Prefix Embeddings for In-context Machine Translation

Suzanna Sia ssia1@jhu.edu
Kevin Duh kevinduh@cs.jhu.edu
Johns Hopkins University

Abstract

The class of large generative pretrained (GPT) language models have demonstrated the ability to translate with in-context examples, a phenomena known as few-shot prompting. However, they have not achieved state-of-art results for translating out of English. In this work, we investigate an extremely lightweight fixed-parameter method for conditioning a large language model to better translate into the target language. Our method introduces additional embeddings, refered to as prefix embeddings which do not interfere with the existing weights of the model. Using unsupervised and weakly supervised methods that train only 0.0001% of the model parameters, the simple method improves up to around 5 BLEU points over the baseline when a single prompt example is provided, and up to around 2 BLEU points when 20 prompt examples are provided across 3 domains and 3 languages. We analyze the resulting embeddings' training dynamics, where they lie in the embedding space, and show that these conditional prefixes can be used for both in-context translation and diverse generation of the monolingual target sentence.

1 Introduction

Under the paradigm of in-context learning,[1] large language models have been shown to generate translations when provided with several priming examples, each of which consists of a source sentence and the translated target sentence. These examples, also known as "prompts", are prefixed to the test source sentence, which then conditions the model to generate the test target sentence. Table 1 shows an example of this format, where `[S1]` and `[S2]` are separator tokens prefixing source and target sentence respectively.

This prompt-and-translate phenomena, or *in-context translation*, presents itself as a new paradigm for Machine Translation applications. First, the ability to adapt to different task specifications using prompts suggest that the same model can be used in multiple settings and domains. While there have been several multilingual translation models (Fan et al., 2021; Xue et al., 2021; Ma et al., 2021), the ability to perform unrelated tasks such as Question-Answering in addition to Translation is relatively new. This also presents an interesting shift from supervised Neural Machine Translation (NMT) in terms of data requirements. These models are trained on massive amounts of web text which are not explicitly parallel.[2] In contrast, modern NMT models are trained with millions of lines of parallel text. Unsuprisingly, the lack of supervision comes at a cost. Translating out of English for in-context models still lags behind state-of-art possibly due to low data quality and/or disproportionate amounts of English.

[1] This has also been termed 'few-shot prompting' (Brown et al., 2020), but the field is increasingly converging on 'in-context learning' (Bommasani et al., 2021).

[2] This does not preclude the possibility that parallel sentences may exist in various forms in the crawled web text.

[S1] So at this point, music diverged	[S2] Donc à partir de là, la musique a divergé.
[S1] The actual rigging on the reins on the horse are made from the same sort of thing.	[S2] Les attaches sur les rennes du cheval sont faites du même genre de choses.
...	
[S1] And that was done with a particle.	[S2]

Table 1: A single continuous input sequence presented to the model for decoding a single test source sentence "And that was done with a particle". Given the entire sequence as input, the model proceeds to generate the target sequence after the final [S2]. $[\cdots]$ refers to several more [S1] en [S2] fr pairs.

In this work, we propose the training of **target language prefix embeddings to improve in-context translation.** Targetting specific languages has been explored in NMT models Yang et al. (2021) but much less so for the in-context setting. In contrast to fine-tuning, we do not change existing model weights. This falls into the class of 'fixed-parameter' methods where the original parameters of the model are held fixed and additional parameters are introduced which influence the activation states of the model. Our proposed method differs from the various approaches to "prefix tuning" (Li and Liang, 2021; Qin and Eisner, 2021; Asai et al., 2022; Lester et al., 2021) in that these all require explicit task supervision. Learning the weights of these prefix embeddings is technically straightforward using gradient descent optimisation machinery. We show that these embeddings can be trained unsupervised (subsection 3.2)[3] and also explore the use of a very small set of bitext sentences for weakly supervised training (subsection 3.3). Experiments were conducted across 3 en-fr domains (subsection 4.1) and from English into three languages French (fr), Portugese (pt), and German (de) (subsection 4.2). Overall, for a very small amount of engineering, data collection, and storage effort, training prefix embeddings can give up to 5 BLEU points for the 1-prompt setting, and up to around 2 BLEU points on the 20-prompt setting with a very small amount of bitext (we used 100 parallel sentences).

2 Related Work

Large language models which perform in-context translation Following GPT3 (Brown et al., 2020) which first reported the in-context translation phenomena, subsequent autoregressive Transformer decoder only architectures such as XGLM (Lin et al., 2021) and mGPT (Shliazhko et al., 2022) have explicitly trained in-context models to be multilingual. However decoding out of English still performs more poorly than decoding into English. Hence we focus on the first scenario of decoding out of English.

Prefix Tuning Unlike previous work which directly prefixes the task by prepending to the input (Li and Liang, 2021; Qin and Eisner, 2021; Asai et al., 2022; Lester et al., 2021), we substitute the trained prefixes for the delimiters throughout the prompts before the target language sequence. Our proposed method *prefixes the target sequence, not the task.* This small but significant difference allows monolingual training for the target language without explicit translation task supervision.

