DEEP: DEnoising Entity Pre-training for Neural Machine Translation

Anonymous ACL submission

Abstract

It has been shown that machine translation models usually generate poor translations for named entities that are infrequent in the training corpus. Earlier named entity translation methods mainly focus on phonetic transliteration, which ignores the sentence context for translation and is limited in domain and language coverage. To address this limitation, we propose DEEP, a **DE**noising **E**ntity **P**re-training method that leverages large amounts of monolingual data and a knowledge base to improve named entity translation accuracy within sentences. Besides, we investigate a multi-task learning strategy that finetunes a pre-trained neural machine translation model on both entity-augmented monolingual data and parallel data to further improve entity translation. Experimental results on three language pairs demonstrate that DEEP results in significant improvements over strong denoising auto-encoding baselines, with a gain of up to 1.3 BLEU and up to 9.2 entity accuracy points for English-Russian translation.

1 Introduction

Proper translation of named entities is critically important for accurately conveying the content of text in a number of domains, such as news or encyclopedic text (Knight and Graehl, 1998; Al-Onaizan and Knight, 2002a,b). In addition, a growing number of new named entities (e.g., person, location) appear every day, therefore many of these entities may not exist in the parallel data traditionally used to train MT systems. As a result, even state-of-the-art MT systems struggle with entity translation. For example, Laubli et al. (2020) note that a Chinese-English news translation system that had allegedly reached human parity still lagged far behind human translators on entity translations, and this issue will be further exacerbated in the cross-domain transfer settings or in the case of emerging entities.

Because of this, there have been a number of methods proposed specifically to address the problem of translating entities. As noted by Liu (2015), earlier studies on named entity translation largely focused on rule-based methods (Wan and Verspoor, 1998), statistical alignment methods (Huang et al., 2003, 2004) and Web mining methods (Huang et al., 2005; Wu and Chang, 2007; Yang et al., 2009). However, these methods have two main issues. First, as they generally translate a single named entity without any context in a sentence, it makes it difficult to resolve ambiguity in entities using context. In addition, the translation of entities is often performed in a two-step process of entity recognition then translation, which complicates the translation pipeline and can result in cascading errors (Huang et al., 2003, 2004; Chen et al., 2013).

In this paper, we focus on a simple yet effective method that improves named entity translation within context. Specifically, we do so by devising a data augmentation method that leverages two data sources: monolingual data from the target language and entity information from a knowledge base (KB). Our method also adopts a procedure of pre-training and finetuning neural machine translation (NMT) models that is used by many recent works (Luong and Manning, 2015; Neubig and Hu, 2018; Song et al., 2019; Liu et al., 2020). In particular, pre-training methods that use monolingual data to improve translation for low-resource and medium-resource languages mainly rely on a denoising auto-encoding objective that attempt to reconstruct parts of text (Song et al., 2019) or the whole sentences (Liu et al., 2020) from noised input sentences without particularly distinguishing named entities and other functional words in the sentences. In contrast, our method exploits an entity linker to identify entity spans in the monolingual sentences and link them to a KB (such as Wikidata (Vrandečić and Krötzsch, 2014)) that contains multilingual translations of these entities. We then generate noised sentences by replacing the extracted

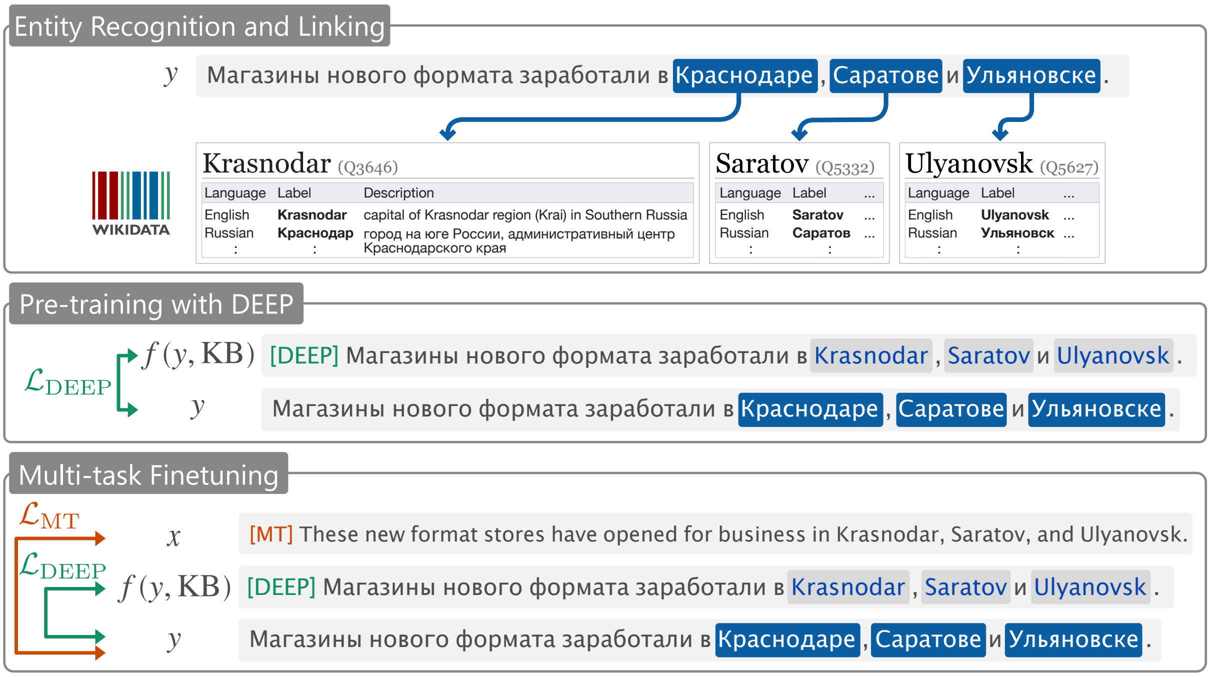

Entity Recognition and Linking
y
Магазины нового формата заработали в Краснодаре , Саратове и Ульяновске .
WIKIDATA
Krasnodar (Q3646)
Language Label Description
English Krasnodar capital of Krasnodar region (Krai) in Southern Russia
Russian Краснодар город на юге России, административный центр Краснодарского края
Saratov (Q5332)
Language Label ...
English Saratov ...
Russian Саратов ...
Ulyanovsk (Q5627)
Language Label ...
English Ulyanovsk ...
Russian Ульяновск ...
Pre-training with DEEP
$\mathcal{L}_{\text{DEEP}}$
$f(y, \text{KB})$
[DEEP] Магазины нового формата заработали в Krasnodar , Saratov и Ulyanovsk .
y
Магазины нового формата заработали в Краснодаре , Саратове и Ульяновске .
Multi-task Finetuning
$\mathcal{L}_{\text{MT}}$
x
[MT] These new format stores have opened for business in Krasnodar, Saratov, and Ulyanovsk.
$\mathcal{L}_{\text{DEEP}}$
$f(y, \text{KB})$
[DEEP] Магазины нового формата заработали в Krasnodar , Saratov и Ulyanovsk .
y
Магазины нового формата заработали в Краснодаре , Саратове и Ульяновске .

