Leveraging Word and Phrase Alignments for Multilingual Learning

Junjie Hu

CMU-LTI-21-012

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15123
www.lti.cs.cmu.edu

Thesis Committee:

Dr. Graham Neubig (Chair)	Carnegie Mellon University
Dr. Yulia Tsvetkov	Carnegie Mellon University
Dr. Zachary Lipton	Carnegie Mellon University
Dr. Kyunghyun Cho	New York University

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in Language and Information Technologies.

Keywords: natural language processing, multilingual learning, cross-lingual transfer learning, machine translation, domain adaptation, deep learning

Dedicated to my family
for their wholehearted love and support in all my endeavours.

Abstract

Recent years have witnessed impressive success in natural language processing (NLP) thanks to the advances of neural networks and the availability of large amounts of labeled data. However, many NLP systems predominately have focused on high-resource languages (e.g., English, Chinese) that have large, computationally accessible collections of labeled data for training. While the achievements on high-resource languages are exciting, there are more than 6,900 languages in the world and the majority of them have far fewer resources for training deep neural networks. In fact, it is often expensive, or sometimes infeasible, to collect labeled data written in all possible languages. As a result, this data scarcity issue limits the generalization of NLP systems in many multilingual scenarios. Moreover, as models may be used to process text from a wide range of domains (e.g., social media or medical articles), the data scarcity issue is further exacerbated by the domain shift between the training and test data.

In this thesis, with the goal of improving the generalization ability of NLP models to alleviate the aforementioned challenges, we exploit *word and phrase alignment* to train neural NLP models (e.g., neural machine translation or contextualized language models), and provide evaluation methods for examining the generalization capabilities of such models over diverse application scenarios. This thesis contains two parts. The first part explores *cross-lingual generalization for language understanding*. In particular, we examine the ability of pre-trained multilingual representations to transfer learned knowledge from a high-resource language to other languages. To this end, we first introduce a multi-task benchmark for evaluating the cross-lingual generalization capabilities of multilingual representations across 40 languages and 9 tasks. Second, we leverage word and sentence alignments from parallel data to improve the multilingual representations for language understanding tasks such as those included in our benchmark. The second part of the thesis is devoted to *leveraging alignment information for machine translation*, a popular and useful language generation task. In particular, we focus on learning to translate aligned words and phrases between two languages with fewer parallel sentences. To accomplish this goal, we exploit techniques to obtain aligned words and phrases from monolingual data, knowledge bases or crowdsourcing and use them to improve translation systems.

Acknowledgments

First and foremost, I am grateful to my advisors Graham Neubig and Jaime Carbonell who have provided me with constant supports as well as insightful discussions in my academic development at Carnegie Mellon University. In addition to being an excellent researcher, Graham is also the kindest and most supportive mentor that I have had during my academic career. He has a great sense of research taste, which also motivates me to conduct important research and think deeply about my research agenda. Jaime is one of the most influential researchers that I have met and a role model for my academic career. Both of them will have a long-standing influence on me beyond my PhD.

I would like to thank Kyunghyun Cho, Yulia Tsvetkov and Zachary Lipton for all the help they have provided to me during my Ph.D. study, including serving as my committee members, giving advice on my job search, and having insightful discussions about my thesis.

A huge thanks go to all of my other collaborators and friends throughout my Ph.D. study at CMU: Pengcheng Yin, Wei-cheng Chang, Yuexin Wu, Diyi Yang, Po-Yao Huang, Han Zhao, Yichong Xu, Zhengbao Jiang, Zi-Yi Dou, Mengzhou Xia, Antonios Anastasopoulos, Chunting Zhou, Xuezhe Ma, Xinyi Wang, Austin Matthews, Craig Stewart, Nikolai Vogler, Shruti Rijhwani, Paul Michel, Danish Pruthi, Zaid Sheikh, Wei Wei, Zi Yang, Zhilin Yang, Desai Fan, Shuxin Yao, Hector Liu, Zhiting Hu, Jean Oh, Andrej Risteski, Geoff Gordon, Anatole Gershman, Ruslan Salakhutdinov, and William Cohen.

In addition to my time at CMU, I also would like to thank my MPhil advisors Michael R. Lyu and Irwin King at the Chinese University of Hong Kong, and my collaborators during my internship at Google and Microsoft: Sebastian Ruder, Melvin Johnson, Orhan Firat, Aditya Siddhant, Yu Cheng, Zhe Gan, Jingjing Liu, Jianfeng Gao, Liu Yang, Xiaodong Liu, Yelong Shen.

Last but most importantly, I would like to thank my dad Saihao Hu, my mom Suhua Li, my elder sister Jiana Hu, and my lovely girlfriend Ming Jiang for all of your unconditional love and support during my Ph.D. journey. Needless to say, this thesis would not have been possible without your encouragement along the way. This thesis is dedicated to all of you.

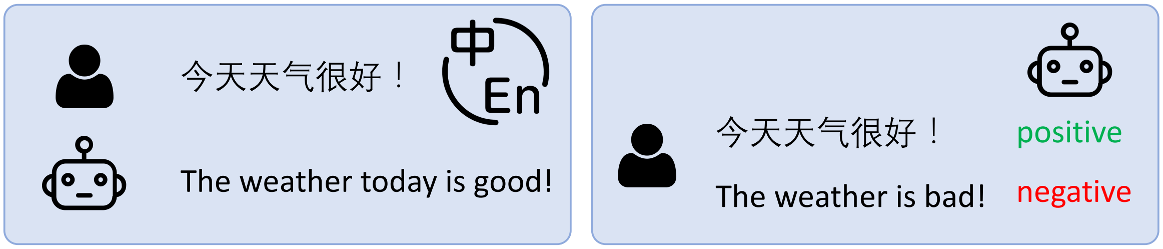
Contents

Chapter 1

Introduction

Over the past decade, the success of NLP systems has been mostly driven by deep neural network models and supervised learning approaches on a large amount of labeled data. For example, neural machine translation (NMT) [18] trained on billions of parallel sentences has become the de facto paradigm of many commercial translation systems such as Google's multilingual NMT system [3, 89] and Microsoft's NMT system [77]. Among these exciting NLP research developments, this thesis particularly focuses on multilingual learning approaches and investigates two main categories of multilingual NLP tasks – *machine translation* (MT) and *cross-lingual language understanding*. Machine translation is the task of translating text from a source language to another target language. The broad genre of cross-lingual language understanding includes a variety of sub-tasks that make predictions over words, phrases, or sentences written in different languages. Figure 1.1a gives an example of Chinese-to-English translation, and Figure 1.1b gives an example of cross-lingual sentiment analysis, a popular cross-lingual language understanding task that predicts whether an input sentence contains positive or negative opinions regarding a particular subject. We kindly refer the reader to the problem definition of both tasks in Section 2.1.

In NLP, most existing systems are English-based or developed for high-resource languages. Given this limitation, there is a pressing urgency to build systems that serve *all* of the human languages to overcome language barriers and enable universal information access for the world's citizens [3, 10, 150]. On the other hand, most existing NLP systems still require a large amount of labeled data for training in order to generalize to diverse unseen data at the test time, and even then generalization is far from certain. In practice, the distributional shift between the training and test data is ubiquitous, especially when NLP systems are deployed to deal with real text data written in different languages (e.g., Nepali or Swahili) or coming from diverse domains (e.g., social media or medical articles). This poses several challenges for the generalization ability of



(a) Chinese to English Translation

(b) Cross-lingual Sentiment Analysis

Figure 1.1: Examples of machine translation (left) and cross-lingual sentence classification (right)

neural network models as follows:

(1) NLP models suffer from data scarcity in multiple languages. One issue related to the discrepancy of data distributions at the training and testing stages is the languages represented in the text. There are about 6,900 living languages in the world [52], while a large majority of labeled datasets are created for English. First of all, it is not efficient to train and maintain separate models for individual languages. At the same time, building NLP systems for most of these languages is challenging due to a stark lack of data. Especially for low-resourced languages, we do not have enough labeled data to train neural models to achieve strong performance. Luckily many languages have similarities in syntax or vocabulary, and multilingual learning approaches that train on multiple languages while leveraging the shared structure of the input space have begun to show promise as ways to alleviate this data scarcity issue. Over the last few years, there has been a move towards *general-purpose multilingual representations* that are applicable to many tasks, both on the word level [13, 57, 125] or the full-sentence level [39, 47]. Despite the fact that such representations are intended to be general-purpose, evaluation of them has been often performed on a very restricted set of tasks and languages. On the other hand, such representations are often trained on large collections of multilingual raw text with the hope of implicitly learning which words and sentences correspond to each other across languages. As a result, training such representations requires huge amounts of computational power and large collections of raw text, which may not be easily accessible to many research groups.

(2) Neural machine translation models are highly susceptible to domain shift. Due to the intrinsic flexibility of natural language, a sentence in one language can be translated into many semantically equivalent sentences in another language. Some of these translations may be more preferable than others depending on the domain (e.g., medical article or news report) of the input text. In particular, it has also been noted that NMT models trained on corpora in a particular domain tend to perform poorly when translating sentences in a significantly different domain [35, 97]. For example, in highly sensitive scenarios such as translation of medical reports [35, 88], if a model is trained on a biased or out-of-distribution dataset that does not accurately represent the real use case, then deploying this model directly without any adaptations could result in skewed predictions, such as translating a medical term “wirkstoff aripiprazol” in German as “a formula” in English rather than the correct translation “substance aripiprazole”. This domain mismatch issue becomes even more problematic if we do not have many training sentences in the target domain. Learning how to translate these domain-specific words or phrases without many training sentences in the target domain remains a big issue for neural machine translation.

This thesis is concerned with solutions to the aforementioned data discrepancy issues that leverage *word and phrase alignment for multilingual learning* in both language understanding and generation. In the first part, we investigate the cross-lingual generalization capability of pre-trained multilingual models for a wide range of NLP tasks. We then leverage parallel sentences between two languages to learn a shared language embedding space that aligns multilingual text data based on their semantics at different granularities. In the second part, when parallel sentences in the target domain are not easily accessible, we investigate methods to extract word or phrase translations from monolingual sentences, knowledge bases, or crowdsourcing, and leverage these aligned words or phrases between two languages to improve the machine translation quality.

1.1 Main Contributions

In this section, we summarize the core contributions of this thesis.

Cross-lingual Generalization Benchmark. (Chapter 3) Much recent progress in applications of machine learning models to NLP has been driven by benchmarks that evaluate models across a wide variety of tasks. However, these broad-coverage benchmarks have been mostly limited to English, and despite an increasing interest in multilingual models, a benchmark that enables the comprehensive evaluation of such methods on a diverse range of languages and tasks is still missing. To this end, we introduce the Cross-lingual TRansfer Evaluation of Multilingual Encoders (XTREME) benchmark, a multi-task benchmark for evaluating the cross-lingual generalization capabilities of multilingual representations across 40 languages and 9 tasks. We demonstrate that while models tested on English reach human performance on many tasks, there is still a sizable gap in the performance of cross-lingually transferred models, particularly on syntactic and sentence retrieval tasks. There is also a wide spread of results across languages. We released the benchmark to encourage research on cross-lingual learning methods that transfer linguistic knowledge across a diverse and representative set of languages and tasks.

Cross-lingual Representation Learning for Language Understanding. (Chapter 4) One of the main experimental findings in the previous chapter is that pre-trained cross-lingual encoders such as mBERT [51] are effective at enabling transfer learning of NLP systems from high-resource languages to low-resource languages. This success comes despite the fact that there is no explicit objective to align the contextual embeddings of words/sentences with similar meanings across languages together in the same space. In this chapter, we present a new method for learning multilingual encoders, AMBER (**A**ligned **M**ultilingual **B**idirectional **E**ncode**R**). AMBER is trained on additional parallel data using two *explicit* alignment objectives that align the multilingual representations at different granularities. We conduct experiments on zero-shot cross-lingual transfer learning for different tasks including sequence tagging, sentence retrieval and sentence classification. Experimental results show that AMBER obtains gains of up to 1.1 average F1 score on sequence tagging and up to 27.3 average accuracy on retrieval over a state-of-the-art XLMR-large model which has 3.2x the parameters of AMBER.

Leveraging Word Alignments for Domain Adaptation of Machine Translation. (Chapter 5) In contrast to Chapter 4, we investigate methods to align words written in two languages *without* parallel sentences, and demonstrate its usage for domain adaptation of neural machine translation (NMT). It has been previously noted that NMT is very sensitive to domain shift. In this chapter, we argue that this is a dual effect of the highly lexicalized nature of NMT, resulting in failure for sentences with large numbers of unknown words, and lack of supervision for domain-specific words. To remedy this problem, we propose an unsupervised adaptation method that fine-tunes a pre-trained out-of-domain NMT model using a pseudo-in-domain corpus. Specifically, we perform lexicon induction to extract an in-domain lexicon, and construct a pseudo-parallel in-domain corpus by performing word-for-word back-translation of monolingual in-domain target sentences. In five domains over twenty pairwise adaptation settings and two model architectures, our method achieves consistent improvements without using any in-domain parallel sentences, improving up to 14 BLEU over unadapted models, and up to 2 BLEU over strong back-translation baselines.

Leveraging Aligned Entities for Machine Translation. (Chapter 6) The previous chapter has shown that machine translation models usually generate poor translations for words or phrases that are infrequent or even unseen in the training corpus. These infrequent words are often named entities that contain key information of the sentences. 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.

Leveraging Phrase Alignment from Crowdsourcing for Machine Translation. (Chapter 7) The previous two chapters have explored methods to extract aligned words or phrases from existing sources such as monolingual data or knowledge base without actively quantifying the importance of aligned words or phrases. In this chapter, we explore an active learning setting where we can actively select in-domain data for human translation, and gradually fine-tune a pre-trained out-of-domain NMT model on the newly translated data. Existing active learning methods for NMT usually select sentences based on uncertainty scores, but these methods require costly translation of full sentences even when only one or two key phrases within the sentence are informative. To address this limitation, we re-examine previous work from the phrase-based machine translation (PBMT) era that selected not full sentences, but rather individual phrases. However, while incorporating these phrases into PBMT systems was relatively simple, it is less trivial for NMT systems, which need to be trained on full sequences to capture larger structural properties of sentences unique to the new domain. To overcome these hurdles, we propose to select *both* full sentences and individual phrases from unlabelled data in the new domain for routing to human translators. In a German-English translation task, our active learning approach achieves consistent improvements over uncertainty-based sentence selection methods, improving up to 1.2 BLEU score over strong active learning baselines.

1.2 Thesis Outline

We begin by giving an overview of prior work in Chapter 2 and outline some general techniques used throughout the thesis. The next 5 chapters are divided into two parts. The first part (Chapter 3, 4) explores methods for zero-shot cross-lingual generalization of natural language understanding tasks. The second part (Chapter 5, 6, 7) shifts the focus towards machine translation – the typical language generation task between two languages – through the lens of translating in-domain words or phrases in an domain adaptation setting. Finally, we conclude this thesis and discuss future research directions in Chapter 8.

Chapter 2

Background

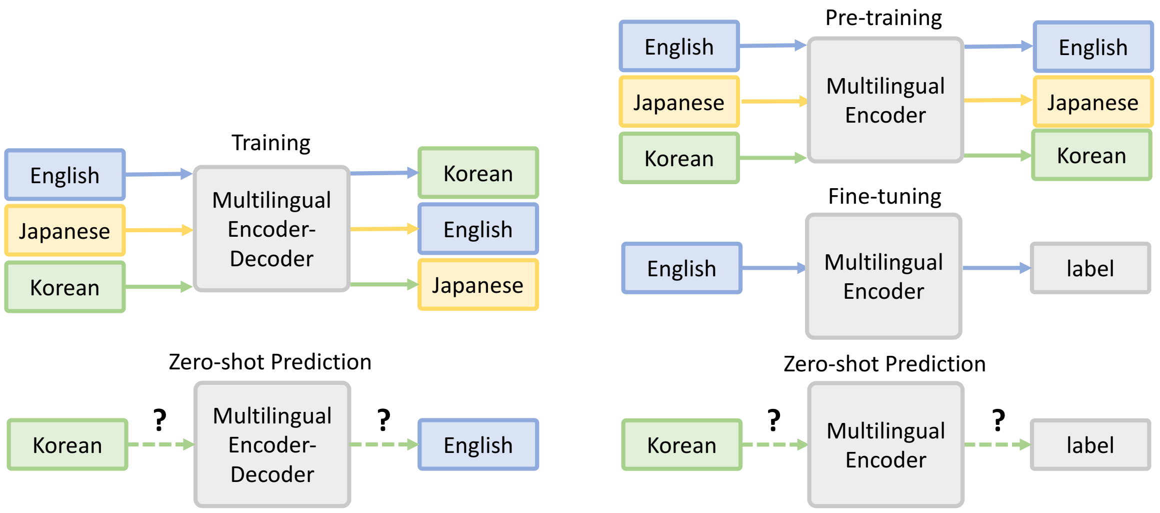
In this chapter, we provide some background knowledge about the multilingual learning problems in this thesis. Section 2.1 situates this thesis in the context of prior work. Section 2.2 introduces neural network models for representation multilingual text. Section 2.3 highlights several popular pre-training methods of neural network models.

2.1 Prior Work

We start with a brief history of machine translation research (Section 2.1.1) and point out related research on multilingual neural machine translation and domain adaptation of neural machine translation. We then highlight the close connection of multilingual neural machine translation to cross-lingual language understanding (Section 2.1.2).

2.1.1 Machine Translation

Earlier research on machine translation dates back to the 1950s. In 1954, A group of researchers at Georgetown University and IBM released a public demonstration of their Georgetown-IBM machine translation system, a rule-based system containing 6 grammar rules and 250 lexical items in its vocabulary. The concept of "interlingual machine translation" that seeks a "language independent representation" was later introduced by Richens [147]. Pioneering studies also include rule-based machine translation, dictionary-based machine translation, etc. Since the 1980s, statistical machine translation (SMT) [25, 176] had attracted more attention due both to methodological advances and less expensive computational power. We refer the reader to Koehn [96] for a detailed description of related methods. As the computational power such as GPU and TPU became more



(a) Zero-shot Translation

(b) Zero-shot Cross-lingual Language Understanding

Figure 2.1: Examples: (a) Training a multilingual encoder-decoder model on English-to-Korean, Japanese-to-English and Korean-to-Japanese parallel data, and performing zero-shot prediction for Korean-to-English translation. (b) Pre-training a multilingual encoder model on raw text in English, Korean and Japanese, fine-tuning the pre-trained multilingual encoder on English labeled data, and performing zero-shot prediction on Korean data.

.

accessible, neural machine translation (NMT) [34, 92, 161] was proposed to learn continuous language representations from a large number of parallel sentences between two languages. Typical NMT models adopt an encoder-decoder model architecture where the encoder summarizes the information of an input source sentence and the decoder generates a corresponding translation in the target language. Bahdanau et al. [18] later introduced the attention mechanism that models the pairwise dependency between the target and source words. Thanks to the attention mechanism, NMT has demonstrated impressive performance when trained on large-scale corpora [23] and became widely used in many commercial systems. Since the advent of neural machine translation, a large number of variants have been developed. We introduce the formal definition of machine translation and highlight the two threads of neural machine translation research closely related to this thesis below.

Problem Definition: Given an input sentence $x = [x_0, x_1, \ldots, x_N]$ in the source language, a machine translation model is trained to predict an output translation $y = [y_0, y_1, \ldots, y_M]$ in the target language. Neural machine translation adopts an encoder-decoder model architecture, where the encoder and decoder can be instantiated by a Transformer model or a recurrent neural network (RNN) model. Given a parallel corpus $\mathcal{D}_{XY}$ of (x, y) pairs, the NMT model is trained by maximum log-likelihood estimation (MLE) of the correct translation given the source input. This optimization problem is also equivalent to minimizing a cross-entropy loss over all training pairs as follows.

$$\mathcal{L}_{\text{MLE}} = - \sum_{(x,y)\sim\mathcal{D}_{XY}} \log P_\theta(y|x), \tag{2.1}$$

where θ denotes the model parameters. The training process usually splits the whole training data into mini-batches of sentence pairs and adopts stochastic gradient descent or its variants to iteratively process each mini-batch.

Multilingual Neural Machine Translation: One particular thread is to extend the traditional bilingual translation from one language to one other language to multilingual machine translation in which a neural network model is trained to translate between multiple languages, e.g., one-to-many [50], many-to-one [106], or many-to-many [59] translations. An early attempt in [50] modified an attention-based encoder-decoder model by adding a separate decoder for each target language and sharing the encoder for a single source language. In [121], multiple encoders and decoders are trained in a multi-task setting for many-to-many translations. Different from these methods that share only parts of the network, Google's multilingual machine translation system [89] proposed to build a shared vocabulary of subwords [157] across languages and train a single encoder-decoder model on large collections of parallel sentences written in multiple language pairs. Due to the shared model architecture across languages, the system demonstrated promising *zero-shot* results on translating between language pairs that it had never seen in the combination of the training data. Since then, in addition to sharing an encoder-decoder architecture for multilingual language generation tasks, similar ideas of sharing an encoder architecture across languages have also been applied to cross-lingual understanding tasks (e.g., sentence classification, sequence tagging). Figure 2.1 shows the comparison of zero-shot predictions by multilingual encoder-decoder models for translation and multilingual encoder models for cross-lingual understanding. Although prior work has examined the zero-shot translation of multilingual encoder-decoder models, zero-shot evaluation of multilingual encoders for language understanding has often been restricted to a disparate set of tasks and typologically similar languages. One of the targets of this thesis is to provide one step further towards understanding the *zero-shot cross-lingual generalization* of these pre-trained multilingual encoders on a variety of NLP tasks across a diverse set of languages (Chapter 3).

Domain Adaptation of Neural Machine Translation: The other thread of research is to enhance the domain robustness of neural machine translation models. It has been noted that NMT models trained on corpora in a particular domain tend to perform poorly when translating sentences in a significantly different domain [35, 97]. As noted by Chu and Wang [35], there are two important distinctions to make in domain adaptation methods for MT. The first is data requirements; *supervised* adaptation relies on small amounts of in-domain parallel data, and *unsupervised* adaptation has no such requirement. There is much work on *supervised* domain adaptation. Luong and Manning [120] propose training a model on an out-of-domain corpus and do fine-tuning with small sized in-domain parallel data to mitigate the domain shift problem. Instead of naively mixing out-of-domain and in-domain data, Britz et al. [24] circumvent the domain shift problem by learning domain discrimination and translation jointly. Joty et al. [90] and Wang et al. [172] address the domain adaptation problem by assigning higher weights to out-of-domain parallel sentences that are close to the in-domain corpus. Despite the effectiveness of these supervised adaptation methods, one of the targets of this thesis is to reduce the heavy reliance on in-domain parallel sentences for adaptation. In particular, we focus on translating infrequent words or phrases in the target domain by leveraging alignment information from monolingual in-domain sentences (Chapter 5), knowledge bases (Chapter 6) and crowdsourcing (Chapter 7). There is also a distinction between *model-based* and *data-based* methods. Model-based methods make explicit changes to the model architecture such as jointly learning domain discrimination and translation [24], interpolation of language modeling and translation [49, 71], and domain control by adding tags and word features [95]. On the other hand, data-based methods perform adaptation either by combining in-domain and out-of-domain parallel corpora for supervised adaptation [60, 120] or by generating pseudo-parallel corpora from in-domain monolingual data for unsupervised adaptation [44, 156]. The thesis mainly focus on data-based methods, e.g., creating data for adaptation during pre-training (Chapter 6) or fine-tuning (Chapter 5, Chapter 7).

2.1.2 Cross-lingual Language Understanding

With the advent of multilingual neural machine translation, concurrently there are several attempts that leverage multilingual raw text to learn cross-lingual representations, both traditional non-contextualized word embeddings [57] and the more recent contextualized word representations [47]. In the following, we first introduce the formal definition of the cross-lingual language understanding task and then highlight the techniques of cross-lingual representation learning as well as several applications of cross-lingual representations in language understanding.

Problem Definition: Given a labeled dataset $(x, y) \sim \mathcal{D}_s$ in a high-resource language s, we aim to train a prediction model (i.e., $\theta : x \mapsto y$) that is able to make predictions for x written in both the source language s and the target language t. First we pre-train the model on text written in both s and t with a pre-training objective (Section 2.3). We then finetune the model on a labeled dataset written in the source language for a downstream task. To test the cross-lingual generalization performance, we directly apply the finetuned model to predict the input data x written in the target language t. We call this zero-shot cross-lingual prediction (Figure 2.1b) as we do not train the model on any labeled data in the target language.

Cross-lingual Representation Learning: Traditional non-contextualized word emebddings are often learned in two steps. First, the source and target word emebddings are obtained separately from learning on available monolingual source and target sentences using techniques such as continuous bag-of-words or skip-gram [126]. We can then perform *supervised* [183] or *unsupervised* [40] learning of a mapping that transforms source embeddings to the target space. In contrast, according to Doddapaneni et al. [48], multilingual contextualized representations are trained mainly by three categories of objectives: *monolingual* objective that relies on monolingual raw text, *parallel-corpora* objective that relies on parallel corpora, and *parallel-resource objective* that relies on parallel resources like word alignments. The most notable monolingual objective is masked language modeling (MLM), the key technique behind popular pre-trained representations such as mBERT [47] and XLM-R [42] (Section 2.3). In addition, there are several attempts to use parallel corpora to align representations of similar text across languages in a shared multilingual encoder space. These objectives are either word-level (e.g., TLM [39], CAMLM [136], CLMLM [87], HICTL [177], CLWA [83]) or sentence-level (e.g., XLCO [31], HICTL, CLSA [83]). In Chapter 4, we introduce two parallel-corpora objectives and compare them with a popular monolingual objective and dictionary-based objectives.

Applications of Cross-lingual Representations: Cross-lingual representations are an essential tool for cross-lingual transfer in downstream language understanding applications. In particular, cross-lingual contextualized word representations have proven effective in reducing the amount of supervision needed in a variety of cross-lingual NLP tasks such as sequence labeling [140], question answering [16], parsing [173], sentence classification [182] and retrieval [187]. This thesis leverages cross-lingual representations in multiple ways. In Chapter 5, we use traditional cross-lingual embedding to perform lexicon induction from monolingual in-domain data. In Chapter 7, we use multilingual contextualized representations for retrieving similar sentences across domains. In Chapter 6, we leverage pre-trained cross-lingual representations for machine translation.

2.2 Multilingual Neural Network Models

Neural networks have demonstrated an impressive ability to learn features automatically from data without manual design. Formally, for text data, we tokenize a text sentence into a sequence of words (or subwords) and denote it as $x = [x_0, x_1, \dots, x_N]$ where each word x_i comes from a fixed vocabulary $\mathcal{V}$ and can be represented as a one-hot vector. A neural network model takes a text sentence as inputs and produce a sequence of hidden vectors for each words (i.e., $\mathrm{H} = [\mathrm{h}_0, \mathrm{h}_1, \dots, \mathrm{h}_N],\ \mathrm{h}_i \in \mathbb{R}^d$), or a single fixed-size vector $\mathrm{h}_s \in \mathbb{R}^d$. Below we describe a parametric model for encoding the text data and a widely-used tokenization technique.

Transformer Architecture: Vaswani et al. [166] proposed the Transformer architecture to encode a word token sequence based on the idea of self-attention. The key idea of self-attention is to model the pairwise relations between all tokens in the sequence. Specifically for an input sequence of word embeddings X, we use three different linear projections to obtain three matrices – queries Q, keys K and values V in Equation (2.2), where the dimension of each vector in these matrices is d. The keys and queries are compared to compute the attention scores that capture the relations between each pair of tokens, and then we derive an output embedding by an average of the values weighted by the attention scores followed by a softmax function in Equation (2.3).

$$Q = W_Q X,\ K = W_K X,\ V = W_V X \tag{2.2}$$

$$Attention(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) V \tag{2.3}$$

The weighted average over the values Q does not consider the order of the sequence. To inject some information about the position of the tokens in the sequence, position embeddings are added to the input embeddings before the self-attention operations. There are many choices of learned or fixed position embeddings (see Gehring et al. [63] for more details). Due to an impressive performance in terms of modeling power and training speed, Transformer networks are widely used in many NLP models.

Tokenization: Processing unknown words has been intensively studied in machine translation. Recently several subword tokenization techniques that tokenize words into smaller frequent subword units have been proposed. For example, the word ‘lower’ could be segmented into two smaller units ‘low’ and ‘er’. Notably among these techniques, Sennrich et al. [157] proposed a segmentation method based on the *byte pair encoding* (BPE) algorithm [61]. The idea behind BPE is to iteratively replaces the most frequent pair of character n-grams in a sequence with a single, unused character n-grams. Specifically, we first initialize the symbol vocabulary with the character vocabulary. Each word is then represented as a sequence of characters plus a special end-of-word symbol '·' which is used to restore the original tokenization. We then iteratively count all symbol pairs and replace each occurrence of the most frequent pair ('A', 'B') with a new, merged symbol 'AB'. The number of the merge operations is a hyperparameter that determines the number of new symbols in the vocabulary, thus the final vocabulary size equals the size of the initial vocabulary size plus the number of the merge operations. Algorithm 1 shows a Python implementation.

2.3 Pre-training of Neural Network Models

While the modeling power of Transformer networks is exciting, training such large neural networks relies on a large amount of data. This poses a challenge for downstream applications for which there are few labeled data in many languages. To address this challenge, several studies have proposed to pre-train neural networks on large collections of freely available raw data in multiple languages. These pre-training techniques can be grouped into two categories: pre-training of *encoder* models and pre-training of *encoder-decoder* models. A typical encoder model is a neural network model (e.g., Transformer) that takes in a sequence of word tokens and produces a sequence of hidden vectors for words. Encoder models are usually used for language understanding tasks (e.g., sequence labeling). In contrast, an encoder-decoder model adds another neural network (e.g., Transformer) as a decoder on top of an encoder model. This allows encoder-decoder models to handle language generation tasks (e.g., machine translation). In the following, we provide several pre-training methods for these two types of models.

2.3.1 Pre-training of Encoder Models

Notably, mBERT [47], XLM [39] and XLM-R [42] have demonstrated impressive power of learning from raw text data in more than 100 languages and have led to state-of-the-art results on a variety of multilingual NLP tasks. The key idea is to pre-train a Transformer encoder network on a large number of unlabeled data or a small number of parallel sentences, and then fine-tune it on a small number of labeled data for adaptation to downstream tasks. We highlight a traditional language modeling objective and two popular pre-training language modeling objectives – *masked language modeling* (MLM) and *translation language modeling* (TLM).

Language Modeling: Traditional language models are trained left-to-right or right-to-left by optimizing the prediction of the next word token given the preceding tokens. The formal definition is as follows:

$$\ell_{\mathrm{MLM}}(x) = -\sum_{t=1}^{|x|} \log P_\theta(x_t | x_{<t}), \tag{2.4}$$

where $x_{<t}$ denotes the proceding tokens before the t-th word in x.

Masked Language Modeling: Different from traditional language models, MLM first randomly masks some percentage of the input tokens and then predicts those masked tokens conditioned on the surrounding context. Specifically, a fraction (e.g., 15%) of input tokens are selected randomly for prediction. These tokens are replaced by a `[MASK]` token 80% of the time, or replaced by a random token 10% of the time or kept unchanged 10% of the time. During prediction, the final hidden vectors of those masked tokens are fed into an output softmax layer over the vocabulary, as in a traditional language model. Finally, the whole neural network is optimized with a cross-entropy loss. Formally, a masked language modeling objective takes a word sequence from a monolingual corpus (e.g., $x \in \mathcal{D}_X$), and optimizes the prediction of randomly masked tokens as follows:

$$\ell_{\mathrm{MLM}}(x) = -\mathbb{E}_{s\sim[1,|x|]} \log P_\theta(x_s | x_{\backslash s}), \tag{2.5}$$

where x_s are the masked tokens randomly sampled from x, and $x_{\backslash s}$ indicates all the other tokens except the masked ones.

Translation Language Modeling: In the standard monolingual setting, while a mix of monolingual corpora contains sequences written in multiple languages, each sequence x only contains word tokens in one of these languages. In contrast, Conneau and Lample [39] proposed a translation language modeling (TLM) objective that uses parallel corpora for pre-training multilingual models. Formally, a TLM objective takes a concatenation of a source-target sentence pair (i.e., $z = [x; y]$) from a parallel corpus $(x, y) \in \mathcal{D}_{XY}$, and optimizes the prediction of randomly masked tokens in z in a similar way as MLM:

$$\ell_{\mathrm{MLM}}(z) = -\mathbb{E}_{s\sim[1,|z|]} \log P_\theta(z_s | z_{\backslash s}). \tag{2.6}$$

2.3.2 Pre-training of Encoder-Decoder Models

Pre-training of encoder-decoder models has been shown effective in low-resource and medium-resource language translations by many recent works [39, 114, 119, 160], where different pre-training objectives are proposed to leverage large amounts of 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 [44, 81, 101, 155]. In the following, we introduce the denoising auto-encoding framework.

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:

$$\mathcal{L}_{\text{DAE}}(\mathcal{D}_Y) = \sum_{y \in \mathcal{D}_Y} \log P_\theta(y \,|\, g(y)), \tag{2.7}$$

where θ denotes the model's learning parameters. For notation simplicity, we drop θ in the rest of the chapters. This formulation encompasses several different previous works in data augmentation for MT, such as monolingual data copying [44], where $g(\cdot)$ is the identity function, back translation [155], where $g(\cdot)$ is a backwards translation model, as well as heuristic noising functions [110, 119, 160] that randomly sample noise according to manually devised heuristics.

In particular, the mBART method [119] is a recently popular method with two types 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.

Algorithm 1 Learning BPE operations

```
import re, collections

def get_stats(vocab):
    pairs = collections.defaultdict(int)
    for word, freq in vocab.items():
        symbols = word.split()
        for i in range(len(symbols)-1):
            pairs[symbols[i],symbols[i+1]] += freq
    return pairs

def merge_vocab(pair, v_in):
    v_out = {}
    bigram = re.escape(’ ’.join(pair))
    p = re.compile(r’(?<!\S)’ + bigram + r’(?!\S)’)
    for word in v_in:
        w_out = p.sub(’’.join(pair), word)
        v_out[w_out] = v_in[word]
    return v_out

vocab = {’l o w </w>’ : 5, ’l o w e r </w>’ : 2,
         ’n e w e s t </w>’:6, ’w i d e s t </w>’:3}
num_merges = 10
for i in range(num_merges):
    pairs = get_stats(vocab)
    best = max(pairs, key=pairs.get)
    vocab = merge_vocab(best, vocab)
    print(best)
```

Chapter 3

Cross-Lingual Generalization Benchmark

In this chapter, we explore the cross-lingual generalization problem – one main data discrepancy problem in which training and test data might be written in different languages. To fully understand this problem, We first propose a benchmark XTREME to evaluate the zero-shot cross-lingual generalization ability of multilingual neural network models trained with the (masked) language modeling objectives. Later on in Chapter 4, we will introduce two *explicit* cross-lingual alignment objectives to align cross-lingual contextualized representations both at the word level and sentence level.

The content in this chapter and follow-up work has been reported in the following papers:

- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalisation. In *Proceddings of International Conference on Machine Learning (ICML) 2020*. [82]

- Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Graham Neubig, and Melvin Johnson. XTREME-R: Towards More Challenging and Nuanced Multilingual Evaluation. *arXiv preprint arXiv:2104.07412 2021, under review*. [151]

3.1 Overview

Over the last few years, there has been a move towards *general-purpose multilingual representations* that are applicable to many tasks, both on the word level [13, 57, 125] or the full-sentence level [39, 47]. Despite the fact that such representations are intended to be general-purpose, evaluation of them has often been performed on a very limited and often disparate set of tasks—typically focusing on translation [39, 66] and classification [41, 154]—and typologically similar languages [40].

To address this problem and incentivize research on truly general-purpose cross-lingual representation and transfer learning, we introduce the Cross-lingual TRansfer Evaluation of Multilingual Encoders (XTREME) benchmark. XTREME covers 40 typologically diverse languages spanning 12 language families and includes 9 tasks that require reasoning about different levels of syntax or semantics.[1] In addition, we introduce *pseudo* test sets as diagnostics that cover all 40 languages by automatically translating the English test set of the natural language inference and question-answering dataset to the remaining languages.