Embedding Tuning vs Prefix Tuning across all layers We adopt the embedding level tuning approach which was shown to be competitive with model tuning with an increasing number of parameters on SuperGLUE tasks (Lester et al., 2021). The focus on training prefix embeddings instead of training additional parameters to directly influence activations across all layers is a design choice primarily to accommodate for very large models. Li and Liang (2021) report

[3]Unsupervised in the terminology of Machine Translation means without parallel bitext sentences.

using 250K-500K of parameter training vs. a 345M Roberta model (Liu et al., 2019), which is 4-7% of the parameter space. If we had applied the same parameter ratio to our current model of 2.7B parameters, this would be equivalent to having to train 195M parameters – which is in the same order of magnitude of the Roberta model. We do acknowledge that embedding tuning is less expressive by virtue of having fewer entry points to influence the model's activations, and leave a middle ground solution such as combining with adaptor layers (Houlsby et al., 2019) to future work.

Language ID token training is a typical method in multilingual models, to condition the model for the source and target language. However these tokens are typically trained together with the rest of the model parameters and is a design choice that needs to be made upfront. In contrast, we use a generic large language model that was pretrained with minimal design choices, and then posthoc train a language specific prefix to condition the model to generate sentences in the target language, with the goal of improving in-context translation.

3 Methods

Our approach is motivated by the knowledge that for very large language models trained on web corpora, there is a weaker target language (being translated into) because English is the dominant language on the web. This trend persists even for explicitly multilingual language models (Lin et al., 2021). Our method therefore aims to condition the language model to decode the weaker target language, by learning a language-specific prefix. We first describe the in-context translation setup at test time (subsection 3.1), followed by unsupervised training (subsection 3.2) and weakly supervised training (subsection 3.3) of the target language prefix embedding. At inference time, the corresponding prefix will be used as the separator token between source and target language. Figure 1 illustrates this process.

3.1 In-context Translation

Let $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}_\mathrm{b}$ be a set of translation pairs that the model has access to at inference time, where $\mathbf{x}$ refers to the source sentence and $\mathbf{y}$ refers to the target sentence. Given the separator tokens `[Sx], [Sy]` and the test source sentence $\mathbf{x}_\mathrm{test}$, we can define a prompt layout format $u(\mathbf{x}_\mathrm{test}, \mathcal{D}_\mathrm{b}, \texttt{[Sx], [Sy]})$ (Table 2), where `[...]` refers to several similarly formatted $\mathbf{x}$, $\mathbf{y}$ examples from $\mathcal{D}_b$. The default in-context learning model autoregressively generates the target sequence by greedily decoding $\hat{\mathbf{y}} = \mathrm{argmax}_\mathbf{y} p(\mathbf{y}|u(\mathbf{x}, \mathcal{D}_\mathrm{b}, \texttt{[Sx], [Sy]}))$. Our goal is to learn a target specific prefix `[S*]` that achieves higher $p(\mathbf{y}|u(\mathbf{x}, \mathcal{D}_\mathrm{b}, \texttt{[Sx], [S*]}))$ for the correct sequence $\mathbf{y}$. We use "$*$" to indicate that the prefix can be of any length.[4]

[Sx] $\mathbf{x}_1$	[Sy] $\mathbf{y}_1$
[Sx] $\mathbf{x}_2$	[Sy] $\mathbf{y}_2$
...	
[Sx] $\mathbf{x}_\mathrm{test}$	[Sy] ?

Table 2: The prompt layout format from $u(\mathbf{x}_\mathrm{test}, \mathcal{D}_\mathrm{b}, \texttt{[Sx], [Sy]})$.

3.2 Unsupervised Training (monolingual)

The primary strategy is simple, train `[S*]` such that it conditions the model to generate sequences $\mathbf{y}$ from the target language. We expand the tokenizer and the corresponding embedding

[4]In practice, we use special tokens such as `[0], [1] ···, [n]` for a prefix of length n and verify that these do not have a collision in the tokenizer namespace.

matrix by the number of prefix tokens, and then prepend the special token [S*] to monolingual sentences during training. A single training sequence is given by "[S*] $\mathbf{y}$", where $\mathbf{y}$ is typically a sentence or paragraph. Given m sequences from a target language training set $\mathbf{y}_1, \cdots, \mathbf{y}_m \in \mathcal{D}_y$, we train the embedding parameters $\theta = \texttt{Embed}(\texttt{[S*]})$, where [S*] indexes the additional rows in the embedding matrix. We use cross-entropy loss as is standard with language modeling, and freeze the parameters of the entire network except for θ.