Figure 1: General workflow of our method. Entities in a sentence is extracted and linked to Wikidata, which includes their translations in many languages. DEEP uses the noise function $f(y, \text{KB})$ that replaces entities with the translations for pre-training. DEEP is also employed during finetuning in a multi-task learning manner.

entity spans with their translations in the knowledge base and pre-train our NMT models to reconstruct the original sentences from the noised sentences. To further improve the entity translation accuracy and avoid forgetting the knowledge learned from pre-training, we also examine a multi-task learning strategy that finetunes the NMT model using both the denoising task on the monolingual data and the translation task on the parallel data.

In the experiments on English-Russian, English-Ukrainian and English-Nepali translations, DEEP outperforms the strong denoising auto-encoding baseline with respect to entity translation accuracy, and obtains comparable or slightly better overall translation accuracy as measured by BLEU. A fine-grained analysis shows that our multi-task finetuning strategy improves the translation accuracy of the entities that do not exist in the finetuning data.

2 Denoising Auto-Encoding (DAE)

Given a set of monolingual text segments for pre-training, i.e., $y \in \mathcal{D}_Y$, a sequence-to-sequence denoising auto-encoder is pre-trained to reconstruct a text segment y from its noised version corrupted by a noise function $g(\cdot)$. Formally, the DAE objective is defined as follows:

$$\mathcal{L}_{\text{DAE}}(\mathcal{D}_Y, \theta) = \sum_{y \in \mathcal{D}_Y} \log P(y \mid g(y); \theta), \qquad (1)$$

where θ denotes the model's learning parameters. For notation simplicity, we drop θ in the rest of the sections. This formulation encompasses several different previous works in data augmentation for MT, such as monolingual data copying (Currey et al., 2017), where $g(\cdot)$ is the identity function, back translation (Sennrich et al., 2016), where $g(\cdot)$ is a backwards translation model, as well as heuristic noise functions (Song et al., 2019; Lewis et al., 2020; Liu et al., 2020) that randomly sample noise according to manually devised heuristics.

In particular, as our baseline we focus on the mBART method (Liu et al., 2020), a recently popular method with two type of heuristic noise functions being used sequentially on each text segment. The first noise function randomly masks spans of text in each sentence. Specifically, a span length is first randomly sampled from a Poisson distribution ($\lambda = 0.35$) and the beginning location for a span in y is also randomly sampled. The selected span of text is replaced by a mask token. This process repeats until 35% of words in the sentence are masked. The second noise function is to permute the sentence order in each text segment with a probability.

3 Denoising Entity Pre-training

Our method adopts a procedure of pre-training and finetuning for neural machine translation. First, we apply an entity linker to identify entities in a monolingual corpus and link them to a knowledge base (§3.1). We then utilize entity translations in the knowledge base to create noisy code-switched data for pre-training (§3.2). Finally, we examine a multi-task learning strategy to further improve the translation of low-frequency entities (§3.3).

3.1 Entity Recognition and Linking

The goal of this part is to identify entities in each monolingual segment and obtain their translations. To this end, we use Wikidata (Vrandečić and Krötzsch, 2014) a public multilingual knowledge base that covers 94M entities.[1] Each entity is represented in surface forms from different languages in which a Wikipedia article exists. Therefore, linking an entity mention t in a target-language segment y to an entity e in Wikidata allows us to obtain the multilingual translations of the entity, that is,

$$\forall t \in y, \exists e \in \text{KB} : T_e = \text{surface}(e, \text{KB}), t \in T_e$$

where T_e denotes a set of multilingual surface forms of e. We can define the translate operation as: $s = \text{lookup}(T_e, X)$ which simply looks for the surface form of e in the source language X. Note that this strategy relies on the fact that translations in higher-resource languages are included in T_e, which we adopt by using English in our experiments. In general, however, T_e does not universally cover all the languages of interest. For entity recognition and linking, we use SLING (Ringgaard et al., 2017),[2] which builds an entity linker for arbitrary languages available in Wikipedia.

3.2 Entity-based Data Augmentation

After obtaining entity translations from the KB, we attempt to explicitly incorporate these translations into the monolingual sentences for pre-training. To do so, we design a entity-based noise function that takes in a sentence y and the KB, i.e., $f(y, \text{KB})$. First, we replace all detected entity spans in the sentence by their translations from the KB:

$$\text{replace}(y, \text{KB}) = \text{swap}(s, t, y), \ \forall t \in y \quad (2)$$

where the swap() function swaps occurrences of one entity span t in y with its translation s in the source language. For example, in the second box of Figure 1, the named entities "Краснодаре, Саратове and Ульяновске" in Russian are replaced by "Krasnodar, Saratov, and Ulyanovsk" in English. After the replacement, we create a noised code-switched segment which explicitly include the translations of named entities in the context of the target language. For some segments that contain fewer entities, their code-switched segments may be similar to them, which potentially results in a easier denoising task. Therefore, we further add noise to these code-switched segments. To do so, if the word count of the replaced entity spans is less than a fraction (35%) of the word count in the segment, we randomly mask the other non-entity words to ensure that about 35% of the words are either replaced or masked in the noised segment. Finally, we follow Liu et al. (2020) to randomly permute the sentence order in y. We then train a sequence-to-sequence model to reconstruct the original sentence y from its noised code-switched sentence as follows:

$$\mathcal{L}_{\text{DEEP}}(\mathcal{D}_Y, \text{KB}) = \sum_{y \in \mathcal{D}_Y} \log P(y \mid f(y, \text{KB}))$$

3.3 Multi-task Finetuning

After pre-training, we continue finetuning the pre-trained model on a parallel corpus $(x, y) \in \mathcal{D}_{XY}$ for machine translation.

$$\mathcal{L}_{\text{MT}}(\mathcal{D}_{XY}) = \sum_{(x,y) \in \mathcal{D}_{XY}} \log P(y \mid x) \quad (3)$$

To avoid forgetting the entity information learned from the pre-training stage, we examine a mutli-task learning strategy to train the model by both the pre-training objective on the monolingual data and the translation objective on the parallel data. Since monolingual segments are longer text sequences than sentences in $\mathcal{D}_{XY}$ and the size of $\mathcal{D}_Y$ is usually larger than that of $\mathcal{D}_{XY}$, simply concatenating both data for multi-task finetuning leads to bias toward denoising longer sequences rather than actually translating sentences. To balance the two tasks, in each epoch we randomly sample a subset of monolingual segments $\mathcal{D}'_Y$ from $\mathcal{D}_Y$, where the total subword count of $\mathcal{D}'_Y$ equals to that of $\mathcal{D}_{XY}$, i.e., $\sum_{y \in \mathcal{D}'_Y} |y| = \sum_{(x,y) \in \mathcal{D}_{XY}} \max(|x|, |y|)$. We then examine the multitask finetuning as follows:

$$\mathcal{L}_{\text{Multi-task}} = \mathcal{L}_{\text{MT}}(\mathcal{D}_{XY}) + \mathcal{L}_{\text{Pre-train}}(\mathcal{D}'_Y) \quad (4)$$

where the pre-training objective $\mathcal{L}_{\text{Pre-train}}$ is either DAE or DEEP with DEEP having an additional input of a knowledge base. Notice that with the sampling strategy for the monolingual data, we double the batch size in the multi-task finetuning setting

[1] June 14, 2021. Creative Commons CC0 License

[2] https://github.com/google/sling, Apache-2.0 License

Lang.	Token	Para.	Entity		
			Type	Count	N
Ru	775M	1.8M	1.4M	337M	123
Uk	315M	654K	524K	140M	149
Ne	19M	26K	17K	2M	34

Table 1: Statistics of Wikipedia corpora in Russian (Ru), Ukrainian (Uk) and Nepali (Ne) for pre-training. N denotes the average subword count of entity spans in a sequence of 512 subwords.

