, which is commonly not the case. Thompson et al. (2019) regularize the training objective with Fischer Information Matrix. Dakwale and Monz (2017) also use the regularization in training, instead based on the predictions of the original model. Similarly, Freitag and Al-Onaizan (2016) use the ensemble of the original and trained model in prediction. In this line, we experiment with the ensemble approach using Transformers but find it underperforms other methods in low-resource adaptation.

Han et al. (2021) find that using parameter-efficient fine-tuning methods, such as using *Adapters* (Houlsby et al., 2019) can increase the robustness of the adapted model. Previous work also applied Adapters in the fine-tuning of generative LLMs (Cooper Stickland et al., 2021; Lai et al., 2022), but do not evaluate the distributional robustness of the final models; Therefore, we include

Adapters as another baseline, but find it also struggling in lower-resource cases, due to the random initialisation of its bottleneck representations. We find that this problem can be avoided using *LoRA* (Hu et al., 2022), which instantiates tuned parameters as *additions* to attention matrices initialised close to zero values, therefore commencing the adaptation with the originally-performing model.

Another problem that commonly arises in the training of generative LMs is referred to as *exposure bias*: while in the *teacher-forced* training, the model's i-th prediction $\Theta(X_j)_i$ is conditioned by the correctly-generated previous tokens from the reference $Y_{j,1..i-1}$, in practice, the model conditions its predictions on its *own* outputs $\Theta(X_j)_{1..i-1}$. We speculate that this discrepancy might be magnified under a domain shift where the model could not have learned to follow the reference closely.

Exposure bias was addressed by *sequential objectives*, such as Minimum Risk Training (MRT) (Ranzato et al., 2016) that optimize the model by the evaluation of complete output sequence (Yang et al., 2018; Wang and Sennrich, 2020; Mi et al., 2020; Unanue et al., 2021). Apart from the specifics of Reinforcement learning, such as fragility to the optimization settings (Pineau et al., 2021), these methods are also more resource-demanding as they require a sequence of predictions for a single update, limiting their applicability in low-resource adaptation. Previous work of Choshen et al. (2020) also shows that gains of sequential methods in adaptation might be similar to a random training signal. Inspired by this finding, we also assess the gains and OOD robustness of our methods against a random-feedback sequential baseline (§4.3).

Closer to us, previous work uses alternative training signal based on comparing model hypotheses to the reference. Xu et al. (2019) build soft alignment between fully-generated hypotheses based on hidden states of bidirectional LSTM encoder-decoder and weigh the predicted probability distribution by such alignment in the training objective. Similarly, Lu et al. (2020) complement *MLE* and sentence-level objective with the objective minimizing a dot-product of the best-matching hidden representations of tokens of a hypothesis and a reference. Chen et al. (2019) and later Zhang et al. (2020a) introduce the matching scheme that uses the Optimal transport cost (Kusner et al., 2015) of the embeddings of reference to the hypothesis as their objective loss.

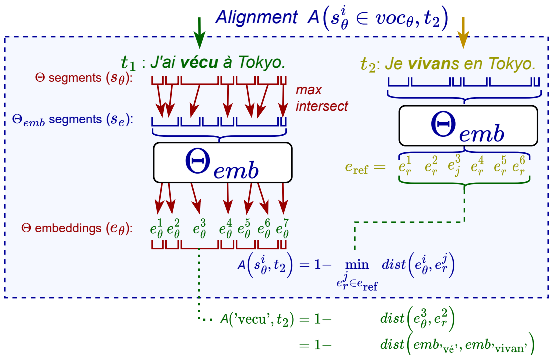
Alignment $A\left(s_\theta^i \in voc_\theta, t_2\right)$
t_1 : J'ai **vécu** à Tokyo.
t_2: Je **vivans** en Tokyo.
Θ segments (s_θ):
max intersect
Θ_{emb} segments (s_e):
Θ_{emb}
Θ_{emb}
$e_{\text{ref}} = e_r^1 \; e_r^2 \; e_j^3 \; e_r^4 \; e_r^5 \; e_r^6$
Θ embeddings (e_θ):
$e_\theta^1 \; e_\theta^2 \; e_\theta^3 \; e_\theta^4 \; e_\theta^5 \; e_\theta^6 \; e_\theta^7$
$A\left(s_\theta^i, t_2\right) = 1 - \min_{e_r^j \in e_{\text{ref}}} dist\left(e_\theta^i, e_r^j\right)$
$A(\text{'vecu'}, t_2) = 1 - dist\left(e_\theta^3, e_r^2\right)$
$= 1 - dist\left(emb_{\text{'vé'}}, emb_{\text{'vivan'}}\right)$

Figure 2: **Token alignment mechanism** represents subwords s_Θ of the trained model Θ with embeddings of a robust, static model Θ_{emb}. Using these representations, we define *Alignment* of any Θ's subword s_Θ^i to another text t_2 through a minimal distance of their embeddings given by the robust embedding model Θ_{emb}.

Referenced work reaches improvements in conventional high-resource training scenarios, whereas our goal is to propose a method for training robust generative models for challenging low-resource settings. This also motivates a primary difference in the design of our methods; That is, to use domain-agnostic representations for constructing training targets, instead of the model's own representations, which are subject of over-specialization in adaptation.

3 Soft Alignment Objectives

This section describes the details of alignment-based objectives[1] that we introduce in this work.

3.1 Token Alignment

Our goal is to circumvent the single-truth assumption of *MLE* with targets respecting the mutual tokens' similarity. Since the representations of the trained models are affected by catastrophic forgetting, we propose to use an alternative, domain-agnostic representation model (Θ_{emb}) to provide the token representations, i.e. embeddings.

However, as the vocabularies of the fine-tuned model Θ and Θ_{emb} are not aligned, to train with representations of a different Θ_{emb}, we need to *match* each subword (token) of the trained model (s_Θ^i) with a subword of the embedding model (s_e^j) having a representation $e^j \in \Theta_{emb}(t)$; (i) We tokenize input text t_1 using both Θ's and Θ_{emb}'s tokenizers, obtaining subwords s_Θ and s_e respectively. (ii) Then,

[1] The implementation of all new objectives is available at: https://github.com/MIR-MU/softalign_objectives

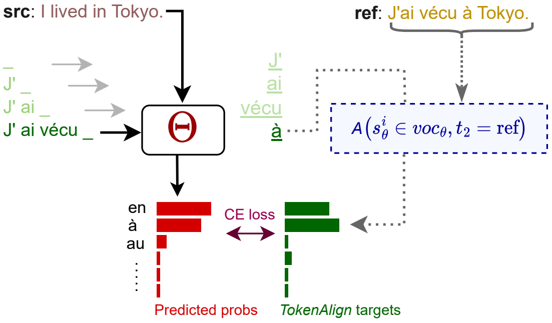

Figure 3: ***TokenAlign* objective** replaces one-hot targets of *MLE* with token Alignments *A* based on a similarity between the embeddings of the candidate and reference tokens (§3.1), encouraging the trained model Θ to respect the ambiguity of prediction, instead of eliminating it.

we compute the character-level positional spans of both subwords lists s_Θ and s_e. Finally, we (iii) *match* each model subword $s^i_\Theta \in s_\Theta$ with embedding subword $s^j_e \in \Theta_{emb}$ such that s^j_e has the largest positional overlap with s^i_Θ. As a result, each Θ's subword s^i_Θ gets assigned an embedding $e^i_\Theta = e^k_r$ of Θ_{emb}, as visualized in Figure 2.