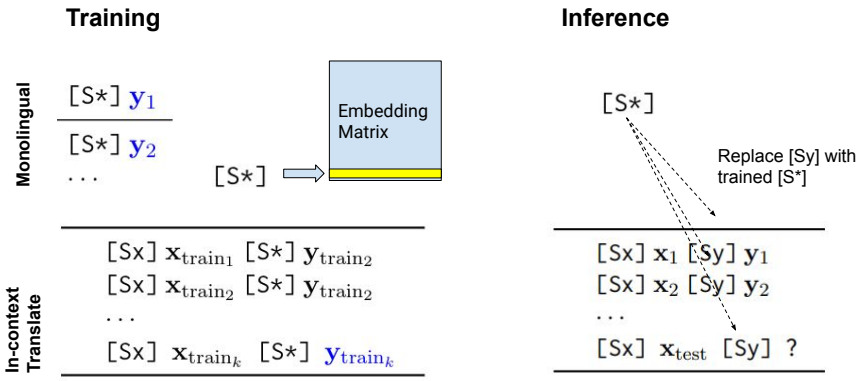


Figure 1: Prompt format for training and inference time. Training loss is computed on the sequences in blue. The token [S*] corresponds to additional row(s) of the embedding matrix which are the only parameters trained by backpropagation. At inference time, we replace [Sy] with the trained [S*] to conditionally generate $\mathbf{y}_{\text{test}}$. Note that [S*] can also be used to generate sequences in the target language directly (subsection 5.4).

3.3 Weakly Supervised Training (In-context Translate with Bitext)

In the previous section, training of θ uses only the target language, without any bilingual supervision for the in-context translation task. To guide θ to a better local optima, we include a very small amount of bitext, 100 parallel sentence pairs which are a subset of the training set. We adopt a weakly supervised setup where we initialise the prefix embeddings using existing tokens, and also where the prefix embeddings are initialised from the monolingual trained prefix embeddings (referred to as *mono-trained-lang* in section 4).[5] An alternative training approach is a multi-task setup where losses from the monolingual language modeling and translation tasks are minimised in alternative batches, however this was thought to be less effective due to the extreme data imbalance of the setting that we consider (30k monolingual sentences to 100 bitext pairs), which might require arbitary reweighting schemes. Since the end-goal is translation, directly tuning towards this is a more straightforward approach.

Figure 1 shows a single in-context translate training sample for the model. Note that loss is computed only for the last target sentence $\mathbf{y}_{\text{train}_k}$. In all our experiments we use $k = 6$ for training, i.e., 5 priming examples. For each datapoint, we randomly sample from the $\mathcal{D}_{\text{train}}$ to construct the prompt set, so that the parameters of [S*] do not overfit to any particular choice of prompt set. Note that for large language models, $|\mathcal{D}_{\text{train}}|$ is less than the number of parameters being trained; a single prefix token has already over 2000 dimensions. We do not

[5]Note that since monolingual data had been used to initialise the prefix, this can be interpreted as a continual semi-supervised learning set up.

expect $\mathcal{D}_{\text{train}}$ to allow the model to learn a mapping for translations, and its role is merely to weakly supervise the training of the monolingual prefix towards loss basins that are compatible with the prompt-translate paradigm.

4 Experiments

We organise our experiments investigating the effects of prefix embedding tuning 1) across three en-fr domains, medical, social media, and TED talks, 2) across three languages in TED Talks.[6] In both sets of experiments, we explore three basic initializations (described in **Prefix Embedding Initialisations**). We also use priming examples of various sizes to investigate if the effects persist across different prompt sizes. To account for prompt selection and ordering effects, all inference runs were repeated with 5 randomly sampled prompt sets from the training data, where each of the source sentences in the prompt examples are between 10 to 20 words long. Scores are reported using SacreBLEU(Post, 2018).[7]

Model We use GPTNeo2.7B (32 layers, 20 heads) (Black et al., 2021) which has been pretrained on The Pile (Gao et al., 2020). The Pile contains Europarl which has been fed into the model at a document level and not a sentence level.[8] To our knowledge, there has not been any reports of sentence level parallel corpora in the training dataset of this model. Note that unlike most dedicated Machine Translation models which have an encoder-decoder architecture, this model is trained autoregressively and is decoder only. [9]

Data We adopt three datasets; multilingual TED talks (Duh, 2018), MED (Bawden et al., 2019), and MTNT (Michel and Neubig, 2018). We use 30,000 monolingual sentences for unsupervised training of the prefix embeddings (`mono` in results table). For TED, MED and MTNT, the monolingual sentences are obtained from their bitext training data. We use 100 bitext sentence pairs for the weakly supervised case (`bitext` in results table). These bitext sentence pairs served as a self-contained prompt set and training data instances as described in subsection 3.3. In both the unsupervised and weakly supervised scenarios, During testing, we sample sentence pairs for prompts examples from the training set. The sentence pairs in weakly supervised training, validation, and inference time prompt selection are all separate splits; there is no overlap between prompt sets seen across these phases.

