Breaking Language Barriers: Enhancing Multilingual Representation for Sentence Alignment and Translation

Zhuoyuan Mao

February 2024

Abstract

In a diverse linguistic landscape where over 7,100 languages are spoken, vast swathes of digital content remain isolated within language silos, creating significant barriers to global communication. Bridging these gaps is the purview of multilingual representation learning, an emerging field within natural language processing (NLP) that seeks to develop computational models capable of understanding and translating across multiple languages. This specialized area of research aims to dismantle linguistic barriers, facilitating the flow of information and ideas in our increasingly interconnected world.

This thesis delves into the intricacies of multilingual representation learning, concentrating on two pivotal tasks: multilingual sentence embedding (MSE) learning and multilingual neural machine translation (NMT). These tasks are key objectives of multilingual representation learning due to their profound impact on facilitating communication across language barriers. MSE learning enables the alignment of semantically similar sentences from different languages, serving as a key enabler for applications such as cross-lingual information retrieval and parallel corpus construction. Meanwhile, multilingual NMT extends the boundaries of language translation to a multilingual context, which is crucial for real-time interpretation and content localization. Ultimately, MSE learning and multilingual NMT encapsulate the primary objectives of multilingual representation learning, underpinning innovative applications that help dissolve language barriers, thus granting more equitable access to information and fostering cross-cultural understanding in a multilingual world.

Within multilingual representation learning, specifically for applications in alignment and translation tasks, three major challenges persist: (1) high computa-

tional demands, which refers to the significant computational overhead incurred in scaling up the language coverage of a multilingual model; (2) data scarcity, the lack of sufficient and diverse language data, particularly for low-resource languages; (3) limitations in Transformer architecture, meaning the current Transformer models are not fully appropriate for the complexities of processing multiple languages. Addressing these challenges is crucial for further advancement in this field. To this end, this thesis seeks to provide solutions to these existing challenges while also exploring potential approaches for enhancing recent large multilingual language models (LLMs). Eventually, we expect to pave the way for more advanced and efficient multilingual representation learning, thus broadening the reach of NLP techniques to a wider audience.

Specifically, to tackle the challenge of high computation demands associated with expanding the language support in training MSE models, we first introduce efficient and effective massively multilingual sentence embedding, using cross-lingual token-level reconstruction and sentence-level contrastive learning as training objectives. Compared with related studies, the proposed model can be efficiently trained using significantly fewer parallel sentences and GPU computation resources. Secondly, we introduce a novel distilled MSE model to streamline the inference process for MSE models. Precisely, we systematically explore learning language-agnostic sentence embeddings with lightweight models. We demonstrate that a thin-deep encoder can construct robust low-dimensional sentence embeddings for 109 languages. With our proposed distillation methods, we achieve further improvements by incorporating knowledge from a teacher model.

To tackle the challenge of data scarcity in low-resource languages, we first introduce innovative sequence-to-sequence pre-training objectives for low-resource NMT to leverage the linguistic knowledge to compensate for the lack of training data. The proposed methods employ phrase structure masking and reordering tasks. Secondly, we propose word-level contrastive learning to leverage statistical word alignments for low-resource multilingual NMT, without the requirement to use high-quality bilingual dictionaries. Additionally, we introduce contrastive alignment instructions to address the challenge of the lack of data in low-resource languages. AlignInstruct emphasizes cross-lingual supervision via a cross-lingual

discriminator built using statistical word alignments, which is empirically demonstrated superior to NMT instruction tuning baseline methods.

To tackle the challenge of limitations in Transformer architecture for zero-shot NMT, we first unveil a novel Transformer architecture that constructs universal interlingua representations atop Transformer encoder. This development significantly enhances the performance of zero-shot NMT than standard Transformer architectures. Moreover, we comprehensively explore the effects of layer normalization on zero-shot NMT. Our results demonstrate that post-layer normalization consistently outperforms pre-layer normalization for zero-shot NMT, regardless of the language tag and residual connection settings.