Having Θ's subwords' representations from a robust embedding model Θ_{emb}, we finally define an *Alignment* $\mathcal{A}$ of any subword $s^i_\Theta \in \Theta$ to another text t_2 as:

$$\mathcal{A}(s^i_\Theta, t_2) = 1 - \min_{e^j_r \in \Theta_{emb}(t_2)} dist(e^i_\Theta, e^j_r) \qquad (3)$$

where *dist* is any distance measure defined for the chosen embedding system. In our experiments, we use standard Euclidean distance as the measure. We provide a more detailed description and complexity analysis of the Alignment algorithm $\mathcal{A}$ in Appendix C.

3.2 *TokenAlign* Objective

TokenAlign is designed as a minimal adjustment to *MLE* (Eq. (2)) using the alignment *A* as the target of each candidate token of Θ's vocabulary. Instead of penalisation, this encourages the model to upweight predictions that do not match the reference token, but still can be accurately matched to the reference text (Figure 3):

$$\mathcal{L}_{TAlign}(\Theta) = \min\left(-\log\frac{\exp(\Theta(X_j, Y_{j,1..i-1}))}{\exp(\mathcal{A}(voc_\Theta, Y_j))}\right) \qquad (4)$$

where voc_Θ is the vocabulary of Θ, and $\mathcal{A}(s^{1..|\Theta|}_\Theta, Y_j)$ are the *alignments* for each token of the vocabulary (s^i_Θ) to the reference text Y_j.

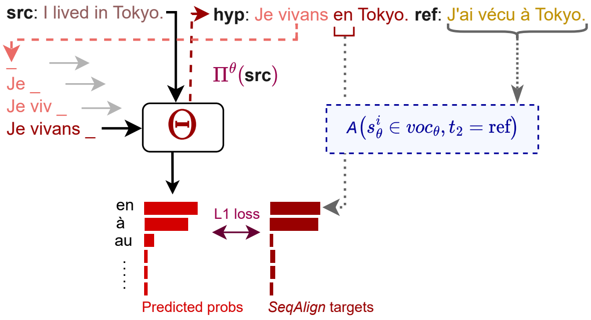

Figure 4: ***SeqAlign* objective** further replaces the reference prefixes in the training with Θ's own-generated hypotheses. This additionally adapts the model to condition the predictions based on its own outputs, instead of the reference.

Note that none of $\mathcal{A}$'s components is updated in training.

Relying on the same training approach as with the conventional *MLE* objective, *TokenAlign* presents an alternative of the *MLE* of similar data and compute efficiency (compared in Appendix B). However, *TokenAlign* still does not address the exposure bias as the model Θ is only updated conditionally to the previous *reference* tokens $Y_{1..i-1}$ as the prefixes, rather than its own outputs.

3.3 *SeqAlign* Objective

Alignment $\mathcal{A}$ allows us to assess Θ's prediction quality on a token level, but without dependence on the exact ordering of reference tokens. Thus, we no longer need to keep the prefixes synchronized with reference and can construct targets for an arbitrary prefix. Hence, instead of taking prediction prefixes from reference Y_j, *SeqAlign* constructs the prefixes from the hypothesis generated by the trained model Θ itself (Fig. 4).

We create the self-generated *hypothesis* by using Θ's outputs as a probability distribution and construct a generation strategy Π^Θ that *samples* next token(s) from this distribution. A desideratum of such generation strategy (compared to a greedy search) is that the prefixes of generated hypotheses are diverse but still realistically likely to occur during Θ's generation.

Additionally, instead of generating a *single* hypothesis for each input, we can obtain a *set of hypotheses* $\hat{Y}_j \sim \Pi^\Theta(X_j)$ that can be used by *SeqAlign* to condition the updates of Θ. The sampling generation strategy is inspired by the previous work, using sampling to construct full hypotheses (Neubig, 2016; Shen et al., 2017; Edunov et al., 2018).

Identically to *TokenAlign*, *SeqAlign* associates *all* the vocabulary tokens voc_Θ with their alignment quality $\mathcal{A}(s_\Theta^{1..|\Theta|}, Y_j)$ and uses the alignment as target distribution. However, motivated by the empirical results, instead of the Cross-Entropy, we minimise *absolute distance* ($L1$) as *SeqAlign*'s training objective:

$$\mathcal{L}_{SAlign}(\Theta) = \min\Big(\Theta(X_j, \hat{Y}_{j,1..i-1}) - \mathcal{A}(voc_\Theta, Y_j)\Big) \\ \text{where } \hat{Y}_j \sim \Pi^\Theta(X_j) \qquad (5)$$

Note that we further analyse the impact of the loss formulation in the ablation in Section 4.3.

3.4 Embeddings Contextualization

Computing alignment $\mathcal{A}$ using *context-insensitive* embedding model Θ_{emb}, such as GloVe (Pennington et al., 2014) or FastText (Bojanowski et al., 2017) requires no further adjustments. However, using more expressive *context-sensitive* embedding models, such as BERT (Devlin et al., 2018) for computing $\mathcal{A}$ as a target for any possible output token faces the following issues.

(i) Inference of representations *on the fly* within the training process is expensive. Consider an example of obtaining contextual representations for each possible next token in generating a 10-token hypothesis, requiring $10^{|\Theta|}$ inferences of Θ_{emb}, where $|\Theta|$ is a size of the vocabulary of Θ, commonly in ranges of 30,000–50,000 tokens.

(ii) A full context required to infer bidirectional contextual embeddings remains incomplete throughout the generation. The embeddings could be inferred within a synthetic context or using a unidirectional embedding model instead, but we find that both these approaches significantly alter tokens' pairwise distances.

In the *SeqAlign* objective, we address these issues by embedding only the top-n *highest-scored tokens* of Θ in each prediction step (denoted $\Theta^{\uparrow n}$). By fixing $n = 3$, we need to infer the contextual embeddings of only $\sum_{k=1}^{K} 3|\Pi_k(X_j)|$ of the highest-scored tokens for each sampled hypothesis $\Pi_k(X_j)$. In our experiments, we also keep the *number of sampled hypotheses* K fixed to $K = 10$ and we do *not* adjust Θ by the scores of the tokens other than the top ones. As the context, we use the complete hypothesis from which the token $s^i_\Theta \in \Theta^{\uparrow n}$ is sampled. Therefore, the targets $\mathcal{A}$ for our distance-based objectives are adjusted to:

$$\mathcal{A}'(s^i_\Theta, t_2) = \begin{cases} \mathcal{A}(s^i_\Theta, t_2) & \text{if } s^i_\Theta \in \Theta^{\uparrow n} \\ 0 & \text{otherwise} \end{cases} \qquad (6)$$

In *TokenAlign*, which requires embeddings of *all* tokens of the vocabulary, we address the computational overhead in a *decontextualization* process. We obtain the decontextualized embedding e^i for each subword s^i_e as an *average* of the contextualized embeddings corresponding to *all* the occurrences of s^i_e in the texts of the training domain X:

$$e^i_{dec} = \frac{1}{\#s^i_e} \sum_{X_j \in X;\, s^i_e \in X_j} \Theta_{emb}(X_j)^i \qquad (7)$$

where $\#s^i_e$ is the number of occurrences of a subword s^i_e in X.

While such a process also causes qualitative decay of the contextual representations, it has been shown that decontextualized representations still outperform context-agnostic (FastText) embeddings in machine translation evaluation (Štefánik et al., 2021). Despite that, we quantify the impact of decontextualization as one of our ablations (§4.3).

Throughout all our experiments, we use the embeddings of multilingual BERT model (Devlin et al., 2019) as Θ_{emb}, extracted from the 9-th hidden layer, motivated by the previous work of Zhang et al. (2020b) showing this model to best correlate with a human evaluation of generative LLMs.