Lang.	Train	Dev	Test	Coverage (F)		Coverage (T)	
				Type	Count	Type	Count
En-Ru	235K	3.0K	3.0K	88%	94%	88%	91%
En-Uk	200K	2.3K	2.5K	87%	94%	91%	94%
En-Ne	563K	2.6K	2.8K	35%	25%	44%	27%

Table 2: Statistics of the parallel train/dev/test data for finetuning. Coverage (F/T) represent the percentage of entity types and counts in the **F**inetuning (**T**est) data that are covered by the pre-training data.

with respect to that in the single-task finetuning setting. Therefore, we make sure that the models are finetuned on the same amount of parallel data in both the single-task and multi-task settings, and the gains from the mutlitask setting sorely come from the additional task on the monolingual data.

To distinguish the tasks during finetuning, we replace the start token (`[BOS]`) in a source sentence or a noised segment by the corresponding task tokens for the translation or denoising task (`[MT]`, `[DAE]` or `[DEEP]`). We initialize these task embeddings by the start token embedding and append them to the word embedding matrix of the encoder.

4 Experimental Setting

Pre-training Data: We conduct our experiments on three language pairs: English-Russian, English-Ukrainian and English-Nepali. We use Wikipedia articles as the monolingual data for pre-training and report the data statistics in Table 1. We tokenize the text using the same sentencepiece model as Liu et al. (2020), and train on sequences of 512 subwords.

Finetuning & Test Data: We use the news commentary data from the English-Russian translation task in WMT18 (Specia et al., 2018) for finetuning and evaluate the performance on the WMT18 test data from the news domain. For English-Ukrainian, we use the TED Talk transcripts from July 2020 in the OPUS repository (Tiedemann, 2012) for finetuning and testing. For English-Nepali translation, we use the FLORES dataset in Guzmán et al. (2019) and follow the paper's setting to finetune on parallel data in the OPUS repository. Table 2 shows the data statistics of the parallel data for finetuning. Notice that from the last four columns of Table 2, the entities in the pre-training data cover at least 87% of the entity types and 91% of the entity counts in both finetuning and test data except the En-Ne pair.

Architecture: We use a standard sequence-to-sequence Transformer model (Vaswani et al., 2017) with 12 layers each for the encoder and decoder. We use a hidden unit size of 512 and 12 attention heads. Following Liu et al. (2020), we add an additional layer-normalization layer on top of both the encoder and decoder to stabilize training at FP16 precision. We use the same sentencepiece model and the vocabulary from Liu et al. (2020).

Methods in Comparison: We compare methods in the single task and multi-task setting as follows:

- **Random → MT**: We include a comparison with a randomly initialized model without pre-training and finetune the model for each translation task.
- **DAE → MT**: We pre-train a model by DAE using the two noise functions in Liu et al. (2020) and finetune the model for each translation task.
- **DEEP → MT**: We pre-train a model using our proposed DEEP objective and finetune the model on the translation task.
- **DAE → DAE+MT**: We pre-train a model by the DAE objective and finetune the model for both the DAE task and translation task.
- **DEEP → DEEP+MT**: We pre-train a model by the DEEP objective and finetune the model for both the DEEP task and translation task.

Learning & Decoding: We pre-train all models for 50K steps using the default parameters in Liu et al. (2020) except that we use a smaller batch of 64 text segments, each of which has 512 subwords. We use Adam (ϵ=1e-6, β_2=0.98) and a polynomial learning rate decay scheduling with a maximum step at 500K. All models are pre-trained on one TPUv3 (128GB) for about 12 hours for 50K steps.[3] We apply the noise function on the monolingual data on the fly for each epoch, and this takes only a few minutes by multiprocessing in `Fairseq` (Ott et al., 2019). We then reset the learning rate scheduler and finetune our pre-trained models on the MT

[3] As we show in Figure 4, models pre-trained for 50K steps provide a reasonably good initialization.

parallel data for 40K steps. Single-task (multi-task) finetuning takes about 16 (32) hours on 2 RTX 3090 GPUs. We set the maximum number of tokens in each batch to 65,536 in the single task setting and double the batch size in the multi-task setting to ensure that models in both settings are trained on an equal amount of parallel data, and thus any performance gain can only be attributed to monolingual data during finetuning. We use 2,500 warm-up steps to reach a maximum learning rate of 3e-5, and use 0.3 dropout and 0.2 label smoothing. After training, we use beam search with a beam size of 5 and report the results in sacreBLEU (Post, 2018) following the same evaluation in Liu et al. (2020).

5 Discussion

5.1 Corpus-level Evaluation

In Table 3, we compare all methods in terms of BLEU (Papineni et al., 2002) and chrF (Popović, 2015) on the test data for three language pairs. First, we find that all pre-training methods significantly outperform the random baseline. In particular, our DEEP method obtains a gain of 3.5 BLEU points in the single task setting for the low-resource En-Ne translation. Second, we compute statistical significance of the BLEU and chrF scores with bootstrap resampling (Koehn, 2004), and we observe significant improvements with the multi-task finetuning strategy over the single-task finetuning for En-Ru and En-Ne. Our DEEP method outperforms the DAE method for En-Ru translation by 1.3 BLEU points in the multi-task setting. It is also worth noting that DEEP obtains higher BLEU points than DAE at the beginning of the multi-task finetuning process, however the gap between both methods decreases as the finetuning proceeds for longer steps (See Appendix A). One possible reason is that models trained by DEEP benefit from the entity translations in the pre-training data and obtain a good initialization for translation at the beginning of the finetuning stage. As the multitask finetuning proceeds, the models trained by both DAE and DEEP rely more on the translation task than the denoising task for translating a whole sentence. Thus the nuance of the entity translations might not be clearly evaluated according to BLEU or chrF.

5.2 Entity Translation Accuracy

Since corpus-level metrics like BLEU or chrF might not necessarily reveal the subtlety of named entity translations, in the section we perform a fine-grained evaluation by the entity translation accuracy which counts the proportion of entities correctly translated in the hypotheses. Specifically, we first use SLING to extract entities for each pair of a reference and a hypothesis. We then count the translation accuracy of an entity as the proportion of correctly mentioning the right entity in the hypotheses, followed by macro-averaging to obtain the average entity translation accuracy. We also show the accuracy scores in Table 3. First, our method in both single- and multi-task settings significantly outperformed the other baselines. In particular, the gains from DEEP are much clear for the En-Uk and En-Ru translations. One possible reason is that Russian or Ukrainian entities extracted from the pre-training data have a relatively higher coverage of the entities in both the finetuning and test data as reported in Table 2. However, SLING might not detect as many entities in Nepali as in the other languages. We believe that future advances on entity linking in low-resource languages could potentially improve the performance of DEEP further. We leave this as our future work.