Preprocessing We preprocess digits to _ as we find that this helps the prefix tuning to converge for the MED and TED domains, without compromising on the ability to copy or generate digits. We run a langid check and restrict training sentence length to 3 to 25 words to avoid trivial sequences and out-of-memory errors.

Prefix Embedding Initialisation We investigate two simple forms of initialisation

- *random* refers to the default behavior of the model when adding new parameters to the embedding. For GPTNeo model, this is drawn from $\mathcal{N}(0, 0.02)$ as the model uses GELU activation units (Hendrycks and Gimpel, 2016). We report results for *random* using the best out of 3 trained prefix embeddings based on the dev set.
- *lang* uses existing words from the vocabulary which is related to the language and the domain. For fr, pt, de, we initialise with the words "French, Portuguese, German", for

[6] Code at `https://github.com/suzyahyah/prefixes_incontext_machinetranslation`.

[7] nrefs:1 | case:lower | eff:no | tok:13a | smooth:exp | version:2.0.0

[8] `https://github.com/thoppe/The-Pile-EuroParl`

[9] We report SOTA results on the datasets although this is not directly comparable because of completely different training data setup of the base model. TED en-fr: 35.9 en-pt: 38.3 en-de: 28.1 (Renduchintala et al., 2019) MED: 39.5 (Bawden et al., 2019) MTNT: 29.7 (Michel and Neubig, 2018)

MTNT, MED and TED we use "social, medical, talks" respectively. This means that for French MED, we would initialise the first prefix with the embedding corresponding to " French" and the second prefix with the embedding corresponding to "medical". [10]

- *mono-trained-lang* are embeddings initialised from monolingual training (the previous bullet point) for further (weakly) supervised training using 100 additional parallel sentences.

Validation Loss For both monolingual training and weakly supervised bitext training, we use the prompt-translation paradigm as the validation loss. This avoids overfitting to the monolingual target sentence at the expense of being able to translate in the in-context setup. The set of translation prompts for the validation set are randomly drawn from within that set itself, removing dependency on any particular prompt set used at inference time. It may be possible to achieve better performance if practitioners were to use the same prompt set at train, validation and test time.

Training Details We apply early stopping with patience over 5 epochs and threshold 0.001 loss. We adopt 4 gradient accumulation steps with a batch size of 8 for an effective batch size of 32 for the monolingual training, and 4 gradient accumulation steps with a batch size of 2 for an effective batch size of 8 for the weakly supervised bitext training to avoid out-of-memory errors. All experiments can be run with a single NVIDIA-TITAN RTX GPUs (24GB). Monolingual training takes about 1 hour per epoch and can range from 8-20 hours for convergence.

Prompt Format (u) We tried several manual variants of `[Sx]` and `[Sy]` but did not optimise over this extensively. Our preliminary experiments showed that using untrained *lang* tokens in the separator performed slightly better, i.e., using the token "French" as `[Sy]` performed better than a separator choice such as ' `A`:'. We also experimented with prepending the entire prompt sequence with Natural Language Instructions: "Translate English to French" but found that this did not help consistently across datasets, hence we opted to exclude it to simplify design choices and isolate the effects of the trained prefix.

4.1 Results for Performance Across Domains [Table 3]

We present the results for 1-prompt and 20-prompt setting in Table 3. The 1-prompt example shows the extreme case of having no bitext data. While this is perhaps an overly restrictive assumption especially in industrial settings, the goal of this experimental setting is to illustrate the effect of the extreme monolingual scenario. The 20-prompt setting simulates a "saturated" prompt setting, which we also investigate with more prompt intervals in Figure 2.

Unsupervised (monolingual) Prefix Training helps 1-prompt setting. Across all domains, unsupervised (`mono`) prefix training tends to improve BLEU score. This improvement is much more prominent in the 1-prompt setting, with improvements of around 5 BLEU points across the three data domains of MED, TED and MTNT. Recall that the `mono` trained *lang* initialised token embedding has no knowledge of translation and only serves to condition the model to generate the target language.

Weakly supervised (bitext) Prefix Training helps the 20-prompt "saturated" setting. A very small amount of supervision with 100 examples can be used to do better than the baseline (0.3 to 1.3 BLEU point gains).[11] It is not always clear whether initialising from a *mono-trained-lang* embedding helps as the performance is the same for TED and MED, but slightly better (0.5 gains) for MTNT. Looking at the 1-prompt case for `bitext`, *mono-trained-lang* always does

[10] Note that having two words does not necessarily correspond to having two tokens.