4 Experiments

We evaluate the impact of the proposed training objectives in the domain adaptation experiments in machine translation, where the distributional robustness in adaptation may bring well-measurable benefits. We compare our results with the adaptation using the commonly-used *MLE* objective (§2), and selected parameter-efficient methods shown to mitigate forgetting. We use the novel objectives as the weighted *complements* of the *MLE* objective (Eq. (2)), thus optimising both the objectives in parallel:

$$\mathcal{L}_{*Align}(\Theta) = \mathcal{L}_{MLE}(\Theta) + \alpha \cdot \mathcal{L}_{NewObj}(\Theta) \qquad (8)$$

4.1 Datasets

We choose the data configurations of our experiments to allow the reader to extrapolate trends and conclusions invariant to the following covariates.

Domains. To assess the distributional robustness of the models, we train and evaluate among *all* pairs of these OPUS domains (Tiedemann, 2012): *Wikimedia*, *OpenSubtitles*, *Bible*, *TEDTalks*, *DGT/Law* and *EMEA/Medical*. We choose the set of domains that reflects both minor (*Wikimedia* → *OpenSubtitles*) and major (*EMEA/Medical* → *Bible*) domain shifts between the training and evaluation. Our selection reflects on real-world settings where practitioners commonly adapt a general-purpose model to a *specialized* domain such as *law* or *medicine*, but need to keep an operational level of quality on any input.

Data size. We focus on the applications where the size of parallel corpora available for adaptation range from *very low-resource* (50,000 aligned sentences, *Bible*) to *medium-resource* (5,100,000 sentences, *DGT/Law*).

Language pairs. Our evaluated language pairs are: *Estonian* → *English*, *German* → *English English* → *Czech*, *English* → *Ukrainian*, *English* → *German* and *English* → *Chinese*. We pick the *English-centric* pairs in order to maximize the number of out-of-domain evaluation sources for the adapted language pair. Our settings cover target languages of Latin, Cyrillic, and Chinese alphabets.

4.2 Experimental Setup

Data configuration As the OPUS sources do not contain standard splits, we split the data into train-validation-test. We first de-duplicate the samples and draw 500 validation and 1,000 test samples from each domain.

Training We perform the adaptations from the bilingual Transformer-base models of Vaswani et al. (2017) using the checkpoints of Tiedemann and Thottingal (2020) pre-trained for a translation of the corresponding language pair on a mixture of OPUS sources.

We perform a hyperparameter search over the parameters of *learning rate*, *objectives weights* α, and objective-specific *batch size*. We detail the values and ranges of this search in Appendix A.

After fixing the objectives' parameters, we set up the experiments to closely resemble the traditional training process; We run each experiment until early-stopping by in-domain validation BLEU, with the patience of 20 evaluations, i.e., 10,000 updates and evaluate the model with the best validation score for testing. If the model does not improve over the first 10,000 updates, we evaluate the resulting model after the 10,000 updates.

We implement our experiments using Adaptor library (Štefánik et al., 2022), allowing the release of all our experiments in a transparent and self-containing form.[2]

Evaluation To discourage the effect of the random variance in the performance of the trained model, we report all test scores as the *average* of the performance in the interval of 5 *preceding* and 5 *succeeding* checkpoints, resulting in a single, average test evaluation for each domain.

We collect evaluations of BLEU in the default settings of SacreBLEU (Post, 2018), obtaining a single (average) evaluation of in-domain (ID) BLEU and a set of corresponding evaluations for *all* listed domains *other* than the in-domain (OOD). Given the availability of the sources, this results in four OOD evaluations for all pairs except (en→ukr) and (en→zh) with the datasets for two OOD evaluations.

To enable mutual comparability, we finally normalize both ID and OOD results by the performance of the initial checkpoint and report the change of performance in percentage. We report a single scalar value, or an interval in a form <*mean*±*range covering all results*>.

Baselines In addition to *MLE*, we compare the proposed methods to four existing methods reported to enhance LLMs'robustness. (i) Label smoothing (Szegedy et al., 2016) with $\alpha = 0.1$ used widely also for training MT models distributes a constant portion of expected probability among all possible predictions. (ii) *Adapters* (Houlsby et al., 2019) freezes pre-trained model parameters and fine-tunes a small set of newly-initialized bottleneck parameters. Instead, (iii) *LoRA* avoids Adapters' issue of breaking the model in the initial training phase by initializing the new parameters that are trained as an *addition* to the model's original, frozen parameters. (iv) We also implement and evaluate the Ensemble approach of (Freitag and Al-Onaizan, 2016), but find this approach unable to bring adaptation gains in either of our relatively low-resource adaptation cases. We detail the settings of our baselines in Appendix A.

[2] All our experiments can be reproduced by running a single line of code; refer to the Section `experiments` in `https://github.com/MIR-MU/softalign_objectives`

	Δ BLEU	Bible (de→en) 62,000 pairs	TEDTalks (en→zh) 155,000 pairs	Opensubs (en→ukr) 877,000 pairs	Wiki (en→cze) 1,003,000 pairs	Medical/EMEA (est→en) 1,021,000 pairs	Law/DGT (en→de) 5,105,000 pairs	**Average** (BLEU)	**Average** (BERTScr)
Orig. BLEU		21.89	29.01	26.12	34.04	54.85	33.56		
MLE	ID	− 8%	+ 7%	+ 4%	+ 9%	+38%	− 1%	+ **8.31%**	+ 9.19‰
(Bahdanau et al., 2015)	OOD	−53% ± 36%	−23% ± 23%	−15% ± 9%	−15% ± 5%	−35% ± 10%	−19% ± 11%	−26.87%	−37.34‰
MLE + Smoothing	ID	− 6%	+**30**%	− 6%	+ 9%	+17%	+ 0%	+ 7.43%	+ 3.77‰
(Szegedy et al., 2016)	OOD	−85% ± 31%	−39% ± 26%	−25% ± 9%	−13% ± 22%	−49% ± 16%	−27% ± 26%	−41.86%	−54.13‰
Adapters	ID	− **5**%	−27%	−14%	+ 1%	+13%	− 0%	− 5.41%	−15.23‰
(Houlsby et al., 2019)	OOD	−91% ± 20%	−80% ± 2%	−53% ± 9%	−46% ± 25%	−77% ± 19%	−45% ± 43%	−65.39%	−94.97‰
LoRA	ID	− 8%	+ 2%	+ 2%	+**14%**	+ 8%	+ 6%	+ 3.98%	+ 5.85‰
(Hu et al., 2022)	OOD	− 7% ± 7%	−21% ± 20%	− **1%** ± 1%	−7% ± 5%	− 4% ± 11%	+2% ± 14%	− 5.15%	− 3.78‰
TokenAlign	ID	−21%	+ 2%	+ **8%**	+12%	+**45%**	+ 1%	+ 8.17%	+ 6.83‰
(ours)	OOD	− 2% ± 1%	−**10%** ±12%	− **1%** ±1%	− **6%** ±6%	− 6% ± 7%	+ **6%** ±20%	− 3.25%	− **0.98**‰
SeqAlign	ID	−23%	+ 7%	− 8%	+ 8%	+31%	+ **7%**	+ 3.67%	+**15.46‰**
(ours)	OOD	− **1**% ±1%	−20% ± 22%	− 2% ± 3%	−12% ± 5%	− **1%** ± 2%	+ 3% ± 13%	− **1.44%**	− 1.53‰

Table 1: **Evaluation of adaptation quality and robustness**: A change of BLEU score relative to the original model, when adapting pre-trained Transformer on the titled domain, as measured on a held-out set of the training domain (in-domain, ID) and other listed domains available for the same language pair (out-of-domain, OOD). **Bold** denotes the best Average ID and OOD results, and per-domain results, where adaptation brings ID improvements. The results are evaluated using SacreBLEU (Post, 2018) and BERTScore (Zhang et al., 2020b).