XTREME focuses on the *zero-shot cross-lingual transfer* scenario, where annotated training data is provided in English but none is provided in the language to which systems must transfer.[2] We evaluate a range of state-of-the-art machine translation (MT) and multilingual representation-based approaches to performing this transfer. We find that while state-of-the-art models come close to human performance in English on many of the tasks we consider, performance drops significantly when evaluated on other languages. Overall, performance differences are highest for syntactic and sentence retrieval tasks. Further, while models do reasonably well in most languages in the Indo-European family, we observe lower performance particularly for Sino-Tibetan, Japonic, Koreanic, and Niger-Congo languages.

In sum, our contributions are the following: (i) We release a suite of 9 cross-lingual benchmark tasks covering 40 typologically diverse languages. (ii) We provide an online platform and leaderboard for the evaluation of multilingual models. (iii) We provide a set of strong baselines, which we evaluate across all tasks, and release code to facilitate adoption. (iv) We provide an extensive analysis of limitations of state-of-the-art cross-lingual models.

[1] By typologically diverse, we mean languages that span a wide set of linguistic phenomena such as compounding, inflection, derivation, etc. which occur in many of the world's languages.

[2] This is done both for efficiency purposes (as it only requires testing, not training, on each language) and practical considerations (as annotated training data is not available for many languages).

Task	Corpus	|Train|	|Dev|	|Test|	Test sets	|Lang.|	Task	Metric	Domain
Classification	XNLI	392,702	2,490	5,010	translations	15	NLI	Acc.	Misc.
	PAWS-X	49,401	2,000	2,000	translations	7	Paraphrase	Acc.	Wiki / Quora
Struct. pred.	POS	21,253	3,974	47-20,436	ind. annot.	33 (90)	POS	F1	Misc.
	NER	20,000	10,000	1,000-10,000	ind. annot.	40 (176)	NER	F1	Wikipedia
QA	XQuAD	87,599	34,726	1,190	translations	11	Span extraction	F1 / EM	Wikipedia
	MLQA			4,517–11,590	translations	7	Span extraction	F1 / EM	Wikipedia
	TyDiQA-GoldP	3,696	634	323–2,719	ind. annot.	9	Span extraction	F1 / EM	Wikipedia
Retrieval	BUCC	-	-	1,896–14,330	-	5	Sent. retrieval	F1	Wiki / news
	Tatoeba	-	-	1,000	-	33 (122)	Sent. retrieval	Acc.	misc.

Table 3.1: Characteristics of the datasets in XTREME for the zero-shot transfer setting. For tasks that have training and dev sets in other languages, we only report the English numbers. We report the number of test examples per target language and the nature of the test sets (whether they are translations of English data or independently annotated). The number in brackets is the size of the intersection with our selected languages. For NER and POS, sizes are in sentences. Struct. pred.: structured prediction. Sent. retrieval: sentence retrieval.

3.2 XTREME

3.2.1 Design principles

Given XTREME's goal of providing an accessible benchmark for the evaluation of cross-lingual transfer learning on a diverse and representative set of tasks and languages, we select the tasks and languages that make up the benchmark based on the following principles:

Task difficulty Tasks should be sufficiently challenging so that cross-language performance falls short of human performance.

Task diversity Tasks should require multilingual models to transfer their meaning representations at different levels, e.g. words, phrases and sentences. For example, while classification tasks require sentence-level transfer of meaning, sequence labeling tasks like part-of-speech (POS) tagging or named entity recognition (NER) test the model's transfer capabilities at the word level.

Training efficiency Tasks should be trainable on a single GPU for less than a day. This is to make the benchmark accessible, in particular to practitioners working with low-resource languages under resource constraints.

Multilinguality We prefer tasks that cover as many languages and language families as possible.

Sufficient monolingual data Languages should have sufficient monolingual data for learning useful pre-trained representations.

Accessibility Each task should be available under a permissive license that allows the use and redistribution of the data for research purposes.

3.2.2 Tasks

XTREME consists of nine tasks that fall into four different categories requiring reasoning on different levels of meaning. We give an overview of all tasks in Table 3.1, and describe the task details as follows.

XNLI The Cross-lingual Natural Language Inference corpus [41] asks whether a premise sentence entails, contradicts, or is neutral toward a hypothesis sentence. Crowd-sourced English data is translated to ten other languages by professional translators and used for evaluation, while the MultiNLI [179] training data is used for training.

PAWS-X The Cross-lingual Paraphrase Adversaries from Word Scrambling [188] dataset requires to determine whether two sentences are paraphrases. A subset of the PAWS dev and test sets [195] was translated to six other languages by professional translators and is used for evaluation, while the PAWS training set is used for training.

POS We use POS tagging data from the Universal Dependencies v2.5 [134] treebanks, which cover 90 languages. Each word is assigned one of 17 universal POS tags. We use the English training data for training and evaluate on the test sets of the target languages.

NER For NER, we use the `Wikiann` [137] dataset. Named entities in Wikipedia were automatically annotated with LOC, PER, and ORG tags in IOB2 format using a combination of knowledge base properties, cross-lingual and anchor links, self-training, and data selection. We use the balanced train, dev, and test splits from Rahimi et al. [143].

XQuAD The Cross-lingual Question Answering Dataset [16] requires identifying the answer to a question as a span in the corresponding paragraph. A subset of the English SQuAD v1.1 [144] dev set was translated into ten other languages by professional translators and is used for evaluation.

MLQA The Multilingual Question Answering [111] dataset is another cross-lingual question answering dataset similar to XQuAD. The evaluation data for English and six other languages were obtained by automatically mining target language sentences that are parallel to sentences in English from Wikipedia, crowd-sourcing annotations in English, and translating the question and aligning the answer spans in the target languages. For both XQuAD and MLQA, we use the SQuAD v1.1 training data for training and evaluate on the test data of the corresponding task.

TyDiQA-GoldP We use the gold passage version of the Typologically Diverse Question Answering [37] dataset, a benchmark for information-seeking question answering, which covers nine languages. The gold passage version is a simplified version of the primary task, which uses only the gold passage as context and excludes unanswerable questions. It is thus similar to XQuAD and MLQA, while being more challenging as questions have been written without seeing the answers, leading to 3× and 2× less lexical overlap compared to XQuAD and MLQA respectively. We use the English training data for training and evaluate the test sets of the target languages.

BUCC The goal of the second and third shared task of the workshop on Building and Using Parallel Corpora [201, 202] is to extract parallel sentences from a comparable corpus between English and four other languages. The dataset provides train and test splits for each language. For simplicity, we evaluate representations on the test sets directly without fine-tuning and calculate similarity using cosine similarity.[3]

Tatoeba We use the Tatoeba dataset [11], which consists of up to 1,000 English-aligned sentence pairs covering 122 languages. We find the nearest neighbor using cosine similarity and calculate the error rate.

3.2.3 Languages

As noted in Section 3.2.1, we choose our target languages based on availability of monolingual data, and typological diversity. We use the number of articles in Wikipedia as a proxy for the amount of monolingual data available online. In order to strike a balance between language diversity and availability of monolingual data, we select all languages out of the top 100 Wikipedias[4] with the most articles as of December 2019.[5] We first select all languages that appear in at least three of our benchmark datasets. This leaves us with 19 languages, most of which are Indo-European or major world languages. We now select 21 additional languages that appear in at least one dataset and come from less represented language families. Wherever possible, we choose at least two languages per family.[6]

In total, XTREME covers the following 40 languages (shown with their ISO 639-1 codes for brevity) belonging to 12 language families and two isolates: af, ar, bg, bn, de, el, en, es, et, eu, fa, fi, fr, he, hi, hu, id, it, ja, jv, ka, kk, ko, ml, mr, ms, my, nl, pt, ru, sw, ta, te, th, tl, tr, ur, vi, yo, and zh. We provide a detailed overview of these languages in terms of their number of Wikipedia

[3] Results can be improved using more sophisticated similarity metrics [11].

[4] https://meta.wikimedia.org/wiki/List_of_Wikipedias

[5] This also has the benefit that they are covered by state-of-the-art methods such as mBERT and XLM.

[6] For the Austro-Asiatic, Kartvelian, and Kra-Dai families as well as for isolates, we only obtain one language.

articles, linguistic features, and coverage in XTREME in the appendix.

While XTREME covers these languages in the sense that there is gold standard data in at least one task in each language, this does not mean that it covers all aspects of each language that are necessary for transfer. Languages may reveal different characteristics based on the task, domain, and register in which they are used. XTREME thus only serves as a glimpse into a model's true cross-lingual generalization capability.

3.2.4 Pseudo test data for analyses

XTREME covers 40 languages overall. Evaluation across the majority of languages is only possible for a subset of tasks, i.e. POS, NER, and Tatoeba. As additional diagnostics and to enable a broader comparison across languages for a more diverse set of tasks, we automatically translate the English portions of a representative classification and QA task to the remaining languages using an in-house translation system.[7] We choose XNLI and XQuAD as both have test sets that are translations of the English data by professional translators.

We first verify that performance on the translated test sets is a good proxy for performance on the gold standard test sets. We report the detailed results in the appendix. For XQuAD, the automatically translated test sets underestimate mBERT's true performance by 3.0 F1 / 0.2 EM points, similar to the 2.6 F1 points reported by Agić and Schluter [1] when translating the test data to other languages.[8] For XNLI, the automatically translated test sets overestimate the true prediction accuracy by 2.4 points. In order to measure the translation quality between the human-translated test data and our pseudo test data, we compute the BLEU score, and the chrF score [142], which is suitable for measuring the translation quality of some languages such as Chinese and Russian. For the 14 languages in XNLI, we obtain average scores of 34.2 BLEU and 58.9 chrF scores on our pseudo test data compared to the reference translations, which correlate with a Pearson's ρ of 0.57 and 0.28 respectively with mBERT performance.

Translating the English data to the remaining languages yields 40-way parallel pseudo test data that we employ for analyses in Section 3.4.

[7] Details of our translation system are provided in the appendix.

[8] Note that even human translated test sets may underestimate a model's true cross-lingual generalization ability as such *translationese* has been shown to be less lexically diverse than naturally composed language [98].

3.3 Experiments

3.3.1 Training and evaluation setup

XTREME focuses on the evaluation of multilingual representations. We do not place any restriction on the amount or nature of the monolingual data used for pretraining multilingual representations. However, we request authors to be explicit about the data they use for training, in particular any cross-lingual signal. In addition, we suggest authors should not use any additional labeled data in the target task beyond the one that is provided.

For evaluation, we focus on *zero-shot cross-lingual transfer* with English as the source language as this is the most common setting for the evaluation of multilingual representations and as many tasks only have training data available in English. Although English is not generally the best source language for cross-lingual transfer for all target languages [113], this is still the most practically useful setting. A single source language also facilitates evaluation as models only need to be trained once and can be evaluated on all other languages.[9]

Concretely, pretrained multilingual representations are fine-tuned on English labeled data of an XTREME task. The model is then evaluated on the test data of the task in the target languages.

3.3.2 Baselines

We evaluate a number of strong baselines and state-of-the-art models. The approaches we consider learn multilingual representations via self-supervision or leverage translations—either for representation learning or for training models in the source or target language. We focus on models that learn deep contextual representations as these have achieved state-of-the-art results on many tasks. For comparability among the representation learning approaches, we focus on models that learn a multilingual embedding space between all languages in XTREME. We encourage future work to focus on these languages to capture as much language diversity as possible.

mBERT Multilingual BERT [47] is a transformer model [166] that has been pretrained on the Wikipedias of 104 languages using masked language modeling (MLM).

XLM XLM [39] uses a similar pretraining objective as mBERT with a larger model, a larger shared vocabulary, and trained on the same Wikipedia data covering 100 languages.

XLM-R XLM-R Large [42] is similar to XLM but was trained on more than a magnitude more

[9] Future work may also consider multi-source transfer, which is interesting particularly for low-resource languages, and transfer to unknown languages or unknown language-task combinations.

data from the web covering 100 languages.

MMTE The massively multilingual translation encoder is part of an NMT model that has been trained on in-house parallel data of 103 languages extracted from the web [10]. For transfer, we fine-tune the encoder of the model [158].

Translate-train For many language pairs, an MT model may be available, which can be used to obtain data in the target language. To evaluate the impact of using such data, we translate the English training data into the target language using our in-house MT system. We then fine-tune mBERT on the translated data. We provide details on how we align answer spans in the source and target language for the QA tasks in the appendix. We do not provide translation-based baselines for structured prediction tasks due to an abundance of in-language data and a requirement for annotation projection.

Translate-train multi-task We also experiment with a multi-task version of the translate-train setting where we fine-tune mBERT on the combined translated training data of all languages jointly.

Translate-test Alternatively, we train the English BERT-Large [47] model on the English training data and evaluate it on test data that we translated from the target language to English using our in-house MT system.

In-language model For the POS, NER, and TyDiQA-GoldP tasks where target-language training data is available, we fine-tune mBERT on monolingual data in the target language to estimate how useful target language labeled data is compared to labeled data in a source language.

In-language few-shot In many cases, it may be possible to procure a small number of labeled examples in the target language [54]. To evaluate the viability of such an approach, we additionally compare against an mBERT model fine-tuned on 1,000 target language examples for the tasks where monolingual training data is available in the target languages.

In-language multi-task For the tasks where monolingual training data is available, we additionally compare against an mBERT model that is jointly trained on the combined training data of all languages.

Human performance For XNLI, PAWS-X, and XQuAD, we obtain human performance estimates from the English datasets they are derived from, MNLI, PAWS-X, and SQuAD respectively [131, 144, 195].[10] For TyDiQA-GoldP, we use the performance estimate of Clark et al. [37]. For MLQA, as answers are annotated using the same format as SQuAD, we employ the same human performance estimate. For POS tagging, we adopt 97% as a canonical estimate of human per-

[10] Performance may differ across languages due to many factors but English performance still serves as a reasonable proxy.

Table 3.2: Hyper-parameters of baseline and state-of-the-art models. We do not use XLM-15 and XLM-R-Base in our experiments.

Model	Parameters	Langs	Vocab size	Layers
BERT-large	345M	1	28,996	24
mBERT	172M	104	119,547	12
MMTE	192M	103	64,000	6
XLM-15	250M	15	95,000	12
XLM-100	570M	100	200,000	12
XLM-R-Base	270M	100	250,002	12
XLM-R-Large	550M	100	250,002	24

formance based on Manning [123]. We are not able to obtain human performance estimates for NER as annotations have been automatically generated and for sentence retrieval as identifying a translation among a large number of documents is too time-consuming.

3.3.3 Hyper-parameters

Table 4.1 summarizes the hyper-parameters of baseline and state-of-the-art models. We refer to XLM-100 as XLM, and XLM-R-large as XLM-R in our paper to simplify the notation. All hyper-parameter tuning is done on English validation data. We encourage authors evaluating on XTREME to do the same.

mBERT We use the cased version, which covers 104 languages, has 12 layers, 768 hidden units per layer, 12 attention heads, a 110k shared WordPiece vocabulary, and 110M parameters.[11] The model was trained using Wikipedia data in all 104 languages, oversampling low-resource languages with an exponential smoothing factor of 0.7. We generally fine-tune mBERT for two epochs, with a training batch size of 32 and a learning rate of 2e-5. For training BERT models on the QA tasks, we use the original BERT codebase. For all other tasks, we use the Transformers library [180].

XLM and XLM-R We use the XLM and XLM-R Large versions that cover 100 languages, use a 200k shared BPE vocabulary, and that have been trained with masked language modeling.[12] We fine-tune both for two epochs with a learning rate of 3e-5 and an effective batch size of 16. In

[11] https://github.com/google-research/bert/blob/master/multilingual.md

[12] https://github.com/facebookresearch/XLM

Model	Avg	Pair sentence		Structured prediction		Question answering			Sentence retrieval	
		XNLI	PAWS-X	POS	NER	XQuAD	MLQA	TyDiQA-GoldP	BUCC	Tatoeba
Metrics		Acc.	Acc.	F1	F1	F1 / EM	F1 / EM	F1 / EM	F1	Acc.
Cross-lingual zero-shot transfer (models are trained on English data)										
mBERT	59.6	65.4	81.9	70.3	62.2	64.5 / 49.4	61.4 / 44.2	59.7 / 43.9	56.7	38.7
XLM	55.5	69.1	80.9	70.1	61.2	59.8 / 44.3	48.5 / 32.6	43.6 / 29.1	56.8	32.6
XLM-R Large	68.1	79.2	86.4	72.6	65.4	76.6 / 60.8	71.6 / 53.2	65.1 / 45.0	66.0	57.3
MMTE	59.3	67.4	81.3	72.3	58.3	64.4 / 46.2	60.3 / 41.4	58.1 / 43.8	59.8	37.9
Translate-train (models are trained on English training data translated to the target language)										
mBERT	-	74.0	86.3	-	-	70.0 / 56.0	65.6 / 48.0	55.1 / 42.1	-	-
mBERT, multi-task	-	75.1	88.9	-	-	72.4 / 58.3	67.6 / 49.8	64.2 / 49.3	-	-
Translate-test (models are trained on English data and evaluated on target language data translated to English)										
BERT-large	-	76.5	84.4	-	-	76.3 / 62.1	72.9 / 55.3	72.1 / 56.0	-	-
In-language models (models are trained on the target language training data)										
mBERT, 1000 examples	-	-	-	87.6	77.9	-	-	58.7 / 46.5	-	-
mBERT	-	-	-	89.8	88.3	-	-	74.5 / 62.7	-	-
mBERT, multi-task	-	-	-	91.5	89.1	-	-	77.6 / 68.0	-	-
Human	-	92.8	97.5	97.0	-	91.2 / 82.3	91.2 / 82.3	90.1 / -	-	-

Table 3.3: Overall results of baselines across all XTREME tasks. Translation-based baselines are not meaningful for sentence retrieval. We provide in-language baselines where target language training data is available. Note that for the QA tasks, translate-test performance is not directly comparable to the other scores as a small number of test questions were discarded and alignment is measured on the English data.

contrast to XLM, XLM-R does not use language embeddings. We use the Transformers library for training XLM and XLM-R models on all tasks.

3.3.4 Results

Overall results We show the main results in Table 3.3. XLM-R is the best-performing zero-shot transfer model and generally improves upon mBERT significantly. The improvement is smaller, however, for the structured prediction tasks. MMTE achieves performance competitive with mBERT on most tasks, with stronger results on XNLI, POS, and BUCC.

If a strong MT system is available, translating the training sets provides improvements over using the same model with zero-shot transfer. Translating the test data provides similar benefits compared to translating the training data and is particularly effective for the more complex QA tasks, while being more expensive during inference time. While using an MT system as a black

Model	XNLI	PAWS-X	XQuAD	MLQA	TyDiQA-GoldP	Avg	POS	NER
mBERT	16.5	14.1	25.0	27.5	22.2	21.1	25.5	23.6
XLM-R	10.2	12.4	16.3	19.1	13.3	14.3	24.3	19.8
Translate-train	7.3	9.0	17.6	22.2	24.2	16.1	-	-
Translate-test	6.7	12.0	16.3	18.3	11.2	12.9	-	-

Table 3.4: The cross-lingual transfer gap (lower is better) of different models on XTREME tasks. The transfer gap is the difference between performance on the English test set and the average performance on the other languages. A transfer gap of 0 indicates perfect cross-lingual transfer. For the QA datasets, we only show EM scores. The average gaps are computed over the sentence classification and QA tasks.

box leads to strong baselines, the MT system could be further improved in the context of data augmentation.

For the tasks where in-language training data is available, multilingual models trained on in-language data outperform zero-shot transfer models. However, zero-shot transfer models nevertheless outperform multilingual models trained on only 1,000 in-language examples on the complex QA tasks as long as more samples in English are available. For the structured prediction tasks, 1,000 in-language examples enable the model to achieve performance that is similar to being trained on the full labeled dataset, similar to findings for classification [54]. Finally, multi-task learning on the Translate-train and In-language setting generally improves upon single language training.

Cross-lingual transfer gap For a number of representative models, we show the cross-lingual transfer gap, i.e. the difference between the performance on the English test set and all other languages in Table 3.4.[13] While powerful models such as XLM-R reduce the gap significantly compared to mBERT for challenging tasks such as XQuAD and MLQA, they do not have the same impact on the syntactic structured prediction tasks. On the classification tasks, the transfer learning gap is lowest, indicating that there may be less headroom for progress on these tasks. The use of MT reduces the gap across all tasks. Overall, a large gap remains for all approaches, which indicates much potential for work on cross-lingual transfer.

[13] This comparison should be taken with a grain of salt, as scores across languages are not directly comparable for the tasks where test sets differ, i.e. POS, NER, MLQA, and TyDiQA-GoldP and differences in scores may not be linearly related.

3.4 Analyses

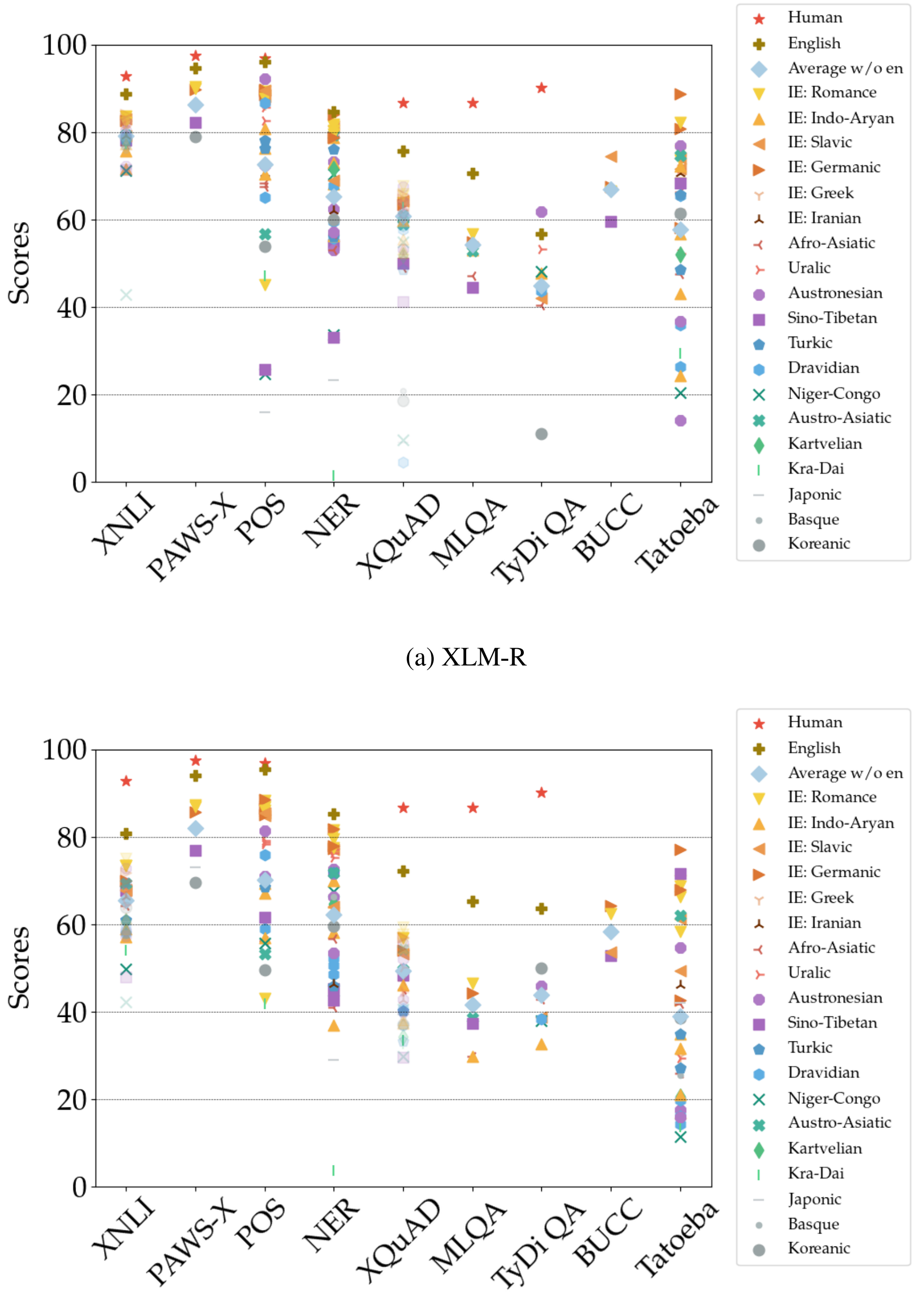
We conduct a series of analyses investigating the limitations of state-of-the-art cross-lingual models.

Best zero-shot model analysis We show the performance of the best zero-shot transfer model, XLM-R Large broken down by task and language in Figure 3.1a. The figure illustrates why it is important to evaluate general-purpose multilingual representations across a diverse range of tasks and languages: On XNLI, probably the most common standard cross-lingual evaluation task, and PAWS-X, scores cluster in a relatively small range—even considering pseudo test sets for XNLI. However, scores for the remaining tasks have a significantly wider spread, particularly as we include pseudo test sets. For TyDiQA-GoldP, English performance is lowest in comparison; the high performance on members of the Austronesian and Uralic language families (Indonesian and Finnish) may be due to less complex Wikipedia context passages for these languages. Across tasks, we generally observe higher performance on Indo-European languages and lower performance for other language families, particularly for Sino-Tibetan, Japonic, Koreanic, and Niger-Congo languages. Some of these difficulties may be due to tokenization and an under-representation of ideograms in the joint sentencepiece vocabulary, which has been shown to be important in a cross-lingual model's performance [16, 42]. We observe similar trends for mBERT, for which we show the same graph in the appendix.

Correlation with pretraining data size We calculate the Pearson correlation coefficient ρ of the model performance and the number of Wikipedia articles (see the appendix) in each language and show results in Figure 3.2.[14] For mBERT, which was pretrained on Wikipedia, we observe a high correlation for most tasks ($\rho \approx 0.8$) except for the structured prediction tasks where $\rho \approx 0.35$. We observe similar trends for XLM and XLM-R, with lower numbers for XLM-R due to the different pretraining domain (see Table 3.5). This indicates that current models are not able to fully leverage the information extracted from the pretraining data to transfer to syntactic tasks.

Analysis of language characteristics We analyze results based on different language families and writing scripts in Figure 3.3. For mBERT, we observe the best transfer performance on branches of the Indo-European language family such as Germanic, Romance and Slavic languages. In contrast, cross-lingual transfer performance on low-resource language families such as Niger-Congo and Kra-Dai is still low. Looking at scripts, we find that the performance on syntactic tasks differs among popular scripts such as Latin and ideogram scripts. For example in

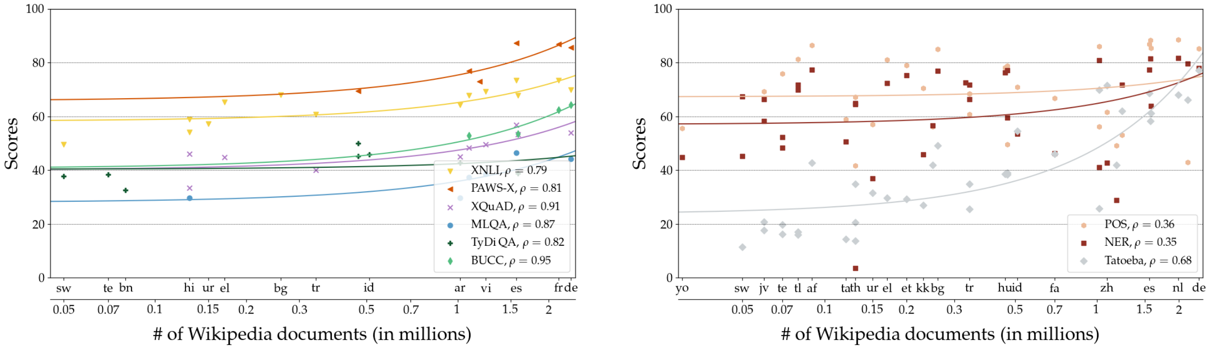
[14] We observe similar correlations when using the number of tokens in Wikipedia instead.



(a) XLM-R

(b) mBERT

Figure 3.1: An overview of XLM-R's and mBERT's performances on the XTREME tasks across all languages in each task. We highlight an estimate of human performance, performance on the English test set, the average of all languages excluding English, and the family of each language. Performance on pseudo test sets for XNLI and XQuAD is shown with slightly transparent markers.



(a)

(b)

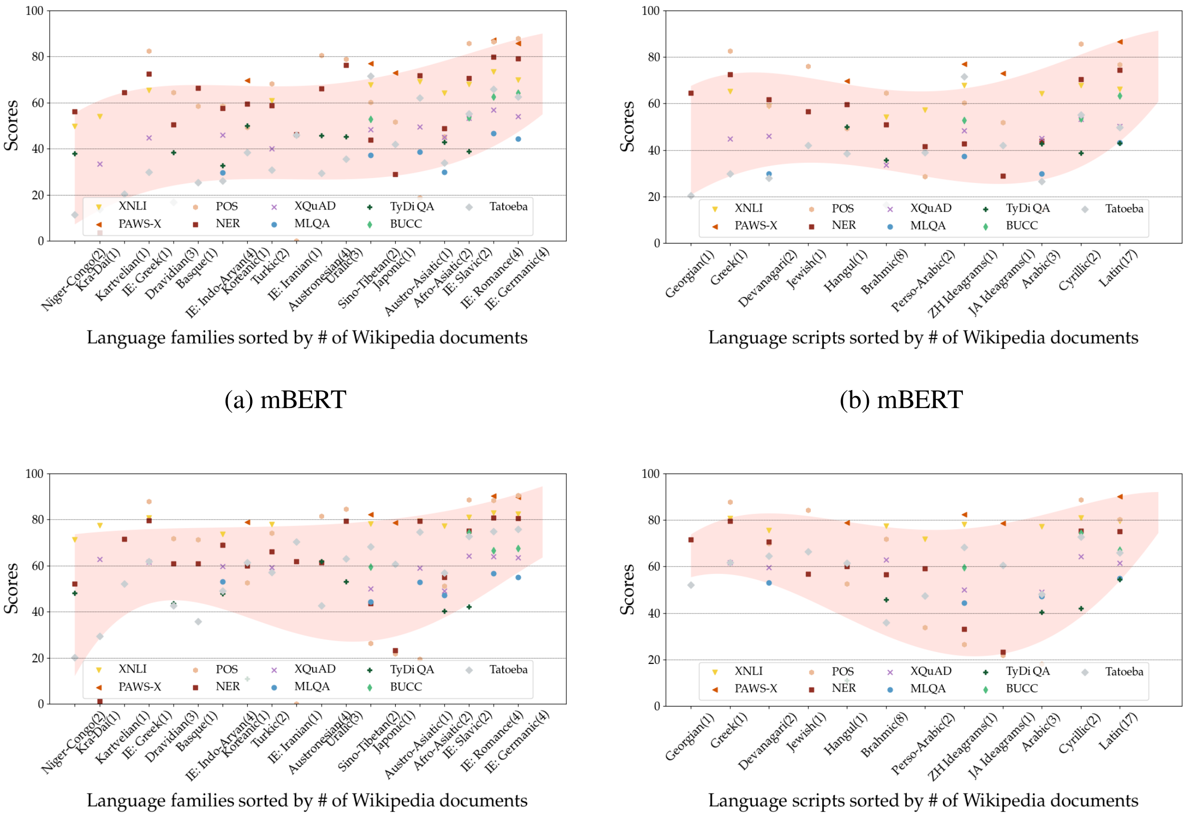
Figure 3.2: Performance of mBERT across tasks and languages in comparison to the number of Wikipedia articles for each language. We show tasks with a Pearson correlation coefficient $\rho > 0.7$ on the left and others on the right. Numbers across tasks are not directly comparable. We remove the x axis labels of overlapping languages for clarity. We additionally plot the linear fit for each task (curved due to the logarithmic scale of the x axis).

	XNLI	PAWS-X	POS	NER	XQuAD	MLQA	TyDiQA-GoldP	BUCC	Tatoeba
mBERT	0.79	0.81	0.36	0.35	0.80	0.87	0.82	0.95	0.68
XLM	0.80	0.76	0.32	0.29	0.74	0.73	0.52	0.61	0.68
XLM-R	0.75	0.79	0.22	0.27	0.50	0.76	0.14	0.36	0.49

Table 3.5: Pearson correlation coefficients (ρ) of zero-shot transfer performance and Wikipedia size across datasets and models.

the NER task, mBERT performs better on data in Latin script than that in Chinese or Japanese ideograms. This indicates that the current models still have difficulty transferring word-level syntactic information across languages written in different scripts.

Errors across languages For XNLI and XQuAD where the other test sets are translations from English, we analyze whether approaches make the same type of errors in the source and target languages. To this end, we explore whether examples that are correctly and incorrectly predicted in English are correctly predicted in other languages. On the XNLI dev set, mBERT correctly predicts on average 71.8% of examples that were correctly predicted in English. For examples that were misclassified, the model's performance is about random. On average, predictions on XNLI are consistent between English and another language for 68.3% of examples. On the XQuAD test



(a) mBERT

(b) mBERT

(c) XLM-R

(d) XLM-R

Figure 3.3: Performance of mBERT (a,b) and XLM-R (c,d) across tasks grouped by language families (left) and scripts (right). The number of languages per group is in brackets and the groups are from low-resource to high-resource on the x-axis. We additionally plot the 3rd order polynomial fit for the minimum and maximum values for each group.

	trigram, seen	trigram, unseen	4-gram, seen	4-gram, unseen
en	90.3	63.0	88.1	67.5
avg w/o en	50.6	12.1	44.3	18.3
difference	39.7	50.9	43.7	49.2

Table 3.6: Accuracy of mBERT on POS tag trigrams and 4-grams in the target language dev data that appeared and did not appear in the English training data. We show the performance on English, the average across all other languages, and their difference.

set, mBERT correctly predicts around 60% of examples that were correctly predicted in English and 20% of examples that were incorrectly predicted. While some of these are plausible spans, more work needs to focus on achieving consistent predictions across languages.

Generalization to unseen tag combinations and entities We analyze possible reasons for the less successful transfer on structured prediction tasks. The Universal Dependencies dataset used for POS tagging uses a common set of 17 POS tags for all languages, so a model is not required to generalize to unseen tags at test time. However, a model may be required to generalize to unseen tag *combinations* at test time, for instance due to differences in word order between languages. We gauge how challenging such generalization is by computing a model's accuracy for POS tag n-grams in the target language dev data that were not seen in the English training data. We calculate values for tag trigrams and 4-grams and show accuracy scores for mBERT in Table 3.6. We observe the largest differences in performance for unseen trigrams and 4-grams, which highlights that existing cross-lingual models struggle to transfer to the syntactic characteristics of other languages. For NER, we estimate how well models generalize to unseen entities at test time. We compute mBERT's accuracy on entities in the target language dev data that were not seen in the English training data. We observe the largest difference between performance on seen and unseen entities for Indonesian and Swahili. Isolating for confounding factors such as entity length, frequency, and Latin script, we find the largest differences in performance for Swahili and Basque. Together, this indicates that the model may struggle to generalize to entities that are more characteristic of the target language. We show the detailed results for both analyses in the appendix.

Generalization to unseen entities We show the performance of mBERT on entities in the target language NER dev data that were seen and not seen in the English NER training data in Table

	af	de	el	en	es	et	eu	fi	fr	he	hu	id	it	ka	ms	nl	pt	ru	sw	tr	vi
(a) Seen	94.7	88.3	91.4	91.9	76.3	88.3	83.6	85.3	90.5	78.2	90.7	89.4	88.4	92.3	88.6	93.5	88.6	83.9	96.3	85.2	91.4
(b) Not seen	82.1	80.2	74.8	84.6	80.4	78.9	69.4	79.8	80.1	56.5	78.3	58.0	81.5	70.2	75.0	82.9	82.3	68.5	66.6	73.7	73.4
(a) − (b)	12.6	8.1	16.5	7.2	-4.1	9.4	14.1	5.5	10.4	21.7	12.3	31.5	6.9	22.1	13.6	10.6	6.4	15.4	29.7	11.6	18.0
(c) Short	86.5	82.9	80.3	88.2	86.6	81.7	72.5	83.9	88.6	66.3	83.7	85.8	87.2	72.5	89.1	87.6	87.8	78.0	65.7	83.1	84.6
(d) Latin	83.6	81.2	87.5	86.2	80.0	79.5	70.3	80.3	81.1	77.2	79.9	61.8	82.6	89.6	76.3	84.2	83.0	83.8	70.0	75.0	74.9
(e) Freq	87.3	80.6	81.9	91.6	83.4	79.4	68.8	85.7	77.3	66.8	86.0	56.5	88.8	74.3	81.3	87.1	84.4	76.5	49.1	81.9	78.6
min((a) − (c–e))	7.4	5.4	3.9	0.3	3.7	6.6	11.0	0.4	1.9	1.0	4.7	3.6	0.4	2.7	0.5	5.9	0.8	0.1	26.4	2.2	6.8

Table 3.7: Comparison of accuracies for entities in the target language NER dev data that were seen in the English NER training data (a); were not seen in the English NER training data (b); only consist of up to two tokens (c); only consist of Latin characters (d); and occur at least twice in the dev data (e). We only show languages where the sets (a–e) contain at least 100 entities each. We show the difference between (a) and (b) and the minimum difference between (a) and (c-e).