5.3 Fine-grained Analysis on Entity Translation Accuracy

In this section, we further analyze the effect on different categories of entities using our method.

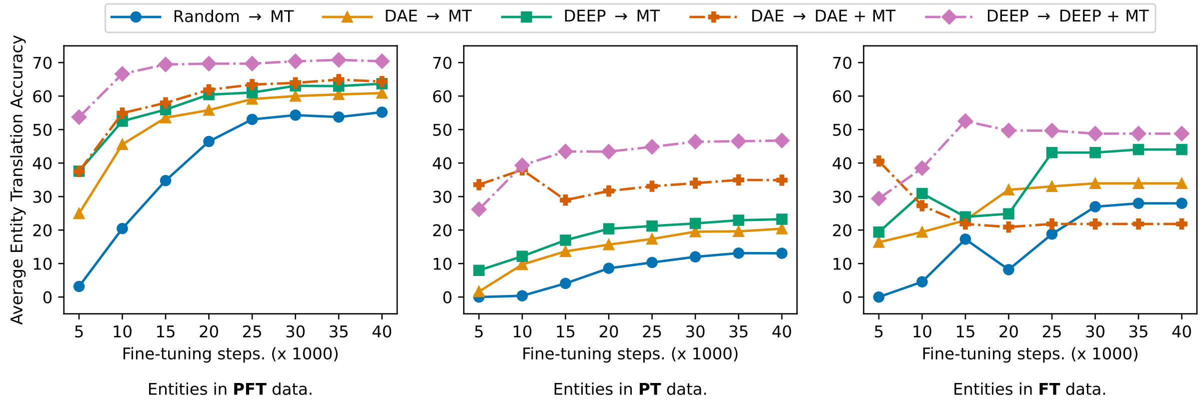
Performance of Entity Groups over Finetuning: The model is exposed to some entities more often than others at different stages: pre-training, finetuning and testing, which raises a question: *how is the entity translation affected by the exposure during each stage?* To answer this question, we divide the entities appearing in the test data into three groups:

- **PFT**: entities appearing in the pre-training, finetuning, and test data.
- **PT**: entities only in the pre-training and test data.
- **FT**: entities only in the finetuning and test data.

We show the English-to-Russian entity translation accuracy for each group over finetuning steps in Figure 2. Overall, accuracies are higher for the entities in the finetuning data (**PFT**, **FT**), which is due to the exposure to the finetuning data. Our proposed method consistently outperformed baseline counterparts in both single- and multi-task settings. The differences in accuracy are particularly large at earlier finetuning steps, which indicates the utility of our method in lower-resource settings with little finetuning data. The effect of multi-task finetuning is most notable for entities in **PT**. Multi-task

Pre-train → Finetune	BLEU			chrF			Entity Translation Acc.		
	En-Uk	En-Ru	En-Ne	En-Uk	En-Ru	En-Ne	En-Uk	En-Ru	En-Ne
Random → MT	17.1	15.0	7.7	37.0	36.8	24.3	49.5	31.1	20.9
DAE → MT	19.5	18.5	10.5	**39.2**	40.4	26.8	56.7	37.7	26.0
DEEP → MT	19.4	18.5	11.2*	**39.2**	40.7*	27.7*	57.7	40.6*	**28.6***
DAE → DAE+MT	19.4	18.5	11.2	39.1	41.0	27.8	58.8	47.2	27.9
DEEP → DEEP+MT	**19.7**	**19.6***	**11.5**	39.1	**42.4***	**28.2***	**61.9***	**56.4***	28.3

Table 3: BLEU, Entity translation accuracy, and chrF in single- and multi-task settings. Largest numbers in each column are bold-faced. * indicates statistical significance of DEEP with $p < 0.05$ to DAE in the respective settings.



Entities in **PFT** data.

Entities in **PT** data.

Entities in **FT** data.

Figure 2: Entity translation accuracy scores in different entity sets for Russian. **PFT**, **PT**, **FT** correspond to entities appearing in (i) pre-training, finetuning and test data, (ii) pre-training and test data (iii) finetuning and test data.

finetuning continuously exposes the model to the pre-training data, which prevents the model from forgetting the learned entity translations from **PT**.

Performance according to Entity Frequency: We further analyze the entity translation accuracy scores using entity frequencies in each group introduced above. This provides a more fine-grained perspective on *how frequent or rare entities are translated*. To do so, we take Russian hypotheses from a checkpoint with 40K steps of finetuning, bin the set of entities in three data (*i.e.* **PFT**, **PT**, **FT**) according to frequencies in each of the data. We then calculate the entity translation accuracy within each bin by comparing them against reference entities in the respective sentences. Figure 3 shows the accuracy gain of each pre-training methodologies from **Random → MT** (*i.e.* no pre-training) on test data, grouped by the entity frequency bins in pre-training and finetuning data. Note that leftmost column and the bottom row represent **PT**, **FT**, respectively. As observed earlier, the proposed method improves more over most frequency bins, with greater differences on entities that are less frequent in finetuning data. This tendency is observed more significantly for the multi-task variant (**DEEP → DEEP + MT**), where the gains are mostly from entities that never appeared in finetuning data (*i.e.* leftmost column). Multi-task learning with DEEP therefore prevents the model from forgetting the entity translations learned at pre-training time. Analytical results on Ukrainian and Nepali are in Appendix B.

5.4 Optimization Effects on DEEP

Finetuning Data Size vs Entity Translation: While DEEP primarily focuses on a low-resource setting, the evaluation with more resources can highlight potential use in broader scenarios. To this end, we expand the finetuning data for English-Russian translation with an additional 4 million sentence pairs from ParaCrawl (Bañón et al., 2020), a parallel data collected from web pages. Although web pages might contain news text, the ParaCrawl data cover more general domains. We finetune models on the combined data and evaluate with BLEU and entity translation accuracy. Table 4 shows the comparisons across different finetuning data sizes. When the model is initialized with pre-training methods, we observed decreased BLEU points and the increased entity translation accuracy scores. This is partly due to the discrepancy of domains between our finetuning data (news) and ParaCrawl. Regardless, DEEP is consistently equal to or better than DAE in all tested settings.