4.3 Ablation Experiments

In a set of additional experiments, we estimate the impact of the crucial components of the soft alignment objectives on adaptation accuracy and robustness. While these assessments provide an ablation study verifying our design decisions, they also assess the impact of different design aspects on the robustness of generative language models.

Impact of teacher forcing Teacher forcing, i.e. replacing the model's own outputs with the preceding tokens of the reference (§2) circumvents the problem of aligning the model's generated output to the reference. We suspect that the discrepancy between the training and generation can be magnified under the distribution shift and hence, can be one of the causes of the catastrophic forgetting.

To assess the impact of teacher forcing on robustness, we design an objective that uses the model's generated outputs as prefixes, but contrary to *SeqAlign*, it provides *non-informative* training signal. We implement the experiment by replacing the *SeqAlign*'s alignment $\mathcal{A}$ (in Eq. (5)) with *randomly-generated* alignment A_{rand} as target:

$$\mathcal{L}_{\mathit{SRand}}(\Theta) = \min\Big[\Theta(X_j, \hat{Y}_{j,1..i-1}) - \mathcal{A}_{\mathit{rand}}\Big] \tag{9}$$

Additionally to the assessment of the impact of teacher forcing removal, this experiment also quantifies the importance of the embedding-based training signal of *SeqAlign*.

Impact of decontextualization While the *TokenAlign* utilize the *decontextualized* grounding embeddings (§4.3), the decontextualization likely affects the quality of target distribution. However, as we discussed in Section 3.4, it is not computationally feasible to simply infer the contextualized embeddings for each candidate token of the generated hypotheses. Hence, to compare the contextualized and decontextualized versions of the same system, we adjust the *SeqAlign*'s alignment $\mathcal{A}'$ (Eq. (6)) to utilize the *decontextualized* embeddings (Eq. (7)) instead of the contextualized ones:

$$\begin{aligned} \mathcal{L}_{\mathit{SeqAlign\text{-}dec}}(\Theta) &= \mathcal{L}_{\mathit{SeqAlign}}(\Theta, \mathcal{A}'_{dec}) \\ \mathcal{A}'_{dec}(s^i_\Theta, t_2) &= \min_{e^j_{dec} \in \Theta_{dec}(t_2)} \mathrm{D}(e^i_{dec}, e^j_{dec}) \end{aligned} \tag{10}$$

All other parameters of *SeqAlign* remain unchanged, as described in Section 4.2.

Impact of the loss formulation Following the previous work on sequential objectives (§2), *SeqAlign* utilize the distance-based loss, but since we use token-level alignment, similarly to standard *MLE*, we could also formulate the objective using Cross Entropy (*CE*).

This ablation evaluates the impact of the loss formulation by introducing an analogous objective to *SeqAlign-dec* (Eq. (10)), but utilizing the *CE* loss instead of $L1$ distance:

$$\mathcal{L}_{\mathit{SCE}}(\Theta) = \min\left(-\log\frac{\exp(\Theta(X_j, \Pi^{\Theta}_{1..i-1}(X_j)))}{\exp(\mathcal{A}_{dec}(voc_\Theta, Y_j))}\right) \tag{11}$$

ΔBLEU:	ID	OOD
0. *MLE*	$+\ 8\% \pm 31\%$	$-27\% \pm 29\%$
1. *TokenAlign*	$+\ 8\% \pm 30\%$	$-\ 3\% \pm\ \ 9\%$
2. *SeqAlign*	$+\ 3\% \pm 27\%$	$-\ 1\% \pm\ \ 8\%$
3. *SRand*	$+\ 3\% \pm 31\%$	$-\ 6\% \pm\ \ 5\%$
4. *SeqAlign-dec*	$+\ 5\% \pm 31\%$	$-\ 6\% \pm 27\%$
5. *SeqAlign-CE*	$+\ 4\% \pm 32\%$	$-17\% \pm 44\%$

Table 2: **Results of Ablation experiments**: Average change of BLEU scores relative to the original model, when adapting the Transformer-base model with a given objective. The intervals cover the averages of 6 in-domain and 20 out-of-domain evaluations (§4.2).

We sample the prefixes from the model's own hypotheses using the same generation strategy Π^{Θ} as in other sequential objectives. We use the decontextualized objective as the reference to avoid the overhead of inference of contextual embeddings for the full vocabulary.

5 Results

Table 1 compares the results of adaptation using a selection of baseline methods and our two main objectives: *TokenAlign* and *SeqAlign*, as trained on a selected domain and evaluated on a held-out set of the same domain (ID) and other domains (OOD). The domains are ordered by ascending size of the training data. Table 2 additionally includes the objectives from our Ablation experiments. More detailed, per-domain ablations results can be found in Table 4 in Appendix D.

Alignment-based objectives improve robustness; Both *TokenAlign* and *SeqAlign* consistently improve the model robustness (OOD) over the *MLE* in *all* the evaluated cases and on average deliver more robust models compared to all other methods. In addition, comparing *TokenAlign* to instances of *MLE*, we also see the advances in the adaptation quality (ID), in four out of five cases where *MLE* is able to deliver any ID improvements. In OOD evaluations, *SeqAlign* is slightly more robust than *TokenAlign*, presenting a more robust, yet also technically more complex alternative.

While the average results confirm our main hypothesis that circumventing *MLE*'s assumption of a single-truth prediction can improve the model's distributional robustness, we see a large variance in the performance of our methods similar to *MLE*. The in-domain results of *SeqAlign* also dispute our assumption that self-generation of prefixes could compensate for the scarcity of natural in-domain data; *SeqAlign*'s ID performance on the two smallest domains is inferior to both *MLE* instances, while it is very efficient in the higher-resource *Law/DGT*.

Avoiding teacher-forcing improves robustness; A comparison of the results of *SRand* and *MLE* in Table 2 shows that the mere exposition of the model to its own hypotheses reduces the forgetting of *MLE* by 77% in average ($-27\% \rightarrow -6\%$). However, constructing the non-informative targets for self-generated inputs also causes a decay in adaptation quality ($+8\% \rightarrow +3\%$).

Alignment-based targets complement avoiding teacher-forcing; Robustness improvements of *SeqAlign* over *SRand* (Table 2) might be attributed to the semantically-grounded Alignment targets (§3.1). While the aggregate in-domain results of *SeqAlign* and *SRand* in Table 2 are very close, the per-domain results (Table 4 in Appendix D) reveal that their results vary over domains and the suggested ID tie of *SRand* to *SeqAlign* is largely attributed to *SRand*'s better results on *Bible*, where both objectives fail to improve ID nevertheless.

Decontextualization does not carry a large qualitative drop; Both objectives grounding their targets in decontextualized embeddings (*TokenAlign* and *SeqAlign-dec*) show relatively good average results on both ID and OOD (Table 2), but *TokenAlign* is the only method reaching adaptation accuracy comparable to *MLE* in average. A comparison of *SeqAlign* to its decontextualized instance (*SeqAlign-dec*) specifically evaluates the impact of decontextualization, in the settings of absolute distance loss and no teacher forcing. We see that while the decontextualization leads to a larger loss in the robustness ($-1\% \rightarrow -6\%$), *SeqAlign-dec* slightly outperforms *SeqAlign* on the in-domain ($+3\% \rightarrow +5\%$). Per-domain results (Table 4 in Appendix D) show that this is attributed mainly to the superior adaptation performance of *SeqAlign-dec* in the low-resource *Opensubs (en→ukr)* case, suggesting that the embeddings' averaging within decontextualization (§4.3) works well also with small amounts of texts.