3.7. For simplicity, we count an entity as occurring in the English training data if a subset of at least two tokens matches with an entity in the English training data. As most matching entities in the target language data only consist of up to two tokens, are somewhat frequent, and consist only of Latin characters, we provide the performance on all entities fitting each criterion respectively for comparison. For all target languages in the table except Spanish, entities that appeared in the English training data are more likely to be tagged correctly than ones that did not. The differences are largest for two languages that are typologically distant to English, Indonesian (id) and Swahili (sw). For most languages, entities that appear in the English training data are similarly likely to be correctly classified as entities that are either frequent, appear in Latin characters, or are short. However, for Swahili and Basque (eu), mBERT does much better on entities that appeared in the English training data compared to the comparison entities. Another interesting case is Georgian (ka), which uses a unique script. The NER model is very good at recognizing entities that are written in Latin script but performs less well on entities in Georgian script.

Sentence representations across all layers For sentence retrieval tasks, we analyze whether the multilingual sentence representations obtained from all layers are well-aligned in the embedding spaces. Without fine-tuning on any parallel sentences at all, we explore three ways of extracting the sentence representations from all the models: (1) the embeddings of the first token in the last layer, also known as [CLS] token; (2) the average word embeddings in each layer; (3) the concatenation of the average word embeddings in the bottom, middle, and top 4 layers, i.e., Layer 1 to 4 (bottom), Layer 5 to 8 (middle), Layer 9 to 12 (top). Figure 3.4 shows the F1 scores

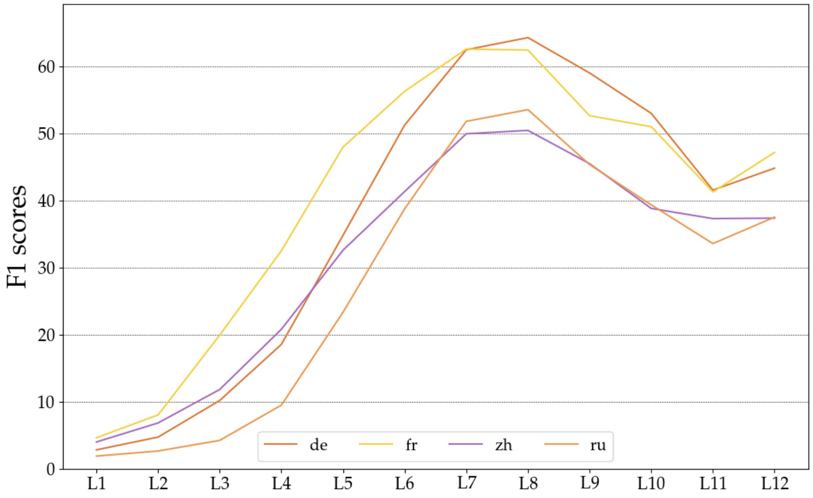


Figure 3.4: Comparison of mBERT's sentence representations by averaging word embeddings in each layer in the BUCC task.

of the average word embeddings in each layer of mBERT in the BUCC task. We observe that the average word embeddings in the middle layers, e.g., Layer 6 to 8, perform better than that in the bottom or the top layers. In Table 3.8, we show the performance of these three types of sentence embeddings in the BUCC task. The embeddings of the CLS token perform relatively bad in cross-lingual retrieval tasks. We conjecture that the CLS embeddings highly abstract the semantic meaning of a sentence, while they lose the token-level information which is important for matching two translated sentences in two languages. With respect to the concatenation of average word embeddings from four continuous layers, We also observe that embeddings from the middle layers perform better than that from the bottom and top layers. Average word embeddings in the middle individual layer perform comparative to the concatenated embeddings from the middle four layers.

3.5 Related Work

Cross-lingual Representations: Early work focused on learning cross-lingual representations using either parallel corpora [68, 122] or a bilingual dictionary to learn a linear transformation [57, 125]. Later approaches reduced the amount of supervision required using self-training [13] and unsupervised strategies such as adversarial training [40], heuristic initialisation [14], and optimal transport [192]. Building on advances in monolingual transfer learning [47, 79, 124,

Type	de	fr	zh	ru
CLS	3.88	4.73	0.89	2.15
Layer 6	51.29	56.32	41.38	38.81
Layer 7	62.51	62.62	49.99	51.84
Layer 8	64.32	62.46	50.49	53.58
Layer 1-4	6.98	12.3	12.05	4.33
Layer 5-8	63.12	63.42	52.84	51.67
Layer 9-12	53.97	52.68	44.18	43.13

Table 3.8: Three types of sentence embeddings from mBERT in BUCC tasks: (1) CLS token embeddings in the last layer; (2) Average word embeddings in the middle layers, i.e., Layer 6, 7, 8; (3) the concatenation of average word embeddings in the continuous four layers, i.e., Layer 1-4 (bottom layers), Layer 5-8 (middle layers), Layer 9-12 (top layers).

139], multilingual extensions of pretrained encoders have recently been shown to be effective for learning deep cross-lingual representations [39, 55, 140, 182].

Cross-lingual Evaluation: One pillar of the evaluation of cross-lingual representations has been translation, either on the word level (*bilingual lexicon induction*) or on the sentence level (*machine translation*). In most cases, evaluation has been restricted to typologically related languages and similar domains; approaches have been shown to fail in less favorable conditions [66, 74, 168]. Past work has also reported issues with common datasets for bilingual lexicon induction [45, 93] and a weak correlation with certain downstream tasks [66]. Translation, however, only covers one facet of a model's cross-lingual generalization ability. For instance, it does not capture differences in classification performance that are due to cultural differences [128, 159].

On the other hand, cross-lingual approaches have been evaluated on a wide range of tasks, including dependency parsing [152], named entity recognition [143], sentiment analysis [20], natural language inference [41], document classification [154], and question answering [16, 111]. Evaluation on a single task is problematic as past work has noted potential issues with standard datasets: MLDoc [154] can be solved by matching keywords [16], while MultiNLI, the dataset from which XNLI [41] was derived, contains superficial cues that can be exploited [72]. Evaluation on multiple tasks is thus necessary to fairly compare cross-lingual models. Benchmarks covering multiple tasks like GLUE [171] and SuperGLUE [170] have arguably spurred research in monolingual transfer learning. In the cross-lingual setting, such a benchmark not only needs to cover a diverse set of tasks but also languages. XTREME aims to fill this gap.

3.6 Discussion and Future Work

As we have highlighted in our analysis, a model's cross-lingual transfer performance varies significantly both between tasks and languages. XTREME is a first step towards obtaining a more accurate estimate of a model's cross-lingual generalization ability. While XTREME is still inherently limited by the data coverage of its constituent tasks for many low-resource languages, XTREME nevertheless provides significantly broader coverage and more fine-grained analysis tools to encourage research on cross-lingual generalization ability of models. We have released the code for XTREME and scripts for fine-tuning models on tasks in XTREME, which should be to catalyze future research. Since the release of our benchmark and code repository, there are several follow-up works built on XTREME. Notably, a series of pre-training techniques and multilingual contextualized representations have been released, including mT5 [185], InfoXLM [31], MARGE [109]. In addition, a series of multilingual datasets have been curated and followed the similar setups as XTREME, such as XCOPA [141], LAReQA [149], [115]. Moreover, XTREME also encourages a thread of more fine-grained analysis of cross-lingual generalization of multilingual representations [105, 151, 174, 196].

Chapter 4

Leveraging Word and Sentence Alignment for Language Understanding

In the previous chapter, we propose a benchmark for evaluating the cross-lingual generalization of pre-trained multilingual encoders. As shown by the results in the previous chapter, pre-trained cross-lingual encoders such as mBERT [47] and XLM-R [42] have proven impressively effective at enabling transfer-learning of NLP systems from high-resource languages to low-resource languages. This success comes despite the fact that there is no explicit objective to align the contextual embeddings of words/sentences with similar meanings across languages together in the same space. In this chapter, we present a new method for learning an **A**ligned **M**ultilingual **B**idirectional **E**ncode**R** (AMBER). AMBER is trained on additional parallel data using two *explicit* alignment objectives that align the multilingual representations at different granularities. We conduct experiments on zero-shot cross-lingual transfer learning for different tasks including sequence tagging, sentence retrieval and sentence classification. Experimental results on the tasks in the XTREME benchmark [82] show that AMBER obtains gains of up to 1.1 average F1 score on sequence tagging and up to 27.3 average accuracy on retrieval over the XLM-R-large model which has 3.2x the parameters of AMBER. Our code and models are available at `http://github.com/junjiehu/amber`.

This work is first appeared in:

- Junjie Hu, Melvin Johnson, Orhan Firat, Aditya Siddhant, and Graham Neubig. Explicit Alignment Objectives for Multilingual Bidirectional Encoders. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

4.1 Overview

Some attempts at training multilingual representations [42, 47] simply train a (masked) language model on monolingual data from many languages. These methods can only *implicitly* learn which words and structures correspond to each-other across languages in an entirely unsupervised fashion, but are nonetheless quite effective empirically [43, 91]. On the other hand, some methods directly leverage multilingual parallel corpora [39, 55, 87, 124], which gives some degree of supervision implicitly aligning the words in the two languages. However, the pressure on the model to learn clear correspondences between the contextualized representations in the two languages is still implicit and somewhat weak. Because of this, several follow-up works [27, 152, 175] have proposed methods that use word alignments from parallel corpora as the supervision signals to align multilingual contextualized representations, albeit in a *post-hoc* fashion.

In this chapter, we propose a training regimen for learning contextualized word representations that encourages symmetry at both the word and sentence levels *at training time*. Our word-level alignment objective is inspired by work in machine translation that defines objectives encouraging consistency between the source-to-target and target-to-source attention matrices [38]. Our sentence-level alignment objective encourages prediction of the correct translations within a mini-batch for a given source sentence, which is inspired by work on learning multilingual sentence representations [178, 187]. In experiments, we evaluate the zero-shot cross-lingual transfer performance of AMBER on four different NLP tasks in the XTREME benchmark [82] including part-of-speech (POS) tagging, paraphrase classification, and sentence retrieval. We show that AMBER obtains gains of up to 1.1 average F1 score on cross-lingual POS tagging, up to 27.3 average accuracy score on sentence retrieval, and achieves competitive accuracy in paraphrase classification when compared with the XLM-R-large model. This is despite the fact that XLM-R-large is trained on data 23.8x as large[1] and has 3.2x parameters of AMBER. This shows that compared to large amounts of monolingual data, even a small amount of parallel data leads to significantly better cross-lingual transfer learning.

4.2 Cross-lingual Alignment

This section describes two objectives for training contextualized embeddings. We denote the monolingual and parallel data as $\mathcal{D}_M$ and $\mathcal{D}_{XY}$ respectively, where M can be either the source

[1] AMBER is trained on 26GB parallel data and 80GB monolingual Wikipedia data, while XLM-R-large is trained on 2.5TB monolingual CommonCrawl data.

language X or the target language Y.

4.2.1 Sentence Alignment

Our first proposed objective encourages cross-lingual alignment of sentence representations. For a source-target sentence pair (x, y) in the parallel corpus, we separately calculate sentence embeddings denoted as $\mathrm{c}_x, \mathrm{c}_y$ by averaging the embeddings in the final layer as the sentence embeddings.[2] We then encourage the model to predict the correct translation y given a source sentence x. To do so, we model the conditional probability of a candidate sentence y being the correct translation of a source sentence x as:

$$P(y|x) = \frac{e^{\mathrm{c}_x^T \mathrm{c}_y}}{\sum_{y' \in \mathcal{D}_M \cup \mathcal{D}_{XY}} e^{\mathrm{c}_x^T \mathrm{c}_{y'}}}, \tag{4.1}$$

where y' can be any sentence in any language. Since the normalization term in Equation (4.1) is intractable, we approximate $P(y|x)$ by sampling y' within a mini-batch $\mathcal{B}$ rather than $\mathcal{D}_M \cup \mathcal{D}_{XY}$. We then define the sentence alignment loss as the average negative log-likelihood of the above probability:

$$\ell_{\mathrm{SA}}(x, y) = -\log P(y|x). \tag{4.2}$$

4.2.2 Bidirectional Word Alignment

Our second proposed objective encourages alignment of word embeddings by leveraging the attention mechanism in the Transformer model. Motivated by the work on encouraging the consistency between the source-to-target and target-to-source translations [38, 78], we create two different attention masks as the inputs to the Transformer model, and obtain two attention matrices in the top layer of the Transformer model. We compute the target-to-source attention matrix $\mathrm{A}_{y \to x}$ as follows:

$$\mathrm{g}_{y_i}^{l} = \mathrm{Attn}(\mathrm{Q} = \mathrm{g}_{y_i}^{l-1}, \mathrm{KV} = \mathrm{g}_{[y_{<i};x]}^{l-1}; W^l), \tag{4.3}$$

$$\mathrm{g}_{x_j}^{l} = \mathrm{Attn}(\mathrm{Q} = \mathrm{g}_{x_j}^{l-1}, \mathrm{KV} = \mathrm{g}_{x}^{l-1}; W^l), \tag{4.4}$$

$$\mathrm{Attn}(\mathrm{QKV}; W) = \mathrm{softmax}(\mathrm{QW}^q (\mathrm{KW}^k)^T)\mathrm{VW}^v, \tag{4.5}$$

$$A_{y \to x}[i, j] = \mathrm{g}_{y_i}^{L} \cdot \mathrm{g}_{x_j}^{L}, \tag{4.6}$$

[2] In comparison, mBERT encodes a sentence pair jointly, then uses the CLS token embedding to perform its next sentence prediction task.

where $\mathrm{g}_{y_t}^{l}$ is the embedding of the t-th word in y on the l-th layer, $A_{y\to x}[i,j]$ is the (i,j)-th value in the attention matrix from y to x, and $W = \{\mathrm{W}^q, \mathrm{W}^k, \mathrm{W}^v\}$ are the linear projection weights for Q, K, V respectively. We compute the source-to-target matrix $\mathrm{A}_{x\to y}$ similarly by switching x and y as follow:.

$$\mathrm{g}_{x_j}^{l} = \mathrm{Attn}(\mathrm{Q} = \mathrm{g}_{x_j}^{l-1}, \mathrm{KV} = \mathrm{g}_{[x_{<j};y]}^{l-1}; W^l), \tag{4.7}$$

$$\mathrm{g}_{y_j}^{l} = \mathrm{Attn}(\mathrm{Q} = \mathrm{g}_{y_i}^{l-1}, \mathrm{KV} = \mathrm{g}_{y}^{l-1}; W^l), \tag{4.8}$$

$$\mathrm{Attn}(\mathrm{QKV}; W) = \mathrm{softmax}(\mathrm{QW}^q(\mathrm{KW}^k)^T)\mathrm{VW}^v, \tag{4.9}$$

$$A_{x\to y}[j,i] = \mathrm{g}_{x_j}^{L} \cdot \mathrm{g}_{y_i}^{L}. \tag{4.10}$$

To encourage the model to align source and target words in both directions, we minimize the distance between the forward and backward attention matrices. Similarly to Cohn et al. [38], we maximize the trace of two attention matrices, i.e., $\mathrm{tr}\,({\mathrm{A}_{y\to x}}^T \mathrm{A}_{x\to y})$. Since the attention scores are normalized in $[0,1]$, the trace of two attention matrices is upper bounded by $\min(|x|,|y|)$, and the maximum value is obtained when the two matrices are identical. Since the Transformer generates multiple attention heads, we average the trace of the bidirectional attention matrices generated by all the heads denoted by the superscript h.

$$\ell_{\mathrm{WA}}(x,y) = 1 - \frac{1}{H}\sum_{h=1}^{H} \frac{\mathrm{tr}\,({\mathrm{A}_{y\to x}^{h}}^T \mathrm{A}_{x\to y}^{h})}{\min(|x|,|y|)}. \tag{4.11}$$

Notably, in the target-to-source attention in Eq (4.3), with attention masking we enforce a constraint that the t-th token in y can only perform attention over its preceding tokens $y_{<t}$ and the source tokens in x. This is particularly useful to control the information access of the query token y_t, in a manner similar to that of the decoding stage of NMT. Without attention masking, the standard Transformer performs self-attention over all tokens, i.e., $Q = K = \mathrm{g}_z^h$, and minimizing the distance between the two attention matrices by Equation (4.11) might lead to a trivial solution where $\mathrm{W}^q \approx \mathrm{W}^k$.

4.2.3 Combined Objective

Finally we combine the masked language modeling objective with the alignment objectives and obtain the total loss in Equation (4.12). Notice that in each iteration, we sample a mini-batch of sentence pairs from $\mathcal{D}_M \cup \mathcal{D}_{XY}$.

$$\mathcal{L} = \mathbb{E}_{(x,y)\in\mathcal{D}_M\cup\mathcal{D}_{XY}}\ \ell_{\mathrm{MLM}}([x;y]) + \mathbb{E}_{(x,y)\in\mathcal{D}_{XY}}\ [\ell_{\mathrm{SA}}(x,y) + \ell_{\mathrm{WA}}(x,y)], \tag{4.12}$$

Model	Data	Langs	Vocab	Layers	Parameters	Ratio
AMBER	Wiki & MT	104	120K	12	172M	1.0
mBERT	Wiki	104	120K	12	172M	1.0
XLM-15	Wiki & MT	15	95K	12	250M	1.5x
XLM-100	Wiki	100	200K	12	570M	3.3x
XLM-R-base	CommonCrawl	100	250K	12	270M	1.6x
XLM-R-large	CommonCrawl	100	250K	24	550M	3.2x
Unicoder	CommonCrawl & MT	100	250K	12	270M	1.6x

Table 4.1: Details of baseline and state-of-the-art models.

where ℓ_{MLM} denotes a masked language model objective (Section 2.3).

4.3 Experiments

4.3.1 Training setup

Following the setting of Hu et al. [82], we focus on the *zero-shot cross-lingual transfer* setting where we fine-tune models on English annotations and apply the models to predict on non-English data.

Models: Table 4.1 shows details of models in comparison. We adopt the same architecture as mBERT for AMBER. Notably, AMBER, XLM-15 and Unicoder are trained on the additional parallel data, while the others are trained only on monolingual data. Besides, XLM-R-base/large models have 2.6x/4.8x the parameters of AMBER and are trained on the larger CommonCrawl corpus. We use a simple setting for our AMBER variants in the ablation study to show the effectiveness of our proposed alignment objectives without other confounding factors such as model sizes, hyper-parameters and tokenizations in different existing studies.

Pre-training: We train AMBER on the Wikipedia data for 1M steps first using the default hyper-parameters as mBERT[3] except that we use a larger batch of 8,192 sentence pairs, as this has proven effective in Liu et al. [118]. We then continue training the model by our objectives for another 1M steps with a batch of 2,048 sentence pairs from Wikipedia corpus and parallel corpus which is used to train XLM-15 [39]. We use the same monolingual data as mBERT and follow

[3] https://github.com/google-research/bert

Conneau and Lample [39] to prepare the parallel data with one change to maintain truecasing. We set the maximum number of subwords in the concatenation of each sentence pair to 256 and use 10k warmup steps with the peak learning rate of 1e-4 and a linear decay of the learning rate. We train AMBER on TPU v3 for about 1 week.

4.3.2 Datasets

Cross-lingual Part-Of-Speech (POS) contains data in 13 languages from the Universal Dependencies v2.3 [134].

PAWS-X [188] is a paraphrase detection dataset. We train on the English data [195], and evaluate the prediction accuracy on the test set translated into 4 other languages.

XNLI [41] is a natural language inference dataset in 15 languages. We train models on the English MultiNLI training data [179], and evaluate on the other 14.

Tatoeba [11] is a testbed for parallel sentence identification. We select the 14 non-English languages covered by our parallel data, and follow the setup in Hu et al. [82] finding the English translation for a given non-English sentence with maximum cosine similarity.

4.3.3 Result Analysis

In Table 4.2, we show the average results over all languages in all the tasks, and show detailed results for each language in Appendix B.2. First, we find that our re-trained mBERT (AMBER with MLM) performs better than the publicly available mBERT on all the tasks, confirming the utility of pre-training BERT models with larger batches for more steps [118]. Second, AMBER trained by the word alignment objective obtains a comparable average F1 score with respect to the best performing model (Unicoder) in the POS tagging task, which shows the effectiveness of the word-level alignment in the syntactic structure prediction tasks at the token level. Besides, it is worth noting that Unicoder is initialized from the larger XLM-R-base model that is pre-trained on a larger corpus than AMBER, and Unicoder improves over XLM-R-base on all tasks. Third, for the sentence classification tasks, AMBER trained with our explicit alignment objectives obtain a larger gain (up to 2.1 average accuracy score in PAWS-X, and 3.9 average accuracy score in XNLI) than AMBER with only the MLM objective. Although we find that AMBER trained with only the MLM objective falls behind existing XLM/XLM-R/Unicoder models with many more parameters, AMBER trained with our alignment objectives significantly narrows the gap of classification accuracy with respect to XLM/XLM-R/Unicoder. Finally, for sentence retrieval

Model	POS	PAWS-X	XNLI	Tatoeba
mBERT (public)	68.5	86.2	65.4	45.6
XLM-15	68.8	88.0	72.6	**77.2**
XLM-100	69.5	86.4	69.1	36.6
XLM-R-base	68.8	87.4	73.4	57.6
XLM-R-large	70.0	**89.4**	**79.2**	60.6
Unicoder	**71.7**	88.1	74.8	72.2
AMBER (MLM)	69.8	87.1	67.7	52.6
AMBER (MLM+TLM)	70.5	87.7	70.9	68.2
AMBER (MLM+TLM+WA)	**71.1**	89.0	71.3	68.8
AMBER (MLM+TLM+WA+SA)	70.5	**89.2**	**71.6**	**87.9**

Table 4.2: Overall results on POS, PAWS-X, XNLI, Tatoeba tasks. Bold numbers highlight the highest scores across languages on the existing models (upper part) and AMBER variants (bottom part).

tasks, we find that XLM-15 and Unicoder are both trained on additional parallel data, outperforming the other existing models trained only on monolingual data. Using additional parallel data, AMBER with MLM and TLM objectives also significantly improves over AMBER with the MLM objective by 15.6 average accuracy score, while combining our word-level alignment objective yields a marginal improvement over AMBER with MLM and TLM objectives. However, adding the sentence-level alignment objective, AMBER trained by the combined objective can further improve AMBER with the MLM and word-level alignment objectives by 19.1 average accuracy score. This confirms our intuition that the explicit sentence-level objective can effectively leverage the alignment supervision in the parallel corpus, and encourage contextualized sentence representations of aligned pairs to be close according to the cosine similarity metric.

4.3.4 How does alignment help by language?

In Figure 4.1, we investigate the improvement of the alignment objectives over the MLM objective on low-resource and high-resource languages, by computing the performance difference between AMBER trained with alignment objectives and AMBER (MLM). First, we find that AMBER trained with alignment objectives significantly improves the performance on languages with relatively small amounts of parallel data, such as Turkish, Urdu, Swahili, while the improvement

Methods	en	bg	de	el	es	fr	Avg.
Cao et al. [27]	80.1	73.4	73.1	71.4	75.5	74.5	74.7
AMBER (full)	84.7	74.3	74.2	72.5	76.9	76.6	76.5

Table 4.3: F1 scores of AMBER trained with all objectives and Cao et al. [27] on 6 languages on XNLI.

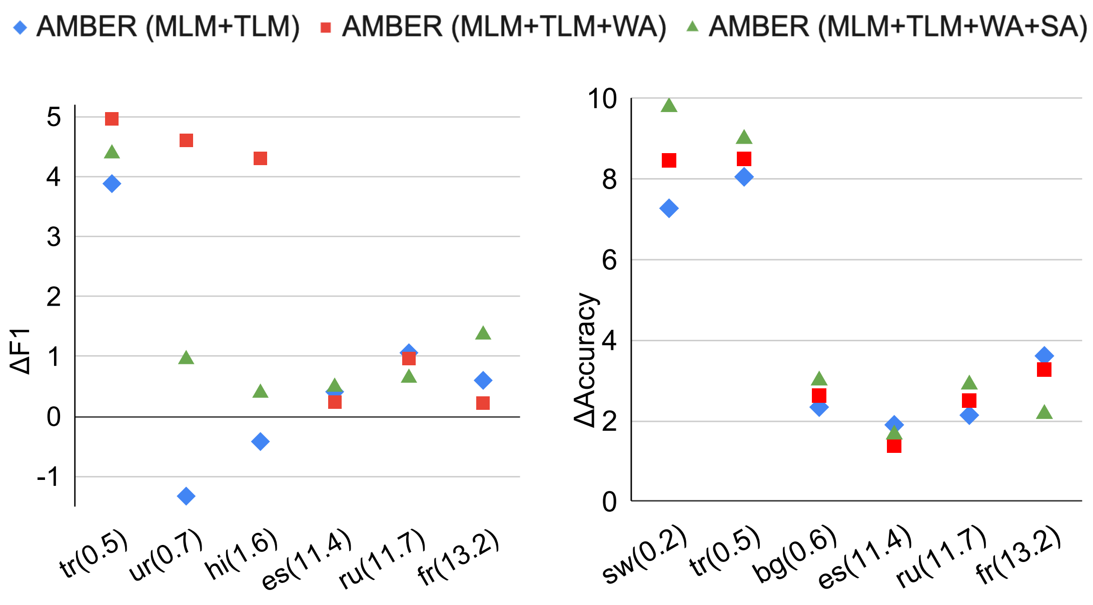
on high-resource languages is marginal. Through a further analysis (Appendix B.2), we observe that AMBER (MLM) performs worse on these low-resource and morphologically rich languages than on high-resource Indo-European languages, while AMBER trained with alignment objectives can effectively bridge the gap. Moreover, AMBER trained with our word-level alignment objective yields the highest improvement on these low-resource languages on the POS task, and AMBER trained with sentence-level alignment performs the best on XNLI.

4.3.5 Alignment with Attention vs Dictionary

Recent studies [27, 175] have proposed to use a bilingual dictionary to align cross-lingual word representations. Compared with these methods, our word-level alignment objective encourages the model to automatically discover word alignment patterns from the parallel corpus in an end-to-end training process, which avoids potential errors accumulated in separate steps of the pipeline. Furthermore, an existing dictionary may not have all the translations for source words, especially for words with multiple senses. Even if the dictionary is relatively complete, it also requires a heuristic way to find the corresponding substrings in the parallel sentences for alignment. If we use a word alignment tool to extract a bilingual dictionary in a pipeline, errors may accumulate, hurting the accuracy of the model. Besides, Wang et al. [175] is limited in aligning only fixed contextual embeddings from the model's top layer. Finally, we also compare AMBER trained with all the objectives and Cao et al. [27] on a subset of languages on XNLI in Table 4.3. We find that our full model obtains a gain of 1.8 average F1 score.

4.4 Related Work

Cross-lingual Alignment: While cross-lingual alignment is a long-standing challenge dating back to the early stage of research in word alignment [26], cross-lingual embeddings [42, 47, 57, 184] are highly promising in their easy integration into neural network models for a variety of



(a) Part-of-Speech

(b) XNLI

Figure 4.1: Performance difference between AMBER trained with alignments on parallel data and AMBER (MLM). Languages are sorted by no. of parallel data (Million) used for training AMBER with alignments.

cross-lingual applications. In particular, to improve cross-lingual transfer, some attempts directly leverage multilingual parallel corpus to train contextualized representations [39, 55, 87, 124] with the hope of aligning words implicitly. The other line of work uses word alignments from parallel corpora as the alignment supervision in a post-hoc fashion [27, 175]. Notably, AMBER does not rely on any word alignment tools, and explicitly encourages the correspondence both on the word and sentence level.

4.5 Discussion and Future Work

In this chapter, we demonstrate the effectiveness of our proposed explicit alignment objectives in learning better cross-lingual representations for downstream tasks. Nonetheless, several challenging and promising directions can be considered in the future. First, most existing multilingual models tokenize words into subword units, which makes the alignment less interpretable. How to align a span of subword units with meaningful semantics at the phrase level deserves further investigation. Second, several studies [64, 112] have shown that attention may fail to capture word alignment for some language pairs, and a few works [6, 108] proposed neural word alignment to improve the word alignment quality. Incorporating such recent advances into the alignment objective is one future direction. Third, how to fine-tune a well-aligned multilingual model on English annotations without catastrophic forgetting of the alignment information is a potential way to improve cross-lingual generalization on the downstream applications.

Chapter 5

Leveraging Word Alignment for Domain Adaptation of Machine Translation

In the previous chapter, we introduce methods to leverage word and sentence alignments for multilingual language understanding tasks. In this chapter, we switch our focus to the other data discrepancy problem – domain shift for language generation, and use neural machine translation (NMT), i.e., a typical language generation task, as a concrete example. In particular we focus on translations of out-of-vocabulary (OOV) words and propose an unsupervised adaptation method which fine-tunes a pre-trained out-of-domain NMT model using a pseudo-in-domain corpus. Later on, in Chapter 7, we extend the adaption setting by introducing human translations with a limited amount of annotation budget, and propose a hybrid active learning strategy to improve the domain robustness of NMT. This work has been published in:

- Junjie Hu, Mengzhou Xia, Graham Neubig, Jaime Carbonell. Domain Adaptation of Neural Machine Translation by Lexicon Induction. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics 2019 [81].

5.1 Overview

Previous work in the context of phrase-based statistical machine translation [46] has noted that unseen (OOV) words account for a large portion of translation errors when switching to new domains. However, this problem of OOV words in cross-domain transfer is under-examined in the context of NMT, where both training methods and experimental results will differ greatly. In

Code/scripts are released at `https://github.com/junjiehu/dali`.

this chapter, we try to fill this gap, examining domain adaptation methods for NMT specifically focusing on correctly translating unknown words. Specifically, we tackle the task of *data-based, unsupervised* adaptation, a strict unsupervised setting where we have no in-domain parallel sentences.

To remedy this problem, we propose a new data-based method for unsupervised adaptation that specifically focuses on the unknown word problem: **domain adaptation by lexicon induction (DALI)**. Our proposed method leverages large amounts of monolingual data to find translations of in-domain unseen words, and constructs a pseudo-parallel in-domain corpus via word-for-word back-translation of monolingual in-domain target sentences into source sentences. More specifically, we leverage existing supervised [183] and unsupervised [40] lexicon induction methods that project source word embeddings to the target embedding space, and find translations of unseen words by their nearest neighbors. For supervised lexicon induction, we learn such a mapping function under the supervision of a seed lexicon extracted from out-of-domain parallel sentences using word alignment. For unsupervised lexicon induction, we follow Conneau et al. [40] to infer a lexicon by adversarial training and iterative refinement.

In the experiments on German-to-English translation across five domains (Medical, IT, Law, Subtitles, and Koran), we find that DALI improves both RNN-based [18] and Transformer-based [166] models trained on an out-of-domain corpus with gains as high as 14 BLEU. When the proposed method is combined with back-translation, we can further improve performance by up to 4 BLEU. Further analysis shows that the areas in which gains are observed are largely orthogonal to back-translation; our method is effective in translating in-domain unseen words, while back-translation mainly improves the fluency of source sentences, which helps the training of the NMT decoder.

5.2 Domain Adaptation by Lexicon Induction

Our method works in two steps: (1) we use lexicon induction methods to learn an in-domain lexicon from in-domain monolingual source data $\mathcal{D}_{\text{src-in}}$ and target data $\mathcal{D}_{\text{tgt-in}}$ as well as out-of-domain parallel data $\mathcal{D}_{\text{parallel-out}}$, (2) we use this lexicon to create a pseudo-parallel corpus for MT.

5.2.1 Lexicon Induction

Given separate source and target word embeddings, X, Y $\in \mathbb{R}^{d \times N}$, trained on all available monolingual source and target sentences across all domains, we leverage existing lexicon induction methods that perform *supervised* [183] or *unsupervised* [40] learning of a mapping $f(\mathrm{X}) = \mathrm{WX}$ that transforms source embeddings to the target space, then selects nearest neighbors in embedding space to extract translation lexicons.

Supervised Embedding Mapping Supervised learning of the mapping function requires a seed lexicon of size n, denoted as $L = \{(s, t)_i\}_{i=1}^{n}$. We represent the source and target word embeddings of the i-th translation pair $(s, t)_i$ by the i-th column vectors of $\mathrm{X}^{(n)}, \mathrm{Y}^{(n)} \in \mathbb{R}^{d \times n}$ respectively. Xing et al. [183] show that by enforcing an orthogonality constraint on $\mathrm{W} \in O_d(\mathbb{R})$, we can obtain a closed-form solution from a singular value decomposition (SVD) of $\mathrm{Y}^{(n)}\mathrm{X}^{(n)^T}$:

$$\begin{aligned} \mathrm{W}^* &= \arg \max_{\mathrm{W} \in O_d(\mathbb{R})} \|\mathrm{Y}^{(n)} - \mathrm{WX}^{(n)}\|_F = \mathrm{UV}^T \\ \mathrm{U\Sigma V}^T &= \mathrm{SVD}(\mathrm{Y}^{(n)}\mathrm{X}^{(n)^T}). \end{aligned} \tag{5.1}$$

In a domain adaptation setting we have parallel out-of-domain data $\mathcal{D}_{\text{parallel-out}}$, which can be used to extract a seed lexicon. Algorithm 2 shows the procedure of extracting this lexicon. We use the word alignment toolkit GIZA++ [135] to extract word translation probabilities $P(t|s)$ and $P(s|t)$ in both forward and backward directions from $\mathcal{D}_{\text{parallel-out}}$, and extract lexicons $L_{\text{fw}} = \{(s, t),\ \forall P(t|s) > 0\}$ and $L_{\text{bw}} = \{(s, t),\ \forall P(s|t) > 0\}$. We take the union of the lexicons in both directions and further prune out translation pairs containing punctuation that is non-identical. To avoid multiple translations of either a source or target word, we find the most common translation pairs in $\mathcal{D}_{\text{parallel-out}}$, sorting translation pairs by the number of times they occur in $\mathcal{D}_{\text{parallel-out}}$ in descending order, and keeping those pairs with highest frequency in $\mathcal{D}_{\text{parallel-out}}$.

Unsupervised Embedding Mapping For unsupervised training, we follow Conneau et al. [40] in mapping source word embeddings to the target word embedding space through adversarial training. Details can be found in the reference, but briefly a discriminator is trained to distinguish between an embedding sampled from WX and Y, and W is trained to prevent the discriminator from identifying the origin of an embedding by making WX and Y as close as possible.

Algorithm 2 Supervised lexicon extraction

```
    Input: Parallel out-of-domain data $\mathcal{D}_{\text{parallel-out}}$
    Output: Seed lexicon $L = \{(s,t)\}_{i=1}^{n}$
Run GIZA++ on $\mathcal{D}_{\text{parallel-out}}$ to get $L_{\text{fw}}, L_{\text{bw}}$
$L_g = L_{\text{fw}} \cup L_{\text{bw}}$
Remove pairs with punctuation only in either $s$ and $t$ from $L_g$
Initialize a counter $C[(s,t)] = 0\ \forall (s,t) \in L_g$
for (src, tgt) $\in \mathcal{D}_{\text{parallel-out}}$ do
    for $(s,t) \in L_g$ do
        if $s \in$ src and $t \in$ tgt then
            $C[(s,t)] = C[(s,t)] + 1$
Sort $C$ by its values in the descending order
$L = \{\}, S = \{\}, T = \{\}$
for $(s,t) \in C$ do
    if $s \notin S$ and $t \notin T$ then
        $L = L \cup \{(s,t)\}$
        $S = S \cup \{s\},\ T = T \cup \{t\}$
return $L$
```

Induction Once we obtain the matrix W either from supervised or unsupervised training, we map all the possible in-domain source words to the target embedding space. We compute the nearest neighbors of an embedding by a distance metric, Cross-Domain Similarity Local Scaling (CSLS; Conneau et al. [40]):

$$\mathrm{CSLS}(\mathrm{Wx}, \mathrm{y}) = 2\cos(\mathrm{Wx}, \mathrm{y}) - r_T(\mathrm{Wx}) - r_S(\mathrm{y})$$

$$r_T(\mathrm{Wx}) = \frac{1}{K} \sum_{\mathrm{y}' \in \mathcal{N}_T(\mathrm{Wx})} \cos(\mathrm{Wx}, \mathrm{y}')$$

where $r_T(\mathrm{Wx})$ and $r_S(\mathrm{y})$ measure the average cosine similarity between their K nearest neighbors in the source and target spaces respectively.