Acknowledgments

As my five years of graduate study at Kyoto University are coming to a close, I find myself at a loss for where to begin the acknowledgments. Over these five years, I delved deeper into NLP, going through cycles of literature review, brainstorming, experimentation, writing and revising papers, submissions, rebuttals, preparing camera-ready manuscripts, attending conferences, and now applying for the doctoral degree. While it might sound challenging, in reality, the continuous support and encouragement from my teachers, lab mates, and friends have been a cornerstone throughout this journey. Even in the loneliest moments, I relied on my introverted intuition, extroverted thinking, and introverted feeling to motivate myself to keep exploring and advancing. This period of long-term, systematic thinking and researching was more a spiritual journey than a hardship. It transitioned me from not knowing what I liked or could do to a clearer understanding of these aspects. Nevertheless, I still believe there is so much more to learn, and I am eager to use the action-oriented approach I have developed over these five years to explore more of reality and sparks of inspiration. Now, I would like to express my heartfelt gratitude to all those who supported and encouraged me on this journey:

Firstly, special thanks to my supervisors, Professor Sadao Kurohashi, Associate Professor Chenhui Chu, and former Associate Professor Fabien Cromières. Their professional guidance, patient tutoring, and selfless support have been the driving force in my research journey. They always provided encouragement and constructive advice when I confronted challenges, helping me to maintain the right direction and a positive attitude.

I also extend my gratitude to the other teachers, colleagues, and lab mates

in the lab, including Associate Professor Yugo Murawaki, Assistant Professor Fei Cheng, Assistant Professor Yin-Jou Huang, Dr. Hirokazu Kiyomaru, Terumi Kosugi, Dr. Tareq Alkhaldi, Dr. Qianying Liu, Haiyue Song, Takashi Kodama, Nobuhiro Ueda, Kazumasa Omura, Shuichiro Shimizu, Zhen Wan, Zhengdong Yang, Yahan Yu, Shunya Kato, Youyuan Lin, Yikun Sun, and Sirou Chen, for their endless help and support in both academic discussions and daily life.

Next, I am grateful to my external co-authors, Dr. Raj Dabre from NICT and Dr. Yibin Shen from Meituan, for their collaboration and constructive support in research. Additionally, I am thankful for the guidance and experiences shared by Dr. Prakhar Gupta and Associate Professor Martin Jaggi from EPFL, Dr. Qian Chen from SenseTime, Dr. Tetsuji Nakagawa from Google and the Google Translate team, and Dr. Yen Yu from Apple and the Siri APAC team, for their mentorship in my unfamiliar academic and professional environments and joint efforts in completing high-quality research and papers.

Furthermore, I would like to express my gratitude to Kyoto University and Japan Society for the Promotion of Science for their financial support of my research. Without their assistance, my research endeavors would have been challenging. I also thank my friends who share my love for anime, hiking, and skiing, for the ordinary joys of daily life.

Lastly, my deepest thanks go to my family, especially my parents, for their unconditional support and encouragement of my academic pursuits and dreams.

As this academic journey comes to a close, the experiences and lessons learned will continue to guide me as I embark on new chapters in life. My sincere thanks to everyone who has supported and helped me along the way, making this journey all the more meaningful and enriching.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Multilingual representation learning is a field within natural language processing (NLP) focusing on developing computational models that are adept at understanding and processing a variety of languages. This field strives to overcome language barriers, making diverse information accessible to people who speak different languages. In an era dominated by digital content, much of which is confined within linguistic silos, achieving this goal is not only an academic exercise but also a crucial step towards fostering a more inclusive and globally connected society.

At its core objectives, multilingual representation learning is concerned with developing algorithms that allow machines to align and translate diverse languages efficiently, minimizing the need for human efforts. This thesis hones in on two fundamental tasks within this domain: multilingual sentence embedding (MSE) learning and multilingual neural machine translation (NMT). MSE learning, which seeks to align sentences conveying similar meanings across languages, is critical in overcoming language barriers. For instance, a cross-lingual search feature, an application of MSE learning, enables users to search for information in one language and retrieve relevant results in another. Additionally, MSE is instrumental in constructing multi-way parallel sentences, which are invaluable not only for educational purposes but also for training translation models. In parallel, multilingual NMT plays an essential role in real-time interpretation and content localization scenes, the importance of which is profound and cannot be overstated in bridging linguistic divides.

These complex MSE learning and multilingual NMT tasks demand sophisticated deep-learning models capable of capturing the nuances of language beyond simple words, and this thesis ultimately endeavors to develop such neural models and propose novel methods to existing intricate challenges, paving the way for more robust multilingual representation learning for sentence alignment and translation.