Loss formulation impacts model robustness; A comparison of *SeqAlign-dec* and *SeqAlign-CE* in Table 2 assesses the impact of changing objectives' loss formulation from $L1$ to Cross Entropy (CE). We see that changing a distance-based loss to CE causes a significant drop in OOD robustness ($-6\% \rightarrow -17\%$), comparable to the drop of the traditional *MLE*, also built upon CE loss (-21%). However, the superior OOD performance of CE-based *TokenAlign* contradicts that CE loss itself could be a pivotal cause of catastrophic forgetting.

6 Conclusion

Our work sets out to explore the alternatives between the efficient yet naïve *MLE* objective and expressive but resource-demanding sequential objectives, by building the training signal from the semantic token representations. We build an alignment mechanism applicable with an arbitrary representation model and propose objectives that utilize a domain-agnostic embedding model as its target. We find that using semantically-grounded targets in adaptation persists robustness of the model much better than other methods, without compromises in in-domain performance.

We additionally explore the impact of selected design choices on the robustness of generative LLMs in the ablation experiments. Among others, we find that a major part of the model's robustness can be persisted merely by including the model's own outputs among the inputs, attributing a part of adaptation forgetting to exposure bias. Future work might also build upon the qualitative assessment of the impact of decontextualization, resolving the computational overhead of applying the contextualized embeddings in dynamic contexts.

We look forward to future work that will explore the potential of applying semantically-grounded objectives in a more robust and data-efficient training of LLMs for many other applications, including the pre-training stages.

While our experiments do not evaluate such settings, we note that our methods complement the model-centric ones, including recent parameter-efficient training strategies (Valipour et al., 2023; Dettmers et al., 2023). Given the encouraging results of *LoRA* (Table 1), we believe that future work combining parameter-efficient methods with semantically-grounded objectives like ours can mitigate forgetting of domain and task adaptation even further.

Limitations

We experiment with a range of adaptation domains that we draw systematically to capture the covariates enumerated in Section 4.1. However, future work should acknowledge that these are not all the covariates responsible for the success of adaptation and the robustness of the final model. Following is the non-exhaustive list of possible covariates that we do not control in this work. (i) the adapted model size, (ii) the size of pre-training data, (iii) pre-training configuration parameters, but also (iv) the broad variance of adapted language pair(s); (v) the variance of mutual similarity of languages within the pair, and hence (vi) the difficulty of training the translation model.

The evaluation of our experiments did not consider the effect of *randomness* of the training process. Despite the fact that our experiments were run with a fixed random seed and initial value, making our results deterministically reproducible, the variance of the results among the experiments of different random seeds was not investigated due to the related infrastructural costs. However, all our results are aggregated over a larger set of checkpoints and/or domains, ranging from 10 (IDs in Table 1) to 720 (OODs in Table 2), as described in Section 4.2.

The alignment scheme proposed in Section 3.1 might have blind spots; for instance, in the cases utilizing decontextualized embeddings, where both the hypothesis and reference contain multiple occurrences of the same word, the alignment scheme will make the prediction of the same target token equally *good*, regardless of the position. In future work, this imperfection could be addressed by using the Optimal transport algorithm (Kusner et al., 2015) within the Alignment, similarly to Zhang et al. (2020a).

Acknowledgements

We thank the anonymous reviewers of our work for providing us with qualified feedback that significantly shaped the resulting form of this paper.

We acknowledge the Centre for Biomedical Image Analysis at Masaryk University supported by MEYS CR (LM2023050 and CZ.02.1.01/0.0/0.0/18_046/0016045 Czech-BioImaging) for their support in obtaining the results presented in this paper.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representations, ICLR 2015*, San Diego, USA.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural Machine Translation by Jointly Learning to Align and Translate. ArXiv:1409.0473v7.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the ACL*, 5:135–146.

Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019. Improving Sequence-to-Sequence Learning via Optimal Transport. *ArXiv*, abs/1901.06283.

Leshem Choshen, Lior Fox, Zohar Aizenbud, and Omri Abend. 2020. On the weaknesses of reinforcement learning for neural machine translation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An Empirical Comparison of Domain Adaptation Methods for Neural Machine Translation. In *Proceedings of the 55th Annual Meeting of the ACL (Volume 2: Short Papers)*, pages 385–391, Vancouver, Canada. ACL.

Asa Cooper Stickland, Alexandre Berard, and Vassilina Nikoulina. 2021. Multilingual domain adaptation for NMT: Decoupling language and domain information with adapters. In *Proceedings of the Sixth Conference on Machine Translation*, pages 578–598, Online. ACL.

Praveen Dakwale and Christof Monz. 2017. Fine-Tuning for Neural Machine Translation with Limited Degradation across In- and Out-of-Domain Data. In *Proceedings of the XVI Machine Translation Summit (Vol. 1: Research Track)*, pages 156–169, Nagoya, Japan.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805v2.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of the 2019 Conference of the NAACL: Human Language Technologies*, pages 4171–4186, Minneapolis, USA. ACL.

Sergey Edunov, Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Classical Structured Prediction Losses for Sequence to Sequence Learning. In *Proceedings of the 2018 Conference of the NAACL: Human Language Technologies, Volume 1 (Long Papers)*, pages 355–364, New Orleans, Louisiana. ACL.

Thierry Etchegoyhen, Anna Fernández Torné, Andoni Azpeitia, Eva Martínez Garcia, and Anna Matamala. 2018. Evaluating Domain Adaptation for Machine Translation Across Scenarios. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. ELRA.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast Domain Adaptation for Neural Machine Translation. *ArXiv*.

Ian J. Goodfellow, Mehdi Mirza, Xia Da, Aaron C. Courville, and Yoshua Bengio. 2014. An Empirical Investigation of Catastrophic Forgeting in Gradient-Based Neural Networks. *CoRR*, abs/1312.6211.

Wenjuan Han, Bo Pang, and Ying Nian Wu. 2021. Robust Transfer Learning with Pretrained Language Models through Adapters. In *Proceedings of the 59th Annual Meeting of the ACL and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 854–861, Online. ACL.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From Word Embeddings To Document Distances. In *Proc. of International Conference on Machine Learning*, volume 37, pages 957–966, Lille, France. PMLR.

Wen Lai, Alexandra Chronopoulou, and Alexander Fraser. 2022. m^4Adapter: Multilingual Multi-Domain Adaptation for Machine Translation with a Meta-Adapter. In *Findings of the ACL: EMNLP 2022*, Abu Dhabi, United Arab Emirates. ACL.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proc. of the 58th Annual Meeting of the ACL*, pages 7871–7880.

Wenjie Lu, Leiying Zhou, Gongshen Liu, and Quanhai Zhang. 2020. A mixed learning objective for neural machine translation. In *Proceedings of the 19th Chinese National Conference on Computational Linguistics*, pages 974–983, Haikou, China. Chinese Information Processing Society of China.

Chenggang Mi, Lei Xie, and Yanning Zhang. 2020. Improving Adversarial Neural Machine Translation for Morphologically Rich Language. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(4):417–426.

Graham Neubig. 2016. Lexicons and Minimum Risk Training for Neural Machine Translation: NAIST-CMU at WAT 2016. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 119–125, Osaka, Japan. The COLING 2016 Organizing Committee.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. 2015. In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning. ArXiv:1412.6614.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the EMNLP*, pages 1532–1543, Doha, Qatar. ACL.

Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Lariviere, Alina Beygelzimer, Florence d'Alche Buc, Emily Fox, and Hugo Larochelle. 2021. Improving Reproducibility in Machine Learning Research (A Report from the NeurIPS 2019 Reproducibility Program). *Journal of Machine Learning Research*, 22(164):1–20.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. ACL.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(146):1–67.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence Level Training with Recurrent Neural Networks. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2–4, 2016, Conference Track Proceedings*.

Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M. Rush. 2021. Learning from others' mistakes: Avoiding dataset biases without modeling them. *CoRR*, abs/2012.01300v1.

Shoetsu Sato, Jin Sakuma, Naoki Yoshinaga, Masashi Toyoda, and Masaru Kitsuregawa. 2020. Vocabulary Adaptation for Domain Adaptation in Neural Machine Translation. In *Findings of the ACL: EMNLP 2020*, pages 4269–4279. ACL.

Danielle Saunders. 2021. Domain Adaptation and Multi-Domain Adaptation for Neural Machine Translation: A Survey. *CoRR*, abs/2104.06951.

Christophe Servan, Josep Maria Crego, and Jean Senellart. 2016. Domain specialization: a post-training domain adaptation for Neural Machine Translation. *ArXiv*, abs/1612.06141.

Shiqi Shen, Yang Liu, and Maosong Sun. 2017. Optimizing Non-Decomposable Evaluation Metrics for Neural Machine Translation. *Journal of Computer Science and Technology*, 32:796–804.

Michal Štefánik, Vít Novotný, Nikola Groverová, and Petr Sojka. 2022. Adaptor: Objective-Centric Adaptation Framework for Language Models. In *Proceedings of the 60th Annual Meeting of the ACL: System Demonstrations*, pages 261–269, Dublin, Ireland. ACL.

Michal Štefánik, Vít Novotný, and Petr Sojka. 2021. Regressive ensemble for machine translation quality evaluation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 1041–1048. ACL.

Christian Szegedy, V. Vanhoucke, S. Ioffe, Jonathon Shlens, and Z. Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *IEEE Conf. CVPR*, pages 2818–2826, Los Alamitos, USA. IEEE.

Brian Thompson, Jeremy Gwinnup, Huda Khayrallah, Kevin Duh, and Philipp Koehn. 2019. Overcoming Catastrophic Forgetting During Domain Adaptation of Neural Machine Translation. In *Proceedings of the 2019 Conference of the NAACL: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2062–2068, Minneapolis, Minnesota. ACL.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proc. of the Eighth International Conf. LREC*, pages 2214–2218, Istanbul, Turkey. ELRA.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. EAMT.

Inigo Jauregi Unanue, Jacob Parnell, and Massimo Piccardi. 2021. BERTTune: Fine-Tuning Neural Machine Translation with BERTScore. In *Proceedings of the 59th Annual Meeting of the ACL and the 11th IJCNL, Volume 2: Short Papers*, pages 915–924. ACL.

Michael Ustaszewski. 2019. Exploring Adequacy Errors in Neural Machine Translation with the Help of Cross-Language Aligned Word Embeddings. In *Proceedings of the Human-Informed Translation and Interpreting Technology Workshop (HiT-IT 2019)*, pages 122–128, Varna, Bulgaria. Incoma Ltd., Shoumen, Bulgaria.

Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. 2023. DyLoRA: Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation. In *Proceedings of the 17th Conference of the European Chapter of the ACL*, pages 3274–3287, Dubrovnik, Croatia. ACL.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proc. of the 31st NIPS conference*, volume 30 of *NIPS '17*, pages 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Chaojun Wang and Rico Sennrich. 2020. On Exposure Bias, Hallucination and Domain Shift in Neural Machine Translation. In *Proceedings of the 58th Annual Meeting of the ACL*, pages 3544–3552. ACL.

Yiyu Wang, Jungang Xu, and Yingfei Sun. 2022. End-to-End Transformer Based Model for Image Captioning.

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. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proc. of the 2020 Conf. EMNLP: System Demonstrations*, pages 38–45. ACL.

Weijia Xu, Xing Niu, and Marine Carpuat. 2019. Differentiable Sampling with Flexible Reference Word Order for Neural Machine Translation. In *Proceedings of the 2019 Conference of the NAACL: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2047–2053, Minneapolis, Minnesota. ACL.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Improving Neural Machine Translation with Conditional Sequence Generative Adversarial Nets. In *Proceedings of the 2018 Conference of the NAACL: Human Language Technologies, Volume 1 (Long Papers)*, pages 1346–1355, New Orleans, Louisiana. ACL.

Ruiyi Zhang, Changyou Chen, Xinyuan Zhang, Ke Bai, and Lawrence Carin. 2020a. Semantic Matching for Sequence-to-Sequence Learning. In *Findings of the ACL: EMNLP 2020*, pages 212–222. ACL.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. BERTScore: Evaluating Text Generation with BERT. In *Proc. of International Conference on Learning Representations*.

A Hyperparameters

For each of the evaluated objectives, we perform a hyperparameter search independently over the selected parameters in the denoted range, based on the best in-domain validation BLEU reached in the adaptation to *Wikimedia* domain.

(1) **learning rate**: ranging from $2 \cdot 10^{-7}$ to $2 \cdot 10^{-4}$, with step 10. (2) **objectives ratio** α (Eq. (8)): we manually set the weight of the additional objective such that the loss values for both components of the final loss are approximately balanced, based the first 10 valuations. We do not perform further tuning and use the same weights over all experiments. (3) **Batch size**: For *ML* experiments, we fix the effective batch size to 60, we pick the optimal batch size for *TokenAlign* and *SeqAlign* objectives over $[1, 5, 10, 20]$.

Other parameters that we adjust and remain fixed over the experiments are the following: **warmup steps** $= 1,000$, **LR schedule** as *constant decay*. Distance-based objectives including *SeqAlign* introduce two new parameters: (i) K: a number of the sampled hypotheses and (ii) n: a number of most-likely tokens to align. To keep the computation time feasible, we do not perform further tuning and set these parameters to $K = 10$ and $n = 3$ over all the experiments. All other parameters can be retrieved from the defaults of TrainingArguments of Transformers (Wolf et al., 2020), version 4.10.2.

We treat the optimized hyperparameters as *independent*; hence we optimize each variable separately. Our configuration results in experimenting with 9 hyperparameter search runs for each objective, including *MLE* baseline.

We also tune selected parameters of *Adapters* and *LoRA* implementations based on their original papers: (i) A compressed representation size ratio $\frac{t}{h}$ to model hidden state size h is chosen from $t \in [2, 4, 16, 32]$, (ii) a learning rate is chosen from $\text{LR} \in [2 \cdot 10^{-3}, 2 \cdot 10^{-4}, 2 \cdot 10^{-5}]$. We pick as optimal $h = 32$, $h = 16$ and $\text{LR} = 2 \cdot 10^{-4}$, $\text{LR} = 2 \cdot 10^{-5}$ for *Adapters* and *LoRA*, respectively.

B Computational Requirements

We performed the adaptation of each of the proposed objectives on a server with a single *NVidia Tesla A100*, 80 GB of graphic memory, 512 GB of RAM and 64-core processor (*AMD EPYC 7702P*). We also tested to train all our experiments using lower configuration using a single *NVidia Tesla T4*, 16 GB of graphic memory, 20 GB of RAM, and a single core of *Intel(R) Xeon(R)* processor.