To ensure the quality of the extracted lexicons, we only consider mutual nearest neighbors, i.e., pairs of words that are mutually nearest neighbors of each other according to CSLS. This significantly decreases the size of the extracted lexicon, but improves the reliability.

5.2.2 NMT Data Generation and Training

Finally, we use this lexicon to create pseudo-parallel in-domain data to train NMT models. Specifically, we follow Sennrich et al. [156] in back-translating the in-domain monolingual target sentences to the source language, but instead of using a pre-trained target-to-source NMT system, we simply perform word-for-word translation using the induced lexicon L. Each target word in the target side of L can be deterministically back-translated to a source word, since we take the nearest neighbor of a target word as its translation according to CSLS. If a target word is not mutually nearest to any source word, we cannot find a translation in L and we simply copy this target word to the source side. We find that more than 80% of the words can be translated by the induced lexicons. We denote the constructed pseudo-parallel in-domain corpus as $\mathcal{D}_{\text{pseudo-parallel-in}}$.

During training, we first pre-train an NMT system on an out-of-domain parallel corpus $\mathcal{D}_{\text{parallel-out}}$, and then fine-tune the NMT model on a constructed parallel corpus. More specifically, to avoid overfitting to the extracted lexicons, we sample an equal number of sentences from $\mathcal{D}_{\text{parallel-out}}$, and get a fixed subset $D'_{\text{parallel-out}}$, where $|D'_{\text{parallel-out}}| = |\mathcal{D}_{\text{pseudo-parallel-in}}|$. We concatenate $D'_{\text{parallel-out}}$ with $\mathcal{D}_{\text{pseudo-parallel-in}}$, and fine-tune the NMT model on the combined corpus.

5.3 Experimental Results

5.3.1 Data

We follow the same setup and train/dev/test splits of Koehn and Knowles [97], using a German-to-English parallel corpus that covers five different domains. Data statistics are shown in Table 5.1. Note that these domains are very distant from each other. Following Koehn and Knowles [97], we process all the data with byte-pair encoding [157] to construct a vocabulary of 50K subwords. To build an unaligned monolingual corpus for each domain, we randomly shuffle the parallel corpus and split the corpus into two parts with equal numbers of parallel sentences. We use the target and source sentences of the first and second halves respectively. We combine all the unaligned monolingual source and target sentences on all five domains to train a skip-gram model using *fasttext* [22]. We obtain source and target word embeddings in 512 dimensions by running 10 epochs with a context window of 10, and 10 negative samples.

Corpus	Words	Sentences	W/S
Medical	12,867,326	1,094,667	11.76
IT	2,777,136	333,745	8.32
Subtitles	106,919,386	13,869,396	7.71
Law	15,417,835	707,630	21.80
Koran	9,598,717	478,721	20.05

Table 5.1: Corpus statistics over five domains.

5.3.2 Main Results

We first compare DALI with other adaptation strategies on both RNN-based and Transformer-based NMT models.

Table 5.2 shows the performance of the two models when trained on one domain (columns) and tested on another domain (rows). We fine-tune the unadapted baselines using pseudo-parallel data created by DALI. We use the unsupervised lexicon here for all settings, and leave a comparison across lexicon creation methods to Table 5.3. Based on the last two columns in Table 5.2, DALI substantially improves both NMT models with average gains of 2.79-7.54 BLEU over the unadapted baselines.

We further compare DALI with two popular data-based unsupervised adaptation methods that leverage in-domain monolingual target sentences: (1) a method that copies target sentences to the source side (Copy; Currey et al. [44]) and (2) back-translation (BT; Sennrich et al. [156]), which translates target sentences to the source language using a backward NMT model. We compare DALI with supervised (DALI-S) and unsupervised (DALI-U) lexicon induction. Finally, we (1) experiment with when we directly extract a lexicon from an in-domain corpus using GIZA++ (DALI-GIZA++) and Algorithm 2, and (2) list scores for when systems are trained directly on in-domain data (In-domain). For simplicity, we test the adaptation performance of the LSTM-based NMT model, and train an LSTM-based NMT with the same architecture on out-of-domain corpus for English-to-German back-translation.

First, DALI is competitive with BT, outperforming it on the medical domain, and underperforming it on the other three domains. Second, the gain from DALI is orthogonal to that from BT – when combining the pseudo-parallel in-domain corpus obtained from DALI-U with that from BT, we can further improve by 2-5 BLEU points on three of four domains. Second, the gains through the usage of both DALI-U and DALI-S are surprisingly similar, although the lexicons induced by these two methods have only about 50% overlap. Detailed analysis of two lexicons

Domain	Method		Medical	IT	Subtitles	Law	Koran	Avg.	Gain
Medical	LSTM	Unadapted	46.19	4.62	2.54	7.05	1.25	3.87	+4.31
		DALI	-	11.32	7.79	9.72	3.85	**8.17**	
	XFMR	Unadapted	49.66	4.54	2.39	7.77	0.93	3.91	+4.79
		DALI	-	10.99	8.25	11.32	4.22	**8.70**	
IT	LSTM	Unadapted	7.43	57.79	5.49	4.10	2.52	4.89	+5.98
		DALI	20.44	-	9.53	8.63	4.85	**10.86**	
	XFMR	Unadapted	6.96	60.43	6.42	4.50	2.45	5.08	+5.76
		DALI	19.49	-	10.49	8.75	4.62	**10.84**	
Subtitles	LSTM	Unadapted	11.36	12.27	27.29	10.95	10.57	11.29	+2.79
		DALI	21.63	12.99	-	11.50	10.17	**16.57**	
	XFMR	Unadapted	16.51	14.46	30.71	11.55	12.96	13.87	+3.85
		DALI	26.17	17.56	-	13.96	13.18	**17.72**	
Law	LSTM	Unadapted	15.91	6.28	4.52	40.52	2.37	7.27	+4.85
		DALI	24.57	10.07	9.11	-	4.72	**12.12**	
	XFMR	Unadapted	16.35	5.52	4.57	46.59	1.82	7.07	+6.17
		DALI	26.98	11.65	9.14	-	5.15	**13.23**	
Koran	LSTM	Unadapted	0.63	0.45	2.47	0.67	19.40	1.06	+6.56
		DALI	12.90	5.25	7.49	4.80	-	**7.61**	
	XFMR	Unadapted	0.00	0.44	2.58	0.29	15.53	0.83	+7.54
		DALI	14.27	5.24	9.01	4.94	-	**8.37**	

Table 5.2: BLEU scores of LSTM based and Transformer (XFMR) based NMT models when trained on one domain (columns), and tested on another domain (rows). The last two columns show the average performance of unadapted baselines and DALI, and the average gains.

can be found in Section 5.3.5.

5.3.3 Word-level Translation Accuracy

Since our proposed method focuses on leveraging word-for-word translation for data augmentation, we analyze the word-for-word translation accuracy for unseen in-domain words. A source word is considered as an unseen in-domain word when it never appears in the out-of-domain corpus. We examine two questions: (1) How much does each adaptation method improve the translation accuracy of unseen in-domain words? (2) How does the frequency of the in-domain word affect its translation accuracy?

To fairly compare various methods, we use a lexicon extracted from the in-domain parallel data with the GIZA++ alignment toolkit as a reference lexicon L_g. For each unseen in-domain source word in the test file, when the corresponding target word in L_g occurs in the output, we

	Medical	Subtitles	Law	Koran
Unadapted	7.43	5.49	4.10	2.52
Copy	13.28	6.68	5.32	3.22
BT	18.51	11.25	11.55	**8.18**
DALI-U	20.44	9.53	8.63	4.90
DALI-S	19.03	9.80	8.64	4.91
DALI-U+BT	**24.34**	**13.35**	**13.74**	8.11
DALI-GIZA++	28.39	9.37	11.45	8.09
In-domain	46.19	27.29	40.52	19.40

Table 5.3: Comparison among different methods on adapting NMT from IT to {Medical, Subtitles, Law, Koran} domains, along with two oracle results

consider it as a "hit" for the word pair.

First, we compare the percentage of successful in-domain word translations across all adaptation methods. Specifically, we scan the source and reference of the test set to count the number of valid hits C, then scan the output file to get the count C_t in the same way. Finally, the hit percentage is calculated as $\frac{C_t}{C}$. The results on experiments adapting IT to other domains are shown in Figure 5.1. The hit percentage of the unadapted output is extremely low, which confirms our assumption that in-domain word translation poses a major challenge in adaptation scenarios. We also find that all augmentation methods can improve the translation accuracy of unseen in-domain words but our proposed method can outperform all others in most cases. The unseen in-domain word translation accuracy is quantitatively correlated with the BLEU scores, which shows that correctly translating in-domain unseen words is a major factor contributing to the improvements seen by these methods.

Second, to investigate the effect of frequency of word-for-word translation, we bucket the unseen in-domain words by their frequency percentile in the pseudo-in-domain training dataset, and calculate the average translation accuracy of unseen in-domain words within each bucket. The results are plotted in Figure 5.2 in which the x-axis represents each bucket within a range of frequency percentile, and the y-axis represents the average translation accuracy. With the increasing frequency of words in the pseudo-in-domain data, the translation accuracy also increases, which is consistent with our intuition that the neural network would be able to remember high-frequency tokens better. Since the absolute value of the occurrences is different among all domains, the numerical values of accuracy within each bucket vary across domains, but all lines follow the ascending pattern.

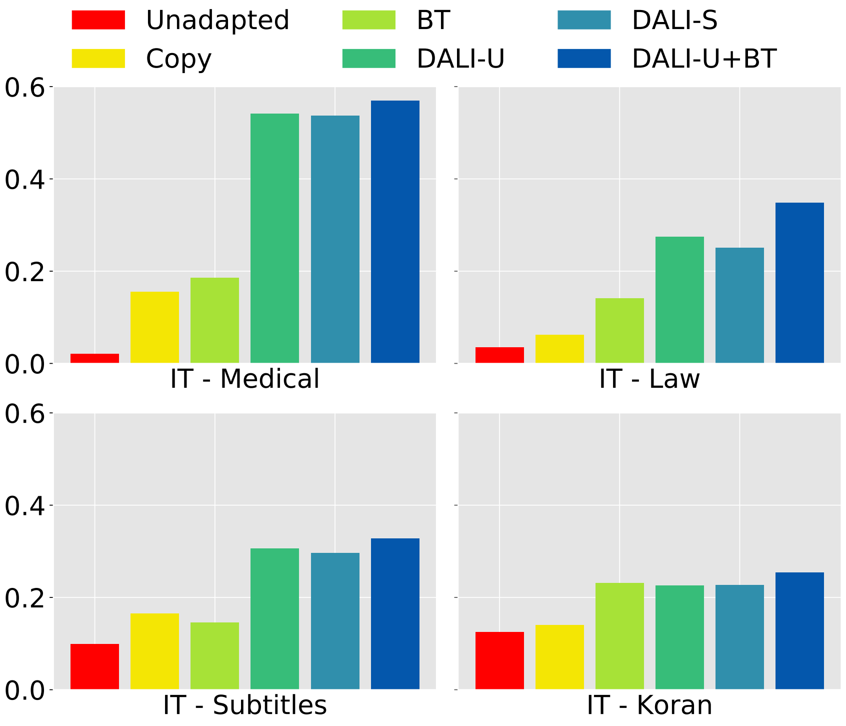


Figure 5.1: Translation accuracy of in-domain words of the test set on several data augmentation baseline and our proposed method with IT as the out domain

5.3.4 When do Copy, BT and DALI Work?

From Figure 5.1, we can see that Copy, BT and DALI all improve the translation accuracy of in-domain unseen words. In this section, we explore exactly what types of words each method improves on. We randomly pick some in-domain unseen word pairs which are translated 100% correctly in the translation outputs of systems trained with each method. We also count these word pairs' occurrences in the pseudo-in-domain training set. The examples are demonstrated in Table 5.5.

We find that in the case of Copy, over 80% of the successful word translation pairs have the same spelling format for both source and target words, and almost all of the rest of the pairs share subword components. In short, and as expected, Copy excels on improving the accuracy of words that have identical forms on the source and target sides.

As expected, our proposed method mainly increases the translation accuracy of the pairs in our induced lexicon. It also leverages the subword components to successfully translate compound words. For example, "monotherapie" does not occur in our induced lexicon, but the model is still able to translate it correctly based on its subwords "mono@@" and "therapie" by leveraging the

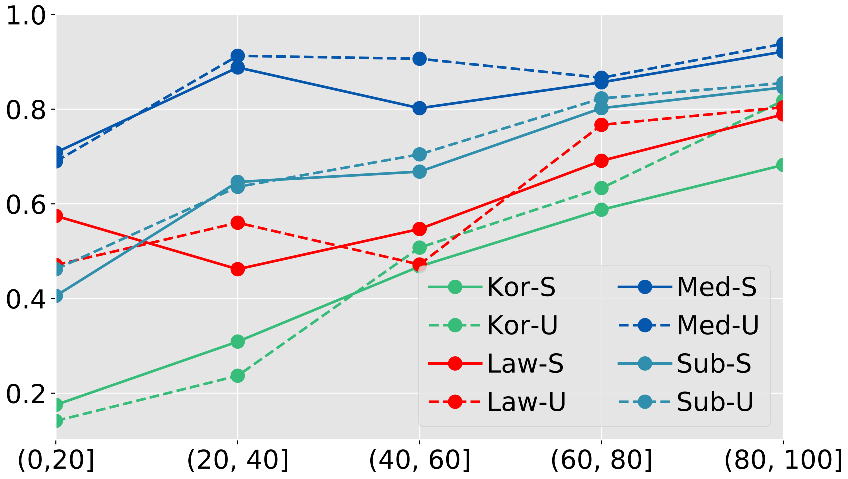


Figure 5.2: Translation accuracy of in-domain unseen words in the test set with regards to the frequency percentile of lexicon words inserted in the pseudo-in-domain training corpus.

BT-S	es ist eine Nachricht , die die aktive Substanz **enthält** .	BT-T	Invirase is a **medicine** containing the active substance saquinavir .
Test-S	ABILIFY ist ein **Arzneimittel** , das den Wirkstoff Aripiprazol **enthält** .	Test-T	Prevenar is a **medicine** containing the design of Arixtra .

Table 5.4: An example that shows why BT could translate the OOV word “Arzneimittel” correctly into “medicine”. “enthált” corresponds to the English word “contain”. Though BT can’t translate a correct source sentence for augmentation, it generates sentences with certain patterns that could be identified by the model, which helps translate in-domain unseen words.

successfully induced pair “therapie” and “therapy”.

It is more surprising to find that adding a back-translated corpus significantly improves the model’s ability to translate in-domain unseen words correctly, even if the source word never occurs in the pseudo-in-domain corpus. Even more surprisingly, we find that the majority of the correctly translated source words are not segmented at all, which means that the model does not leverage the subword components to make correct translations. In fact, for most of the correctly translated in-domain word pairs, the source words are never seen during training. To further analyze this, we use our BT model to do word-for-word translation for these individual words without any other context, and the results turn out to be extremely bad, indicating that the model does not actually find the correspondence of these word pairs. Rather, it relies solely on the decoder to make the correct translation on the target side for test sentences with related target sentences in

Type	Word Pair	Count
Copy	(tremor, tremor)	452
	(347, 347)	18
BT	(ausschuss, committee)	0
	(apotheker, pharmacist)	0
	(toxizität, toxicity)	0
DALI	(müdigkeit, tiredness)	444
	(therapie, therapy)	9535
	(monotherapie, monotherapy)	0

Table 5.5: 100% successful word translation examples from the output of the IT to Medical adaptation task. The Count column shows the number of occurrences of word pairs in the pseudo-in-domain training set.

the training set. To verify this, Table 5.4 demonstrates an example extracted from the pseudo-in-domain training set. BT-T shows a monolingual in-domain target sentence and BT-S is the back-translated source sentence. Though the back translation fails to generate any in-domain words and the meaning is unfaithful, it succeeds to generate a similar sentence pattern as the correct source sentence, which is "... ist eine (ein) ... , die (das) ... enthält .". The model can easily detect the pattern through the attention mechanism and translate the highly related word "medicine" correctly.

From the above analysis, it can be seen that the improvement brought by the augmentation of BT and DALI is largely orthogonal. The former utilizes the highly related contexts to translate unseen in-domain words while the latter directly injects reliable word translation pairs into the training corpus. This explains why we get further improvements over either single method alone.

5.3.5 Lexicon Coverage

Intuitively, with a larger lexicon, we would expect a better adaptation performance. In order to examine this hypothesis, we do experiments using pseudo-in-domain training sets generated by our induced lexicon with various coverage levels. Specifically, we split the lexicon into 5 folds randomly and use a portion of it comprising folds 1 through 5, which correspond to 20%, 40%, 60%, 80% and 100% of the original data. We calculate the coverage of the words in the Medical test set comparing with each pseudo-in-domain training set. We use each training set to train a model and get its corresponding BLEU score. From Figure 5.3, we find that the proportion of the

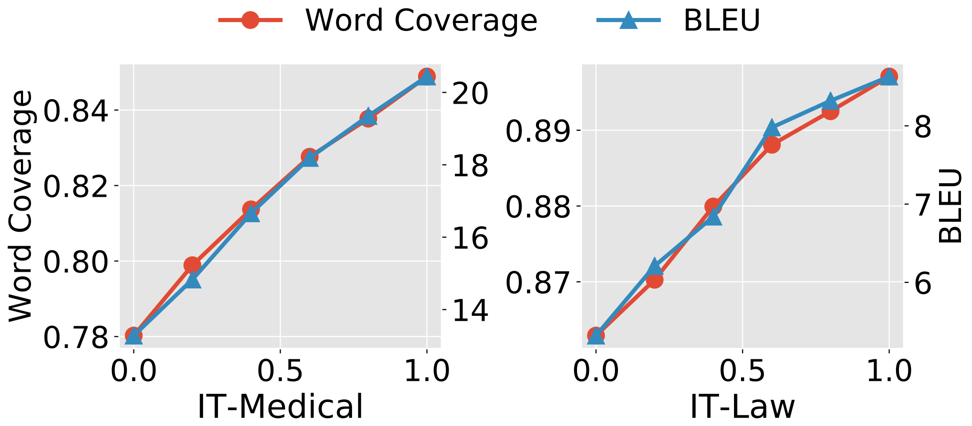


Figure 5.3: Word coverage and BLEU score of the Medical test set when the pseudo-in-domain training set is constructed with different level of lexicon coverage.

Source	ABILIFY ist ein Arzneimittel , das den Wirkstoff Aripiprazol enthält .	BLEU
Reference	abilify is a medicine containing the active substance aripiprazole .	1.000
Unadapted	the time is a figure that corresponds to the formula of a formula .	0.204
Copy	abilify is a casular and the raw piprexpression offers .	0.334
BT	prevenar is a medicine containing the design of arixtra .	0.524
DALI	abilify is a arzneimittel that corresponds to the substance ariprazole .	0.588
DALI+BT	abilify is a arzneimittel , which contains the substance aripiprazole .	0.693

Table 5.6: Translation outputs from various data augmentation method and our method for IT→Medical adaptation.

used lexicon is highly correlated with both the known word coverage in the test set and its BLEU score, indicating that by inducing a larger and more accurate lexicon, further improvements can likely be made.

5.3.6 Semi-supervised Adaptation

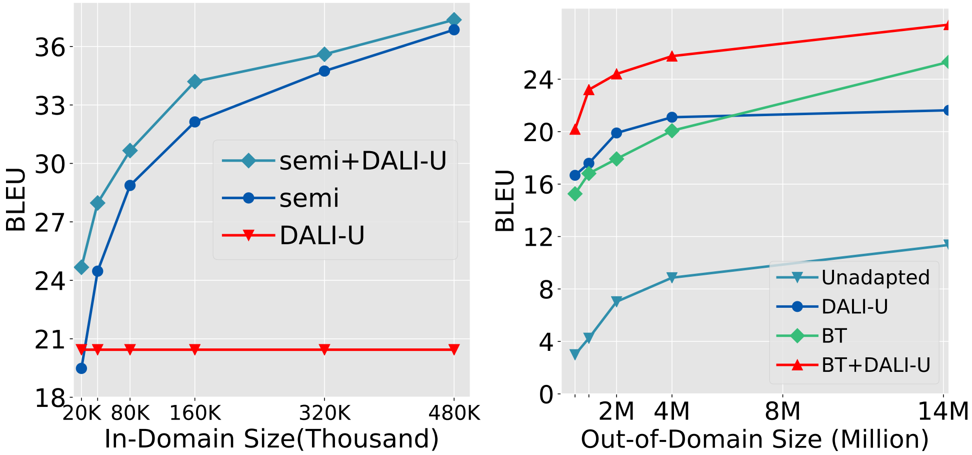
Although we target *unsupervised* domain adaptation, it is also common to have a limited amount of in-domain parallel sentences in a semi-supervised adaptation setting. To measure the efficacy of DALI in this setting, we first pre-train an NMT model on a parallel corpus in the IT domain, and adapt it to the medical domain. The pre-trained NMT obtains 7.43 BLEU scores on the medical test set. During fine-tuning, we sample 330,278 out-of-domain parallel sentences, and concatenate them with 547,325 pseudo-in-domain sentences generated by DALI and the real in-domain sentences. We also compare the performance of fine-tuning on the combination of the out-of-domain parallel sentences with only real in-domain sentences. We vary the number of real in-domain sentences in the range of [20K, 40K, 80K, 160K, 320K, 480K]. In Figure 5.4a, semi-supervised adaptation outperforms unsupervised adaptation after we add more than 20K real in-domain sentences. As the number of real in-domain sentences increases, the BLEU scores on the in-domain test set improve, and fine-tuning on both the pseudo and real in-domain sentences further improves over fine-tuning sorely on the real in-domain sentences. In other words, given a reasonable number of real in-domain sentences in a common semi-supervised adaptation setting, DALI is still helpful in leveraging a large number of monolingual in-domain sentences.

5.3.7 Effect of Out-of-Domain Corpus

The size of data that we use to train the unadapted NMT and BT NMT models varies from hundreds of thousands to millions, and covers a wide range of popular domains. Nonetheless, the unadapted NMT and BT NMT models can both benefit from training on a large out-of-domain corpus. We examine the question: how does fine-tuning on weak and strong unadapted NMT models affect the adaptation performance? To this end, we compare DALI and BT on adapting from subtitles to medical domains, where the two largest corpora in subtitles and medical domains have 13.9 and 1.3 million sentences. We vary the size of the out-of-domain corpus in a range of $[0.5, 1, 2, 4, 13.9]$ million, and fix the number of in-domain target sentences to 0.6 million. In Figure 5.4b, as the size of out-of-domain parallel sentences increases, we have a stronger upadapted NMT which consistently improves the BLEU score of the in-domain test set. Both DALI and BT also benefit from adapting a stronger NMT model to the new domain. Combining DALI with BT further improves the performance, which again confirms our finding that the gains from DALI and BT are orthogonal to each other. Having a stronger BT model improves the quality of synthetic data, while DALI aims at improving the translation accuracy of OOV words by explicitly injecting their translations.

5.3.8 Effect of Domain Coverage

We further test the adaptation performance of DALI when we train our base NMT model on the WMT14 German-English parallel corpus. The corpus is a combination of Europarl v7, Common Crawl corpus and News Commentary, and consists of 4,520,620 parallel sentences from a wider range of domains. In Table 5.7, we compare the BLEU scores of the test sets between the unadapted NMT and the adapted NMT using DALI-U. We also show the percentage of source



(a) IT-Medical (b) Subtitles-Medical

Figure 5.4: Effect of training on increasing number of in-domain (a) and out-of-domain (b) parallel sentences

words or subwords in the training corpus of five domains being covered by the WMT14 corpus. Although the unadapted NMT system trained on the WMT14 corpus obtains higher scores than that trained on the corpus of each individual domain, DALI still improves the adaptation performance over the unadapted NMT system by up to 5 BLEU score.

5.3.9 Qualitative Examples

Finally, we show outputs generated by various data augmentation methods. Starting with the unadapted output, we can see that the output is totally unrelated to the reference. By adding the

Domain	Base	DALI	Word	Subword
Medical	28.94	30.06	44.1%	69.1%
IT	18.27	23.88	45.1%	77.4%
Subtitles	22.59	22.71	35.9%	62.5%
Law	24.26	24.55	59.0%	73.7%
Koran	11.64	12.19	83.1%	74.5%

Table 5.7: BLEU scores of LSTM based NMT models when trained on WMT14 De-En data (Base), and adapted to one domain (DALI). The last two columns show the percentage of source word/subword overlap between the training data on the WMT domain and other five domains.

copied corpus, words that have the same spelling in the source and target languages e.g. “abilify” are correctly translated. With back translation, the output is more fluent; though keywords like “abilify” are not well translated, in-domain words that are highly related to the context like “medicine” are correctly translated. DALI manages to translate in-domain words like “abilify” and “substance”, which are added by DALI using the induced lexicon. By combining both BT and DALI, the output becomes fluent and also contains correctly translated in-domain keywords of the sentence.

5.4 Related Work

Data Augmentation: Early studies on data-based methods such as self-enhancing [101, 153] translate monolingual source sentences by a statistical machine translation system, and continue training the system on the synthetic parallel data. Recent data-based methods such as back-translation [156] and copy-based methods [44] mainly focus on improving fluency of the output sentences and translation of identical words, while our method targets OOV word translation. In addition, there have been several attempts to do data augmentation using monolingual source sentences [33, 191]. Besides, model-based methods change model architectures to leverage monolingual corpus by introducing an extra learning objective, such as auto-encoder objective [30] and language modeling objective [145]. Another line of research on using monolingual data is unsupervised machine translation [15, 102, 103, 189]. These methods use word-for-word translation as a component, but require a careful design of model architectures, and do not explicitly tackle the domain adaptation problem. Our proposed data-based method does not depend on model architectures, which makes it orthogonal to these model-based methods.

Out-of-Vocabulary Word Translations: Daumé III and Jagarlamudi [46] induce lexicons for unseen words and construct phrase tables for statistical machine translation. However, it is nontrivial to integrate lexicon into NMT models that lack explicit use of phrase tables. With regard to NMT, Arthur et al. [17] use a lexicon to bias the probability of the NMT system and show promising improvements. Luong and Manning [120] propose to emit OOV target words by their corresponding source words and do post-translation for those OOV words with a dictionary. Fadaee et al. [56] proposes an effective data augmentation method that generates sentence pairs containing rare words in synthetically created contexts, but this requires parallel training data not available in the fully unsupervised adaptation setting. Arcan and Buitelaar [9] leverage a domain-specific lexicon to replace unknown words after decoding. Zhao et al. [197] design a contextual memory module in an NMT system to memorize translations of rare words. Kothur et al. [99] treats an annotated lexicon as parallel sentences and continues training the NMT system on the lexicon. Though all these works leverage a lexicon to address the problem of OOV words, none specifically target translating in-domain OOV words under a domain adaptation setting.

5.5 Discussion and Future Work

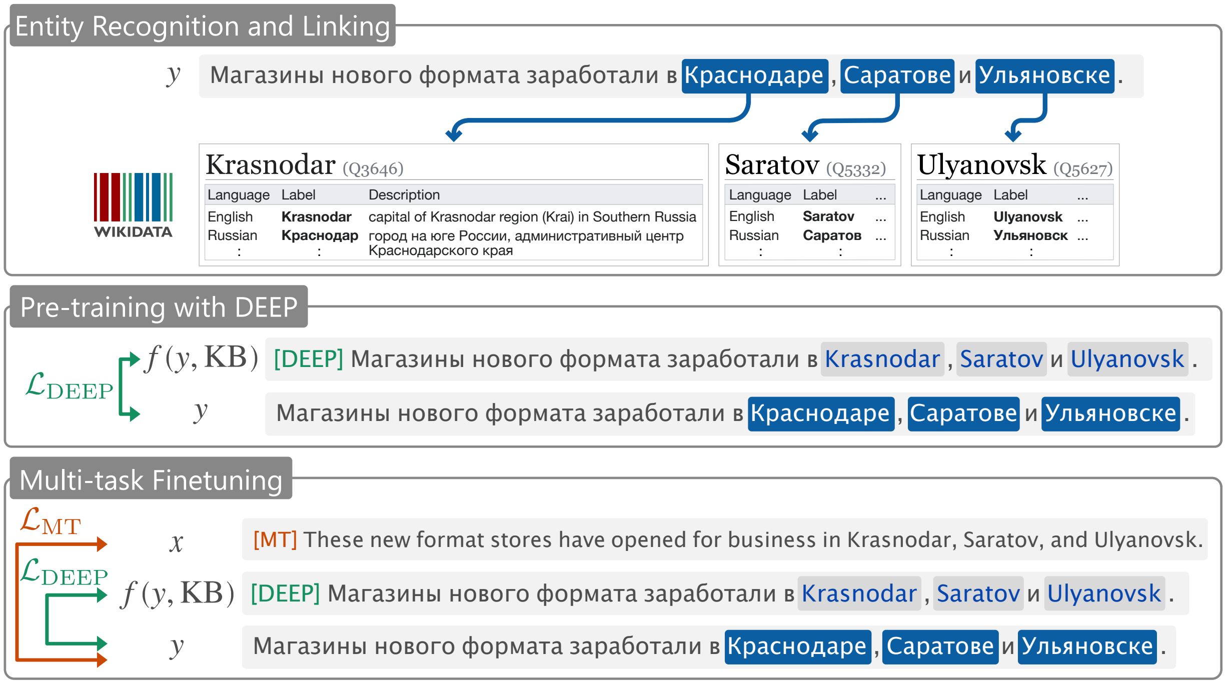
In this chapter, we propose a *data-based, unsupervised adaptation* method that focuses on domain adaption by lexicon induction (DALI) for mitigating unknown word problems in NMT. We conduct extensive experiments to show consistent improvements of two popular NMT models through the usage of our proposed method. Further analysis shows that our method is effective in fine-tuning a pre-trained NMT model to correctly translate unknown words when switching to new domains.

Chapter 6

Leveraging Aligned Entities for Machine Translation

In the previous chapter, we have shown that neural machine translation models usually perform poorly on out-of-vocabulary words when switching domains. We also discover that many of these out-of-vocabulary words are infrequent named entities. 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 a **DE**noising **E**ntity **P**re-training method (DEEP) 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 fine-tunes 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] This work is written in:

- Junjie Hu, Hiroaki Hayashi, Kyunghyun Cho, Graham Neubig. DEEP: DEnoising Entity Pre-training for Neural Machine Translation. *(Under review)*.


Entity Recognition and Linking
y
WIKIDATA
Pre-training with DEEP
$\mathcal{L}_{\text{DEEP}}$
$f(y, \text{KB})$
y
Multi-task Finetuning
$\mathcal{L}_{\text{MT}}$
$\mathcal{L}_{\text{DEEP}}$
x
$f(y, \text{KB})$
y

Магазины нового формата заработали в Краснодаре, Саратове и Ульяновске.

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**	...
:	:	

[DEEP] Магазины нового формата заработали в Krasnodar , Saratov и Ulyanovsk .

Магазины нового формата заработали в Краснодаре, Саратове и Ульяновске.

[MT] These new format stores have opened for business in Krasnodar, Saratov, and Ulyanovsk.

[DEEP] Магазины нового формата заработали в Krasnodar , Saratov и Ulyanovsk .

Магазины нового формата заработали в Краснодаре, Саратове и Ульяновске.

Figure 6.1: 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 fine-tuning in a multi-task learning manner.

6.1 Overview

A 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 [4, 5, 94]. In addition, a growing number of new named entities (e.g., person name, location) appear every day, and as a consequence many of these entities may not exist in the parallel data traditionally used to train MT systems. As a consequence, even state-of-the-art MT systems struggle with entity translation. For example, Laubli et al. [104] note that a Chinese-English news translation system that had allegedly reached human parity still lagged far behind human translators on accurate translations of entities, and this problem will be further exacerbated in the settings of cross-domain transfer 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 [117], earlier studies on named entity translation

[1] All code/data/models will be released upon acceptance.

largely focused on rule-based methods [169], statistical alignment methods [84, 85] and Web mining methods [86, 181, 186]. 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 [29, 84, 85].

In this chapter, 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 fine-tuning neural machine translation (NMT) models that is used by many recent works [119, 120, 132, 160]. 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 attempts to reconstruct parts of text [160] or the whole sentences [119] 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 [167]) that contains multilingual translations of these entities. We then generate noised sentences by replacing the extracted 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 fine-tunes 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 fine-tuning strategy improves the translation accuracy of the entities that do not exist in the fine-tuning data.

6.2 Denoising Entity Pre-training

Our method adopts a procedure of pre-training and fine-tuning for neural machine translation. First, we apply an entity linker to identify entities in a monolingual corpus and link them to a knowledge base (Section 6.2.1). We then utilize entity translations in the knowledge base to create noisy code-switched data for pre-training (Section 6.2.2). Finally, we examine a multi-task learning strategy to further improve the translation of low-frequency entities (Section 7.4).

6.2.1 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 [167] a multilingual knowledge base that covers 94M entities.[2] 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,

$$\forall t \in y, \exists e \in \text{KB} : T_e = \text{surface}(e, \text{KB}),\ t \in T_e, \tag{6.1}$$

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 [148],[3] which builds an entity linker for arbitrary languages available in Wikipedia.

6.2.2 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 an 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:

$$\text{replace}(y, \text{KB}) = \text{swap}(s, t, y),\ \forall t \in y, \tag{6.2}$$

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 6.1, the named entities "Краснодаре, Саратове and Ульяновске" in Russian are replaced by their English translations "Krasnodar, Saratov, and Ulyanovsk". After the replacement, we create a noised code-switched segment which explicitly includes the translations of named entities in the context of the target language. For

[2] Statistics as of June 14, 2021.

[3] https://github.com/google/sling.

some segments that contain fewer entities, their code-switched segments may be similar to them, which potentially results in an 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 then randomly mask the other non-entity words to make sure that about 35% of the words are either replaced or masked in the noised segment. Finally, we follow Liu et al. [119] 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:

$$\mathcal{L}_{\text{DEEP}}(\mathcal{D}_Y, \text{KB}) = \sum_{y \in \mathcal{D}_Y} \log P(y \mid f(y, \text{KB})). \tag{6.3}$$

6.2.3 Multi-task fine-tuning

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

$$\mathcal{L}_{\text{MT}}(\mathcal{D}_{XY}) = \sum_{(x,y) \in \mathcal{D}_{XY}} \log P(y \mid x). \tag{6.4}$$

To avoid forgetting the entity information learned from the pre-training stage, we examine a multitask 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 fine-tuning leads to a 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 fine-tuning as follows:

$$\mathcal{L}_{\text{Multi-task}} = \mathcal{L}_{\text{MT}}(\mathcal{D}_{XY}) + \mathcal{L}_{\text{Pre-train}}(\mathcal{D}'_Y), \tag{6.5}$$

where the pre-training objective $\mathcal{L}_{\text{Pre-train}}$ is either DAE defined in Equation 2.7 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 fine-tuning setting with respect to that in the single-task fine-tuning setting. Therefore, we make sure that the models are fine-tuned

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 6.1: 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.

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 fine-tuning, 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 (i.e., "[MT]", "[DAE]" or "[DEEP]"). We initialize the additional task embeddings by the start token embedding and append these task embeddings to the word embedding matrix of the encoder.