[11] How much supervision is required? We separately find that increasing from 100 to 1000 training examples performs within 0.1 BLEU points of the `bitext` *mono-trained-lang* (last column of Table 3.

exp	direction	nprompts	untrained *lang*	mono (unsupervised) *random*	mono (unsupervised) *lang*	bitext (supervised) *lang*	bitext (supervised) *mono-trained-lang*
MED	en-fr	1	8,8 (1.6)	**13.3*(2.4)**	12.0*(4.8)	7.6 (4.8)	10.7 (4.1)
MTNT	en-fr	1	10.7 (3.5)	7.3 (4.2)	**15.5*(2.5)**	14.2 (2.6)	18.4 (1.3)
TED	en-fr	1	12.7 (4.7)	16.4*(3.8)	**17.7*(2.1)**	18.8 (1.1)	19.1 (0.9)
MED	en-fr	20	17.9 (0.7)	11.5 (1.4)	18.1 (0.8)	**18.4*(0.5)**	**18.4*(0.5)**
MTNT	en-fr	20	21.0 (0.6)	1.5 (0.9)	21.2 (0.3)	21.5*(0.4)	**22.0*(0.4)**
TED	en-fr	20	22.5 (0.2)	21.2 (0.9)	22.2 (0.2)	**22.8 (0.2)**	**22.8 (0.1)**

Table 3: BLEU points across different domains of Medical (MED), Social Media (MTNT) and TED Talks. We report the average of 5 random prompt sets with standard deviation. The best result is in bold row-wise, and (*) indicates $p < 0.01$ for a paired permutation test (1000 rounds) against the baseline (untrained). For 1-prompt case, this assumes that there is no bitext available, although we report bitext results (in lightgray) for the sake of completeness. The number of prefixes tokens for all experiments in this table is 2.

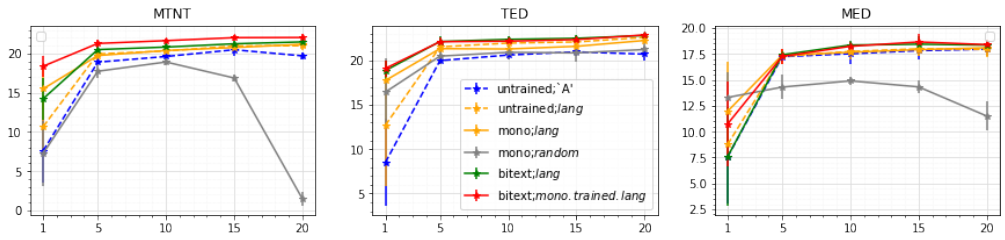


Figure 2: k-shot performance with trained prefixes on MTNT, TED and MED datasets for en-fr. Plots show $k = \{1, 5, 10, 15, 20\}$ examples on the x-axis. Baselines include lang token without further training (untrained; *lang*), an "A" token without further training (untrained; *'A'*), and the best out of 3 randomly initialised embeddings (mono; *random*).

better than *lang*, indicating that there is still an effect of having higher scores over the target language logits, but that this effect vanishes with increasing prompts.

Plateau effect across increasing number of prompt sets. We observe a plateau effect after 5 prompts which is consistent with the hypothesis that the primary role of prompts is task location rather than instruction (Reynolds and McDonell, 2021). With regards to individual prompt set selection, we find that improvements occur across all prompt sets that had been randomly selected, strongly suggesting that the improvements from training the target language prefix are orthogonal (or can be used independently) of prompt selection and ordering effects to achieve better translation results. We further analyse the improvements at a sentence level in subsection 5.3.

Random initialisation has high variance. If not correctly initialised, the prefix might not converge to a good local optima for in-context translation and can result in worse performances than baseline despite having low perplexity on the monolingual training set. Very occasionally we might observe a stronger performance, in the case of mono *random* in the 1-prompt setting). This suggests that the primary reason why a monolingual trained prefix can still have good performance when used in the in-context translation setting, is because it still retains properties from the word embedding that it is initialised from that are compliant with the activation states for in-context translation.

4.2 Performance Across Languages

In this section we presents experiments with GPTNeo2.7B on TED talks for German(de) and Portugese (pt). Overall the results are encouraging for prefix training; we observe improvements in the 1-prompt setting with `mono` *lang*, and in the 20-prompt setting with `bitext` *mono-trained-lang* with en-de and en-pt. The limited en-pt gain may be explained by the already high scores on the `untrained` *lang* at 22.0 BLEU, but it remains unclear why the improvements are limited for en-de.

			`untrained`	`mono` (unsupervised)	`bitext` (supervised)	
exp	direction	nprompts	*lang*	*lang*	*lang*	*mono-trained-lang*
TED	en-fr	1	12.7 (4.7)	**17.7*****(2.1)**	18.8 (1.1)	19.1 (0.9)
TED	en-de	1	8.5 (2.2)	**9.7*****(2.2)**	13.6 (0.6)	15.1 (0.5)
TED	en-pt	1	22.0 (0.7)	**22.9*****(0.8)**	24.3 (0.8)	25.0 (1.2)
TED	en-fr	20	22.5 (0.2)	22.2 (0.2)	**22.8 (0.2)**	**22.8 (0.1)**
TED	en-de	20	16.6 (0.3)	16.1 (0.6)	17.4*(0.2)	**17.6*****(0.2)**
TED	en-pt	20	24.9 (0.4)	25.8*(0.9)	**27.2*****(0.4)**	26.8*(0.2)

Table 4: BLEU points across different language directions translating from English (en) to French (fr), Portugese (pt), German (de). We report the average of 5 random prompt sets with standard deviation. The best result is in bold row-wise and (*) indicates $p < 0.01$ for a paired permutation test (1000 rounds) against the baseline (`untrained`). For 1-prompt case, this assumes that there is no bitext available, although we report bitext results (in lightgray) for the sake of completeness. The number of prefixes tokens for all experiments in this table is 2.