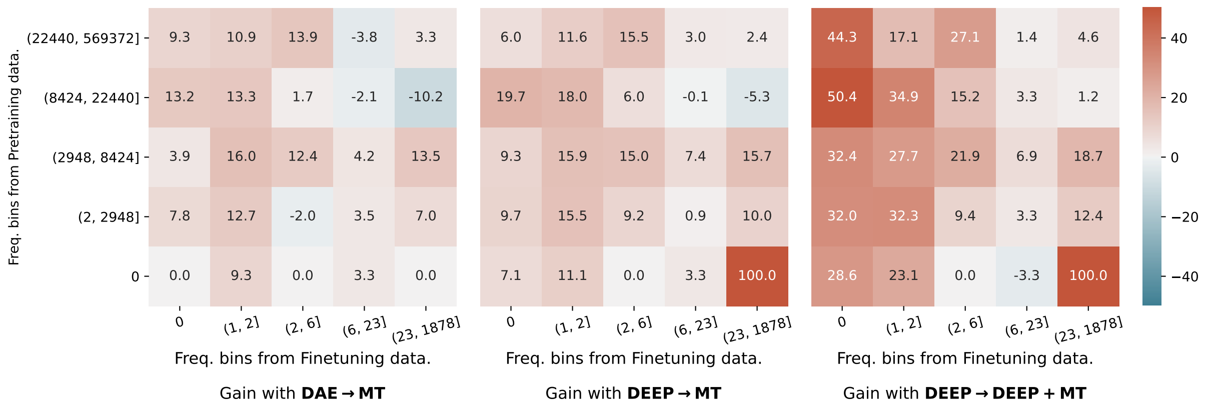


Figure 3: Gain from Random → MT in entity translation accuracy for each model.

Methods	0.24M		4.25M	
	BLEU	Acc.	BLEU	Acc.
Random → MT	15.0	31.1	15.7	39.4
DAE → MT	18.5	37.7	16.3	53.7
DEEP → MT	18.5	40.6	17.2	53.9

Table 4: Model comparisons across different finetuning data sizes. The results on the right are obtained after finetuning on the combined news commentary and ParaCrawl data.

Pre-training Steps vs Entity Translation: Since DEEP leverages entity-augmented monolingual data, the model trained by DEEP revisits more entities in different context as the pre-training steps increase. To analyze the efficiency of learning entity translations during pre-training, we focus on the question: *how many pre-training steps are needed for named entity translation?* To do so, we take the saved checkpoints trained by DEEP from various pre-training steps, and apply the single-task finetuning strategy on the checkpoints for another 40K steps. We plot the entity translation accuracy and BLEU on the test data in Figure 4. We find that the checkpoint at 25K steps has already achieved a comparable entity translation accuracy with respect to the checkpoint at 150K steps. This shows that DEEP is efficient to learn the entity translations as early as in 25K steps. Besides, both the BLEU and entity translation accuracy keep improving as the pre-training steps increase to 200K steps.

5.5 Qualitative Analysis

In this section, we select two examples that contain entities appearing only in the pre-training and testing data. The first example contains three location names. We find that the model trained by the single-task DAE predicts the wrong places which provide the wrong information in the translated sentence. In addition, the model trained by the multitask DAE just copies the English named entities (i.e., “Krasnodar”, “Saratov” and “Ulyanovsk”) to the target sentence without actual translation. In contrast, our method predicts the correct translation for “Krasnodar” in both single-task and multi-task setting, while the multi-task DEEP translates all entities correctly. In the second example, although our method in the single-task setting predicts wrong for all the entities, the model generates partially correct translations such as “Барнале” for “Барнауле” and “Красно @-@ Молгскиском” for “Красноармейском”. Notice that DEEP in the multi-task setting translates the correct entities “asphalt” and “Krasnoarmeyskiy” which convey the key information in this sentence. In contrast, the translation produced by the multi-task DAE method literally means “Барнаул (Barnaul), новый (new) миф (myth) на (at) Krasnoarmey Prospekt, выращивающий (grow) Krasnoarmeski.”, which is incomprehensible due to the entity translation errors.

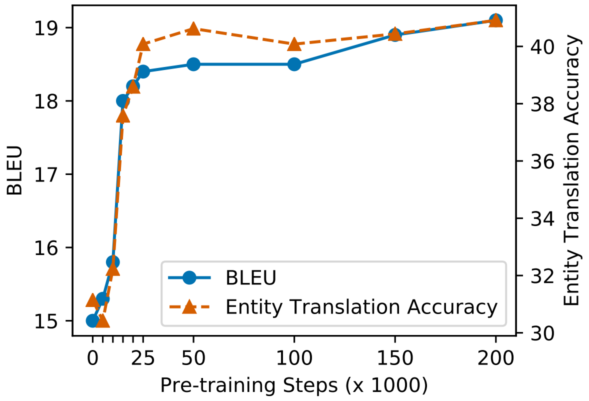


Figure 4: English-to-Russian BLEU and Entity translation accuracy scores after finetuning from variable pre-training steps. Finetuning is performed for 40K steps.

Src:	These new format stores have opened for business in **Krasnodar**, **Saratov**, and **Ulyanovsk**.
Ref:	Магазины нового формата заработали в Краснодаре, Саратове и Ульяновске.
①	Эти новые форматовые магазины открылись для бизнеса в **Анридаре**, **Кристофе** и **Куьянме**.
②	Эти новые формат @-@ магазины открылись для бизнеса в **Краснодаре**, **Сараабане** и в **Уругянском университете**.
③	Эти новые магазины форматов открылись для бизнеса в **Krasnodar**, **Saratov** и **Ulyanovsk**.
④	Эти новые форматные магазины открылись для бизнеса в **Краснодаре**, **Саратове** и **Ульяновске**.
Src:	In **Barnaul**, the new **asphalt** on **Krasnoarmeyskiy** Prospekt is being dug up
Ref:	В **Барнауле** вскрывают новый **асфальт** на проспекте **Красноармейском**
①	В **Барнауле** новое, как **разворачивающееся** на **железнополярном** Происсе, растет.
②	В **Барнале**, новое, как **разразилось** на **Красно @-@ Молгскиском** Просвещении, растет.
③	**Барнаул**, новый миф на **Krasnoarmey** Prospekt, выращивающий Krasnoarmeski.
④	В **Барнауле** новый **асфальт** на **Красноармейском** проспекте выращивание растет.

Table 5: Qualitative comparison among four pre-training methods on named entity translations. ①: DAE → MT, ②: DEEP → MT, ③: DAE → DAE+MT, ④: DEEP → DEEP+MT.

6 Related Work

Named Entity Translation has been extensively studied for decades (Arbabi et al., 1994; Knight and Graehl, 1998). Earlier studies focus on rule-based methods using phoneme or grapheme (Wan and Verspoor, 1998; Al-Onaizan and Knight, 2002b), statistical methods that align entities in parallel corpus (Huang et al., 2003, 2004; Zhang et al., 2005) and Web mining methods built on top of a search engine (Huang et al., 2005; Wu and Chang, 2007; Yang et al., 2009). Recently, Finch et al. (2016); Hadj Ameur et al. (2017); Grundkiewicz and Heafield (2018) used NMT to transliterate named entities *without any sentence context*. Another line of research (Ugawa et al., 2018; Li et al., 2018; Torregrosa et al., 2020; Modrzejewski et al., 2020; Zhou et al., 2020) only performs entity recognition and uses entity tags (e.g., person) which are not directly informative to the translation task, in contrast to the entity translations obtained by entity linking in our work. Besides, these methods modify model architecture to integrate entity tag embeddings or knowledge graph entity embeddings (Moussallem et al., 2019), which also require extracting entity information for both training and test data. In contrast, we focus on data augmentation methods to improve name entity translation *within context*, so our method is easily applicable to any architectures and test data without preprocessing.