This introduction begins with a detailed background of multilingual representation learning in the context of MSE learning and multilingual NMT tasks (Section 1.1). Following this, the ensuing sections meticulously dissect the inherent challenges and introduce the proposed innovative methods within this domain. Specifically, challenges of high computational demands, data scarcity, and limitations in Transformer architecture are discussed in Section 1.2. Subsequent chapters illuminate the pathways to overcoming these challenges in multilingual NLP.

1.1 Background of Multilingual Representation Learning

1.1.1 Two Fundamental Tasks and Advancements

This section begins with a comprehensive introduction to two fundamental tasks of multilingual representation learning: MSE learning and multilingual NMT, which involve acquiring the ability to align and translate sentences, respectively. It also provides an overview of the methodologies employed in training multilingual representations for these two tasks. Following this, it delves into the recent advancements in large language models (LLMs), posited as potential universal solutions for both tasks in multilingual NLP.

Multilingual Sentence Embedding Learning

MSE learning effectively aligns sentences from diverse languages within a shared semantic space, known as multilingual sentence embedding or language-agnostic sentence embedding. Such embedding space ensures that each sentence, regardless

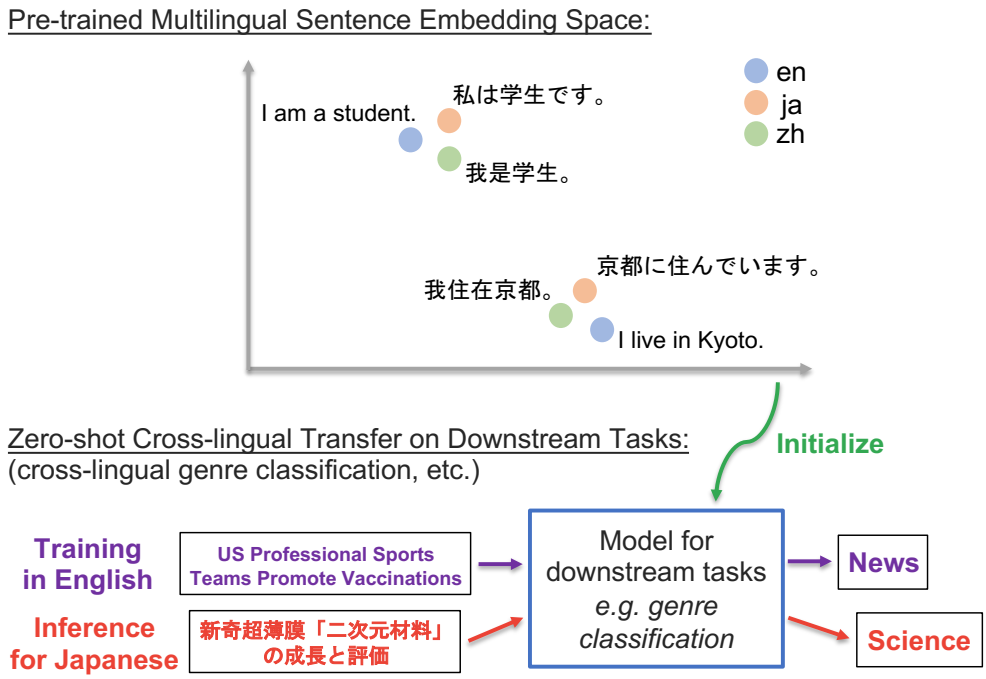

Figure 1.1: An example of a multilingual sentence embedding space and its applications in zero-shot cross-lingual transfer.

of its original language, is mapped into a common semantic domain. For example, the sentence "I am a student" in English should correspond to an equivalent MSE as its Japanese counterpart, "私は学生です", or its Chinese version, "我是学生", as illustrated in Figure 1.1.

With the constructed shared semantic space across languages, MSE models can be applied to various downstream NLP tasks, such as cross-lingual sentence retrieval and cross-lingual sentence classification. These tasks are facilitated without the prerequisite of initial training or reliance on a monolingual model. Significantly, MSE models offer substantial benefits to low-resource languages, which typically lack sufficient training data, thereby reducing the need for extensive human effort in curating large datasets for language-specific model training. For example, as depicted in Figure 1.1, once the pre-trained MSE model is fine-tuned using English datasets for genre classification tasks, this fine-tuned model can be directly applied for inference in Japanese. This is possible because the pre-trained

MSE model is designed to be language-agnostic. Section 1.1.2 delves deeper into the application paradigms of MSE models, specifically focusing on their implementation in various cross-lingual downstream tasks.