6.3 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 6.1. We tokenize the text using the same sentencepiece model as Liu et al. [119], and train on sequences of 512 subwords.

fine-tuning & Test Data: We use the news commentary data from the English-Russian translation task in WMT18 for fine-tuning 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 [162] for fine-tuning and testing. For English-Nepali translation, we use the FLORES dataset in Guzmán et al. [74] and follow the paper's setting to fine-tune on parallel data in the OPUS repository. Table 6.2 shows the data statistics of the parallel data for fine-tuning. Notice that from the last four columns of Table 6.2, the entities in the pre-training data cover at least 87% of the entity types and 91% of the entity counts in both fine-tuning and test data except the En-Ne pair.

Lang.	Train	Dev	Test	PF / F		PT / 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 6.2: Statistics of the parallel train/dev/test data for fine-tuning. Type and Count under PF/F (PT/T) show the percentage of entity types and counts in the fine-tuning (test) data that are covered by the pre-training data.

Architecture: We use a standard sequence-to-sequence Transformer model [165] 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. [119], 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. [119].

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 fine-tune the model for each translation task.
- **DAE → MT**: We pre-train a model by DAE using the two noising functions in Liu et al. [119] and fine-tune the model for each translation task.
- **DEEP → MT**: We pre-train a model using our proposed DEEP objective and fine-tune the model on the translation task.
- **DAE → DAE+MT**: We pre-train a model by the DAE objective and fine-tune the model for both the DAE task and translation task.
- **DEEP → DEEP+MT**: We pre-train a model by the DEEP objective and fine-tune the model for both the DEEP task and translation task.

Learning & Decoding: We pre-train all models for 50K steps first using the default parameters in Liu et al. [119] except that we use a smaller batch of 64 text segments, each of which has 512 subwords. We use the Adam optimizer (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 approximately 12 hours for 50K steps.[4] We then reset the learning rate scheduler and continue fine-tuning our pre-trained models on the MT parallel data for 40K steps. 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. 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 BLEU following the evaluation in Liu et al. [119].

6.4 Discussion

6.4.1 Corpus-level Evaluation

In Table 6.3, we compare all methods in terms of BLEU 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 substantial gain of 3.5 BLEU points in the single task setting for the low-resource En-Ne translation. Second, we observe improvements with the multi-task fine-tuning strategy over the single-task fine-tuning for all language pairs. 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 fine-tuning process, however the gap between both methods decreases as the fine-tuning proceeds for longer steps (See Figure 6.2). 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 fine-tuning stage. As the multitask fine-tuning 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.

6.4.2 Entity Translation Accuracy

Since corpus-level metrics like BLEU 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

[4] As we show in Figure 6.9, models pre-trained for 50K steps have provided a reasonably good initialization.

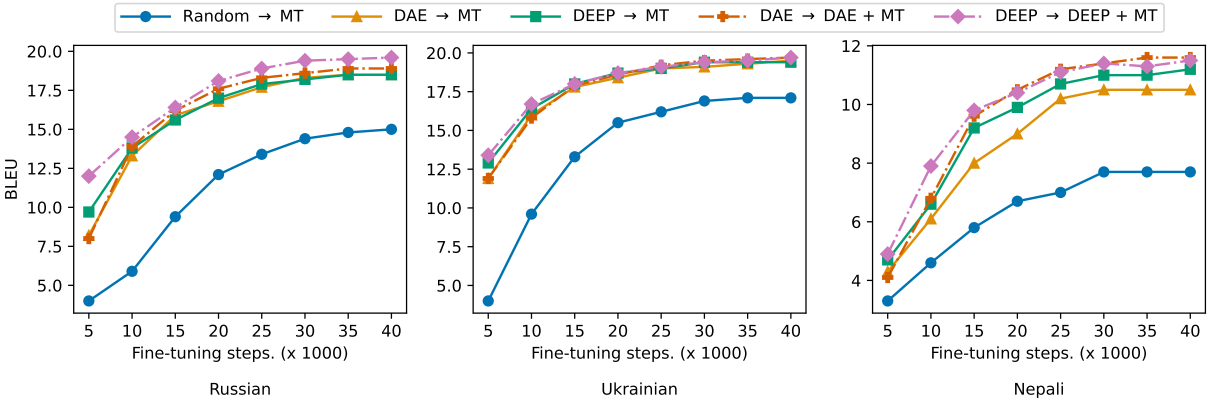


Figure 6.2: **BLEU** scores for 3 language pairs over various fine-tuning steps.

Pre-train → fine-tune	En-Uk	En-Ru	En-Ne
Random → MT	17.1	15.0	7.7
DAE → MT	19.5	18.5	10.5
DEEP → MT	19.4	18.5	11.2
DAE → DAE+MT	**19.7**	18.9	**11.6**
DEEP → DEEP+MT	**19.7**	**19.6**	11.5

Table 6.3: BLEU in single- and multi-task settings.

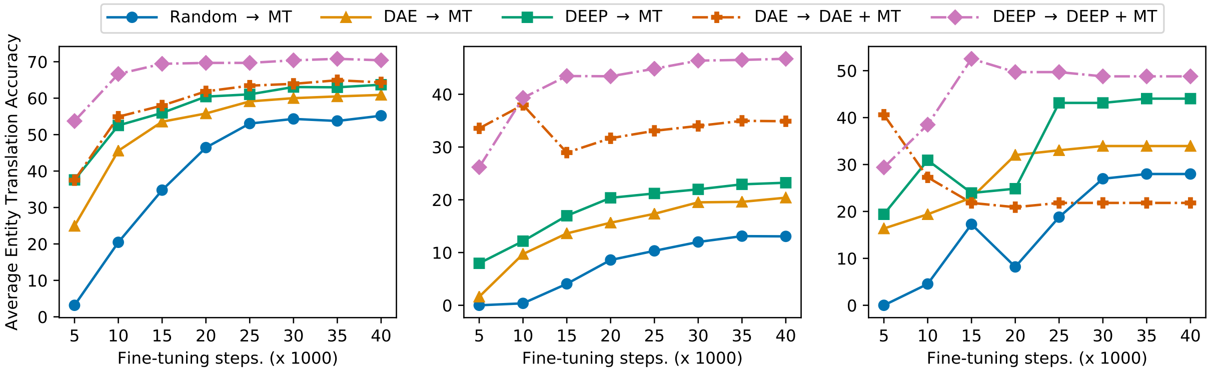
hypotheses, followed by macro-averaging to obtain the average entity translation accuracy. We show the results in Table 6.4. First, our method in both single- and multi-task settings significantly outperformed the other baselines. In particular, the gains from DEEP are much clear for the En-Uk and En-Ru 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 fine-tuning and test data as reported in Table 6.2. 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.

6.4.3 Fine-grained Analysis on Entity Translation Accuracy

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

Pre-train → fine-tune	En-Uk	En-Ru	En-Ne
Random → MT	49.5	31.1	20.9
DAE → MT	56.7	37.7	26.0
DEEP → MT	57.7	40.6	**28.6**
DAE → DAE+MT	58.8	47.2	27.9
DEEP → DEEP+MT	**61.9**	**56.4**	28.3

Table 6.4: Entity translation accuracy in single- and multi-task settings.



Entities in **PFT** data.

Entities in **PT** data.

Entities in **FT** data.

Figure 6.3: Entity translation accuracy scores aggregated over different entity sets for Russian. **PFT**, **PT**, **FT** data correspond to entities appearing in (i) pre-training, fine-tuning and test data, (ii) only pre-training and test data (iii) only fine-tuning and test data.

Performance of Entity Groups over fine-tuning: The model is exposed to some entities more often than others at different stages: pre-training, fine-tuning 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, fine-tuning, and test data.
- **PT**: entities only in the pre-training and test data.
- **FT**: entities only in the fine-tuning and test data.

We show the English-to-Russian entity translation accuracy scores for each group over fine-tuning steps in Figure 6.3. Overall, accuracies are higher for the entities that appear in the fine-tuning data (**PFT**, **FT**), which is due to the exposure to the fine-tuning data. Our proposed method consistently outperformed baseline counterparts in both single- and multi-task settings. The differences in accuracy are particularly large at earlier fine-tuning steps, which indicates the utility of our method in lower-resource settings with little fine-tuning data. The effect of multi-task fine-tuning is most notable for entities in **PT**. Multi-task fine-tuning continuously exposes the model to the pre-training data, which as a result 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 the translated hypotheses from a checkpoint with 40K steps of fine-tuning, 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.

Russian: Figure 6.4 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 fine-tuning 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 fine-tuning 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 fine-tuning data (*i.e.* leftmost column). Multi-task learning with DEEP therefore prevents the model from forgetting the entity translations learned at pre-training time.

Ukrainian: As seen in Figure 6.7, the general trend for the entity translation accuracy according to entity groups are similar to that of Russian. Notice that empty cells in the heatmaps are due to no entities that meet the conditions in those cells. 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 fine-tuning data (See the left bottom of Figure 6.5).

Nepali: While outperforming at the beginning of fine-tuning, Figure 6.8 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

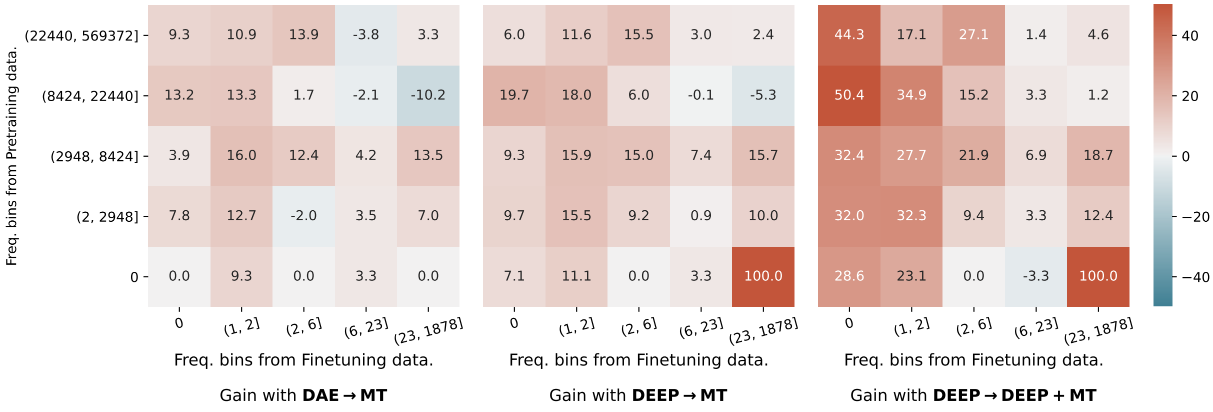


Figure 6.4: Gain from Random → MT in entity translation accuracy for each model.

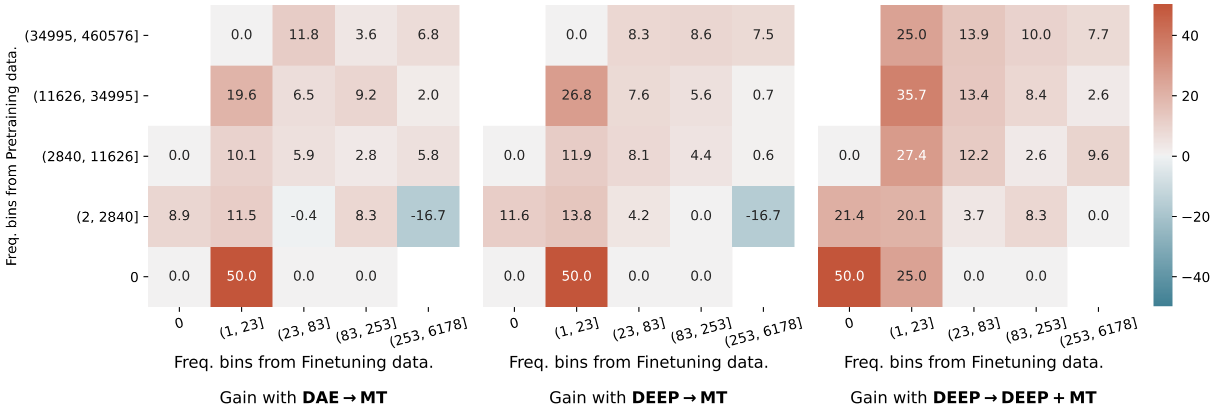
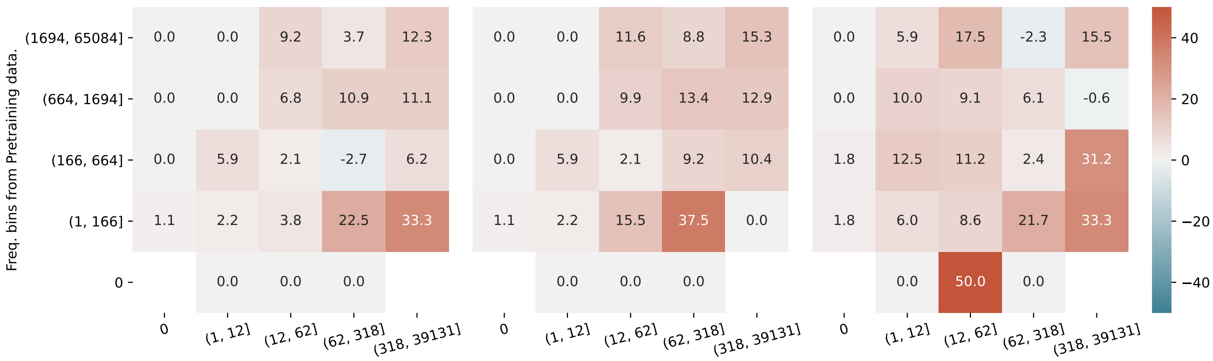


Figure 6.5: Gain from **Random** → **MT** in entity translation accuracy for **Ukrainian** for each model.

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 6.6 shows the results. In the single-task setting, DEEP improves the translations of frequent entities appearing in both the pre-training and fine-tuning 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 fine-tuning data.

6.4.4 Optimization Effects on DEEP

fine-tuning Data Size vs Entity Translation: While DEEP primarily focuses on the application in a low-resource setting, the evaluation with more resources can highlight potential use in broader scenarios. To this end, we expand the fine-tuning data for English-Russian translation

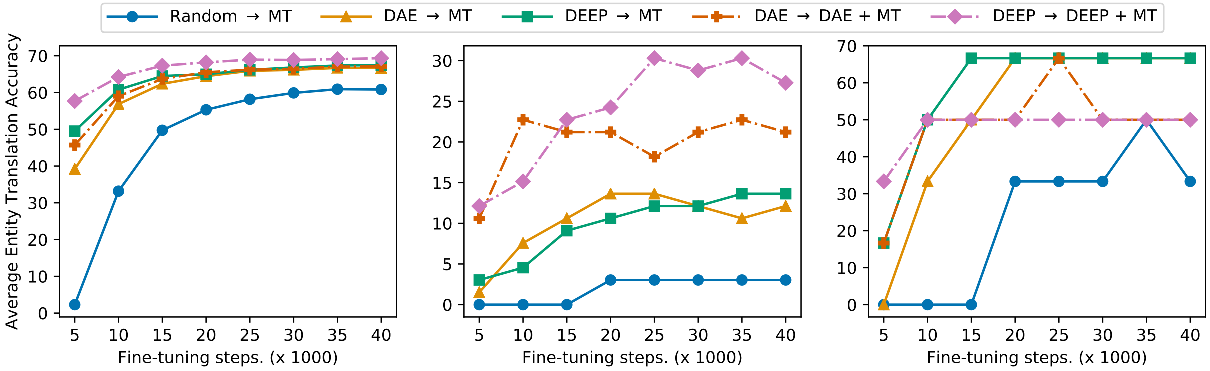


Gain with **DAE → MT**

Gain with **DEEP → MT**

Gain with **DEEP → DEEP + MT**

Figure 6.6: Gain from **Random → MT** in entity translation accuracy for **Nepali** for each model.

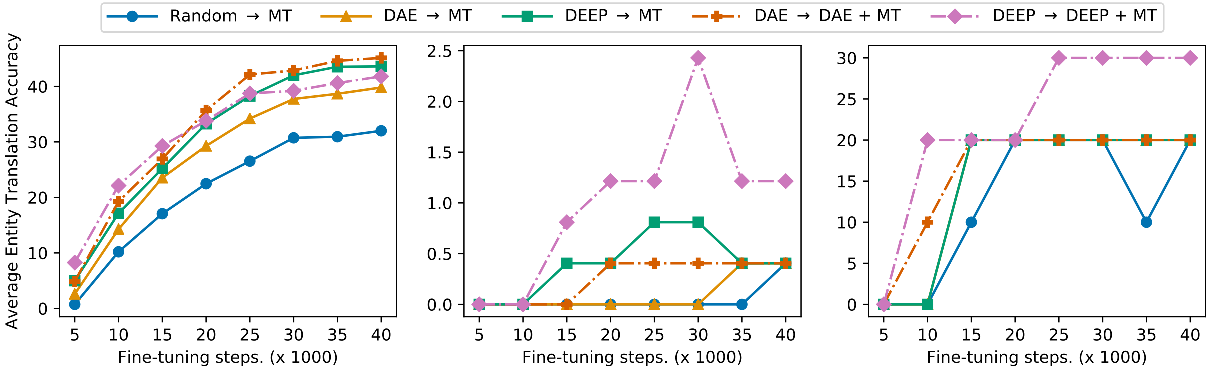


Entities in **PFT** data.

Entities in **PT** data.

Entities in **FT** data.

Figure 6.7: **Entity translation accuracy** aggregated over different entity sets for **Ukrainian**.



Entities in **PFT** data.

Entities in **PT** data.

Entities in **FT** data.

Figure 6.8: **Entity translation accuracy** aggregated over different entity sets for **Nepali**.

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 6.5: Model comparisons across different fine-tuning data sizes. The results on the right are obtained after fine-tuning on the combined news commentary and ParaCrawl data.

with an additional 4 million sentence pairs from ParaCrawl [19], a parallel data collected from web pages. Although web pages might contain news text, the ParaCrawl data cover more general domains. We fine-tune models on the combined data and evaluate with BLEU and entity translation accuracy. Table 6.5 shows the model comparisons across different fine-tuning data sizes. When the model is initialized with pre-training methods, we observed decreased BLEU points and the increased entity translation accuracy scores. On the one hand, this is partly due to the discrepancy in terms of domains between our fine-tuning data (news) and ParaCrawl. Regardless, DEEP is consistently equal to or better than DAE in all tested settings.

Pre-training Steps vs Entity Translation: Since DEEP leverages entity-augmented monolingual data, the model trained by DEEP revisits more entities in different contexts as the pre-training steps increase. To analyze the efficiency of learning name entity translations during the pre-training stage, we focus on the question: *how many pre-training steps are needed for named entity translation?* To examine this question, we take the saved checkpoints trained by DEEP from various pre-training steps, and apply the single-task fine-tuning strategy on the checkpoints for another 40K steps. We plot the entity translation accuracy and BLEU of the test data in Figure 6.9. 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 keeps improving as the pre-training steps increase to 200K steps.

6.4.5 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

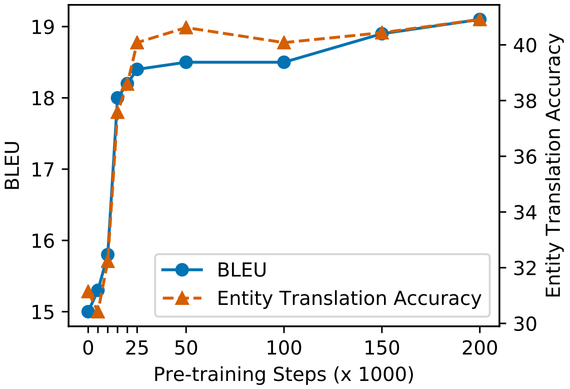


Figure 6.9: English-to-Russian BLEU and Entity translation accuracy scores after fine-tuning with respect to variable pre-training steps. fine-tuning is performed for 40K steps.

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.

6.5 Related Work

Named Entity Translation: Earlier studies on named entity translation [8, 94] focus on rule-based methods using phoneme or grapheme [4, 169], statistical methods that align entities in parallel corpus [84, 85, 193] and Web mining methods built on top of a search engine [86, 181, 186]. Recently, neural models have been applied for named entity translations. Finch et al. [58], Grundkiewicz and Heafield [70], Hadj Ameur et al. [75] used neural machine translation to transliterate named entities. Torregrosa et al. [163], Ugawa et al. [164] integrated named entity tags to neural

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 6.6: 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.

machine translation models. In this chapter, without changing model architectures, we focus on data augmentation methods to improve named entity translation within context. In addition, while recent work shows that continue fine-tuning a pre-trained encoder with the same pre-training objective improves language understanding tasks [73], this fine-tuning 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 [49, 71, 191, 200], while we examine a multi-task fine-tuning strategy with an entity-based denoising task in this work and demonstrate substantial improvements for named entity translations.

6.6 Discussion and Future Work

In this chapter, 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 settings. Despite the effectiveness, several challenging and promising directions can be considered in the future. First, recent works on integrating knowledge graphs [198, 199] in neural machine translation 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, fine-tuning pre-trained models on in-domain text data is a potential way to improve entity translations across domains.

Chapter 7

Leveraging Phrase Alignment for Machine Translation

In this chapter, we further explore the application scenario where we have a limited budget to annotate a small amount of in-domain phrases or sentences for domain adaptation of NMT. This work first appeared in:

- Junjie Hu, Graham Neubig. Phrase-level Active Learning for Neural Machine Translation. *arXiv preprint arXiv:2106.11375 2021 (Under review).*

7.1 Overview

One typical way to address the domain shift problem for machine translation is adding in-domain data to the MT training process [36, 120]. However, this data may not be available *a priori*, and hiring professional translators with knowledge of specific domains (such as medicine or law) is usually costly. As a result, active learning approaches [21, 62, 76] have been widely adopted to reduce the annotation cost by translating a smaller representative subset of the in-domain data, with the hope that models trained on this translated subset approximate those trained on a much larger labeled set. In general, active learning (AL) approaches iterate between two steps: *data selection/annotation*, and *model update*. With regards to data selection for machine translation, most existing works [76, 138, 190] focus on selecting *sentences* that are most useful for training either phrase-based machine translation (PBMT) or neural machine translation (NMT) models.

However, even the most informative sentences inevitably involve segments that the MT system can already translate well, and asking the translator to also translate these segments is not

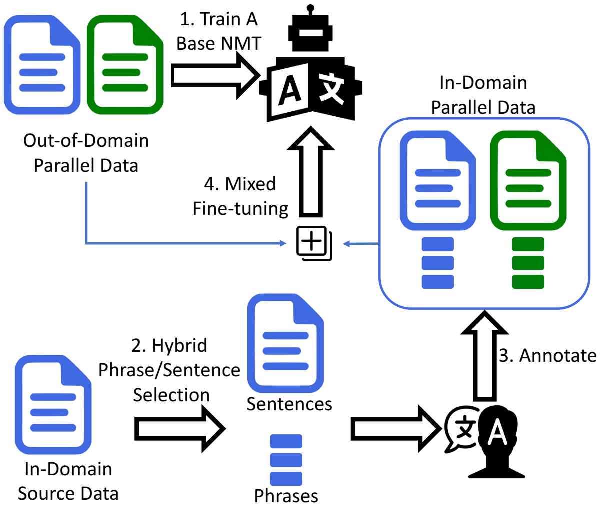


Figure 7.1: Overview of the active learning process

cost-effective. There have been a few works used in conjunction with older PBMT models that ameliorate this problem through phrase-based selection techniques [21, 46, 127], which select only *individual phrases*, maximizing information gain. However, while these translated phrases can be easily integrated into PBMT by adding them to the existing phrase table, incorporating them into NMT models is less simple because NMT has no concept of a "phrase table" and must be trained on full sentences similar to those that must be translated.

In this chapter, we propose a method for incorporating phrase-based active learning into NMT. Specifically, we first describe sentence-based and phrase-based selection strategies, then propose a hybrid strategy that combines both methods. We also describe several ways to incorporate this translated data into the training of NMT systems. We conduct experiments on German-English translation by adapting NMT models trained on WMT parallel data to the medicine and IT domains. Experimental results show that the hybrid selection strategy obtains more stable translation performance than either phrase-based or sentence-based selection strategy.

7.2 Problem Definition

In the setting of active learning for domain adaptation, we are given an out-of-domain labelled corpus $(x, y) \in \mathcal{L}$ and an in-domain unlabelled corpus $x \in \mathcal{U}$. We define a phrase as a contiguous sequence of words up to some length limit N, and denote a set of possible phrases in a sentence x by $\cup_{n\in[1,N]} n\text{-gram}(x)$, where we set $N = 4$ in all experiments below. To obtain translations of unlabelled data, we assume access to professional translators $O(\cdot)$ who can translate source-side sentences $\mathcal{S}$ and/or phrases $\mathcal{P}$ selected from $\mathcal{U}$, i.e., $\mathcal{O}(x)\ \forall x \in \mathcal{S} \subset \mathcal{U}$, and $\mathcal{O}(p)\ \forall p \in \mathcal{P} \subset \mathcal{P}_{\mathcal{U}} = \cup_{x\in\mathcal{U}} \cup_{n\in[1,N]} n\text{-gram}(x)$. We assume that translating sentences or phrases requires cost $c(\cdot)$, and annotation must be performed within a fixed budget $B = \sum_{x\in\mathcal{S}} c(x) + \sum_{p\in\mathcal{P}} c(p)$. This active learning procedure consists of two main steps: selection/translation (Section 7.3) and fine-tuning (Section 7.4).

Algorithm 3 Active Learning for Domain Adaptation of Machine Translation

```
procedure ActiveAdaptation($\mathcal{U}, \mathcal{L}, B$)
    Inputs: the unlabelled set $\mathcal{U}$, the labelled set $\mathcal{L}$, and a budget $B$.
    Train a MT model $\theta$ on $\mathcal{L}$.
    $\mathcal{S}, \mathcal{P} \leftarrow$ Selection($\mathcal{U}, \mathcal{L}, B$)
    Translate $\mathcal{S}$ by $\mathcal{L}_s = \{(x, \mathcal{O}(x)) | x \in \mathcal{S}\}$
    Translate $\mathcal{P}$ by $\mathcal{L}_p = \{(p, \mathcal{O}(p)) | p \in \mathcal{P}\}$
    $\mathcal{L}_r \leftarrow$ Obtain parallel data from $\mathcal{L}$ (Section 7.4)
    Fine-tune $\theta$ on $\mathcal{L}_s \cup \mathcal{L}_p \cup \mathcal{L}_r$
return $\theta$
```

Algorithm 4 Hybrid Phrase/Sentence Selection

```
procedure Selection($\mathcal{U}, \mathcal{L}, B$)
    Inputs: the unlabelled set $\mathcal{U}$, the labelled set $\mathcal{L}$, and a budget $B$.
    Initialize $\mathcal{S} = \{\}$, $\mathcal{P} = \{\}$
    Allocate the budget: $B_s, B_p \leftarrow B$
    while $\sum_{x\in\mathcal{S}} c(x) < B_s$ do
        $x \leftarrow \text{argmax}_{x\in\mathcal{U}}\, \phi(x, \cdot)$
        $\mathcal{U} = \mathcal{U} \setminus \{x\}$
        $\mathcal{S} = \mathcal{S} \cup \{x\}$
    Construct $\mathcal{P}_{\mathcal{U}}, \mathcal{P}_{\mathcal{L}}$ by strategies (Section 7.3.2)
    while $\sum_{p\in\mathcal{P}} c(p) < B_p$ do
        $p \leftarrow \text{argmax}_{p\in\mathcal{P}_{\mathcal{U}}}\, \textbf{occ}(p, \mathcal{U})$
        $\mathcal{P}_{\mathcal{U}} = \mathcal{P}_{\mathcal{U}} \setminus \{p\}$
        $\mathcal{P} = \mathcal{P} \cup \{p\}$
    return $\mathcal{S}, \mathcal{P}$
```

7.3 Active Selection Strategies

7.3.1 Sentence Selection Strategies

Existing sentence-based active learning methods usually define a sentence-level scoring function $\phi(x, \cdot)$, and select sentences with the top scores. Following Zeng et al. [190], we categorize these methods into two classes: data-driven and model-driven methods. Data-driven methods only rely on the unlabeled data $\mathcal{U}$ and the labeled data $\mathcal{L}$, i.e., $\phi(x, \mathcal{U}, \mathcal{L})$, and usually score sentences based on the trade-off between the density and diversity of the selected sentences. The density of the selected sentences determines whether these sentences frequently exist in the unlabeled data, while the diversity of the selected sentences determines whether these sentences cover the variety of the unlabeled data. In contrast, model-driven approaches usually estimate the prediction uncertainty of a source sentence given the current MT model θ, i.e., $\phi(x, \theta, \mathcal{U}, \mathcal{L})$, and select sentences with high uncertainty for training the model. Before getting to our proposed phrase-based strategies in Section 7.3.2 we highlight several existing sentence selection strategies.

Random Sampling: One easy strategy is randomly sampling sentences from the unlabeled data $\mathcal{U}$ for annotation. Although it is simple, this method is an unbiased approximation of the data distribution in $\mathcal{U}$. Therefore, this method remains a strong baseline in the active learning literature [62, 127, 190] if the annotation budget is sufficiently large.

Cosine Similarity between Sentence Embeddings (CSSE): Zhang et al. [194] propose to measure the distance between sentence embeddings. This method takes each unlabeled sentence, estimates its distance in embedding space from the labeled sentences in the out-of-domain corpus, and iteratively selects sentences that are more distant from sentences in the labeled data. In our instantiation of this method, we leverage the pre-trained mBERT model [47] to extract sentence representation e_x of a particular sentence x,[1] and measure a ratio-based distance [12] which is the ratio between the cosine similarity of $(\mathrm{e}_x, \mathrm{e}_{x'})$ and the average cosine similarity with their k nearest neighbors:

$$\text{ratio}(\mathrm{e}_x, \mathrm{e}_{x'}) = \frac{\cos\left(\mathrm{e}_x, \mathrm{e}_{x'}\right)}{\sum\limits_{z \in \mathrm{NN}_k(x)} \frac{\cos\left(\mathrm{e}_x, \mathrm{e}_z\right)}{2k} + \sum\limits_{z \in \mathrm{NN}_k(x')} \frac{\cos\left(\mathrm{e}_{x'}, \mathrm{e}_z\right)}{2k}}, \tag{7.1}$$

[1] We average the word representations from the 7th layer of the mBERT model as the sentence embedding, because the middle-layer representations have proven effective in cross-lingual retrieval tasks [82, 140].

where k is the number of nearest neighbors.

We then compute the margin-based distance between each in-domain sentence and its nearest out-of-domain neighbor within a randomly sampled subset of labeled sentences $\mathcal{L}'$:

$$\phi(x, \cdot) = \text{dist}(x, \mathcal{L}') = \min_{x' \in \mathcal{L}'} \text{ratio}(\text{e}_x, \text{e}_{x'}). \tag{7.2}$$

We approximate the distance between x and out-of-domain corpus $\mathcal{L}$ using a subset $\mathcal{L}'$ for efficiency purposes, because the out-of-domain $\mathcal{L}$ is usually large. Next we use the minimal distance $\text{dist}(x, \mathcal{L}')$ as our scoring function $\phi(x, \cdot)$, and select the unlabeled sentence with the largest distance from (sub-sampled) sentences in the out-of-domain corpus.

Round Trip Translation Likelihood (RTTL): One model-driven method is based on a method referred to as "round trip translation" [76, 190]. The labeled data $\mathcal{L}$ is used to train two MT models $\theta_{\text{src-tgt}}, \theta_{\text{tgt-src}}$ that translate between the source and target languages in two directions. Each unlabeled source sentence $x \in \mathcal{U}$ is first translated to $\hat{y}$ in the target language by $\theta_{\text{src-tgt}}$, and then $\hat{y}$ is translated to $\hat{x}$ by $\theta_{\text{tgt-src}}$. This method assumes that if this round-trip translation process fails to recover some of the content on the source side then this is an indication that the sentence may be difficult for the current model and is a good candidate for human annotation. Haffari et al. [76] use a scoring function that computes the similarity between the original sentence x and $\hat{x}$ using the sentence-level BLEU score [28], while Zeng et al. [190] estimate the likelihood of the original source sentence x given $\hat{y}$ by the reverse MT model $\theta_{\text{tgt-src}}$.

$$\hat{y} \approx \operatorname*{argmax}_{y} P_{\theta_{\text{src-tgt}}}(y|x) \tag{7.3}$$

$$\phi(x, \cdot) = \log P_{\theta_{\text{tgt-src}}}(x|\hat{y}) \tag{7.4}$$

7.3.2 Phrase Selection Strategies

A few existing phrase-based active learning methods [21, 127] have been proposed to improve PBMT systems. These methods first determine the possible set of phrases in a sentence, select phrases to be translated according to a scoring metric, and incorporate these in the training of the PBMT system. In the following paragraphs, we introduce two phrase-based selection strategies, and discuss how to integrate this data into NMT in Section 7.4.

***n*-gram Frequency (NGF)** [21]: The most straightforward phrase selection strategy is to select the most frequent phrases in the unlabelled data that *do not* appear in the already labeled data.

First we extract two sets of possible n-grams ($n \leq 4$) from sentences in $\mathcal{U}$ and $\mathcal{L}$, which are defined as $\mathcal{P}_{\mathcal{U}} = \cup_{x \in \mathcal{U}} \cup_{n \in [1,N]} n\text{-gram}(x)$, and $\mathcal{P}_{\mathcal{L}} = \cup_{(x,y) \in \mathcal{L}} \cup_{n \in [1,N]} n\text{-gram}(x)$. We then select the most frequent in-domain phrases from $\mathcal{P}_U$ by

$$p = \underset{p \in \mathcal{P}_{\mathcal{U}}, p \notin \mathcal{P}_{\mathcal{L}}}{\text{argmax}} \ \text{occ}(p, \mathcal{U}), \tag{7.5}$$

where $\text{occ}(p)$ counts the occurrences of p in $\mathcal{U}$.

Semi-Maximal Phrases (NGF-SMP): The two phrase sets $\mathcal{P}_{\mathcal{U}}, \mathcal{P}_{\mathcal{L}}$ extracted by the n-gram Frequency method contain many substrings that also occur in some longer strings. For example, p = "eines der" always co-occurs with the longer p' = "eines der besten" in the WMT14 German-English dataset. To identify the longer strings, Miura et al. [127] proposed the following semi-order relation, which defines the relation between a phrase p' and its sub-string p satisfying the condition that p' occurs at least half the time of p in the corpus $\mathcal{U}$.

$$p \overset{*}{\leq} p' \Leftrightarrow \exists \alpha, \beta : \alpha p \beta = p' \wedge \frac{\text{occ}(p, \mathcal{U})}{2} < \text{occ}(p', \mathcal{U}) \tag{7.6}$$

A phrase p is called a semi-maximal phrase if there does not exist a phrase p' in $\mathcal{U}$ such that $p \overset{*}{\leq} p'$. Therefore, a compact subset of phrases $\mathcal{P}'_{\mathcal{U}}$ can be constructed by containing only semi-maximal phrases in the phrase set $\mathcal{P}_{\mathcal{U}}$ in $\mathcal{U}$:

$$\mathcal{P}'_{\mathcal{U}} = \{p | \nexists p' \in \mathcal{P}_{\mathcal{U}}, p \overset{*}{\leq} p' \wedge p \in \mathcal{P}_{\mathcal{U}}\}. \tag{7.7}$$

By using semi-maximal phrases in $\mathcal{P}'_{\mathcal{U}}$ rather than all phrases in $\mathcal{P}_{\mathcal{U}}$, we remove a large number of phrases that are included in a longer phrase more than half the time, and reduce the redundancy of the selected phrases. Next we can select phrases similarly using Equation (7.5) by replacing the original phrase set $\mathcal{P}_{\mathcal{U}}$ with the sub-set $\mathcal{P}'_{\mathcal{U}}$.

7.3.3 Hybrid Selection Strategy

Phrase-based selection has its benefits, such as efficient annotation of core vocabulary from the target domain. However, at the same time, it lacks the ability to identify larger sentence structures that may nonetheless be unique to the target domain. Modeling this structure is particularly important for NMT (in contrast to PBMT), as NMT directly learns both lexical and syntactic transformations within the same model.

Because of this, we propose a simple yet novel hybrid selection strategy that leverages the benefits of both sentence-based and phrase-based selection strategies. Specifically, we allocate our budget in a way to annotate sentences with B_s words from our set of sentences and B_p words from our set of phrases. Depending on the specific sentence-based and phrase-based selection strategies chosen in the hybrid selection strategy, it is non-trivial to determine which selection strategy improves the in-domain translation performance more than the other one before actual finetuning. Therefore, in our implementation, we assume that we have no prior knowledge about which selection strategies will be most effective, and simply evenly distribute the annotation budget into the sentence-based and phrase-based strategies. We leave more sophisticated allocation strategies as future work, and we discuss some potential avenues briefly in Section 7.7.