Pre-training of Neural Machine Translation has been shown effective by many recent works (Conneau and Lample, 2019; Song et al., 2019; Liu et al., 2020; Lin et al., 2020), where different pre-training objectives are proposed to leverage monolingual data for translation. These methods adopt a denoising auto-encoding framework, which encompasses several different works in data augmentation on monolingual data for MT (Lambert et al., 2011; Currey et al., 2017; Sennrich et al., 2016; Hu et al., 2019). However, named entity translations during pre-training is under-explored. We fill this gap by integrating named entity recognition and linking to the pre-training of NMT. Moreover, while recent work shows that continue finetuning a pre-trained encoder with the pre-training objective improves language understanding tasks (Gururangan et al., 2020), this finetuning paradigm has not been explored for pre-training of a sequence-to-sequence model. Besides, previous works on multitask learning for MT focus on language modeling (Gulcehre et al., 2015; Zhang and Zong, 2016; Domhan and Hieber, 2017; Zhou et al., 2019), while we examine a multi-task finetuning strategy with an entity-based denoising task in this work and demonstrate substantial improvements for named entity translations.

7 Conclusion

In this paper, we propose an entity-based pre-training method for neural machine translation. Our method improves named entity translation accuracy as well as BLEU score over strong denoising auto-encoding baselines in both single-task and multi-task setting. Despite the effectiveness, several challenging questions remain open. First, recent works on integrating knowledge graphs (Zhao et al., 2020a,b) in NMT have shown promising results for translation. Our method links entities to a multilingual knowledge base which contains rich information of the entities such as entity description, relation links, alias. How to leverage these richer data sources to resolve entity ambiguity deserves further investigation. Second, finetuning pre-trained models on in-domain text data is a potential way to improve entity translations across domains.

References

Yaser Al-Onaizan and Kevin Knight. 2002a. Named entity translation. In *Proceedings of HLT 2002*, pages 122–124.

Yaser Al-Onaizan and Kevin Knight. 2002b. Translating named entities using monolingual and bilingual resources. In *Proceedings of ACL 2002*, pages 400–408.

M. Arbabi, S. M. Fischthal, V. C. Cheng, and E. Bart. 1994. Algorithms for arabic name transliteration. *IBM Journal of Research and Development*, 38(2):183–194.

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

Yufeng Chen, Chengqing Zong, and Keh-Yih Su. 2013. A joint model to identify and align bilingual named entities. *Computational Linguistics*, 39(2):229–266.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in NeurIPS*, volume 32.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of WMT 2017*, pages 148–156.

Tobias Domhan and Felix Hieber. 2017. Using target-side monolingual data for neural machine translation through multi-task learning. In *Proceedings of EMNLP 2017*, pages 1500–1505.

Andrew Finch, Lemao Liu, Xiaolin Wang, and Eiichiro Sumita. 2016. Target-bidirectional neural models for machine transliteration. In *Proceedings of the Sixth Named Entity Workshop*, pages 78–82.

Roman Grundkiewicz and Kenneth Heafield. 2018. Neural machine translation techniques for named entity transliteration. In *Proceedings of the Seventh Named Entities Workshop*, pages 89–94.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv:1503.03535*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of ACL 2020*, pages 8342–8360.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. In *Proceedings of EMNLP-IJCNLP 2019*, pages 6098–6111.

Mohamed Seghir Hadj Ameur, Farid Meziane, and Ahmed Guessoum. 2017. Arabic machine transliteration using an attention-based encoder-decoder model. *Procedia Computer Science*, 117:287–297.

Junjie Hu, Mengzhou Xia, Graham Neubig, and Jaime Carbonell. 2019. Domain adaptation of neural machine translation by lexicon induction. In *Proceedings of ACL 2019*, pages 2989–3001.

Fei Huang, Stephan Vogel, and Alex Waibel. 2003. Automatic extraction of named entity translingual equivalence based on multi-feature cost minimization. In *Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition*, pages 9–16.

Fei Huang, Stephan Vogel, and Alex Waibel. 2004. Improving named entity translation combining phonetic and semantic similarities. In *Proceedings of HLT-NAACL 2004*, pages 281–288.

Fei Huang, Ying Zhang, and Stephan Vogel. 2005. Mining key phrase translations from web corpora. In *Proceedings of HLT-EMNLP 2005*, pages 483–490.

Kevin Knight and Jonathan Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4):599–612.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP 2004*, pages 388–395.

Patrik Lambert, Holger Schwenk, Christophe Servan, and Sadaf Abdul-Rauf. 2011. Investigations on translation model adaptation using monolingual data. In *Proceedings of WMT 2011*, pages 284–293.

Samuel Laubli, Sheila Castilho, Graham Neubig, Rico Sennrich, Qinlan Shen, and Antonio Toral. 2020. A set of recommendations for assessing human–machine parity in language translation. *JAIR*, 67.

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 *Proceedings of ACL 2020*, pages 7871–7880.

Zhongwei Li, Xuancong Wang, Ai Ti Aw, Eng Siong Chng, and Haizhou Li. 2018. Named-entity tagging and domain adaptation for better customized translation. In *Proceedings of the Seventh Named Entities Workshop*, pages 41–46.

Zehui Lin, Xiao Pan, Mingxuan Wang, Xipeng Qiu, Jiangtao Feng, Hao Zhou, and Lei Li. 2020. Pre-training multilingual neural machine translation by leveraging alignment information. In *Proceedings of EMNLP 2020*, pages 2649–2663.

Ying Liu. 2015. The technical analyses of named entity translation. In *Proceedings of ISCI 2015*, pages 2028–2037.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *TACL*, 8:726–742.

Minh-Thang Luong and Christopher D. Manning. 2015. Stanford neural machine translation systems for spoken language domain. In *Proceedings of IWSLT 2015*.

Maciej Modrzejewski, Miriam Exel, Bianka Buschbeck, Thanh-Le Ha, and Alexander Waibel. 2020. Incorporating external annotation to improve named entity translation in NMT. In *Proceedings of EAMT*, pages 45–51, Lisboa, Portugal.

Diego Moussallem, Axel-Cyrille Ngonga Ngomo, Paul Buitelaar, and Mihael Arcan. 2019. Utilizing knowledge graphs for neural machine translation augmentation. In *Proceedings of K-CAP 2019*, pages 139–146.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Proceedings of EMNLP 2018*, pages 875–880.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL 2019 (Demo)*, pages 48–53.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of WMT 2015*, pages 392–395.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of WMT 2018*, pages 186–191.

Michael Ringgaard, Rahul Gupta, and Fernando CN Pereira. 2017. Sling: A framework for frame semantic parsing. *arXiv:1710.07032*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of ACL 2016*, pages 86–96.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of ICML 2019*, volume 97, pages 5926–5936.