The pursuit of such dense text embedding has evolved significantly, beginning with the advent of word vector [128] and progressing to sentence embedding and multilingual scenarios. In the monolingual scenario, initial approaches, such as those by Arora et al. [9], advocated for the weighted average of word embeddings to create sentence embeddings, establishing a robust baseline. Subsequent efforts shifted towards leveraging neural models [39, 25] and pre-trained Transformers [172, 144, 250, 226] as backbone architectures.

In the realm of multilingual contexts, Schwenk and Douze [182] pioneered the concept of MSE, leveraging intermediate representations from LSTM [73] encoder-decoder frameworks in NMT. Concurrently, Grégoire and Langlais [63] devised MSE by aligning outputs from LSTM dual encoders (akin to Siamese networks [253]) into a unified representational space. Building on this, España-Bonet et al. [52] experimented with sum pooling of NMT encoder's top hidden states, diverging from the max pooling and last hidden state approach in Schwenk and Douze [182]. Yu et al. [255] introduced a training methodology for MSE that combines bidirectional NMT losses and minimizes the Euclidean distance between translation pair embeddings.

The transition to dual Transformer architectures replacing LSTM was initiated by Guo et al. [66], who first utilized Transformers for constructing MSE in bilingual settings. This was expanded by Chidambaram et al. [29], who incorporated multiple tasks such as conversational response [247], quick-thought [111], natural language inference [21], and translation into the training regimen. Building on these efforts, Yang et al. [244] further refined the training objectives by integrating an AMS loss, enhancing the approach proposed by Guo et al. [66]. Recent studies have shifted their focus towards massively multilingual scenarios, a direction that inherently brings with it the challenge of heightened computational demands. This specific challenge is explored in detail in Section 1.2, where we delve into its implications and potential proposals.

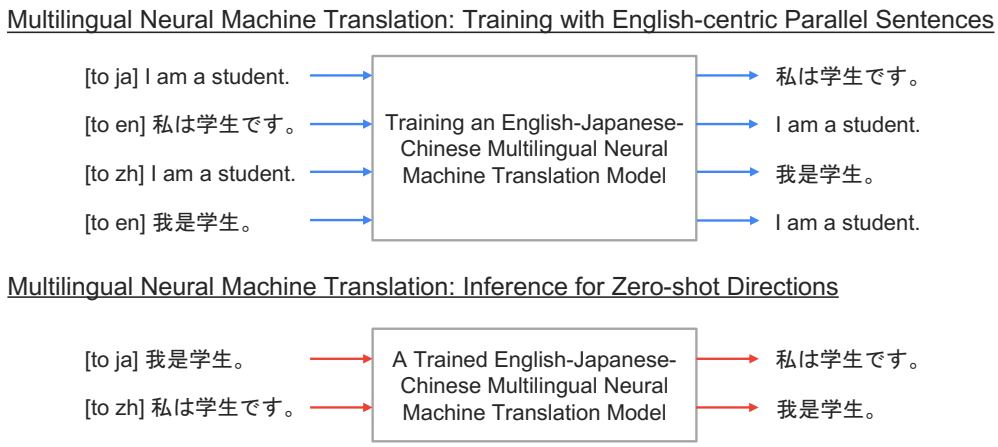

Figure 1.2: An example of English–Chinese–Japanese multilingual neural machine translation model trained with English-centric parallel data and its applications in zero-shot translation.

Multilingual Neural Machine Translation

Multilingual Neural Machine Translation (NMT) distinguishes itself by jointly training a system capable of both accepting input and generating output in multiple languages, a significant departure from traditional bilingual translation systems limited to a single language pair. This multilingual approach offers a more unified and efficient solution for language translation, covering a broader linguistic spectrum. For instance, a bilingual English-Japanese NMT system is restricted to translating the sentence "I am a student" only into Japanese as "私は学生です". In contrast, as depicted in Figure 1.2, a multilingual NMT system can translate the same sentence into several other languages the model supports, such as Chinese. In such multilingual translation scenarios, the introduction of language tags plays a crucial role in specifying desired source or target languages. As illustrated in Figure 1.2, the target language can be determined using a "to target language" token alongside the source sentence [81]. Another efficient approach to language tagging involves specifying both the source and target languages separately in their respective sentences, a method pioneered by mBART [110]. This thesis will adopt these two key language tag settings primarily in multilingual

machine translation, following the insights from previous work investigating the language tag settings [236].