Effect across languages are not equal. Translation into de and pt for the 20-prompt setting under very weak supervision of 100 bitext examples gives around 1 to 2 point gains which is slightly more encouraging than en-fr, suggesting that the performance gains are not equal across languges. Curiously, the corresponding gains from the 1-prompt setting for en-de and en-pt are much smaller around 1 point compared to the 5 point gain for the 1-prompt setting in en-fr.

5 Analysis

5.1 Trained Prefix in Embedding Space

What is the difference between *lang* initialised and *random* initialised prefix embeddings? To get a better of understanding of the local minima, we compare them before and after training. In Table 5 we present the top 20 closest tokens by cosine distance to the prefix (before and after), and in Figure 3 we observe the 'density' of the closest 50 tokens by cosine distance in a PCA plot. A similar pattern emerged across all domains regardless of whether unsupervised or weakly supervised training.

Observations

1. For the *lang* token1, the closest 10 words are in the similar theme of country/language. However after the 10th word, this diverges to a different set of words. From the PCA plots, we can see that the red points (the closest 50 words) are largely a different set of words in a different part of the embedding space.

2. For the *lang* token2, we observe that the top 20 words do not change much unlike *lang* token1. This indicates that *lang* token1 may play a more critical role in conditioning the model. We do not plot *lang* token2 in Figure 3 as this is domain specific and different across different domains.

[Before Training]	
lang token1	French, French, France, french, Spanish, Italian, German, Dutch, Swedish, Belgian, Danish, Portuguese, Russian, Frenchman, Japanese, Paris, Turkish, Irish, Polish, Norwegian
lang token2	social, social, Social, Social, socially, societal, socio, SOC, soc, Facebook, cultural, facebook, FB, Soci, Twitter, sociop, civic, twitter, hugely, Instagram
random token1	exponent, Occ, ashi, 070, Redd, multiplication, Consumer, ost, grinning, promul, pos, crafted, apex, Import, justifying, 778, Ing, std, spit, grad
random token2	Apply, EN, round, ail, private, fruit, su, San, marks, akra, wi, atin, tar, arb, ank, ADVERTISEMENT, gi, ORN, ize
[After Training]	
lang token1	French, French, french, France, Italian, German, France, Spanish, Russian, Dutch, Paris, scrut, amazingly, showcasing, fueling, meticulously, nurturing, boosters, fiercely, British
lang token2	social, social, Social, Social, socially, societal, socio, SOC, soc, Facebook, FB, facebook, twitter, Twitter, Soci, cultural, incess, sociop, Instagram, hugely
random token1	452, 647, 339, Maurit, 467, 751, 466, 146, bustling, 338, 383, 546, 626, 340, 604, 267, 287, 649, 447
random token2	soDeliveryDate, istg, Skydragon, ÛÛ, srfN,‾‾‾‾‾, embedreportprint, = = , quickShipAvailable, natureconservancy, guilcon,externalToEVA, RandomRedditorWithNo, largeDownload

Table 5: Top 20 tokens by cosine similarity to the prefix token before and after training. *lang* token1, *rand* token1 and 2 are the same across MTNT, TED and MED datasets. *lang* token2 is a domain specific word, in this case "social" for the prefix trained in MTNT dataset. Note that the trained prefix token1 and token2 are concatenated as a prefix of length 2.

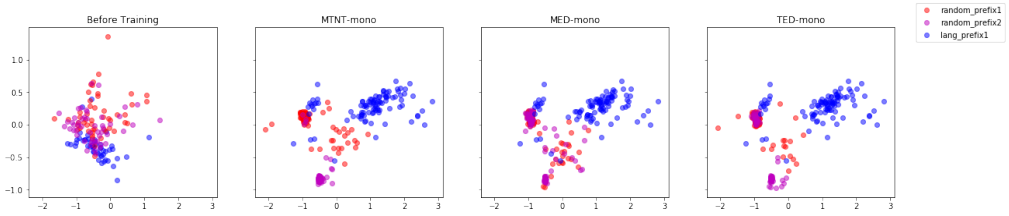


Figure 3: Each point in the plot corresponds to a token which is closest to the prefix by cosine similarity in the full embedding space dimensionality before and after training. The top 50 nearest tokens are plot in 2D as reduced by PCA. Orange and red are tokens closest to random prefix 1 and 2 respectively, while blue are tokens closest to the lang prefix 1, which corresponds to the token "French".