Lucia Specia, Frédéric Blain, Varvara Logacheva, Ramón F. Astudillo, and André F. T. Martins. 2018. Findings of the WMT 2018 shared task on quality estimation. In *Proceedings of WMT 2018*, pages 689–709.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of LREC 2012*, pages 2214–2218.

Daniel Torregrosa, Nivranshu Pasricha, Maraim Masoud, Bharathi Raja Chakravarthi, Juan Alonso, Noe Casas, and Mihael Arcan. 2020. Aspects of terminological and named entity knowledge within rule-based machine translation models for under-resourced neural machine translation scenarios. *arXiv:2009.13398*.

Arata Ugawa, Akihiro Tamura, Takashi Ninomiya, Hiroya Takamura, and Manabu Okumura. 2018. Neural machine translation incorporating named entity. In *Proceedings of COLING 2018*, pages 3240–3250.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in NeurIPS*, volume 30.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Stephen Wan and Cornelia Maria Verspoor. 1998. Automatic English-Chinese name transliteration for development of multilingual resources. In *ACL-COLING 1998*, pages 1352–1356.

Jian-Cheng Wu and Jason S. Chang. 2007. Learning to find English to Chinese transliterations on the web. In *Proceedings of EMNLP-CoNLL 2007*, pages 996–1004.

Fan Yang, Jun Zhao, and Kang Liu. 2009. A Chinese-English organization name translation system using heuristic web mining and asymmetric alignment. In *Proceedings of ACL-IJCNLP 2009*, pages 387–395.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *Proceedings of EMNLP 2016*, pages 1535–1545.

Min Zhang, Haizhou Li, Jian Su, and Hendra Setiawan. 2005. A phrase-based context-dependent joint probability model for named entity translation. In *IJCNLP 2005*.

Yang Zhao, Lu Xiang, Junnan Zhu, Jiajun Zhang, Yu Zhou, and Chengqing Zong. 2020a. Knowledge graph enhanced neural machine translation via multi-task learning on sub-entity granularity. In *Proceedings of COLING 2020*, pages 4495–4505.

Yang Zhao, Jiajun Zhang, Yu Zhou, and Chengqing Zong. 2020b. Knowledge graphs enhanced neural machine translation. In *Proceedings of IJCAI 2020*, pages 4039–4045.

Leiying Zhou, Wenjie Lu, Jie Zhou, Kui Meng, and Gongshen Liu. 2020. Incorporating named entity information into neural machine translation. In *Proceedings of NLPCC 2020*, pages 391–402.

Shuyan Zhou, Xiangkai Zeng, Yingqi Zhou, Antonios Anastasopoulos, and Graham Neubig. 2019. Improving robustness of neural machine translation with multi-task learning. In *Proceedings of WMT 2019*, pages 565–571.

Appendix

A Finetuning BLEU Curves

We report BLEU score for three language pairs calculated from checkpoints at different finetuning steps in Figure 5. For all language pairs, all pre-training methods result in a significant increase in terms of BLEU throughout the finetuning in both single-task and multi-task setting. In particular, the differences in BLEU between DEEP and the other baselines are most significant at the beginning of the finetuning stage.

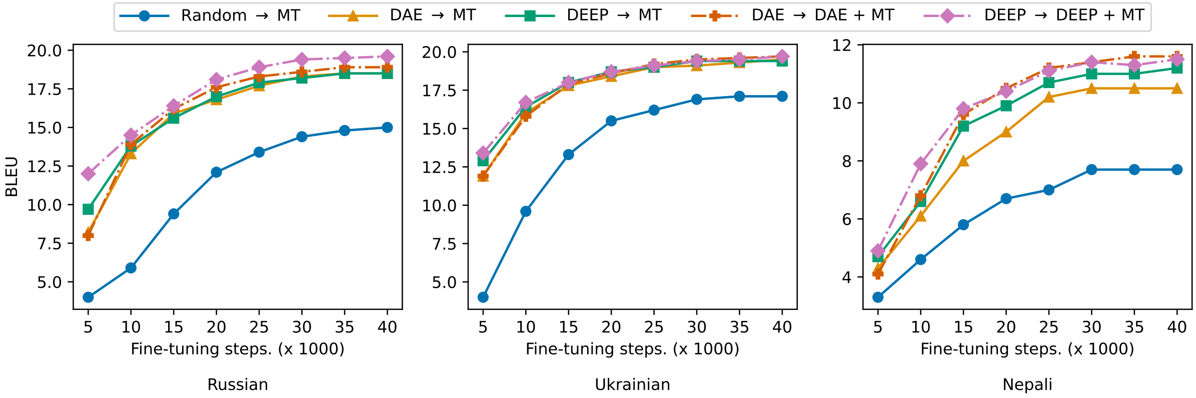


Figure 5: **BLEU** scores for 3 language pairs over various finetuning steps.

B Entity Translation Accuracy for other languages

We show the entity translation accuracy performance over various finetuning steps for Ukrainian and Nepali in Figure 6, 7, and show the gains of three pre-training methods over the random baseline with respect to the entity frequencies in Figure 8, 9. Empty cells in the heatmaps are due to no entities that meet the conditions in those cells.

Ukrainian: As seen in Figure 6, the general trend for the entity translation accuracy according to entity groups are similar to that of Russian. While DEEP achieves the highest accuracy in **FT**, the results for **FT** is less reliable due to a small sample size of entities in **FT**. In terms of the gain from **Random → MT** according to the entity frequency, we observe a consistent improvement of our multi-task DEEP on translating low-frequent entities in the finetuning data (See the left bottom of Figure 8).

Nepali: While outperforming at the beginning of finetuning, Figure 7 shows that **DEEP → DEEP+MT** eventually under-performed for translations of entities in **PFT** data. Moreover, the accuracy is considerably lower on entities in **PT**, which suggests that the degree of forgetting is much more conspicuous in Nepali. The gain from **Random → MT** with respect to the entity frequency exhibited a different trend from Russian and Ukrainian. Figure 9 shows the results. In the single-task setting, DEEP improve the translations of frequent entities appearing in both the pre-training and finetuning data. Despite the multi-task learning that introduces additional exposure to entities that are more frequent in the pre-training data, the largest gain comes from entities that are less frequent in the pre-training data but frequent in the finetuning data.

C Scientific Artifacts

In Table 6, we provide the detailed information about the scientific artifacts (e.g., data, code, tools) used in our paper. We have checked the data used in this work to make sure that we do not intentionally use private or sensitive information or offensive content for deriving the observations and conclusions from our work. Although WikiData may contain the name of some individual people (e.g., famous people that have Wikipedia webpages), we do not use their sensitive information in our analysis.