7.4 Training with Sentences and Phrases

After data selection, we fine-tune the base NMT model on the newly translated data. This is essentially an extreme form of domain adaptation where we adapt a base NMT model trained on out-of-domain data to a new domain. Specifically, we adopt a strategy of *mixed fine-tuning* [120], which continues training a pre-trained out-of-domain model on both in-domain data and a certain amount of out-of-domain data to prevent overfitting to relatively small in-domain data. Compared to the standard domain adaptation setting where we have only a small number of in-domain sentences, our phrase-level active learning setting has the additional difficulty of having to use short translations of individual phrases. In the following, we describe both methods to choose which data to use in mixed fine-tuning, and how to incorporate phrasal translations.

7.4.1 Data Mixing

For data mixing, we sample a subset $\mathcal{L}_r$ of data directly from the labeled set $\mathcal{L}'$, and concatenate $\mathcal{L}_r$ with the newly annotated sentences $\mathcal{L}_s$ and phrases $\mathcal{L}_p$ for mixed fine-tuning (Line 8 in Algorithm 3). Specifically, we define a distribution function ψ over $\mathcal{L}'$, and either sample by $(x, y) \sim \psi$ or greedily take the most likely data by $(x, y) = \text{argmax}_{(x,y)\in\mathcal{L}'} \psi(x, y)$ iteratively for M times to obtain the subset $\mathcal{L}_r$ of M parallel data.

Random Sampling: The most simple way to select out-of-domain data is to randomly sample sentences from the out-of-domain corpus $\mathcal{L}'$, i.e., $(x, y) \sim \text{Uniform}(\mathcal{L}')$. Although it is simple, this has been popularly used in the literature of domain adaption for NMT [35].

Retrieve Similar Sentences: Recently Aharoni and Goldberg [2] showed that pre-trained language models implicitly learn sentence embeddings that cluster by domains, and proposed a data selection method that has proven more effective than methods based on the likelihood of an in-domain language model [129]. Since our base NMT model is pre-trained on out-of-domain corpus, we need to adapt the model to the domain of the unlabeled data. Instead of random sampling, we adopt the selection method in Aharoni and Goldberg [2] to retrieve parallel sentences from $\mathcal{L}'$ that are close to the in-domain sentences in $\mathcal{U}$. To do so, we leverage the contextualized sentence representations, and measure the distance of a source sentence in $\mathcal{L}'$ w.r.t. the unlabeled corpus $\mathcal{U}$ by $\text{ratio}(x, \mathcal{U}),\ \forall x \in \mathcal{L}'$. Next, we iteratively retrieve labeled data from $\mathcal{L}'$ that have the smallest distance scores to their nearest neighbors, i.e., $(x, y) = \text{argmax}_{(x,y)\in\mathcal{L}'} \text{ratio}(x, \mathcal{U})$.

7.4.2 Incorporating Phrasal Translations

In addition to obtaining real parallel data from $\mathcal{L}'$ for mixed fine-tuning, we create synthetic parallel data $(\hat{x}, \hat{y})$ by incorporating phrasal translations into existing context from $\mathcal{L}'$. Specifically, for an unlabeled sentence $x \in \mathcal{U}$ containing a newly annotated phrase p_x, we retrieve the similar sentence pair (x^*, y^*) from $\mathcal{L}'$ by

$$(x^*, y^*) = \underset{(x',y')\in\mathcal{L}'}{\text{argmax}}\ \text{ratio}(\mathrm{e}_x, \mathrm{e}_{x'}) \tag{7.8}$$

We then alter (x^*, y^*) with the newly annotated phrase pair (p_x, p_y) to create synthetic sentence pair $(\hat{x}, \hat{y})$. Similar to data mixing, we concatenate the set of synthetic data with the annotated sentences $\mathcal{L}_s$ and phrases $\mathcal{L}_p$ for mixed fine-tuning.

Switch Phrases: Inspired by existing data augmentation methods [56], we examine a data augmentation method that switches out phrases in the out-of-domain sentence pairs in $\mathcal{L}'$ by the newly annotated phrase pairs from $\mathcal{U}$. First, we define the following operation $\text{Switch}(x, p, i)$ that returns a new sentence by substituting the phrase at the i-th position in x^* by p_x.

$$\text{Switch}(x^*, p_x, i) = [x^*_{<i}; p_x; x^*_{\geq i+|p|}] \tag{7.9}$$

Next, we enumerate all possible positions in x^* for switching phrases, and then apply the in-domain language model trained on $\mathcal{U}$ to select the most probably synthetic sentence by

$$\hat{x} = \underset{\substack{x' = \text{Switch}(x^*, p_x, i) \\ \forall 0 \le i < |x^*| - |p|, \;\; p_x \in \cup_{n \in [1,N]} n\text{-gram}(x)}}{\text{argmax}} P_{\text{LM}}(x'), \tag{7.10}$$

where p_x is a phrase in the unlabeled sentence x.

To synthesize the corresponding $\hat{y}$ from the retrieved target sentence y^*, we apply a word alignment model trained on $\mathcal{L}$ to find the index j for the translation of the replaced phrase $x^*_{i:i+|p_x|}$ in y^*, and substitute the phrase at the j-th position in y^* by p_y to obtain $\hat{y} = \text{Switch}(y^*, p_y, j)$.

Contextualized Phrases: The other idea is to augment the context of a newly annotated phrase pair (p_x, p_y), since a phrase p_x lacks larger sentence structure. Specifically, we define the contextualized operation that augments a phrase p_x in x by appending it to the retrieved sentence x^*.

$$\text{Contextualize}(x^*, p_x) = [x^*, p_x] \tag{7.11}$$

We then enumerate all annotated phrases in x, and apply an in-domain language model to find the most probable annotated phrase pair (p_x, p_y) that synthesizes $\hat{x}$. The corresponding $\hat{y}$ can be obtained by $\text{Contextualize}(y^*, p_y)$.

$$\hat{x} = \underset{\substack{x' = [x^*, p_x] \\ \forall p_x \in \cup_{n \in [1,N]} n\text{-gram}(x)}}{\text{argmax}} P_{\text{LM}}(x') \tag{7.12}$$

7.5 Experiments

7.5.1 Experimental Setting

Dataset: We use the WMT14 German-English data as the out-of-domain labeled data for training our base NMT model, and take the source sentences of two parallel corpora in the medicine and IT domains [97] as the unlabeled data. As pointed out in Aharoni and Goldberg [2], there is overlap between the training data and the test data in the original split of the two corpora provided by Koehn and Knowles [97], so we follow them in removing the duplicated sentences in the in-domain data, and re-splitting two new test sets in order to prevent the model from memorizing the selected in-domain training data that could potentially be included in the test data. Table 7.1 shows the data statistics.

Data	Domain	Lang	#Sentences	#Words	Vocab	Avg Len
$\mathcal{L}$	WMT14	De	4.4M	108.0M	1.9M	24.4
		En		114.5M	955.3K	25.8
$\mathcal{U}$	Medicine	De	227.2K	3.8M	114.3K	16.8
	IT	De	190.6K	2.1M	114.6K	11.5

Table 7.1: Data statistics of the out-of-domain labeled data in WMT14 and the in-domain unlabeled data in the medicine and IT domains.

Model: As our NMT model, we use a 6-layer 512-unit Transformer network [166] implemented in `Fairseq`,[2] and use a subword vocabulary of 5,000 for both languages constructed by Byte Pair Encoding [157].

Training: We train the base model with Adam for 10 epochs with 4K warmup steps and a peak learning rate of 1e-3, and decay the learning rate based on the inverse square root of the number of update steps [166].

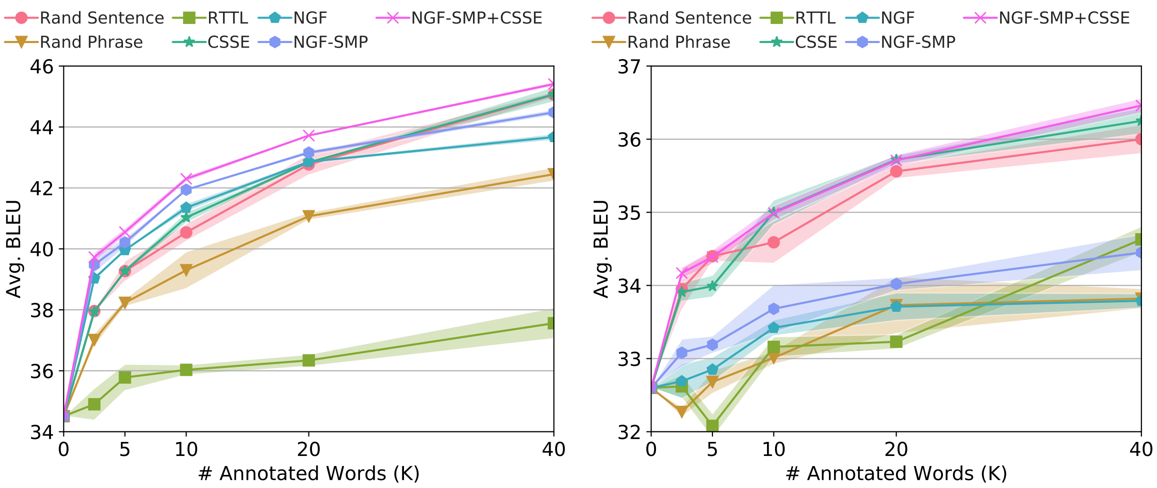
For active learning, we set our annotation budgets by number of words translated (following the prevailing translation market practice to charge for jobs by the word), and investigate the budgets from 2.5K words up to 40K words.[3] After data selection (Section 7.3), we obtain a set $\mathcal{L}_r$ of M parallel sentences (Section 7.4), and set the size $M = |\mathcal{L}_p|$ where $\mathcal{L}_p$ is selected by NGF-SMP. We then fix $\mathcal{L}_r$ for mixed fine-tuning in all experiments, and continue fine-tuning the base model on a mixture of the newly-translated data and $\mathcal{L}_r$ for 5 more epochs.

7.5.2 Word-level Translation Accuracy

Since our selection and mixed fine-tuning methods focus on leveraging phrasal translations for domain adaptation, we perform a fine-grained analysis on the word-level translation accuracy of the NMT systems due to the domain shift. A source word is defined as an unseen in-domain word when it never appears in the out-of-domain corpus. If phrase selection strategies select more in-domain words, we would expect a higher translation accuracy of such in-domain words by the adapted NMT systems using phrase selection. As a result, we compare the translation accuracy of in-domain words by the NMT models using different selection strategies in Figure 7.3. As shown in the figure, NGF-SMP significantly improves the translation accuracy of the in-domain words with a small annotation budget. In contrast, CSSE falls short of the other compared methods

[2] https://github.com/pytorch/fairseq

[3] At current market rates, this would cost from 491 to 7,092 USD for German-English translation by professional translators at https://translated.com/.



(a) WMT14→Medicine

(b) WMT14→IT

Figure 7.2: Average BLEU score over 3 runs for adapting a base NMT to the Medicine and IT domains.

when the annotation budget is less than 80K words. Moreover, we find that the hybrid selection strategy of NGF-SMP and CSSE can combine the merits of both methods, and obtain an even higher accuracy when the budget is greater than 40K annotated words. Qualitatively, the example in Table 7.2 shows the translations for a source sentence with all words appearing in the medical domain. The NMT model adapted by CSSE translates the first half of the source sentence by picking the correct word “exercised”, while the NMT model adapted by NGF-SMP generates the correction translation “somnolence” in the second half of the output. The NMT model using the hybrid of NGF-SMP and CSSE strategies translates both words correctly.

7.5.3 How Does Each Selection Strategy Help?

We examine the question of which selection strategy (Section 7.3) best improves accuracy on in-domain test data. For mixed fine-tuning, in this section we use the retrieved out-of-domain parallel data for a fair comparison among all active selection strategies. Figure 7.2 shows the average BLEU score and the standard deviation of the adapted MT systems to two new domains over 3 independent runs.[4]

[4] To obtain a stable result, we independently run the active learning procedure with different selection strategies 3 times, collect new translation data, and concatenate them with the same data retrieved from out-of-domain labeled data

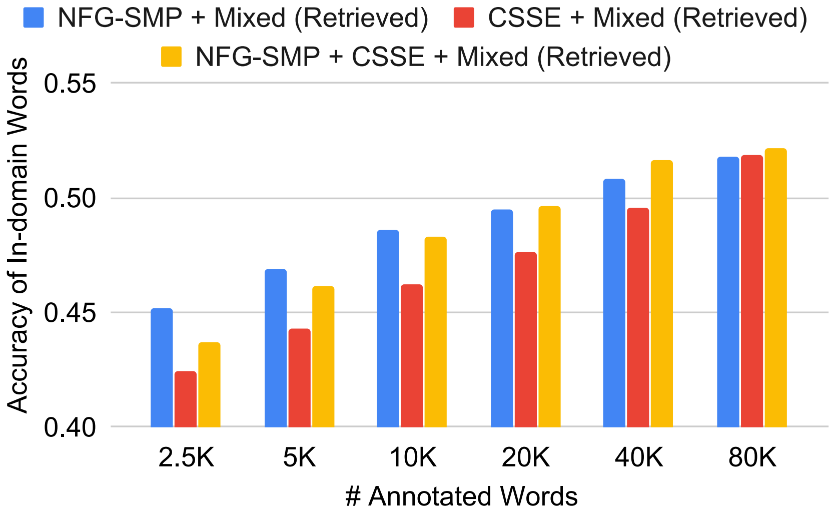


Figure 7.3: Translation accuracy of in-domain words in the test set from the medicine domain

Comparing among sentence selection strategies in Figure 7.2, CSSE performs slightly better than the random sentence selection baseline on adapting the NMT model to the IT domain with smaller standard deviation values, and performs comparably on adapting to the medicine domain. However, we observe that RTTL performs worst, and we conjecture that this is due to the usage of the base NMT models that are trained on the out-of-domain parallel data in both directions. The errors accumulated from the round trip translation process lead to an inaccurate estimation of the uncertainty score for a source sentence. Table 7.3 shows the top 5 sentences selected by RTTL. The selected sentences in the medicine domain are short phrases rather than complete sentences, and those selected in the IT domain contain duplicate phrases such as “bewerten mitâ”.

For phrase-based selection methods, NGF-SMP significantly outperform the random phrase selection strategy. Further, NGF-SMP even outperforms sentence selection methods when the annotation budget is small (less than 20k words) for adaption to the medicine domain. As we increase the annotation budget to 40K annotated words, sentence selection strategies outperform phrase selection strategies. This indicates that if we keep training NMT systems on shorter phrase pairs when the annotation budget is sufficient, the NMT systems would be limited by lack of longer sentence structures. In Figure 7.2b, we also find that NMT models trained with phrasal translations fall short of those trained with sentence translations when adapting to the IT domain. It is hard to train the NMT systems to translate certain phrases correctly without the sentence context. For example, “Persönlichen Ordner” in the IT domain is translated to “home directory” rather than “personal folder” in the sentence “jedes Skript dieses Dialogs hat Schreib-Zugriff auf

	Output	S-BLEU
Source	Jedoch ist Vorsicht geboten, da Berichten zufolge Verwirrung und Somnolenz während der Behandlung auftreten können.	
Reference	However, caution should be exercised as confusion and somnolence have been reported.	
NGF-SMP	However, caution is required, as there are reports of confusion and somnolence during the treatment.	15.71
CCSE	However, caution should be exercised, as confusion and drowsiness may occur during the treatment.	15.62
NGF-SMP+CSSE	However, caution should be exercised as confusion and somnolence may occur during the treatment.	15.71
Source	Schwindel, Parästhesie, Geschmacksstörung	
Reference	Dizziness, paraesthesiae, taste disorder	
NGF-SMP	Dizziness, paraesthesia, taste disturbance	23.27
CSSE	The room was very small and the bathroom was very small.	0.00
NGF-SMP+CSSE	Dizziness, paraesthesia, taste disturbance	23.27
Source	Über Hospitalisierung oder Todesfälle in Verbindung mit Infektionen wurde berichtet.	
Reference	Hospitalisation or fatal outcomes associated with infections have been reported.	
NGF-SMP	There have been reports of Hospitalisation or death associated with infections.	29.79
CSSE	Hospitals or deaths associated with infections have been reported.	54.63
NGF-SMP+CSSE	There have been reports of Hospitalisation or fatality associated with infections.	29.79

Table 7.2: Translations generated by NMT models using different selection strategies. The last column shows the sentence BLEU score of the translations. Translation errors are highlighted in red.

MED	Portugal Lundbeck Portugal Lda Quinta da Fonte Edifício D. Bronchitis Gastrointestinaltrakt : Neugebore 139 B.
IT	Eigenschaften des Stichwortes â % 1â bewerten mitâ Drei Sternenâ keine Speicherplatzinformation aufâ procfsâ bewerten mitâ Einem Sternâ neue und einzelne auswÃ Â hlen

Table 7.3: Top 5 sentences selected by RTTL

Ihren Persönlichen Ordner ”.

Finally, the hybrid selection of NGF-SMP and CSSE strategies outperforms the individual selection strategies over every budget in our set of budgets, i.e., 2.5K, 5K, 10K, 20K, 40K annotated words, improving the best phrase selection strategy NGF-SMP by 0.49 average BLEU points, and the best sentence selection strategy CSSE by 1.11 average BLEU points in the medicine domain. Notably, the phrase-based selection strategy especially helps in the scenario where the context is not required to translate domain-specific words, for example, the name of a medicine or a disease in the medicine domain in the second example. For the adaptation scenario that requires a longer context in some domains such as IT, the hybrid strategy can also significantly outperforms the best phrase-based strategy NGF-SMP by 1.2 average BLEU points, and the best sentence selection strategy CSSE by 0.15 BLEU points. Overall, our hybrid selection strategy is effective to combine the merits of both sentence and phrase selection strategies in the domain adaptation setting.

7.5.4 Analysis on Translation Length

Do Phrasal Annotations Bias NMT? Since phrasal annotations are short and do not contain complex sentence structure, we hypothesis that NMT systems trained on phrasal annotations would be biased towards generating shorter sentences or sentences in different grammatical order w.r.t. the reference sentence. To understand this question, we analyze the length ratio between the translation outputs and the reference sentences in Figure 7.4. We find that the NMT model trained only on annotated phrases selected by NGF-SMP generates shorter sentences than reference sentences. In contrast, adding sentences randomly sampled from the labeled corpus $\mathcal{L}$ make the NMT model generate longer sentences than the reference sentences, while retrieving sentences from $\mathcal{L}$ that are similar to the sentences in $\mathcal{U}$ makes the model produces translation outputs with closed lengths as the reference sentences. Qualitatively, we also show the problem of generating sentences with different structures as the reference sentences in the third example in Table 7.2. In the third example, the NMT model trained with NGF-SMP produces a translation in an active voice, while the reference sentence uses a passive voice.

7.5.5 How Representative Are the Selected Data?

If the selected data has a significant overlap of segments with the in-domain test data, we would expect a better adaptation performance of the NMT trained on the selected data. Therefore we investigate the n-gram overlap between the selected data and the test data when we annotate 5K words from the medicine corpus, and report the average BLEU score of the adapted NMT models trained on the selected data in Table 7.4. Interestingly, we find that there exists a high correlation (ρ ≈0.8) between the n-gram overlap and the average BLEU score, which indicates that the n-

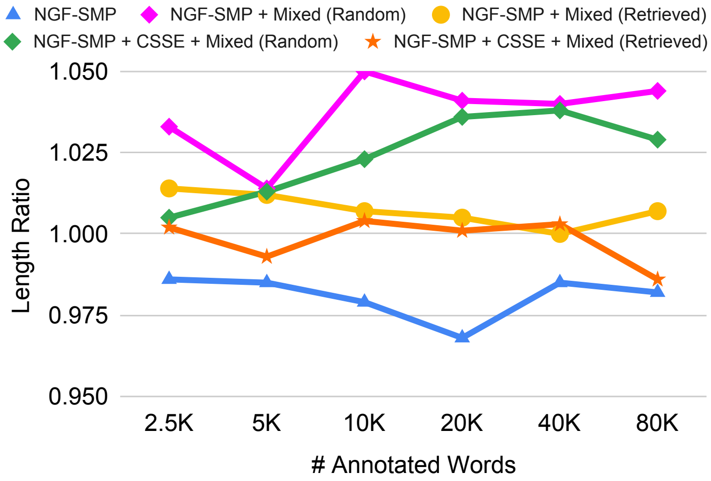


Figure 7.4: Length ratio between the NMT outputs and the reference sentences.

gram overlap with the test set can be used as a good measure of whether the selected data is useful for improving the NMT model in the new domain. Compared to the random phrase selection, NGF-SMP selects phrases with a high overlap with the test data. We also observe that sentence selection strategies cover fewer phrases in the test data than phrase selection strategies. This also corroborates our assumption that asking translators to annotate phrases that the MT system can already translate well is not cost-effective to improve the in-domain translation performance.

7.5.6 How Do Phrasal Translations Help in Mixed Fine-tuning?

We further investigate the effect of mixed fine-tuning using the newly annotated in-domain data and sub-sampled out-of-domain data when comparing with fine-tuning only on the newly annotated data. Table 7.5 shows the average BLEU score and the standard deviation values over 3 independent runs. Compared to fine-tuning on only annotated data, adding randomly sampled sentence pairs from the out-of-domain data helps when the annotation budget is less than 5K annotated words, but hurts when we increase the budget. In contrast, adding sentences retrieved by the similarity in the sentence embedding space not only outperforms fine-tuning only on annotated data and mixed fine-tuning with randomly sampled sentences, but also achieves smaller standard deviation values. On the other hand, mixed fine-tuning on synthetic data by switching phrases performs slightly worse than the mixed fine-tuning on real retrieved data, but outperforms the fine-tuning without any out-of-domain data, especially when the annotation budget is small, e.g., 5K annotated words. Combining synthetic data by switching phrase and real retrieved data for mixed fine-tuning also improve the translation performance over the training only on

Methods	uni-gram	bi-gram	tri-gram	4-gram	Avg. BLEU
OoD Data	79.33	32.65	7.30	1.10	34.51
+ Random Sentence	82.81	38.45	11.62	3.73	39.27
+ RTTL	80.70	35.76	9.85	3.04	35.78
+ CSSE	82.74	38.83	12.01	4.05	39.27
+ Random Phrase	82.36	35.84	7.98	1.15	38.23
+ NGF	84.45	41.82	14.94	6.17	39.96
+ NGF-SMP	85.80	43.13	16.15	7.11	40.21
+ NGF-SMP + CSSE	84.48	41.89	14.98	6.48	40.55
ID Training Data	98.58	87.30	67.61	52.11	57.59
Pearson Correlation	0.90	0.83	0.80	0.78	/

Table 7.4: Percentage of the n-gram in the test sentences that are covered by the selected data with 5K words, the out-of-domain training data and the in-domain training data. The last row shows the Pearson correlation coefficient between *n*-gram overlap and avg. BLEU score.

Out-of-domain Data				In-domain Data		2.5K	5K	10K	20K	40K
Sampled	Retrieved	Switched	Contextualized	NGF-SMP	CSSE					
				✓		39.39 ± 0.14	39.22 ± 0.00	40.56 ± 0.02	41.19 ± 0.25	44.07 ± 0.33
					✓	37.94 ± 0.08	38.68 ± 0.54	40.62 ± 0.59	42.62 ± 0.03	45.00 ± 0.11
				✓	✓	38.94 ± 0.02	39.60 ± 0.09	41.34 ± 0.12	42.44 ± 0.15	44.90 ± 0.06
✓				✓	✓	39.46 ± 0.14	40.51 ± 0.23	40.62 ± 0.49	41.82 ± 0.26	43.78 ± 0.57
	✓			✓	✓	**39.73** ± 0.16	40.55 ± 0.14	**42.30** ± 0.10	**43.72** ± 0.04	**45.41** ± 0.08
		✓		✓	✓	38.93 ± 0.36	40.59 ± 0.17	41.82 ± 0.29	42.70 ± 0.37	45.33 ± 0.04
			✓	✓	✓	35.36 ± 0.38	37.85 ± 0.68	39.96 ± 0.35	42.83 ± 0.11	44.14 ± 0.15
	✓	✓		✓	✓	39.61 ± 0.06	**40.95** ± 0.06	42.19 ± 0.08	43.42 ± 0.17	45.06 ± 0.19
	✓		✓	✓	✓	37.88 ± 0.25	39.52 ± 0.32	41.17 ± 0.28	42.80 ± 0.21	44.28 ± 0.13

Table 7.5: Comparison between mixed fine-tuning methods. Bold indicates highest average BLEU by column.

Methods	IDWT	WT	$\frac{\text{IDWT}}{\text{WT}}$	IDWC	WC	$\frac{\text{IDWC}}{\text{WC}}$
Random Phrase	787	2206	35.68	860	5003	17.19
NGF	489	1053	46.44	889	5002	17.77
NGF-SMP	796	1492	53.35	1076	5001	21.52
Random Sentence	631	1984	31.80	712	5023	14.17
RTTL	592	1338	44.25	961	5023	19.13
CSSE	647	2056	31.47	721	5023	14.35
NGF-SMP + CSSE	667	1755	38.01	859	5035	17.06

Table 7.6: Statistics of the unique in-domain word types and word counts in the selected data with 10K annotated words.

synthetic data. However, the contextualized method performs worst among all mixed fine-tuning methods, which indicates that simply appending existing sentence context to phrasal translations might potentially introduce noise to the training data.

7.5.7 How redundant are the selected data?

To answer this question, we first define “in-domain words” as words that only appear in the in-domain test set but do not exist in the out-of-domain data. We report the statistics of the in-domain word types word counts in the selected data with 10K annotated words in Table 7.6. We find that phrase selection strategies select more unique in-domain word types and counts than sentence selection strategies. This indicates that phrase selection strategies leverage the same amount of budget effectively to annotate more diverse in-domain words than sentence selection strategies.

7.6 Related Work

Active Learning for Machine Translation Pioneering works on active learning for machine translation focus on selecting sentences that are most useful for training PBMT. This includes sentence selection strategies based on maximizing the percentage of unseen *n*-gram [53], *n*-gram frequency, lexical diversity [76], or in-domain coverage [7]. These sentence selection strategies have been used in active learning algorithms to deal with static data in the batch mode [7], or steaming data in the interactive setting [67, 100, 138].

For phrase-level annotations, there have been a few works applying phrase-based selection [21, 127] to PBMT. While the annotated phrases can be easily integrated by adding them with estimated translation probability to the existing phrase table in PBMT, it is less trivial to integrate these phrase-level annotations in NMT. Arthur et al. [17] integrated the word-level translations to NMT by interpolating the probability of the NMT decoder with the estimated lexical probability. However, this approach requires a modification of the NMT model. This chapter investigates the data-driven approaches that augment the training data by leveraging annotated phrases and existing parallel data.

7.7 Discussion and Future Work

In this chapter, we investigate ways to incorporating phrasal translations into training NMT for domain adaptation in the active learning setting. We find that phrasal translation is particularly useful in the adaptation scenario where longer sentence context is not necessarily required to translate in-domain words correctly. In contrast, NMT systems can benefit from learning sentence structure with sentence-based selection strategies. The hybrid selection strategies can combine the merits of both sentence-based and phrase-based selection strategies. Nonetheless, there are several future directions. (1) It is worth exploring how different annotation strategies may result in a difference in cost or time. (2) Although several findings could be generalized to other language pairs, testing our methods on morphologically rich languages is our next step. (3) Our current hybrid strategy simply allocate the annotation budget evenly without assuming any prior knowledge on the strategies and the translation performance. Techniques in multi-armed bandit problems [65] can be used to learn a good allocation strategy.

Chapter 8

Conclusion

In this thesis, we studied multilingual learning methods for machine translation as well as a variety of language understanding tasks such as text classification, sentence retrieval, sequence tagging, and question answering. In particular, we developed methods to enhance the cross-lingual understanding and generation by leveraging word and phrase alignment information from parallel sentences, monolingual raw text, knowledge bases, or human crowsourcing.

In this chapter, we start by summarizing the key contributions of this thesis in Section 8.1, followed by some discussions for the key ideas learned in Section 8.2 and future work in Section 8.3.

8.1 Summary of Contributions

This thesis facilitates multilingual learning research using *word and phrase alignments* for both language understanding and generation given limited direct supervision. In the following, we outline the exploration of the multilingual learning research covered in this thesis.

Cross-lingual Generalization. Before we develop any particular multilingual models, we start by asking a fundamental question: *how could we accurately evaluate existing multilingual models for language understanding?* To answer this question, there are several aspects to consider. First, ideally we should select a diverse set of representative NLP tasks that comprehensively examine the semantic understanding of text data at different granularities such as words, word spans, and sentences. Second, we should choose our target languages that reveal the typological diversity of natural languages. To this end, we introduced XTREME (Chapter 3), a multitask multilingual benchmark for understanding the zero-shot cross-lingual generalization capability of multilingual contextualized representations. A more fine-grained analysis shows that cross-lingual transfer learning between distant languages is still challenging. Existing multilingual contextualized models still fall short of structure prediction tasks across languages. We believe that XTREME opens the door to a more fine-grained understanding of cross-lingual generalization of multilingual models, sheds light on future designs of more effective multilingual model architectures, and facilitates the development of more efficient multilingual pre-training methods. For example, our follow-up work [151] not only extends the diversity of tasks and languages, but also provides a multilingual diagnostic suite (MultiCheckList [146]) and an interactive leaderboard (Explainaboard [116]) for a better understanding of these multilingual models. With the target of learning better multilingual representations for cross-lingual generalization, we further proposed two explicit alignment objectives to align contextualized word and sentence representations in different languages (Chapter 4). We show that a compact model trained by our alignment method obtains substantial gains or comparable performance with respect to large pre-trained multilingual models at word-level and sentence-level language understanding tasks. Our work has inspired several follow-up works on cross-lingual alignment [32, 69].

Domain Adaptation of Machine Translation In addition to learning multilingual representations for language understanding, we also applied these representations for machine translation, a typical and useful language generation task. In Chapter 5, we leverage word alignment techniques to induce lexicons from monolingual data and apply a word-for-word back-translation technique to create synthetic data for adapting a neural machine translation model to a new domain. We find that learning such cross-lingual representations from monolingual text data in the target domain is crucial to obtain translations of unseen words in the target domain. This finding has motivated a series of follow-up works such as detecting domain information by multilingual contextualized representations [2]. Through a further analysis of these unseen words for translation, we discovered that many unseen or raw words in the target domains are named entities. While it might be hard to induce the translations of these named entities from monolingual data, we find that existing knowledge bases such as WikiData contain rich information of multilingual named entities. As a result, we further investigate the aligned entities from a knowledge base to pre-train a neural machine translation on large amounts of raw text in Chapter 6. We explicitly inject translations of named entities into monolingual raw text for pre-training and adopt a multi-task fine-tuning strategy to improve the translation accuracy of low-frequency or unseen entities in the downstream translation tasks. Despite the fact that a knowledge base contains entity translations, there are still plenty of entities that do not have their translations in the knowledge bases. To address this limitation, we further examine an active learning setting to handle the adaption scenario when those aligned words or phrases are not available in Chapter 7. We proposed a hybrid strategy to integrate both phrase-based and sentence-based selection to fully utilize the annotation budget.

8.2 Key Ideas and Suggestions

Based on the exploration of multilingual learning research throughout this thesis, I summarize several key ideas that we have learned and provide suggestions for future researcher in this direction.

Multilingual Representation Learning: A recurring theme in the methods presented in this thesis is that we can learn cross-lingual representations for our downstream NLP tasks including tasks in XTREME (Chapter 3) and lexicon induction task (Chapter 5). Learning such multilingual representations requires supervision from various sources. Throughout this thesis we have introduced many supervision signals from in-domain monolingual data (Chapter 5), parallel sentences (Chapter 4) and aligned entities (Chapter 6). Whether to choose multilingual contextualized representation or non-contextualized representations crucially depends on the task difficulty in a specific application scenario. When we are tackling tasks crucially depending on sentence context (e.g., sentence retrieval) or tasks with a limited amount of labeled data (e.g., question answering), contextualized representations are usually more preferable. On the other hand, there is always a trade-off between the availability of high-quality data and the cross-lingual generalization of multilingual representations. When we have large collections of parallel corpora to train a strong multilingual neural machine translation system (e.g., Google Translate), one particular strong baseline is the *translate-train* method that creates synthetic training data by using the MT system to translate labeled data from a source language to a target language. When we only have a small number of parallel sentences, we find that aligning these multilingual contextualized representations using parallel sentences can be efficient in terms of smaller model size and data size (Chapter 4). When we do not have any parallel sentences, pre-training of neural network models on large collections of monolingual corpora in multiple languages is still effective with respect to the translate-train baseline. Notably, there is a substantial gain from pre-training when we increase the size of monolingual corpora and model parameters for pre-training (Chapter 3).

Data Augmentation: We have been using data augmentation techniques to generate pseudo-parallel data (Chapter 5) and code-switched data (Chapter 6) for training an encoder-decoder based neural machine translation model. Notice that we often use the original human-written sentences on the target side while leaving the noisy synthetic data on the source side. This is critical to ensure that the decoder is trained on clean data so that the decoder will not be able to generate unnatural sentences. In addition, when we have a certain amount of real parallel data either in the target domain (Chapter 6) or out of the target domain (Chapter 5, 7), fine-tuning neural machine translation models jointly on both the synthetic and real parallel data usually yields better performance than fine-tuning on either one of them.

Domain Robustness: It is hard to quantify all errors due to domain shift in language generation systems as natural languages are inherently flexible. For sentence-level machine translation systems, translation errors due to domain shift can be mainly categorized into *word/phrase-level* errors and *sentence-level* errors. While this thesis mainly focuses on translations of domain-specific words or phrases (Chapter 5, 6, 7), existing studies [130] also pointed out sentence-level errors such as hallucination of translations. As a result, judging the quality of machine translation systems based on a single metric such as BLEU is not accurate. A more fine-grained comparison through `compare-mt` [133] would be beneficial for error analysis. In addition to error analysis, the development of techniques to improve domain robustness of NMT models is also important. These techniques crucially depend on the availability of in-domain data for adaptation. Prior studies [36, 120] have shown the effectiveness of fine-tuning NMT models on in-domain parallel sentences. In contrast, this thesis tackles the challenging scenarios in which we do not have lots of in-domain parallel sentences to improve in-domain word or phrase translations. When we only have in-domain monolingual sentences in both the source and target languages, we can perform adaptation through a bilingual in-domain dictionary induced from in-domain monolingual sentences (Chapter 5). When low-frequency words and their translations cannot be easily found in monolingual sentences at the same time, lexicon induction techniques may fail to find a high-quality dictionary. To tackle this issue, we find that entity recognition and linking techniques are useful to find translations of these low-frequency entities in a knowledge base (Chapter 6). When we only have in-domain monolingual sentences in the source language, active learning techniques that obtain human translations of informative in-domain phrases or sentences using an annotation budget are usually helpful (Chapter 7). When we only have a limited annotation budget, phrase-based selection strategies are usually more effective in terms of word translation accuracy. When we have a slightly larger annotation budget, a hybrid strategy that combines both phrase-based and sentence-based strategies performs better than either one of them. When we have a sufficiently large annotation budget, a random selection strategy remains a strong baseline. When we compare our DALI method (Chapter 5) with the active learning strategies (Chapter 7), DALI automatically induces a dictionary from in-domain monolingual data in both the source and target languages, while the phrase-based active learning methods obtain a high-quality dictionary from human translators. As a result, the phrase-based active learning methods should perform better than DALI at the expense of annotation costs.

8.3 Future Work

The thesis of this research has been shown that multilingual learning approaches have the potential to transfer knowledge across languages. While we have made much progress, there are several issues that we still need to address:

Beyond Static Evaluation: Throughout the thesis, we have been focusing on evaluating language understanding and machine translation in a collection of static labeled datasets. While multilingual learning has an amazing potential of breaking down the language barrier of language communication, the research on multilingual interactive systems such as task-oriented dialogue systems, simultaneous speech-to-speech machine translation has not been covered so far in the thesis. It will be interesting to see whether multilingual learning approaches could be integrated into a real human-computer interactive setting. As a concrete next step, one potential effort is to extend existing multilingual contextualized representations to learn from existing English-based task-oriented dialogue dataset using the similar cross-lingual transfer techniques in Chapter 3 and develop active learning approaches to obtain labels in the conversation (Chapter 7).