We benchmark the running times of the time-demanding parts of the adaptation process in the first-mentioned configuration. We find that the proposed decontextualization process required by *TokenAlign*, *SeqAlign-CE* and *SeqAlign-dec* takes in these settings between 50 minutes on the smallest domain to 25 hours on the largest domain. Table 3

Objective	Updates / hour	Updates to converge
MLE	451	15,500
TokenAlign	404	24,000
SeqAlign	287	11,875
SRand	152	10,100
SeqAlign-dec	295	7,500
SeqAlign-CE	585	23,740

Table 3: **Adaptation speed**: Average number of updates per hour and average number of updates to converge that we measure over objectives in our experiments.

shows the average speed of updates and the number of steps that each of the designed objectives requires to converge. Further details on our methodology are described in Section 4.2.

C Details of Alignment Algorithm

Algorithm 1 describes the alignment procedure that we propose to obtain *grounding embeddings* for the tokens of the trained model.

Our approach first *aligns* the model and embeddings vocabulary; Given a text t, we obtain two ordered sequences of textual subwords (tokens): grounding embeddings tokens $s_e(t)$ and model tokens $s_\Theta(t)$. We obtain the *model grounding embeddings* e^i_Θ of each *model* subword $s^i_\Theta \in s_\Theta(t)$ to each *grounding* subword $s_{e,i} \in s_\Theta(t)$ by (i) assigning the *coverage intervals* of t to each model and embedding subword $s_\Theta(t)$ and $s_e(t)$, and (ii) for each model subword $s^i_\Theta \in s_\Theta(t)$, searching for the subword $s^i_e(t)$ with *largest intersection* of the covering intervals $|s^i_\Theta \cap s^j_e|$.

```
proc $align\_to\_grounding(s_\Theta, s_e)$:
    foreach $i \in 1..|s_\Theta|$ do
        while $|s^i_\Theta \cap s^j_e| > best\_cov$ do
            $pairs_i \leftarrow j$
            $best\_cov \leftarrow |s^i_\Theta \cap s^j_e|$
            $j \leftarrow j + 1$
    return $pairs$
```

Algorithm 1: Ability to pair each model token s^i_Θ with the best-matching grounding subword s^j_e allows us to use alignment grounded in domain-agnostic representations. Relying on the consistent ranking of the aligned sequences, the grounding alignment algorithm requires at most $(|s_\Theta| + |s_e|)$ steps to finish.

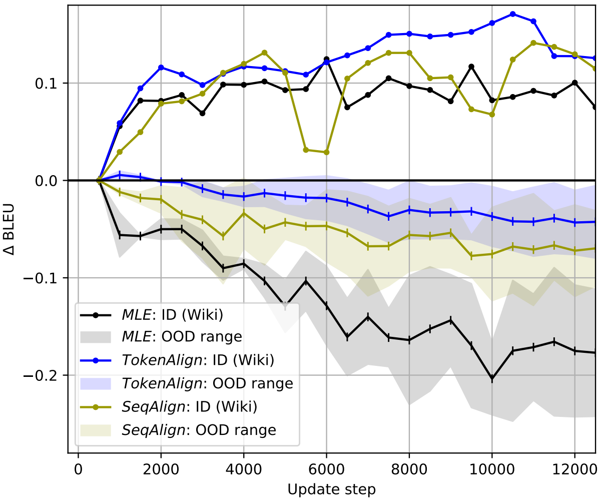

Figure 5: In-domain (ID) and out-of-domain (OOD) change of the original BLEU in domain adaptation of a translation model using ***MLE*** and the two introduced objectives: ***TokenAlign*** and ***SeqAlign***. Adaptation of Transformer-base model on Wikipedia, evaluated on a held-out set of the adapted domain (in-domain, ID) and a variety of out-of-domain (OOD) datasets (§4.2).

D Detailed Results of Ablation Objectives

Table 4 shows a comparison of *all* objectives over all evaluated domains, providing a finer-grained report of results presented in Table 2. Note that in order to eliminate the effect of different scaling of BLEU evaluations in character-segmented BLEU results, we exclude the (en→zh) pair from the ablations. The methodology of results collections is described in Section 4.2. The discussion including these results is present in Section 5.

E Training Validation Reports

We report and compare the change of validation BLEU of our two main objectives, relative to the *MLE* objective over the course of our experiments and overview the results in Figures 6 and 7 for *SeqAlign* and *TokenAlign* objective, respectively. A comparison of all three objectives is in Figure 5.

	Δ BLEU	Bible	Opensubs	Wiki	Medical/EMEA	Law/DGT
		(de→en)	(en→ukr)	(en→cze)	(est→en)	(en→de)
		50,000 pairs	80,000 pairs	100,000 pairs	300,000 pairs	5,100,000 pairs
Orig. BLEU		21.89	26.12	34.04	54.85	33.56
MLE	ID	− 8%	+ 4%	+9%	+38%	− 1%
	OOD	−53% ± 36%	−15% ± 9%	−15% ± 5%	−35% ± 10%	−19% ± 11%
TokenAlign	ID	−21%	+ 8%	**+12%**	**+45%**	+ 1%
	OOD	− 2% ± 1%	**− 1% ± 1%**	**− 6% ± 6%**	− 6% ± 7%	**+ 6% ± 20%**
SeqAlign	ID	−23%	− 8%	+ 8%	+31%	**+ 7%**
	OOD	**− 1% ± 1%**	− 2% ± 3%	−12% ± 5%	**− 1% ± 2%**	+ 3% ± 13%
SRand	ID	−14%	− 7%	+ 8%	+34%	− 7%
	OOD	− 8% ± 2%	− 3% ± 3%	− 9% ± 3%	− 7% ± 5%	− 7% ± 5%
SeqAlign-dec	ID	−26%	**+11%**	+ 5%	+35%	+ 2%
	OOD	−13% ± 8%	− 1% ± 1%	−11% ± 19%	−12% ± 7%	+ 4% ± 17%
SeqAlign-CE	ID	**+ 8%**	+ 9%	+11%	+ 1%	−11%
	OOD	−78% ± 9%	−32% ± 1%	−12% ± 5%	− 1% ± 2%	−14% ± 13%

Table 4: **Evaluation of adaptation quality and robustness over *all* designed objectives**: A change of BLEU score relative to the original model, when adapting pre-trained Transformer-base on a selected domain, as measured on a test set of the training domain (in-domain, ID) and out-of-domain (OOD). The aggregates over all domains are listed in Table 2.