3. *random* token1 and token2 start out in a visually similar density spread to *lang* token1 and the similar tokens are in some generic random space. However after training, the cluster of similar words become very concentrated in the same 2D space (Figure 3). For *random* token1 this is three digit numbers and for *random* token2 this is CamelCased words.

5.2 Validation Loss Across Training Epochs

We present the validation loss under a 5-prompt setting, as training loss for unsupervised and weakly supervised are not directly comparable. We can observe that at the beginning of training, the *lang* initialised prefixes are already performant. This corresponds to the `untrained` *lang* prefix in Table 3. Validation loss increases for the `mono` although this is validation at the 5-prompt validation setting. As reported in Table 4, the trained `mono` prefix give 5 BLEU points at the 1-prompt test setting. For the weakly supervised bitext setting, the loss continues to fall very gradually and consistently under the weakly supervised bitext setting.

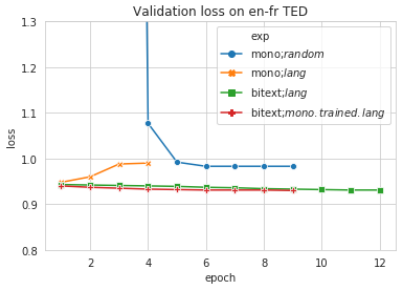
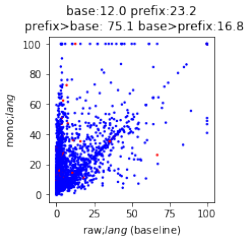


Figure 4: Validation (log) loss plots on en-fr for the TED dataset. The validation loss is the in-context translate loss using 5 prompts. The validation loss for *random* which initially starts at 17.6 is not shown in this chart.

5.3 Sentence Level Analysis for 1-prompt setting

In Table 3 and Table 4, we reported BLEU scores averaged across 5 random prompt sets. We find that when the mono trained prefix method is performant in the 1-shot setting, it does better than the baseline consistently across all 5 random prompt sets. We thus look at the scatter plot of sentence level scores to see whether improvements are coming from across all sentences or from a small group of sentences. Points in red are sentences which did not get translated into the target language in the baseline case. We show the scatterplot for a single prompt set and MTNT domain, as other plots follow a similar pattern.



domain	prefix>base	prefix=base	prefix<base
MED	70.4%	10.4%	19.2%
MTNT	70.0%	7.7%	22.3%
TED	64.5%	12.8%	22.7%

Figure 5: Scatter plot for the 1-prompt setting, for en-fr in MTNT, where each point is a single sentence, Y-axis shows mono trained *lang* prefix versus the baseline (untrained prefix) on the X-axis. We report % of sentences where the prefix underperforms, equals to, and outperforms the baseline, averaged across 5 prompt sets, for unsupervised (monolingual) trained prefix.

Observations

1. **Prefix embedding does better "on average" rather than universally across all sentences.** We quantify the % of sentences where using prefix outperforms baseline and vice versa. Many sentences which score higher with the prefix occurs when the baseline has very low scoring sentences. This likely accounts for the higher BLEU scores. Interestingly, about 20-25% of the sentences across the three domains perform worse with the trained prefix, than without. Overall this suggests that a potential next direction might be in translation reranking methods.

2. Most points in red appear to be above the diagonal, indicating that sentences that were previously not translated into the target language are mostly scoring higher.

5.4 Sampling from Prefix

Using [S*] as the starting token in a sentence, we sample subsequent tokens from the LM using the vanilla softmax probability distribution without any other probability rescoring tricks. We observe that the prefix token conditions the model to generate naturally diverse outputs, indicating that it has a non-peaky distribution over the target language space. This indicates that the prefix is highly flexible in conditioning the model to generate sequences from that language and the domain. This might be potentially useful as a generic domain prefix for other tasks beyond translation, such as generating dialogue in a particular style.

MTNT	Après un peu d'attention, qu'est-ce que vous voudriez (*after a little attention, what would you like*)
	Et quelle est la façon d'interdire que ça ait lieu? Laisse moi parler le pignon, ... *(And what is the way of prohibiting that it takes place? Let me speak the gable, ...)*
MED	Une étude met en évidence une association entre ces facteurs et le degré d'état de santé chez les adolescents... *(A study highlights an association between these factors and the degree of health in adolescents ...)*
	Nous avons interrogé une grande majorité des parents ayant reçu des soins de santé pour connaître le résultat ... *(We interviewed a large majority of parents who received health care to find out the result ...)*
TED	Le taux annuel des demandeurs d'un emploi est de 2,4 %. L'enregistrement de ce taux en janvier... (*The annual rate of job seekers is 2.4 %. Recording this rate in January ...*)
	Sérieusement. Et c'était quand même pas trop. L'air mou et froide comme ça et la réalité se révéla que mes souvenirs *(Seriously. And it was not too much. The soft and cold air like that and the reality turned out that my memories)*

Table 6: Random samples from prefixes trained on monolingual data for french MTNT, MED and TED, together with their English translations (from Google Translate) in italics for readability.