Other Types of Encoding Methods for Multilingual Representations: The previous chapters have shown impressive cross-lingual results for language understanding. However, there is still no clear connection between learning multilingually-aligned representations and linguistic structures of text. Most existing methods rely on a shared vocabulary of subwords and a shared encoder model for cross-lingual transfer, while linguistic features such as dependency trees, character similarity, or phonetic representations have not been fully explored yet. In particular, an important direction for future work is to develop linguistically motivated encoding methods with the hope of reducing the requirement of computational power and data resources. We outline some ideas below:

1. **Tokenization of Multilingual Text:** Currently we still rely on subword techniques to preprocess text data for building a fixed vocabulary before training. This loses some flexibility of jointly learning task-specific tokenization and the task itself. One way is to revisit traditional character-based encoding methods and integrate the tokenization model in an end-to-end learning process.

2. **Sentence Structures:** Most existing multilingual contextualized representations purely rely on a Transformer model to learn the pairwise relations between words in a data-driven way. However, there are many out-of-box parsing models for text in multiple languages, a more principled approach of leveraging structural information from language grammar

would be more efficient for representation learning.

3. **Disentangled Langauge Representations:** Most existing contextualized multilingual representations mainly rely on sharing the entire model architecture for cross-lingual transfer. However, a more principled approach would allow a model to disentangle representations into language-agnostic representations that share among languages, and language-specific representations that uniquely identify languages themselves.

Other Types of Multilingual Resources for Training: Throughout the thesis, we have investigated available data including monolingual data, parallel data, or knowledge bases for learning multilingual representations. Below we outline several more potential resources:

1. **Relations between entities in knowledge bases:** In our previous work regarding learning from knowledge bases, we have mainly focused on training neural network models from aligned entities. However, to capture deeper semantics in human-machine communication, we believe that one key missing component of language learning is the utilization of structured relations between entities.

2. **Multimodal data:** Multimodal data such as video or image data contain rich visual information of entities mentioned in their captions. The intuition is that people have prior knowledge about entities in the world and ground their concepts of the entities to the same real-world objects, even they speak different languages. As a result, a future direction beyond this thesis is to pursue methods to integrate visual data with text-based multilingual models for multilingual multimodal language learning.

Appendix A

Appendix of Chapter 3

A.1 Languages

We show a detailed overview of languages in the cross-lingual benchmark including interesting typological differences in Table A.1. Wikipedia information is taken from Wikipedia[1] and linguistic information from WALS Online[2]. XTREME includes members of the Afro-Asiatic, Austro-Asiatic, Austronesian, Dravidian, Indo-European, Japonic, Kartvelian, Kra-Dai, Niger-Congo, Sino-Tibetan, Turkic, and Uralic language families as well as of two isolates, Basque and Korean.

A.2 Translations for QA datasets

We use an in-house translation tool to obtain translations for our datasets. For the question answering tasks (XQuAD and MLQA), the answer span is often not recoverable if the context is translated directly. We experimented with enclosing the answer span in the English context in quotes [107, 111] but found that quotes were often dropped in translations (at different rates depending on the language). We found that enclosing the answer span in HTML tags (e.g. <b> and </b>) worked more reliably. If this fails, as a back-off we fuzzy match the translated answer with the context similar to [80]. If the minimal edit distance between the closest match and the translated answer is larger than $\min(10, \texttt{answer\_len}/2)$, we drop the example. On the whole, using this combination, we recover more than 97% of all answer spans in training and test data.

[1] https://meta.wikimedia.org/wiki/List_of_Wikipedias
[2] https://wals.info/languoid

Language	ISO 639-1 code	# Wikipedia articles (in millions)	Script	Language family	Diacritics / special characters	Extensive compounding	Bound words / clitics	Inflection	Derivation	# datasets with language
Afrikaans	af	0.09	Latin	IE: Germanic		X				3
Arabic	ar	1.02	Arabic	Afro-Asiatic	X		X	X		7
Basque	eu	0.34	Latin	Basque	X		X	X	X	3
Bengali	bn	0.08	Brahmic	IE: Indo-Aryan	X	X	X	X	X	3
Bulgarian	bg	0.26	Cyrillic	IE: Slavic	X		X	X		4
Burmese	my	0.05	Brahmic	Sino-Tibetan	X	X				1
Dutch	nl	1.99	Latin	IE: Germanic		X				3
English	en	5.98	Latin	IE: Germanic						9
Estonian	et	0.20	Latin	Uralic	X	X		X	X	3
Finnish	fi	0.47	Latin	Uralic				X	X	3
French	fr	2.16	Latin	IE: Romance	X		X			6
Georgian	ka	0.13	Georgian	Kartvelian				X	X	2
German	de	2.37	Latin	IE: Germanic		X		X		8
Greek	el	0.17	Greek	IE: Greek	X	X		X		5
Hebrew	he	0.25	Hebrew	Afro-Asiatic				X		3
Hindi	hi	0.13	Devanagari	IE: Indo-Aryan	X	X	X	X	X	6
Hungarian	hu	0.46	Latin	Uralic	X	X		X	X	4
Indonesian	id	0.51	Latin	Austronesian			X	X	X	4
Italian	it	1.57	Latin	IE: Romance	X		X			3
Japanese	ja	1.18	Ideograms	Japonic			X	X		4
Javanese	jv	0.06	Brahmic	Austronesian	X		X			1
Kazakh	kk	0.23	Arabic	Turkic	X			X	X	1
Korean	ko	0.47	Hangul	Koreanic		X		X	X	5
Malay	ms	0.33	Latin	Austronesian			X	X		2
Malayalam	ml	0.07	Brahmic	Dravidian	X	X	X	X		2
Mandarin	zh	1.09	Chinese ideograms	Sino-Tibetan		X				8
Marathi	mr	0.06	Devanagari	IE: Indo-Aryan			X	X		3
Persian	fa	0.70	Perso-Arabic	IE: Iranian		X				2
Portuguese	pt	1.02	Latin	IE: Romance	X		X			3
Russian	ru	1.58	Cyrillic	IE: Slavic				X		7
Spanish	es	1.56	Latin	IE: Romance	X		X			7
Swahili	sw	0.05	Latin	Niger-Congo			X	X	X	3
Tagalog	tl	0.08	Brahmic	Austronesian	X		X	X		1
Tamil	ta	0.12	Brahmic	Dravidian	X	X	X	X	X	3
Telugu	te	0.07	Brahmic	Dravidian	X	X	X	X	X	4
Thai	th	0.13	Brahmic	Kra-Dai	X					4
Turkish	tr	0.34	Latin	Turkic	X	X		X	X	5
Urdu	ur	0.15	Perso-Arabic	IE: Indo-Aryan	X	X	X	X	X	4
Vietnamese	vi	1.24	Latin	Austro-Asiatic	X					6
Yoruba	yo	0.03	Arabic	Niger-Congo	X					1

Table A.1: Statistics about languages in the cross-lingual benchmark. Languages belong to 12 language families and two isolates, with Indo-European (IE) having the most members. Diacritics / special characters: Language adds diacritics (additional symbols to letters). Compounding: Language makes extensive use of word compounds. Bound words / clitics: Function words attach to other words. Inflection: Words are inflected to represent grammatical meaning (e.g. case marking). Derivation: A single token can represent entire phrases or sentences.

	Test set	es	de	el	ru	tr	ar	vi	th	zh	hi	avg
mBERT	gold	75.6 / 56.9	70.6 / 54.0	62.6 / 44.9	71.3 / 53.3	55.4 / 40.1	61.5 / 45.1	69.5 / 49.6	42.7 / 33.5	58.0 / 48.3	59.2 / 46.0	62.6 / 47.2
	auto	76.1 / 58.7	64.3 / 49.9	57.9 / 42.5	68.3 / 51.8	55.6 / 42.9	62.1 / 48.6	68.6 / 54.3	41.1 / 32.6	48.5 / 47.7	54.1 / 40.9	59.7 / 47.0
translate-train	gold	80.2 / 63.1	75.6 / 60.7	70.0 / 53.0	75.0 / 59.7	68.9 / 54.8	68.0 / 51.1	75.6 / 56.2	36.9 / 33.5	66.2 / 56.6	69.6 / 55.4	68.7 / 54.6
	auto	80.7 / 66.0	71.1 / 58.9	69.3 / 54.5	75.7 / 61.5	71.2 / 59.1	74.3 / 60.7	76.8 / 64.0	79.5 / 74.8	59.3 / 58.0	69.1 / 55.2	72.7 / 61.3

Table A.2: Comparison of F1 and EM scores of mBERT and translate-train (mBERT) baselines on XQuAD test sets (gold), which were translated by professional translators and automatically translated test sets (auto).

Languages	zh	es	de	ar	ur	ru	bg	el	fr	hi	sw	th	tr	vi	avg
auto Acc.	69.1	74.7	72.8	66.5	64.5	71.6	70.2	67.7	74.3	65.1	50.2	54.5	60.0	72.7	66.7
gold Acc.	67.8	73.5	70.0	64.3	57.2	67.8	68.0	65.3	73.4	58.9	49.7	54.1	60.9	69.3	64.3
BLEU	40.92	43.46	30.94	32.35	20.13	22.62	45.04	60.29	47.91	29.55	31.25	10.65	15.39	56.93	34.82
chrF	35.96	67.92	60.28	59.64	48.21	50.38	67.52	75.34	69.58	53.85	59.84	54.89	51.46	69.37	58.87

Table A.3: Comparison of accuracy scores of mBERT baseline on XNLI test sets (gold), which were translated by professional translators and automatically translated test sets (auto) in 14 languages. BLEU and chrF scores are computed to measure the translation quality between gold and automatically translated test sets.

A.3 Performance on translated test sets

We show results comparing the performance of mBERT and translate-train (mBERT) baselines on the XQuAD test sets with automatically translated test sets in Table A.2. Performance on the automatically translated test sets underestimates the performance of mBERT by 2.9 F1 / 0.2 EM points but overestimates the performance of the translate-train baseline by 4.0 F1 / 6.7 EM points. The biggest part of this margin is explained by the difference in scores on the Thai test set. Overall, this indicates that automatically translated test sets are useful as a proxy for cross-lingual performance but may not be reliable for evaluating models that have been trained on translations as these have learned to exploit the biases of *translationese*.

A.4 Generalization to unseen tag combinations

We show the performance of mBERT on POS tag trigrams and 4-grams that were seen and not seen in the English training data in Table A.4.

	trigram, seen	trigram, unseen	4-gram, seen	4-gram, unseen
en	90.3	63.0	88.1	67.5
af	68.1	8.2	64.1	24.2
ar	22.0	0.7	14.9	4.6
bg	63.1	14.6	56.1	23.9
de	77.8	47.2	73.0	48.7
el	59.6	9.1	52.5	14.2
es	68.6	10.6	62.4	24.9
et	60.7	14.4	53.1	31.9
eu	32.8	7.1	28.7	8.1
he	52.7	35.7	44.0	27.4
hi	38.7	13.0	32.6	12.5
hu	55.5	28.8	46.9	23.7
id	60.8	16.6	54.7	21.6
it	75.5	12.8	71.8	23.5
ja	16.3	0.0	12.3	1.0
ko	22.0	2.9	14.7	3.8
mr	31.7	0.0	25.5	3.3
nl	75.5	24.1	71.0	37.8
pt	76.2	14.9	71.2	30.6
ru	69.1	4.8	63.8	20.6
ta	30.3	0.0	24.5	4.2
te	57.8	0.0	48.7	24.7
tr	41.2	6.2	33.9	10.1
ur	30.6	18.3	22.3	10.9
zh	29.0	0.0	21.7	3.9
avg	50.6	12.1	44.3	18.3
diff	39.7	50.9	43.7	49.2

Table A.4: Accuracy of mBERT on the target language dev data on POS tag trigrams and 4-grams that appeared and did not appear in the English training data. We show the average performance across all non-English languages and the difference of said average compared to the English performance on the bottom.

A.5 Results for each task and language

We show the detailed results for all tasks and languages in Tables A.5 (XNLI), A.9 (PAWS-X), A.13 (POS), A.7 (NER), A.10 (XQuAD), A.12 (MLQA), A.11 (TyDiQA-GoldP), A.8 (BUCC), and A.6 (Tatoeba).

Model	en	ar	bg	de	el	es	fr	hi	ru	sw	th	tr	ur	vi	zh	**avg**
mBERT	80.8	64.3	68.0	70.0	65.3	73.5	73.4	58.9	67.8	49.7	54.1	60.9	57.2	69.3	67.8	65.4
XLM	82.8	66.0	71.9	72.7	70.4	75.5	74.3	62.5	69.9	58.1	65.5	66.4	59.8	70.7	70.2	69.1
XLMR	**88.7**	**77.2**	**83.0**	**82.5**	**80.8**	**83.7**	**82.2**	**75.6**	**79.1**	**71.2**	**77.4**	**78.0**	**71.7**	**79.3**	**78.2**	**79.2**
MMTE	79.6	64.9	70.4	68.2	67.3	71.6	69.5	63.5	66.2	61.9	66.2	63.6	60.0	69.7	69.2	67.5
Translate-train (multi-task)	81.9	**73.8**	**77.6**	**77.6**	**75.9**	**79.1**	**77.8**	70.7	**75.4**	**70.5**	70.0	74.3	**67.4**	**77.0**	**77.6**	**75.1**
Translate-train	80.8	73.6	76.6	77.4	75.7	78.1	77.4	**71.9**	75.2	69.4	**70.9**	**75.3**	67.2	75.0	74.1	74.6
Translate-test	**85.9**	73.1	76.6	76.9	75.3	78.0	77.5	69.1	74.8	68.0	67.1	73.5	66.4	76.6	76.3	76.8

Table A.5: XNLI accuracy scores for each language.

Lang.	af	ar	bg	bn	de	el	es	et	eu	fa	fi	fr	he	hi	hu	id	it	ja
BERT	42.7	25.8	49.3	17	77.2	29.8	68.7	29.3	25.5	46.1	39	66.3	41.9	34.8	38.7	54.6	58.4	42
XLM	43.2	18.2	40	13.5	66.2	25.6	58.4	24.8	17.1	32.2	32.2	54.5	32.1	26.5	30.1	45.9	56.5	40
XLMR	**58.2**	**47.5**	**71.6**	**43**	**88.8**	**61.8**	**75.7**	**52.2**	**35.8**	**70.5**	**71.6**	**73.7**	**66.4**	**72.2**	**65.4**	**77**	**68.3**	**60.6**
	jv	ka	kk	ko	ml	mr	nl	pt	ru	sw	ta	te	th	tl	tr	ur	vi	zh
BERT	17.6	20.5	27.1	38.5	19.8	20.9	68	69.9	61.2	11.5	14.3	16.2	13.7	16	34.8	**31.6**	62	**71.6**
XLM	**22.4**	22.9	17.9	25.5	20.1	13.9	59.6	63.9	44.8	12.6	20.2	12.4	**31.8**	14.8	26.2	18.1	47.1	42.2
XLMR	14.1	**52.1**	**48.5**	**61.4**	**65.4**	**56.8**	**80.8**	**82.2**	**74.1**	**20.3**	**26.4**	**35.9**	29.4	**36.7**	**65.7**	24.3	**74.7**	68.3

Table A.6: Tatoeba results (Accuracy) for each language

Lang.	en	af	ar	bg	bn	de	el	es	et	eu	fa	fi	fr	he	hi	hu	id	it	ja	jv
mBERT	**85.2**	77.4	41.1	77.0	70.0	78.0	72.5	77.4	75.4	**66.3**	46.2	77.2	79.6	56.6	65.0	76.4	**53.5**	**81.5**	29.0	**66.4**
XLM	82.6	74.9	44.8	76.7	70.0	78.1	73.5	74.8	74.8	62.3	49.2	79.6	78.5	57.7	66.1	76.5	53.1	80.7	23.6	63.0
XLMR	84.7	**78.9**	**53.0**	**81.4**	**78.8**	**78.8**	**79.5**	**79.6**	**79.1**	60.9	**61.9**	**79.2**	**80.5**	**56.8**	**73.0**	**79.8**	53.0	81.3	23.2	62.5
MMTE	77.9	74.9	41.8	75.1	64.9	71.9	68.3	71.8	74.9	62.6	45.6	75.2	73.9	54.2	66.2	73.8	47.9	74.1	**31.2**	63.9
	ka	kk	ko	ml	mr	ms	my	nl	pt	ru	sw	ta	te	th	tl	tr	ur	vi	yo	zh
mBERT	64.6	45.8	59.6	52.3	58.2	**72.7**	45.2	81.8	80.8	64.0	67.5	50.7	48.5	3.6	71.7	71.8	36.9	71.8	**44.9**	**42.7**
XLM	67.7	**57.2**	26.3	59.4	62.4	69.6	47.6	81.2	77.9	63.5	68.4	53.6	49.6	0.3	**78.6**	71.0	43.0	70.1	26.5	32.4
XLMR	**71.6**	56.2	**60.0**	**67.8**	**68.1**	57.1	**54.3**	**84.0**	**81.9**	**69.1**	**70.5**	**59.5**	**55.8**	1.3	73.2	**76.1**	**56.4**	**79.4**	33.6	33.1
MMTE	60.9	43.9	58.2	44.8	58.5	68.3	42.9	74.8	72.9	58.2	66.3	48.1	46.9	**3.9**	64.1	61.9	37.2	68.1	32.1	28.9

Table A.7: NER results (F1 Score) for each language

Model	de	fr	ru	zh	avg
BERT	62.5	62.6	51.8	50.0	56.7
XLM	56.3	63.9	60.6	46.6	56.8
XLMR	67.5	**66.5**	**73.5**	**56.7**	**66.0**
MMTE	**67.9**	63.9	54.3	53.3	59.8

Table A.8: BUCC results (F1 scores) for each languages.

Model	en	de	es	fr	ja	ko	zh	**avg**
mBERT	94.0	85.7	87.4	87.0	73.0	69.6	77.0	81.9
XLM	94.0	85.9	88.3	87.4	69.3	64.8	76.5	80.9
XLMR	**94.7**	**89.7**	**90.1**	**90.4**	**78.7**	**79.0**	**82.3**	**86.4**
MMTE	93.1	85.1	87.2	86.9	72.0	69.2	75.9	81.3
Translate-train	94.0	87.5	89.4	89.6	78.6	81.6	83.5	86.3
Translate-train (multi-task)	**94.5**	**90.5**	**91.6**	**91.7**	**84.4**	**83.9**	**85.8**	**88.9**
Translate-test	93.5	88.2	89.3	87.4	78.4	76.6	77.6	84.4

Table A.9: PAWS-X accuracy scores for each language.

Model	en	ar	de	el	es	hi	ru	th	tr	vi	zh	**avg**
mBERT	83.5 / 72.2	61.5 / 45.1	70.6 / 54.0	62.6 / 44.9	75.5 / 56.9	59.2 / 46.0	71.3 / 53.3	42.7 / 33.5	55.4 / 40.1	69.5 / 49.6	58.0 / 48.3	64.5 / 49.4
XLM	74.2 / 62.1	61.4 / 44.7	66.0 / 49.7	57.5 / 39.1	68.2 / 49.8	56.6 / 40.3	65.3 / 48.2	35.4 / 24.5	57.9 / 41.2	65.8 / 47.6	49.7 / 39.7	59.8 / 44.3
XLMR	**86.5 / 75.7**	**68.6 / 49.0**	**80.4 / 63.4**	**79.8 / 61.7**	**82.0 / 63.9**	**76.7 / 59.7**	**80.1 / 64.3**	**74.2 / 62.8**	**75.9 / 59.3**	**79.1 / 59.0**	**59.3 / 50.0**	**76.6 / 60.8**
MMTE	80.1 / 68.1	63.2 / 46.2	68.8 / 50.3	61.3 / 35.9	72.4 / 52.5	61.3 / 47.2	68.4 / 45.2	48.4 / 35.9	58.1 / 40.9	70.9 / 50.1	55.8 / 36.4	64.4 / 46.2
Translate-train	83.5 / 72.2	68.0 / 51.1	75.6 / 60.7	70.0 / 53.0	80.2 / 63.1	69.6 / 55.4	75.0 / 59.7	36.9 / 33.5	68.9 / 54.8	75.6 / 56.2	66.2 / 56.6	70.0 / 56.0
Translate-train (multi-task)	86.0 / 74.5	71.0 / 54.1	78.8 / 63.9	74.2 / 56.1	82.4 / 66.2	71.3 / 56.2	78.1 / 63.0	38.1 / 34.5	70.6 / 55.7	78.5 / 58.8	67.7 / 58.7	72.4 / 58.3
Translate-test	**87.9 / 77.1**	**73.7 / 58.8**	**79.8 / 66.7**	**79.4 / 65.5**	**82.0 / 68.4**	**74.9 / 60.1**	**79.9 / 66.7**	**64.6 / 50.0**	**67.4 / 49.6**	**76.3 / 61.5**	**73.7 / 59.1**	**76.3 / 62.1**

Table A.10: XQuAD results (F1 / EM) for each language.

Model	en	ar	bn	fi	id	ko	ru	sw	te	avg
mBERT	**75.3 / 63.6**	62.2 / 42.8	49.3 / 32.7	59.7 / 45.3	64.8 / 45.8	**58.8 / 50.0**	60.0 / 38.8	57.5 / 37.9	49.6 / 38.4	59.7 / 43.9
XLM	66.9 / 53.9	59.4 / 41.2	27.2 / 15.0	58.2 / 41.4	62.5 / 45.8	14.2 / 5.1	49.2 / 30.7	39.4 / 21.6	15.5 / 6.9	43.6 / 29.1
XLM-R	71.5 / 56.8	**67.6 / 40.4**	**64.0 / 47.8**	**70.5 / 53.2**	**77.4 / 61.9**	31.9 / 10.9	**67.0 / 42.1**	**66.1 / 48.1**	**70.1 / 43.6**	**65.1 / 45.0**
MMTE	62.9 / 49.8	63.1 / 39.2	55.8 / 41.9	53.9 / 42.1	60.9 / 47.6	49.9 / 42.6	58.9 / 37.9	63.1 / 47.2	54.2 / 45.8	58.1 / 43.8
Translate-train	75.3 / 63.6	61.5 / 44.1	31.9 / 31.9	62.6 / 49.0	68.6 / 52.0	53.2 / 41.3	53.1 / 33.9	61.9 / 45.5	27.4 / 17.5	55.1 / 42.1
Translate-train (multi-task)	73.2 / 62.5	**71.8 / 54.2**	49.7 / 36.3	68.1 / 53.6	72.3 / 55.2	58.6 / 47.8	64.3 / 45.3	66.8 / 48.9	53.3 / 40.2	64.2 / 49.3
Translate-test	**75.9 / 65.9**	68.8 / 49.6	**66.7 / 48.1**	**72.0 / 56.6**	**76.8 / 60.9**	**69.2 / 55.7**	**71.4 / 54.3**	**73.3 / 53.8**	**75.1 / 59.2**	**72.1 / 56.0**
Monolingual	75.3 / 63.6	80.5 / 67.0	71.1 / 60.2	75.6 / 64.1	**81.3 / 70.4**	59.0 / 49.6	72.1 / 56.2	75.0 / 66.7	80.2 / 66.4	74.5 / 62.7
Monolingual few-shot	63.1 / 50.9	61.3 / 44.8	58.7 / 49.6	51.4 / 38.1	70.4 / 58.1	45.4 / 38.4	56.9 / 42.6	55.4 / 46.3	65.2 / 49.6	58.7 / 46.5
Joint monolingual	**77.6 / 69.3**	**82.7 / 69.4**	**79.6 / 69.9**	**79.2 / 67.8**	68.9 / 72.7	**68.9 / 59.4**	**75.8 / 59.2**	**81.9 / 74.3**	**83.4 / 70.3**	**77.6 / 68.0**

Table A.11: TyDiQA-GoldP results (F1 / EM) for each language.

Model	en	ar	de	es	hi	vi	zh	avg
mBERT	80.2 / 67.0	52.3 / 34.6	59.0 / 43.8	67.4 / 49.2	50.2 / 35.3	61.2 / 40.7	59.6 / 38.6	61.4 / 44.2
XLM	68.6 / 55.2	42.5 / 25.2	50.8 / 37.2	54.7 / 37.9	34.4 / 21.1	48.3 / 30.2	40.5 / 21.9	48.5 / 32.6
XLM-R	**83.5 / 70.6**	**66.6 / 47.1**	**70.1 / 54.9**	**74.1 / 56.6**	**70.6 / 53.1**	**74 / 52.9**	**62.1 / 37.0**	**71.6 / 53.2**
MMTE	78.5 / –	56.1 / –	58.4 / –	64.9 / –	46.2 / –	59.4 / –	58.3 / –	60.3 / 41.4
Translate-train	80.2 / 67.0	55.0 / 35.6	64.4 / 49.4	70.0 / 52.0	60.1 / 43.4	65.7 / 45.5	63.9 / 42.7	65.6 / 47.9
Translate-train (multi-task)	80.7 / 67.7	58.9 / 39.0	66.0 / 51.6	71.3 / 53.7	62.4 / 45.0	67.9 / 47.6	66.0 / 43.9	67.6 / 49.8
Translate-test	**83.8 / 71.0**	**65.3 / 46.4**	**71.2 / 54.0**	**73.9 / 55.9**	**71.0 / 55.1**	**70.6 / 54.0**	**67.2 / 50.6**	**71.9 / 55.3**

Table A.12: MLQA results (F1 / EM) for each language.

Lang.	af	ar	bg	de	el	en	es	et	eu	fa	fi	fr	he	hi	hu	id	it
mBERT	86.6	56.2	85.0	85.2	81.1	95.5	86.9	79.1	60.7	66.7	78.9	84.2	56.2	67.2	78.3	71.0	88.4
XLM	88.5	63.1	85.0	85.8	84.3	95.4	85.8	78.3	62.8	64.7	78.4	82.8	65.9	66.2	77.3	70.2	87.4
XLMR	**89.8**	**67.5**	**88.1**	**88.5**	**86.3**	96.1	**88.3**	**86.5**	**72.5**	**70.6**	**85.8**	**87.2**	**68.3**	**76.4**	**82.6**	72.4	**89.4**
MMTE	86.2	65.9	87.2	85.8	77.7	**96.6**	85.8	81.6	61.9	67.3	81.1	84.3	57.3	76.4	78.1	**73.5**	89.2
	ja	kk	ko	mr	nl	pt	ru	ta	te	th	tl	tr	ur	vi	yo	zh	avg
mBERT	**49.2**	70.5	49.6	69.4	88.6	86.2	85.5	59.0	75.9	41.7	81.4	68.5	57.0	53.2	**55.7**	61.6	71.5
XLM	49.0	70.2	50.1	68.7	88.1	84.9	86.5	59.8	76.8	55.2	76.3	66.4	61.2	52.4	20.5	65.4	71.3
XLMR	15.9	**78.1**	53.9	**80.8**	**89.5**	**87.6**	**89.5**	**65.2**	**86.6**	**47.2**	**92.2**	**76.3**	**70.3**	**56.8**	24.6	25.7	**73.8**
MMTE	48.6	70.5	**59.3**	74.4	83.2	86.1	88.1	63.7	81.9	43.1	80.3	71.8	61.1	56.2	51.9	**68.1**	73.5

Table A.13: POS results (Accuracy) for each language

Appendix B

Appendix of Chapter 4

B.1 Training Details for Reproducibility

Although English is not the best source language for some target languages [113], this zero-shot cross-lingual transfer setting is still practical useful as many NLP tasks only have English annotations. In the following paragraphs, we show details for reproducing our results on a zero-shot cross-lingual transfer setting.

Model: We use the same architecture as mBERT for AMBER, and we build our AMBER trained with the alignment objectives on top of the original mBERT implementation[1].

Pre-training: We first train the model on the Wikipedia data for 1M steps using the default hyper-parameters in the original repository except that we use a larger batch of 8,192 sentence pairs. The max number of subwords in the concatenation of each sentence pair is set to 256. To continue training AMBER with additional objectives on parallel data, we use 10K warmup steps with the peak learning rate of 1e-4, and use a linear decay of the learning rate. All models are pre-trained with our proposed objectives on TPU v3, and we use the same hyper-parameter setting for our AMBER variants in the experiments. We follow the practice of mBERT[2] to sample from multilingual data for training. We select the checkpoint of all models at the 1M step for a fair comparison. It takes about 1 week to finish the pre-training.

[1] https://github.com/google-research/bert

[2] https://github.com/google-research/bert/blob/master/multilingual.md#data-source-and-sampling

models	ar	bg	de	el	en	es	fr	hi	ru	tr	ur	vi	zh	Avg
mBERT (public)	14.9	85.2	89.3	82.8	95.3	85.7	84.1	65.1	86.0	67.5	57.4	18.5	58.9	68.5
XLM-15	17.5	86.1	89.3	85.4	95.7	85.9	84.9	63.9	86.8	69.3	55.1	18.0	57.2	68.8
XLM-100	17.1	85.8	89.3	85.7	95.4	85.3	84.3	67.0	87.1	65.0	62.0	19.2	**60.2**	69.5
XLM-R-base	17.6	**88.5**	91.1	**88.2**	95.8	87.2	85.7	70.1	88.9	72.7	61.6	19.2	27.9	68.8
XLM-R-large	**18.1**	87.4	**91.9**	87.9	**96.3**	**87.8**	**87.3**	**76.1**	**89.9**	**74.3**	**67.6**	**19.5**	26.5	**70.0**
AMBER (MLM, our mBERT)	15.4	86.6	90.1	84.3	95.5	86.5	84.6	68.2	86.8	69.0	59.2	18.7	62.1	69.8
AMBER (MLM+TLM)	**16.0**	**87.2**	**91.5**	**86.4**	**95.7**	86.9	85.2	67.7	**87.9**	72.9	57.9	19.1	**62.7**	70.5
AMBER (MLM+TLM+WA)	14.8	86.9	90.4	84.9	95.6	86.7	84.8	**72.5**	87.8	**73.9**	**63.8**	**19.5**	62.3	**71.1**
AMBER (MLM+TLM+WA+SA)	14.6	87.1	90.6	85.9	95.5	**87.0**	**86.0**	68.6	87.4	73.4	60.2	18.8	61.8	70.5

Table B.1: F1 scores of part-of-speech tag predictions from the Universal Dependency v2.3. Bold numbers highlight the highest scores across languages on the existing models (upper part) and AMBER variants (bottom part).

Fine-tuning: For fine-tuning the models on the downstream applications, we use the constant learning rate of 2e-5 as suggested in the original paper [47]. We fine-tune all the models for 10 epochs on the cross-lingual POS tag prediction task, and 5 epochs on the sentence classification task. We use the batch size of 32 for all the models. All models are fine-tuned on 2080Ti GPUs, and the training can be finished within 1 day.

Datasets: We use the same parallel data that is used to train XLM-15. The parallel data can be processed by this script.[3] All the datasets in the downstream applications can be downloaded by the script.[4]

B.2 Detailed Results

We show the detailed results over all languages on the cross-lingual POS task in Table B.1, on the PAWS-X task in Table B.2, on the XNLI task in Table B.3, and on the Tatoeba retrieval task in Table B.4.

[3] https://github.com/facebookresearch/XLM/blob/master/get-data-para.sh

[4] https://github.com/google-research/xtreme/blob/master/scripts/download_data.sh

Model	de	en	es	fr	zh	Avg
mBERT (public)	85.7	94.0	87.4	87.0	77.0	86.2
XLM-15	88.5	**94.7**	89.3	89.6	78.1	88.0
XLM-100	85.9	94.0	88.3	87.4	76.5	86.4
XLM-R-base	87.0	94.2	88.6	88.7	78.5	87.4
XLM-R-large	**89.7**	**94.7**	**90.1**	**90.4**	**82.3**	**89.4**
AMBER (MLM, our mBERT)	87.3	93.9	87.5	87.8	78.8	87.1
AMBER (MLM+TLM)	87.6	**95.8**	87.4	88.9	78.7	87.7
AMBER (MLM+TLM+WA)	88.9	95.5	88.9	**90.7**	**81.1**	89.0
AMBER (MLM+TLM+WA+SA)	**89.4**	95.6	**89.2**	**90.7**	80.9	**89.2**

Table B.2: Accuracy of zero-shot cross-lingual classification on PAWS-X. Bold numbers highlight the highest scores across languages on the existing models (upper part) and AMBER variants (bottom part).

Models	en	zh	es	de	ar	ur	ru	bg	el	fr	hi	sw	th	tr	vi	avg
mBERT (public)	80.8	67.8	73.5	70.0	64.3	57.2	67.8	68.0	65.3	73.4	58.9	49.7	54.1	60.9	69.3	65.4
XLM-15	84.1	68.8	77.8	75.7	70.4	62.2	75.0	75.7	73.3	78.0	67.3	67.5	70.5	70.0	73.0	72.6
XLM-100	82.8	70.2	75.5	72.7	66.0	59.8	69.9	71.9	70.4	74.3	62.5	58.1	65.5	66.4	70.7	69.1
XLM-R-base	83.9	73.6	78.3	75.2	71.9	65.4	75.1	76.7	75.4	77.4	69.1	62.2	72.0	70.9	74.0	73.4
XLM-R-large	**88.7**	**78.2**	**83.7**	**82.5**	**77.2**	**71.7**	**79.1**	**83.0**	**80.8**	**82.2**	**75.6**	**71.2**	**77.4**	**78.0**	**79.3**	**79.2**
AMBER (MLM, our mBERT)	82.1	71.0	75.3	72.7	66.2	60.1	70.4	71.3	67.9	74.4	63.6	50.1	55.0	64.2	71.6	67.7
AMBER (MLM+TLM)	84.3	71.6	**77.2**	73.9	69.1	59.6	72.5	73.6	70.9	**78.0**	64.7	57.4	65.0	72.2	73.1	70.9
AMBER (MLM+TLM+WA)	84.1	**72.1**	76.6	**74.7**	69.3	**61.5**	72.9	73.9	71.6	77.7	65.7	58.6	65.3	72.7	**73.4**	71.3
AMBER (MLM+TLM+WA+SA)	**84.7**	71.6	76.9	74.2	**70.2**	61.0	**73.3**	**74.3**	**72.5**	76.6	**66.2**	**59.9**	**65.7**	**73.2**	**73.4**	**71.6**

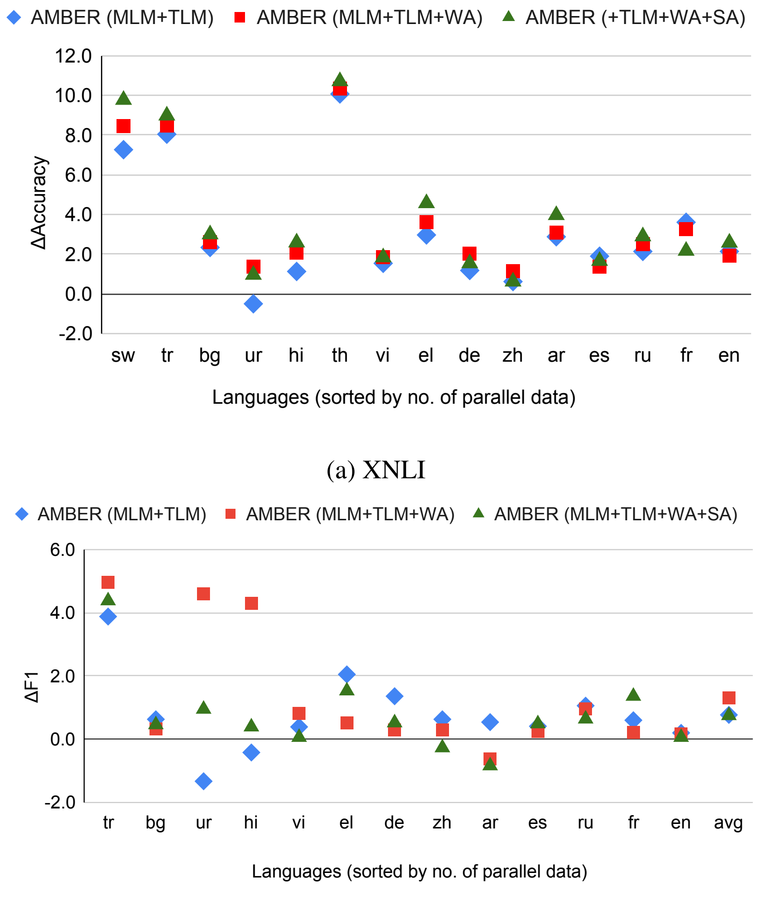
Table B.3: Accuracy of zero-shot crosslingual classification on the XNLI dataset. Bold numbers highlight the highest scores across languages on the existing models (upper part) and AMBER variants (bottom part).