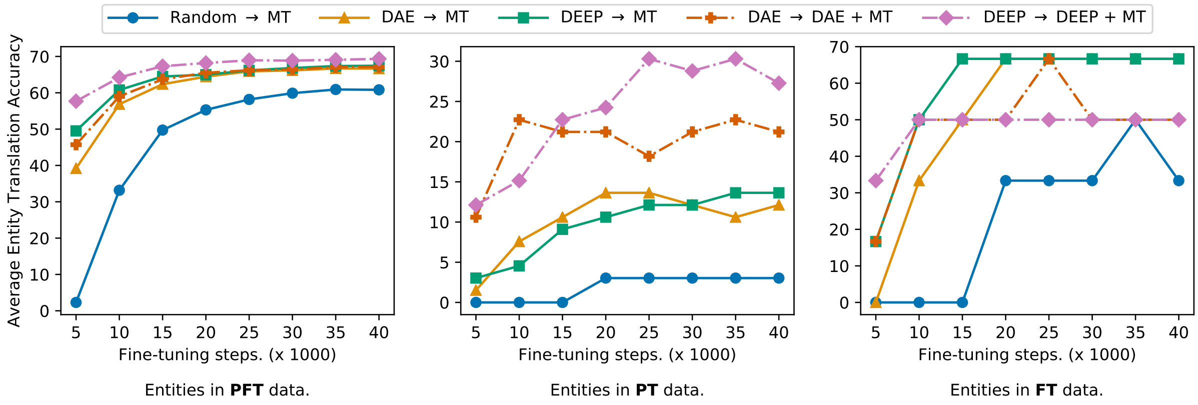


Figure 6: **Entity translation accuracy** aggregated over different entity sets for **Ukrainian**.

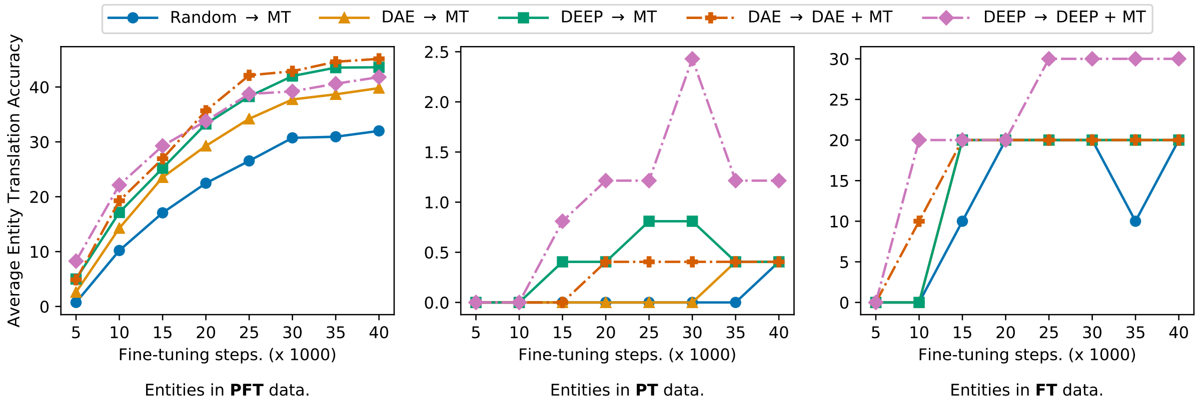


Figure 7: **Entity translation accuracy** aggregated over different entity sets for **Nepali**.

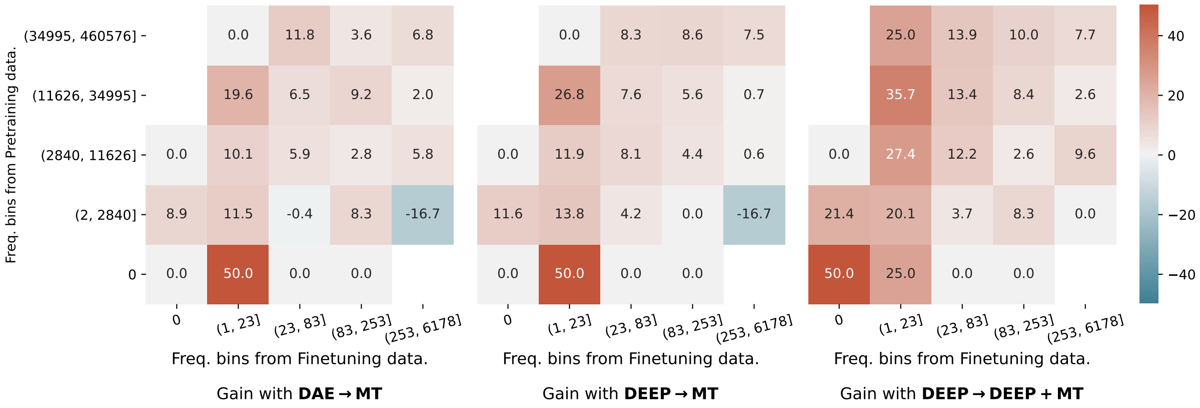
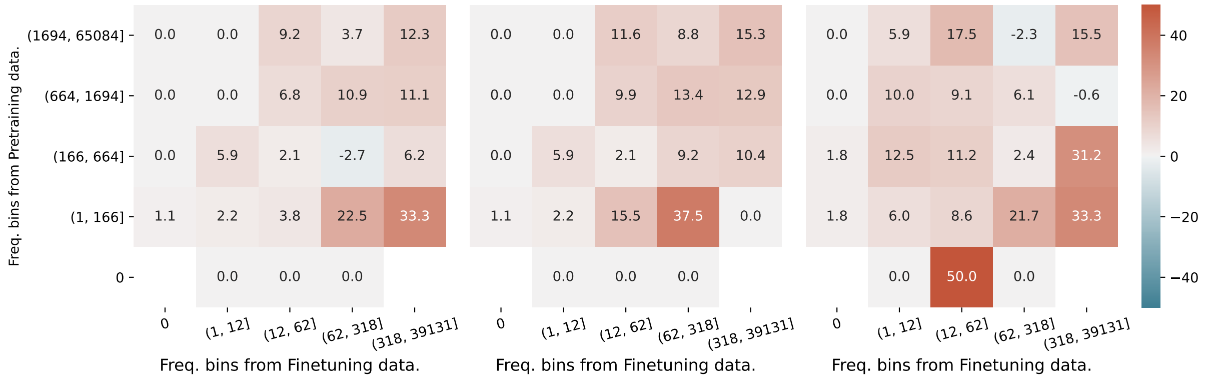


Figure 8: Gain from **Random → MT** in entity translation accuracy for **Ukrainian** for each model.



Gain with **DAE → MT**

Gain with **DEEP → MT**

Gain with **DEEP → DEEP + MT**

Figure 9: Gain from **Random** → **MT** in entity translation accuracy for **Nepali** for each model.

Artifact	License/Term	Documentation
WikiData (Vrandečić and Krötzsch, 2014)	Creative Commons CC0	This resource is a free knowledge base that supports various research and projectw.
Sling (Ringgaard et al., 2017)	Apache-2.0	This tool is intended to use for analyze WikiData and Wikipedia articles.
WMT18 En-Ru Data (Specia et al., 2018)	Open-sourced	This dataset is intended to be used for MT on news texts.
OPUS Data (Tiedemann, 2012)	Open-sourced	This data resource is intended to be used for MT.
FLORES Data (Guzmán et al., 2019)	CC-BY-SA-4.0 License	This dataset is intended to be used for low-resource MT.
Fairseq (Ott et al., 2019)	MIT License	This tool is intended to facilitate deep learning research.

Table 6: Detail information about scientific artifacts used in this paper.