6 Conclusion

In this paper, we show that priming of in-context learning models can be improved using primarily unsupervised methods. To our knowledge, this is the first work which emphasises the target side language during decoding of a large language model for in-context translation. The gains are modest but so are the number of parameters trained. In our experiments we have shown that the simple method gives up to 5 BLEU point gains for monolingual training in the 1-shot setting, and weakly supervised bitext training in the 20-shot setting gives up to around 2 BLEU point gains across 3 domains and 3 languages. Given that we leverage primarily on unsupervised (monolingual) target side training and carefully control for random prompt selection, this could be a generic approach for improving decoding into a weaker target distribution, which is complementary to the vast literature on prompt example selection and optimisation (Liu et al., 2021).

Limitations We have used one model, GPTNeo2.7B, in this set of experiments. Although this accessible off-the-shelf model is considered a replication of GPT3 in terms of architecture and is highly used (88k downloads in the month of January 2022), other factors such as different training data or scale of the model (100B parameter vs 2B parameters) may affect generalisability of the results. There are no known ethical concerns.

Acknowledgements We thank Marc Marone, Tony Ramirez, and Neha Verma for useful comments and discussion, and Benjamin Van Durme for computational resources for some portion of the experiments.

References

Asai, A., Salehi, M., Peters, M. E., and Hajishirzi, H. (2022). Attentional mixtures of soft prompt tuning for parameter-efficient multi-task knowledge sharing. *arXiv preprint arXiv:2205.11961*.

Bawden, R., Bretonnel Cohen, K., Grozea, C., Jimeno Yepes, A., Kittner, M., Krallinger, M., Mah, N., Neveol, A., Neves, M., Soares, F., Siu, A., Verspoor, K., and Vicente Navarro, M. (2019). Findings of the WMT 2019 biomedical translation shared task: Evaluation for MEDLINE abstracts and biomedical terminologies. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 29–53, Florence, Italy. Association for Computational Linguistics.

Black, S., Leo, G., Wang, P., Leahy, C., and Biderman, S. (2021). GPT-Neo: Large scale autoregressive language modeling with Mesh-Tensorflow.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Duh, K. (2018). The Multitarget TED Talks Task. `http://www.cs.jhu.edu/~kevinduh/a/multitarget-tedtalks/`.

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., Baines, M., Celebi, O., Wenzek, G., Chaudhary, V., et al. (2021). Beyond English-centric multilingual Machine Translation. *Journal of Machine Learning Research*, 22:1–48.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., et al. (2020). The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Hendrycks, D. and Gimpel, K. (2016). Gaussian Error Linear Units (GELUs). *arXiv preprint arXiv:1606.08415*.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. (2019). Parameter-efficient Transfer Learning for NLP. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Lester, B., Al-Rfou, R., and Constant, N. (2021). The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Li, X. L. and Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Lin, X. V., Mihaylov, T., Artetxe, M., Wang, T., Chen, S., Simig, D., Ott, M., Goyal, N., Bhosale, S., Du, J., et al. (2021). Few-shot learning with multilingual language models. *arXiv preprint arXiv:2112.10668*.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2021). Pre-train, prompt, and predict: A systematic survey of prompting methods in Natural Language Processing. *arXiv preprint arXiv:2107.13586*.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ma, S., Dong, L., Huang, S., Zhang, D., Muzio, A., Singhal, S., Awadalla, H. H., Song, X., and Wei, F. (2021). Deltalm: Encoder-decoder pre-training for language generation and translation by augmenting pretrained multilingual encoders. *arXiv preprint arXiv:2106.13736*.

Michel, P. and Neubig, G. (2018). MTNT: A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 543–553, Brussels, Belgium. Association for Computational Linguistics.

Post, M. (2018). A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Qin, G. and Eisner, J. (2021). Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5203–5212.

Renduchintala, A., Shapiro, P., Duh, K., and Koehn, P. (2019). Character-aware decoder for translation into morphologically rich languages. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 244–255, Dublin, Ireland. European Association for Machine Translation.

Reynolds, L. and McDonell, K. (2021). Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Shliazhko, O., Fenogenova, A., Tikhonova, M., Mikhailov, V., Kozlova, A., and Shavrina, T. (2022). mGPT: Few-shot learners go multilingual. *arXiv preprint arXiv:2204.07580*.

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2021). mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Yang, Y., Eriguchi, A., Muzio, A., Tadepalli, P., Lee, S., and Hassan, H. (2021). Improving multilingual translation by representation and gradient regularization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7266–7279.