B.3 Detailed Results on Performance Difference by Languages

Figure B.1b and Figure B.1a show the performance difference between AMBER trained with alignment objectives and AMBER trained with only MLM objective on the POS and XNLI tasks over all languages.

Method	ar	bg	de	el	es	fr	hi	ru	sw	th	tr	ur	vi	zh	Avg
mBERT (public)	25.8	49.3	77.2	29.8	68.7	66.3	34.8	61.2	11.5	13.7	34.8	31.6	62.0	71.6	45.6
XLM-15	**63.5**	71.5	**92.6**	**73.1**	**85.5**	**82.5**	**81.0**	**82.0**	**47.9**	**90.3**	**67.6**	**68.4**	**91.1**	**84.1**	**77.2**
XLM-100	18.2	40.0	66.2	25.6	58.4	54.5	26.5	44.8	12.6	31.8	26.2	18.1	47.1	42.2	36.6
XLM-R-base	36.8	67.6	89.9	53.7	74.0	74.1	54.2	72.5	19.0	38.3	61.1	36.6	68.4	60.8	57.6
XLM-R-large	47.5	**71.6**	88.8	61.8	75.7	73.7	72.2	74.1	20.3	29.4	65.7	24.3	74.7	68.3	60.6
AMBER (MLM, our mBERT)	30.7	54.9	81.4	37.7	72.7	72.7	47.5	67.5	15.1	25.7	48.3	42.6	64.6	75.1	52.6
AMBER (MLM+TLM)	47.1	61.8	89.0	53.8	76.3	77.9	72.3	69.8	20.5	83.4	88.1	50.0	86.9	78.0	68.2
AMBER (MLM+TLM+WA)	46.8	63.3	88.8	52.2	78.3	79.5	66.9	71.6	27.4	77.2	86.9	56.5	86.5	81.6	68.8
AMBER (MLM+TLM+WA+SA)	**78.5**	**87.1**	**95.5**	**75.3**	**93.3**	**92.2**	**95.0**	**91.5**	**52.8**	**94.5**	**98.4**	**84.5**	**97.4**	**94.3**	**87.9**

Table B.4: Sentence retrieval accuracy on the Tatoeba dataset. Bold numbers highlight the highest scores across languages on the existing models (upper part) and AMBER variants (bottom part).



(a) XNLI

(b) POS

Figure B.1: Performance difference between AMBER trained with alignments on parallel data and AMBER (MLM) on XNLI and POS task. Languages are sorted by no. of parallel data used for training AMBER with alignments.

Bibliography

[1] Željko Agić and Natalie Schluter. Baselines and test data for cross-lingual inference. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). 3.2.4

[2] Roee Aharoni and Yoav Goldberg. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online, July 2020. Association for Computational Linguistics. 7.4.1, 7.5.1, 8.1

[3] Roee Aharoni, Melvin Johnson, and Orhan Firat. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 1, 1

[4] Yaser Al-Onaizan and Kevin Knight. Translating named entities using monolingual and bilingual resources. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 400–408, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. 6.1, 6.5

[5] Yaser Al-Onaizan and Kevin Knight. Named entity translation. In *Proceedings of the second international conference on Human Language Technology Research*, pages 122–124, 2002. 6.1

[6] Tamer Alkhouli, Gabriel Bretschner, and Hermann Ney. On the alignment problem in multi-head attention-based neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 177–185, Brussels, Belgium, October 2018. Association for Computational Linguistics. 4.5

[7] Sankaranarayanan Ananthakrishnan, Rohit Prasad, David Stallard, and Prem Natarajan. A semi-supervised batch-mode active learning strategy for improved statistical machine translation. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 126–134, Uppsala, Sweden, July 2010. Association for Computational Linguistics. 7.6

[8] M. Arbabi, S. M. Fischthal, V. C. Cheng, and E. Bart. Algorithms for arabic name transliteration. *IBM Journal of Research and Development*, 38(2):183–194, 1994. 6.5

[9] Mihael Arcan and P. Buitelaar. Translating domain-specific expressions in knowledge bases with neural machine translation. *ArXiv*, abs/1709.02184, 2017. 5.4

[10] Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. Massively Multilingual Neural Machine Translation in the Wild: Findings and Challenges. *ArXiv*, abs/1907.05019, 2019. 1, 3.3.2

[11] Mikel Artetxe and Holger Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610, March 2019. 3.2.2, 3, 4.3.2

[12] Mikel Artetxe and Holger Schwenk. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy, July 2019. Association for Computational Linguistics. 7.3.1

[13] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada, July 2017. Association for Computational Linguistics. 1, 3.1, 3.5

[14] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia, July 2018. Association for Computational Linguistics. 3.5

[15] Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. Unsupervised neural machine translation. *International Conference on Learning Representations*, 2018. 5.4

[16] Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the cross-lingual transferability

of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online, July 2020. Association for Computational Linguistics. 2.1.2, 3.2.2, 3.4, 3.5

[17] Philip Arthur, Graham Neubig, and Satoshi Nakamura. Incorporating discrete translation lexicons into neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567, Austin, Texas, November 2016. Association for Computational Linguistics. 5.4, 7.6

[18] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, 2015. 1, 2.1.1, 5.1

[19] 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. ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online, July 2020. Association for Computational Linguistics. 6.4.4

[20] Jeremy Barnes, Roman Klinger, and Sabine Schulte im Walde. Bilingual sentiment embeddings: Joint projection of sentiment across languages. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2483–2493, Melbourne, Australia, July 2018. Association for Computational Linguistics. 3.5

[21] Michael Bloodgood and Chris Callison-Burch. Bucking the trend: Large-scale cost-focused active learning for statistical machine translation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 854–864, Uppsala, Sweden, July 2010. Association for Computational Linguistics. 7.1, 7.3.2, 7.3.2, 7.6

[22] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. ISSN 2307-387X. 5.3.1

[23] Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Philipp Koehn, and Christof Monz. Findings of the 2018 conference on machine translation (WMT18). In *Proceedings of the Third Conference on Machine Translation: Shared Task*

Papers, pages 272–303, Belgium, Brussels, October 2018. Association for Computational Linguistics. 2.1.1

[24] Denny Britz, Quoc Le, and Reid Pryzant. Effective domain mixing for neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 118–126. Association for Computational Linguistics, 2017. 2.1.1

[25] P. Brown, J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, R. Mercer, and P. Roossin. A statistical approach to language translation. In *Coling Budapest 1988 Volume 1: International Conference on Computational Linguistics*, 1988. 2.1.1

[26] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993. 4.4

[27] Steven Cao, Nikita Kitaev, and Dan Klein. Multilingual alignment of contextual word representations. In *International Conference on Learning Representations*, 2020. 4.1, **??**, 4.3, 4.3.5, 4.4

[28] Boxing Chen and Colin Cherry. A systematic comparison of smoothing techniques for sentence-level BLEU. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. 7.3.1

[29] Yufeng Chen, Chengqing Zong, and Keh-Yih Su. A joint model to identify and align bilingual named entities. *Computational Linguistics*, 39(2):229–266, 2013. 6.1

[30] Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974, Berlin, Germany, August 2016. Association for Computational Linguistics. 5.4

[31] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online, June 2021. Association for Computational Linguistics. 2.1.2, 3.6

[32] Zewen Chi, Li Dong, Bo Zheng, Shaohan Huang, Xian-Ling Mao, Heyan Huang, and Furu

Wei. Improving pretrained cross-lingual language models via self-labeled word alignment. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*, Online, August 2021. Association for Computational Linguistics. 8.1

[33] Mara Chinea-Rios, Álvaro Peris, and Francisco Casacuberta. Adapting neural machine translation with parallel synthetic data. In *Proceedings of the Second Conference on Machine Translation*, pages 138–147, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. 5.4

[34] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, October 2014. Association for Computational Linguistics. 2.1.1

[35] Chenhui Chu and Rui Wang. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. 1, 2.1.1, 7.4.1

[36] Chenhui Chu, Raj Dabre, and Sadao Kurohashi. An empirical comparison of domain adaptation methods for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–391, Vancouver, Canada, July 2017. Association for Computational Linguistics. 7.1, 8.2

[37] Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. In *Transactions of the Association of Computational Linguistics*, 2020. 3.2.2, 3.3.2

[38] Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. Incorporating structural alignment biases into an attentional neural translation model. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 876–885, San Diego, California, June 2016. Association for Computational Linguistics. 4.1, 4.2.2, 4.2.2

[39] Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 1, 2.1.2, 2.3.1, 2.3.1, 2.3.2, 3.1, 3.3.2, 3.5, 4.1, 4.3.1, 4.4

[40] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. 2018. 2.1.2, 3.1, 3.5, 5.1, 5.2.1, 5.2.1, 5.2.1

[41] Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. 3.1, 3.2.2, 3.5, 4.3.2

[42] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics. 2.1.2, 2.3.1, 3.3.2, 3.4, 4, 4.1, 4.4

[43] Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online, July 2020. Association for Computational Linguistics. 4.1

[44] Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. 2.1.1, 2.3.2, 2.3.2, 5.3.2, 5.4

[45] Paula Czarnowska, Sebastian Ruder, Edouard Grave, Ryan Cotterell, and Ann Copestake. Don't forget the long tail! a comprehensive analysis of morphological generalization in bilingual lexicon induction. In *Proceedings of EMNLP 2019*, pages 973–982, 2019. 3.5

[46] Hal Daumé III and Jagadeesh Jagarlamudi. Domain adaptation for machine translation by mining unseen words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 407–412, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. 5.1, 5.4, 7.1

[47] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 1, 2.1.2, 2.3.1, 3.1, 3.3.2, 3.5, 4, 4.1, 4.4, 7.3.1, B.1

[48] Sumanth Doddapaneni, Gowtham Ramesh, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M. Khapra. A primer on pretrained multilingual language models. *ArXiv*, abs/2107.00676, 2021. 2.1.2

[49] Tobias Domhan and Felix Hieber. Using target-side monolingual data for neural machine translation through multi-task learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1500–1505, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. 2.1.1, 6.5

[50] Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732, Beijing, China, July 2015. Association for Computational Linguistics. 2.1.1

[51] Zi-Yi Dou, Junjie Hu, Antonios Anastasopoulos, and Graham Neubig. Unsupervised domain adaptation for neural machine translation with domain-aware feature embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1417–1422, Hong Kong, China, November 2019. Association for Computational Linguistics. 1.1

[52] David M Eberhard, Gary F Simons, and Charles D Fennig. *Ethnologue: Languages of Asia*. SIL International, 2019. 1

[53] Matthias Eck, Stephan Vogel, and Alex Waibel. Low cost portability for statistical machine translation based on n-gram frequency and tf-idf. In *International Workshop on Spoken Language Translation (IWSLT) 2005*, 2005. 7.6

[54] Julian Eisenschlos, Sebastian Ruder, Piotr Czapla, Marcin Kadras, Sylvain Gugger, and Jeremy Howard. MultiFiT: Efficient Multi-lingual Language Model Fine-tuning. In *Proceedings of EMNLP 2019*, 2019. 3.3.2, 3.3.4

[55] Akiko Eriguchi, Melvin Johnson, Orhan Firat, Hideto Kazawa, and Wolfgang Macherey. Zero-shot cross-lingual classification using multilingual neural machine translation. *ArXiv*, abs/1809.04686, 2018. 3.5, 4.1, 4.4

[56] Marzieh Fadaee, Arianna Bisazza, and Christof Monz. Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573, Vancouver, Canada, July 2017. Association for Computational Linguistics. 5.4, 7.4.2

[57] Manaal Faruqui and Chris Dyer. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 462–471, Gothenburg, Sweden, April 2014. Association for Computational Linguistics. 1, 2.1.2, 3.1, 3.5, 4.4

[58] Andrew Finch, Lemao Liu, Xiaolin Wang, and Eiichiro Sumita. Target-bidirectional neural models for machine transliteration. In *Proceedings of the Sixth Named Entity Workshop*, pages 78–82, Berlin, Germany, August 2016. Association for Computational Linguistics. 6.5

[59] Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875, San Diego, California, June 2016. Association for Computational Linguistics. 2.1.1

[60] Markus Freitag and Yaser Al-Onaizan. Fast domain adaptation for neural machine translation. *ArXiv*, abs/1612.06897, 2016. 2.1.1

[61] Philip Gage. A new algorithm for data compression. *C Users Journal*, 12(2):23–38, 1994. 2.2

[62] Rashmi Gangadharaiah, Ralf D. Brown, and Jaime Carbonell. Active learning in example-based machine translation. In *Proceedings of the 17th Nordic Conference of Computational Linguistics (NODALIDA 2009)*, pages 227–230, Odense, Denmark, May 2009. Northern European Association for Language Technology (NEALT). 7.1, 7.3.1

[63] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *International Conference on Machine Learning*, pages 1243–1252. PMLR, 2017. 2.2

[64] Hamidreza Ghader and Christof Monz. What does attention in neural machine translation

pay attention to? In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 30–39, Taipei, Taiwan, November 2017. Asian Federation of Natural Language Processing. 4.5

[65] John Gittins, Kevin Glazebrook, and Richard Weber. *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011. 7.7

[66] Goran Glavaš, Robert Litschko, Sebastian Ruder, and Ivan Vulić. How to (properly) evaluate cross-lingual word embeddings: On strong baselines, comparative analyses, and some misconceptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 710–721, Florence, Italy, July 2019. Association for Computational Linguistics. 3.1, 3.5

[67] Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. Active learning for interactive machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 245–254, Avignon, France, April 2012. Association for Computational Linguistics. 7.6

[68] Stephan Gouws, Yoshua Bengio, and Greg Corrado. Bilbowa: Fast bilingual distributed representations without word alignments. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 748–756, Lille, France, 07–09 Jul 2015. PMLR. 3.5

[69] Milan Gritta and Ignacio Iacobacci. Xeroalign: Zero-shot cross-lingual transformer alignment. In *Findings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*, Online, August 2021. Association for Computational Linguistics. 8.1

[70] Roman Grundkiewicz and Kenneth Heafield. Neural machine translation techniques for named entity transliteration. In *Proceedings of the Seventh Named Entities Workshop*, pages 89–94, Melbourne, Australia, July 2018. Association for Computational Linguistics. 6.5

[71] Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. On using monolingual corpora in neural machine translation. *ArXiv*, abs/1503.03535, 2015. 2.1.1, 6.5

[72] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman,

and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. 3.5

[73] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online, July 2020. Association for Computational Linguistics. 6.5

[74] Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6098–6111, Hong Kong, China, November 2019. Association for Computational Linguistics. 3.5, 6.3

[75] Mohamed Seghir Hadj Ameur, Farid Meziane, and Ahmed Guessoum. Arabic machine transliteration using an attention-based encoder-decoder model. *Procedia Computer Science*, 117:287–297, 2017. ISSN 1877-0509. Arabic Computational Linguistics. 6.5

[76] Gholamreza Haffari, Maxim Roy, and Anoop Sarkar. Active learning for statistical phrase-based machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 415–423, Boulder, Colorado, June 2009. Association for Computational Linguistics. 7.1, 7.3.1, 7.6

[77] Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, C. Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, W. Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, F. Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and M. Zhou. Achieving human parity on automatic chinese to english news translation. *ArXiv*, abs/1803.05567, 2018. 1

[78] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma.

Dual learning for machine translation. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 4.2.2

[79] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia, July 2018. Association for Computational Linguistics. 3.5

[80] Tsung-yuan Hsu, Chi-liang Liu, and Hung-yi Lee. Zero-shot Reading Comprehension by Cross-lingual Transfer Learning with Multi-lingual Language Representation Model. In *Proceedings of EMNLP 2019*, pages 5935–5942, 2019. A.2

[81] Junjie Hu, Mengzhou Xia, Graham Neubig, and Jaime Carbonell. Domain adaptation of neural machine translation by lexicon induction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2989–3001, Florence, Italy, July 2019. Association for Computational Linguistics. 2.3.2, 5

[82] Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR, 13–18 Jul 2020. 3, 4, 4.1, 4.3.1, 4.3.2, 1

[83] Junjie Hu, Melvin Johnson, Orhan Firat, Aditya Siddhant, and Graham Neubig. Explicit alignment objectives for multilingual bidirectional encoders. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3633–3643, Online, June 2021. Association for Computational Linguistics. 2.1.2

[84] Fei Huang, Stephan Vogel, and Alex Waibel. 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, Sapporo, Japan, July 2003. Association for Computational Linguistics. 6.1, 6.5

[85] Fei Huang, Stephan Vogel, and Alex Waibel. Improving named entity translation combining phonetic and semantic similarities. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 281–288, Boston, Massachusetts, USA, May 2 - May 7

2004. Association for Computational Linguistics. 6.1, 6.5

[86] Fei Huang, Ying Zhang, and Stephan Vogel. Mining key phrase translations from web corpora. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 483–490, Vancouver, British Columbia, Canada, October 2005. Association for Computational Linguistics. 6.1, 6.5

[87] Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2485–2494, Hong Kong, China, November 2019. Association for Computational Linguistics. 2.1.2, 4.1, 4.4

[88] Ann Irvine, John Morgan, Marine Carpuat, Hal Daumé III, and Dragos Munteanu. Measuring machine translation errors in new domains. *Transactions of the Association for Computational Linguistics*, 1:429–440, 2013. 1

[89] Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017. 1, 2.1.1

[90] Shafiq Joty, Hassan Sajjad, Nadir Durrani, Kamla Al-Mannai, Ahmed Abdelali, and Stephan Vogel. How to avoid unwanted pregnancies: Domain adaptation using neural network models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1259–1270. Association for Computational Linguistics, 2015. 2.1.1

[91] Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. Cross-lingual ability of multilingual bert: An empirical study. In *International Conference on Learning Representations*, 2020. 4.1

[92] Nal Kalchbrenner and Phil Blunsom. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. 2.1.1

[93] Yova Kementchedjhieva, Mareike Hartmann, and Anders Søgaard. Lost in evaluation: Mis-

leading benchmarks for bilingual dictionary induction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3336–3341, Hong Kong, China, November 2019. Association for Computational Linguistics. 3.5

[94] Kevin Knight and Jonathan Graehl. Machine transliteration. *Computational Linguistics*, 24(4):599–612, 1998. 6.1, 6.5

[95] Catherine Kobus, Josep Crego, and Jean Senellart. Domain control for neural machine translation. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 372–378. INCOMA Ltd., 2017. 2.1.1

[96] Philipp Koehn. *Statistical machine translation*. Cambridge University Press, 2009. 2.1.1

[97] Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver, August 2017. Association for Computational Linguistics. 1, 2.1.1, 5.3.1, 7.5.1

[98] Moshe Koppel and Noam Ordan. Translationese and its dialects. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1318–1326, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. 8

[99] Sachith Sri Ram Kothur, Rebecca Knowles, and Philipp Koehn. Document-level adaptation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 64–73, Melbourne, Australia, July 2018. Association for Computational Linguistics. 5.4

[100] Tsz Kin Lam, Shigehiko Schamoni, and Stefan Riezler. Interactive-predictive neural machine translation through reinforcement and imitation. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, pages 96–106, Dublin, Ireland, August 2019. European Association for Machine Translation. 7.6

[101] Patrik Lambert, Holger Schwenk, Christophe Servan, and Sadaf Abdul-Rauf. Investigations on translation model adaptation using monolingual data. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 284–293, Edinburgh, Scotland, July 2011. Association for Computational Linguistics. 2.3.2, 5.4

[102] Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Unsupervised machine translation using monolingual corpora only. *International Conference on Learning Representations*, 2018. 5.4

[103] Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. 5.4

[104] Samuel Laubli, Sheila Castilho, Graham Neubig, Rico Sennrich, Qinlan Shen, and Antonio Toral. A set of recommendations for assessing human–machine parity in language translation. *Journal of Artificial Intelligence Research (JAIR)*, 67, February 2020. 6.1

[105] Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavas. From zero to hero: On the limitations of zero-shot cross-lingual transfer with multilingual transformers. *ArXiv*, abs/2005.00633, 2020. 3.6

[106] Jason Lee, Kyunghyun Cho, and Thomas Hofmann. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378, 2017. 2.1.1

[107] Kyungjae Lee, Kyoungho Yoon, Sunghyun Park, and Seung-won Hwang. Semi-supervised training data generation for multilingual question answering. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). A.2

[108] Joël Legrand, Michael Auli, and Ronan Collobert. Neural network-based word alignment through score aggregation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 66–73, Berlin, Germany, August 2016. Association for Computational Linguistics. 4.5

[109] Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. Pre-training via paraphrasing. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18470–18481. Curran Associates, Inc., 2020. 3.6

[110] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. 2.3.2

[111] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. MLQA:

Evaluating cross-lingual extractive question answering. pages 7315–7330, July 2020. 3.2.2, 3.5, A.2

[112] Xintong Li, Guanlin Li, Lemao Liu, Max Meng, and Shuming Shi. On the word alignment from neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1293–1303, Florence, Italy, July 2019. Association for Computational Linguistics. 4.5

[113] Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135, Florence, Italy, July 2019. Association for Computational Linguistics. 3.3.1, B.1

[114] Zehui Lin, Xiao Pan, Mingxuan Wang, Xipeng Qiu, Jiangtao Feng, Hao Zhou, and Lei Li. Pre-training multilingual neural machine translation by leveraging alignment information. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2649–2663, Online, November 2020. Association for Computational Linguistics. 2.3.2

[115] Dayiheng Liu, Yu Yan, Yeyun Gong, Weizhen Qi, Hang Zhang, Jian Jiao, Weizhu Chen, Jie Fu, Linjun Shou, Ming Gong, Pengcheng Wang, Jiusheng Chen, Daxin Jiang, Jiancheng Lv, Ruofei Zhang, Winnie Wu, Ming Zhou, and Nan Duan. Glge: A new general language generation evaluation benchmark. *ArXiv*, abs/2011.11928, 2020. 3.6

[116] Pengfei Liu, Jinlan Fu, Yanghua Xiao, Weizhe Yuan, Shuaicheng Chang, Junqi Dai, Yixin Liu, Zihuiwen Ye, Zi-Yi Dou, and Graham Neubig. Explainaboard: An explainable leaderboard for nlp. *ArXiv*, abs/2104.06387, 2021. 8.1

[117] Ying Liu. The technical analyses of named entity translation. In *2015 International Symposium on Computers & Informatics*, pages 2028–2037. Atlantis Press, 2015. 6.1

[118] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019. 4.3.1, 4.3.3

[119] Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742,

2020. 2.3.2, 2.3.2, 6.1, 6.2.2, 6.3, 6.3, 6.3, 6.3

[120] Minh-Thang Luong and Christopher D Manning. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the International Workshop on Spoken Language Translation*, 2015. 2.1.1, 5.4, 6.1, 7.1, 7.4, 8.2

[121] Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. Multi-task sequence to sequence learning. In *International Conference on Learning Representations*, 2016. 2.1.1

[122] Thang Luong, Hieu Pham, and Christopher D. Manning. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, 2015. 3.5

[123] Christopher D Manning. Part-of-speech tagging from 97% to 100%: is it time for some linguistics? In *International conference on intelligent text processing and computational linguistics*, pages 171–189. Springer, 2011. 3.3.2

[124] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 3.5, 4.1, 4.4

[125] Tomas Mikolov, Quoc V Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *ArXiv*, abs/1309.4168, 2013. 1, 3.1, 3.5

[126] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. 2.1.2

[127] Akiva Miura, Graham Neubig, Michael Paul, and Satoshi Nakamura. Selecting syntactic, non-redundant segments in active learning for machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 20–29, San Diego, California, June 2016. Association for Computational Linguistics. 7.1, 7.3.1, 7.3.2, 7.3.2, 7.6

[128] Saif M Mohammad, Mohammad Salameh, and Svetlana Kiritchenko. How translation alters sentiment. *Journal of Artificial Intelligence Research*, 55:95–130, 2016. 3.5

[129] Robert C. Moore and William Lewis. Intelligent selection of language model training

data. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (Short Papers)*, Uppsala, Sweden, July 2010. Association for Computational Linguistics. 7.4.1

[130] Mathias Müller, Annette Rios, and Rico Sennrich. Domain robustness in neural machine translation. pages 151–164, October 2020. 8.2

[131] Nikita Nangia and Samuel R. Bowman. Human vs. Muppet: A Conservative Estimate of Human Performance on the GLUE Benchmark. In *Proceedings of ACL 2019*, pages 4566–4575, 2019. 3.3.2

[132] Graham Neubig and Junjie Hu. Rapid adaptation of neural machine translation to new languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 875–880, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. 6.1

[133] Graham Neubig, Zi-Yi Dou, Junjie Hu, Paul Michel, Danish Pruthi, and Xinyi Wang. compare-mt: A tool for holistic comparison of language generation systems. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 35–41, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 8.2

[134] Joakim Nivre, Mitchell Abrams, Željko Agić, Lars Ahrenberg, Lene Antonsen, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, et al. Universal dependencies 2.2. 2018. 3.2.2, 4.3.2

[135] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003. 5.2.1

[136] Xuan Ouyang, Shuohuan Wang, Chao Pang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-m: Enhanced multilingual representation by aligning cross-lingual semantics with monolingual corpora. *ArXiv*, abs/2012.15674, 2020. 2.1.2

[137] Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada, July 2017. Association for Computational Linguistics. 3.2.2

[138] Álvaro Peris and Francisco Casacuberta. Active learning for interactive neural machine translation of data streams. In *Proceedings of the 22nd Conference on Computational Nat-*

ural Language Learning, pages 151–160, Brussels, Belgium, October 2018. Association for Computational Linguistics. 7.1, 7.6

[139] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. 3.5

[140] Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy, July 2019. Association for Computational Linguistics. 2.1.2, 3.5, 1

[141] Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online, November 2020. Association for Computational Linguistics. 3.6

[142] Maja Popović. chrF: character n-gram f-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, 2015. 3.2.4

[143] Afshin Rahimi, Yuan Li, and Trevor Cohn. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy, July 2019. Association for Computational Linguistics. 3.2.2, 3.5

[144] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of EMNLP 2016*, 2016. 3.2.2, 3.3.2

[145] Prajit Ramachandran, Peter Liu, and Quoc Le. Unsupervised pretraining for sequence to sequence learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 383–391, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. 5.4

[146] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics, pages 4902–4912, Online, July 2020. Association for Computational Linguistics. 8.1

[147] Richard H Richens. Interlingual machine translation. *The Computer Journal*, 1(3):144–147, 1958. 2.1.1

[148] Michael Ringgaard, Rahul Gupta, and Fernando CN Pereira. Sling: A framework for frame semantic parsing. *ArXiv*, abs/1710.07032, 2017. 6.2.1

[149] Uma Roy, Noah Constant, Rami Al-Rfou, Aditya Barua, Aaron Phillips, and Yinfei Yang. LAReQA: Language-agnostic answer retrieval from a multilingual pool. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5919–5930, Online, November 2020. Association for Computational Linguistics. 3.6

[150] Sebastian Ruder, Ivan Vulić, and Anders Søgaard. A Survey of Cross-lingual Word Embedding Models. *Journal of Artificial Intelligence Research*, 65:569–631, 2019. 1

[151] Sebastian Ruder, Noah Constant, Jan A. Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Graham Neubig, and Melvin Johnson. Xtreme-r: Towards more challenging and nuanced multilingual evaluation. *ArXiv*, abs/2104.07412, 2021. 3, 3.6, 8.1

[152] Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1599–1613, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 3.5, 4.1

[153] Holger Schwenk. Investigations on large-scale lightly-supervised training for statistical machine translation. In *International Workshop on Spoken Language Translation (IWSLT)*, 2008. 5.4

[154] Holger Schwenk and Xian Li. A corpus for multilingual document classification in eight languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). 3.1, 3.5

[155] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the*

Association for Computational Linguistics (Volume 1: Long Papers), pages 86–96, Berlin, Germany, August 2016. Association for Computational Linguistics. 2.3.2, 2.3.2

[156] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, August 2016. Association for Computational Linguistics. 2.1.1, 5.2.2, 5.3.2, 5.4

[157] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. 2.1.1, 2.2, 5.3.1, 7.5.1

[158] Aditya Siddhant, Melvin Johnson, Henry Tsai, Naveen Ari, Jason Riesa, Ankur Bapna, Orhan Firat, and Karthik Raman. Evaluating the cross-lingual effectiveness of massively multilingual neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8854–8861, 2020. 3.3.2

[159] Laura Smith, Salvatore Giorgi, Rishi Solanki, Johannes Eichstaedt, H. Andrew Schwartz, Muhammad Abdul-Mageed, Anneke Buffone, and Lyle Ungar. Does 'well-being' translate on Twitter? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2042–2047, Austin, Texas, November 2016. Association for Computational Linguistics. 3.5

[160] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. MASS: Masked sequence to sequence pre-training for language generation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936. PMLR, 09–15 Jun 2019. 2.3.2, 2.3.2, 6.1

[161] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 2.1.1

[162] Jörg Tiedemann. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). 6.3

[163] Daniel Torregrosa, Nivranshu Pasricha, Maraim Masoud, Bharathi Raja Chakravarthi, Juan Alonso, Noe Casas, and Mihael Arcan. Aspects of terminological and named entity knowledge within rule-based machine translation models for under-resourced neural machine translation scenarios. *ArXiv*, abs/2009.13398, 2020. 6.5

[164] Arata Ugawa, Akihiro Tamura, Takashi Ninomiya, Hiroya Takamura, and Manabu Okumura. Neural machine translation incorporating named entity. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3240–3250, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. 6.5

[165] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 6.3

[166] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 2.2, 3.3.2, 5.1, 7.5.1

[167] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014. 6.1, 6.2.1

[168] Ivan Vulić, Goran Glavaš, Roi Reichart, and Anna Korhonen. Do we really need fully unsupervised cross-lingual embeddings? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4407–4418, Hong Kong, China, November 2019. Association for Computational Linguistics. 3.5

[169] Stephen Wan and Cornelia Maria Verspoor. Automatic English-Chinese name transliteration for development of multilingual resources. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*, pages 1352–1356, Montreal, Quebec, Canada, August 1998. Association for Computational Linguistics. 6.1, 6.5

[170] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing*

Systems, volume 32. Curran Associates, Inc., 2019. 3.5

[171] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *International Conference on Learning Representations*, 2019. 3.5

[172] Rui Wang, Andrew Finch, Masao Utiyama, and Eiichiro Sumita. Sentence embedding for neural machine translation domain adaptation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 560–566, Vancouver, Canada, July 2017. Association for Computational Linguistics. 2.1.1

[173] Yuxuan Wang, Wanxiang Che, Jiang Guo, Yijia Liu, and Ting Liu. Cross-lingual BERT transformation for zero-shot dependency parsing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5721–5727, Hong Kong, China, November 2019. Association for Computational Linguistics. 2.1.2

[174] Zirui Wang, Zachary C. Lipton, and Yulia Tsvetkov. On negative interference in multilingual models: Findings and a meta-learning treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4438–4450, Online, November 2020. Association for Computational Linguistics. 3.6

[175] Zirui Wang, Jiateng Xie, Ruochen Xu, Yiming Yang, Graham Neubig, and Jaime G. Carbonell. Cross-lingual alignment vs joint training: A comparative study and a simple unified framework. In *International Conference on Learning Representations*, 2020. 4.1, 4.3.5, 4.4

[176] Warren Weaver et al. Translation. *Machine translation of languages*, 14(15-23):10, 1955. 2.1.1

[177] Xiangpeng Wei, Rongxiang Weng, Yue Hu, Luxi Xing, Heng Yu, and Weihua Luo. On learning universal representations across languages. In *International Conference on Learning Representations*, 2021. 2.1.2

[178] John Wieting, Kevin Gimpel, Graham Neubig, and Taylor Berg-Kirkpatrick. Simple and effective paraphrastic similarity from parallel translations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4602–4608, Florence, Italy, July 2019. Association for Computational Linguistics. 4.1

[179] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the*

North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. 3.2.2, 4.3.2

[180] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. 3.3.3

[181] Jian-Cheng Wu and Jason S. Chang. Learning to find English to Chinese transliterations on the web. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 996–1004, Prague, Czech Republic, June 2007. Association for Computational Linguistics. 6.1, 6.5

[182] Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China, November 2019. Association for Computational Linguistics. 2.1.2, 3.5

[183] Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado, May–June 2015. Association for Computational Linguistics. 2.1.2, 5.1, 5.2.1, 5.2.1

[184] Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado, May–June 2015. Association for Computational Linguistics. 4.4

[185] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 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, June 2021. Association for Computational Linguistics. 3.6

[186] Fan Yang, Jun Zhao, and Kang Liu. A Chinese-English organization name translation system using heuristic web mining and asymmetric alignment. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 387–395, Suntec, Singapore, August 2009. Association for Computational Linguistics. 6.1, 6.5

[187] Yinfei Yang, Gustavo Hernandez Abrego, Steve Yuan, Mandy Guo, Qinlan Shen, Daniel Cer, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. Improving multilingual sentence embedding using bi-directional dual encoder with additive margin softmax. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5370–5378. International Joint Conferences on Artificial Intelligence Organization, 7 2019. 2.1.2, 4.1

[188] Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China, November 2019. Association for Computational Linguistics. 3.2.2, 4.3.2

[189] Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. Unsupervised neural machine translation with weight sharing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 46–55, Melbourne, Australia, July 2018. Association for Computational Linguistics. 5.4

[190] Xiangkai Zeng, Sarthak Garg, Rajen Chatterjee, Udhyakumar Nallasamy, and Matthias Paulik. Empirical evaluation of active learning techniques for neural MT. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 84–93, Hong Kong, China, November 2019. Association for Computational Linguistics. 7.1, 7.3.1, 7.3.1, 7.3.1

[191] Jiajun Zhang and Chengqing Zong. Exploiting source-side monolingual data in neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545, Austin, Texas, November 2016. Association for Computational Linguistics. 5.4, 6.5

[192] Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. Earth mover's distance minimization for unsupervised bilingual lexicon induction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1934–1945, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. 3.5

[193] Min Zhang, Haizhou Li, Jian Su, and Hendra Setiawan. A phrase-based context-dependent joint probability model for named entity translation. In *Second International Joint Conference on Natural Language Processing: Full Papers*, 2005. 6.5

[194] Pei Zhang, Xueying Xu, and Deyi Xiong. Active learning for neural machine translation. In *2018 International Conference on Asian Language Processing (IALP)*, pages 153–158. IEEE, 2018. 7.3.1

[195] Yuan Zhang, Jason Baldridge, and Luheng He. PAWS: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 3.2.2, 3.3.2, 4.3.2

[196] Mengjie Zhao, Yi Zhu, Ehsan Shareghi, Ivan Vulić, Roi Reichart, Anna Korhonen, and Hinrich Schütze. A closer look at few-shot crosslingual transfer: The choice of shots matters. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5751–5767, Online, August 2021. Association for Computational Linguistics. 3.6

[197] Yang Zhao, Jiajun Zhang, Zhongjun He, Chengqing Zong, and Hua Wu. Addressing troublesome words in neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 391–400, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. 5.4

[198] Yang Zhao, Lu Xiang, Junnan Zhu, Jiajun Zhang, Yu Zhou, and Chengqing Zong. Knowledge graph enhanced neural machine translation via multi-task learning on sub-entity granularity. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4495–4505, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. 6.6

[199] Yang Zhao, Jiajun Zhang, Yu Zhou, and Chengqing Zong. Knowledge graphs enhanced neural machine translation. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth*

International Joint Conference on Artificial Intelligence, IJCAI-20, pages 4039–4045. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track. 6.6

[200] Shuyan Zhou, Xiangkai Zeng, Yingqi Zhou, Antonios Anastasopoulos, and Graham Neubig. Improving robustness of neural machine translation with multi-task learning. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 565–571, Florence, Italy, August 2019. Association for Computational Linguistics. 6.5

[201] Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. Overview of the second bucc shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, 2017. 3.2.2

[202] Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. Overview of the third bucc shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of 11th Workshop on Building and Using Comparable Corpora*, pages 39–42, 2018. 3.2.2