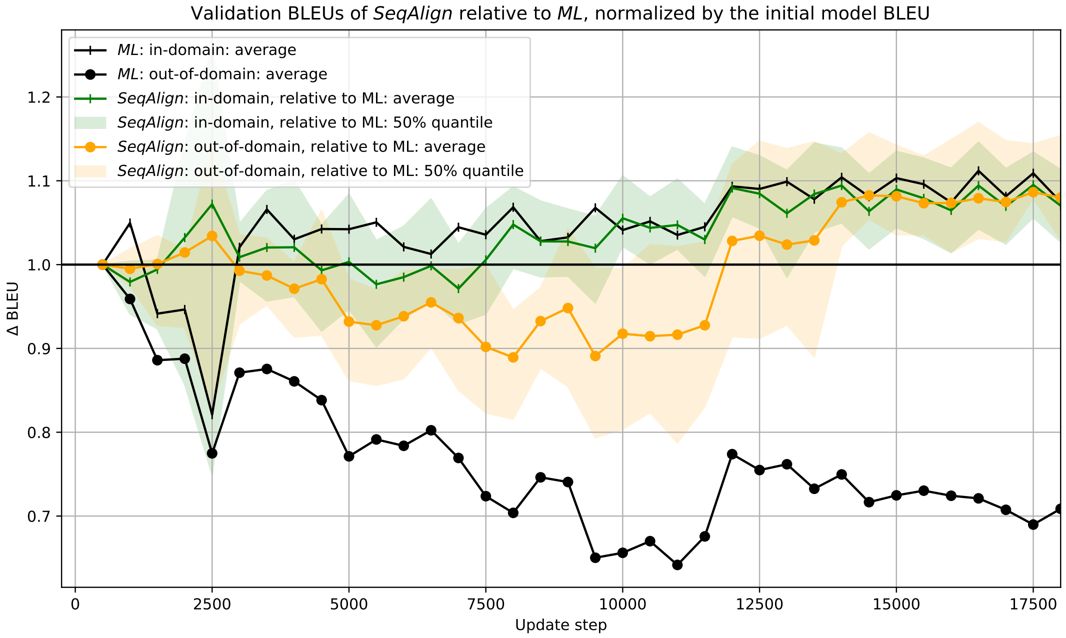

Figure 6: Comparison of **validation BLEU of *MLE* and *SeqAlign* objective** reported over the training on 5 different domains and 20 corresponding out-of-distribution domains until the in-domain early-stopping. For easier comparison, both *MLE* logs are averaged, and reported intervals correspond to the 50%-quantile of difference to the *MLE* run on the corresponding evaluation domain. While the training with *MLE* objective consistently magnifies the *forgetting* of adaptation, the soft objectives report a higher OOD score over all experiments while reaching comparable adaptation gains on the in-domain. Note that the two major gains of *SeqAlign* before steps 12,000 and 14,000 are attributed to the early stopping of specific runs at these points and hence, should be excluded from the conclusions. See Appendix E for further description.

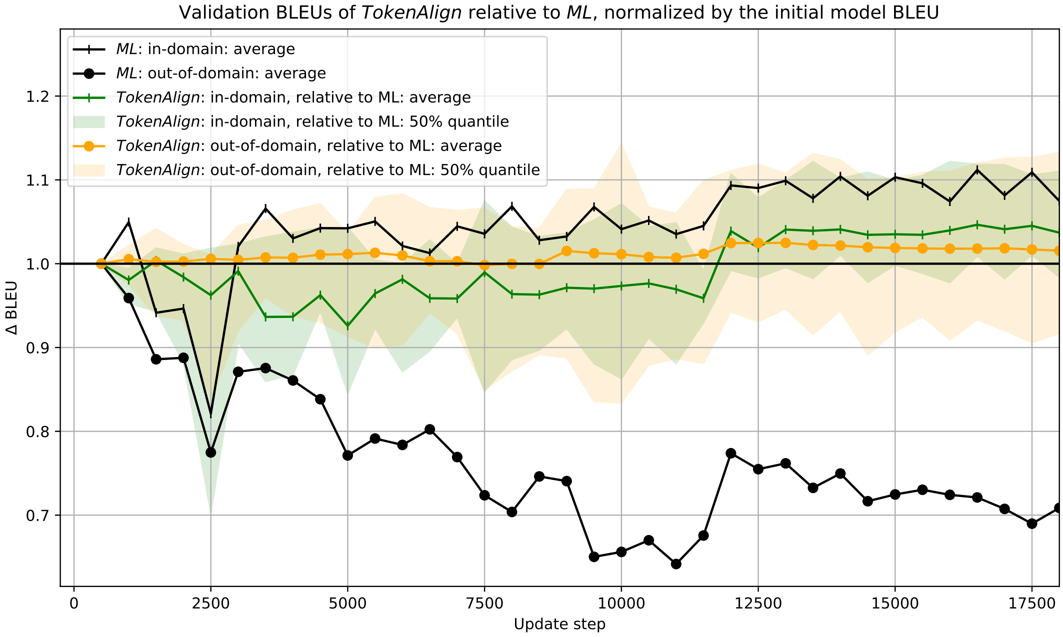

Figure 7: Comparison of **validation BLEU of *MLE* and *TokenAlign* objective** as reported over the training on 5 different domains and 20 corresponding out-of-distribution domains until in-domain early-stopping. See Figure 6 and Appendix E for further description.

The plots aggregate 5 training logs and their corresponding out-of-domain logs into the in-domain and out-of-domain reports, for easy comparability with *MLE*, both in-domain and out-of-domain BLEUs of *MLE* are *averaged* and paired with the corresponding BLEUs of the inspected objective over the shared evaluation domain. Finally, the plots of the inspected objective consist of *50% quantile intervals* and the *average* of BLEU relative to both the *MLE* BLEU and initial model performance. Note that while the relative distances of *MLE* to the corresponding plots of the other objective *always* correspond, some training runs are terminated in the course of the plotted steps, explaining some sudden performance gains in the plot.

While the performance decay of *MLE* by the time of early-stopping by in-domain BLEU is close to linear, *TokenAlign* on average maintains none, or minimal decays of the out-of-domain performance, although the variance of the initial decay significantly varies over domains. This trend implies that the early-stopping strategy based on in-domain performance does not significantly decay the robustness results and favors the deployment of *TokenAlign* in situations where no validation out-of-domain data is present.

The robustness of the model trained using *Seq-Align* behaves differently and the initial robustness decay is more significant. However, the decay soon diverges from *MLE* and noticeably, after the 5,000-th step *all* the robustness evaluations of *Seq-Align* report robustness gains over *MLE*.

Although we restrain from drawing conclusions based exclusively on these plots, the comparisons suggest that while the decay of robustness of *MLE* training is continuous, in the case of soft objectives, the decay gradually slows, while the model incrementally reaches potential in-domain gains similar to *MLE*.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 4.1 (explicit list of controlled covariates in selecting datasets), Section 5: Results (discussion of both the cases where our objectives improved compared to baselines, and where they did not). Section Limitations (enumerating a list of potential uncontrolled covariates of our experiments).

☑ A2. Did you discuss any potential risks of your work?
Section Limitations: We explicitly name the cases for which we are aware that our algorithms will fail.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Introduction lines 57-70.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4.1: Datasets: we enumerate datasets of our experiments; Section 4.2: Experimental setup: we refer and cite the authors of our adapted models.

☑ B1. Did you cite the creators of artifacts you used?
Section 4.1: we cite the introductory OPUS paper Section 4.2: we cite the authors of our base adapted models.

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
We checked that the artifacts of the previous work that we use does not counter our application. We do not deliver any new artifacts directly with our work.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
We thoroughly read the introductory papers of the authors of both our base models and datasets and found no explicit or implicit exclusion of our use of their artifacts. We license the implementation of our objectives under the MIT license, consistently with the authors of the Adaptor library in which we build our implementations.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The data of our experiments are collected from the publicly-available domains, as collected in OPUS. None of the domains that we use in adaptation should contain texts of a private origin. Hence, we do not have a reason to presume that any personal information would be contained in our training resources.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 4.1: Datasets: We explicitly name data covariates that we control in our experiments.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 4.2 (splits) Table 1 & 2 descriptions (number of examples).

C ☑ Did you run computational experiments?

Section 4.2: Experimental Setup (Reference to our base model), Appendix B: Computational demands

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4.2: Experimental Setup (Reference to our base model), Appendix C: Computational demands

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix B: Hyperparameter search, Appendix F: Training validation reports

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Figures 5 and 6: We report a change of validation BLEUs covering 50% of OOD evaluations for both objectives. Tables 1 and 2: test scores, including the intervals covering all the results.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
We used sacrebleu for all the reported evaluations, with no adjustments of the default settings. We state this information in Section 4.2, lines 436-437. The specific use can also be checked within the implementations of our experiments in the attached code repository.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.