A trained multilingual NMT model can be applied to various translation tasks, encompassing supervised and zero-shot directions. We introduce these applications in depth in Section 1.1.2, providing a comprehensive overview along with the downstream tasks of aforementioned MSE learning. As an instance of zero-shot translation, Figure 1.2 illustrates how a multilingual NMT model trained with English-centric parallel sentences can effectively handle zero-shot translation scenarios, such as translating from Chinese to Japanese and vice versa, showcasing the model's versatility in handling diverse linguistic tasks.

The genesis of machine translation can be traced back to the era of bilingual systems, where early models were designed to translate between two specific languages. These systems, often rule-based or statistical [136, 22, 219], laid the groundwork for understanding linguistic structures and translation patterns, albeit within a limited linguistic context. The evolution from these bilingual models to multilingual ones [55, 81, 3, 186, 8, 105, 149] marked a significant leap, expanding the horizon of MT to include multiple languages within a single model framework. This shift streamlined the translation process, facilitated a more comprehensive understanding of linguistic relationships across various languages, and notably improved translation quality for low-resource languages. This improvement is largely attributable to the cross-lingual knowledge transfer, leveraging shared scripts and syntactic structures across languages. Recently, the exploration of language relatedness has emerged as an important research topic in the field of multilingual NMT, as the performance of low-resource languages within a multilingual NMT system is increasingly recognized as being heavily dependent on the extent of cross-lingual transfer capabilities among related languages [142, 3, 8, 53, 42].

In parallel with the historical progression of NMT, the field has witnessed significant shifts in model architecture and methodology. Initially dominated by sequence-based models [208] like RNNs [176] and LSTM [73] networks, the advent of Transformer models revolutionized NMT [219], significantly enhancing translation accuracy and efficiency. Alongside this, multilingual NMT has gained prominence as a research focus, evidenced by various studies [48, 55, 67, 81, 43]. These

efforts seek to augment multilingual NMT performance across multiple language pairs, even as model capacity poses challenges in scaling up languages [3, 210, 261]. Research has predominantly concentrated on strategies like oversampling low-resource languages [8, 230, 213] and incorporating language-specific model components [192, 18, 156, 261, 233, 106, 246] to elevate translation quality for diverse language pairs. Moreover, the balance between shared and language-specific components in models is an area of ongoing exploration [20, 242, 258, 92], illustrating the dynamic evolution of NMT to cater to an expanding linguistic spectrum. However, two critical challenges persist in the realm of multilingual NMT: data scarcity in low-resource languages, particularly in zero-shot NMT scenarios, and limitations in Transformer architecture. These issues are explored in depth in Section 1.2, where we delve into their specifics and discuss potential approaches.

In the Era of LLMs

Recent developments in NLP have heralded a new era with the advent of LLMs, also known as generative AI [23, 31, 180, 216, 132, 145, 5, 217, 196]. These models signify a substantial advancement for multilingual NLP, offering the potential to simultaneously support MSE learning and machine translation tasks within a singular, unified model architecture and parameter set. This significant progress can be largely credited to the innovative paradigm of LLM prompting techniques, which has redefined the capabilities and efficiency in handling multilingual tasks in NLP.

In the case of MSE learning, LLMs excel in generating embeddings that capture the nuanced semantic representations of sentences across various languages. Jiang et al. [78] and Su et al. [205] have pioneered the study of constructing MSE using LLMs. On the other hand, downstream tasks that MSE models can be applied to can also potentially be tackled by prompting LLMs. For multilingual NMT, LLMs also demonstrated remarkable proficiency in accurately and contextually translating text between multiple languages [222, 31, 2, 259, 104, 257, 59, 152, 131, 264, 23, 180, 69, 126] but still suffer from the issue of poor performance for low-resource languages. This multi-functionality of LLMs, underpinned by their vast and diverse training data, positions them as a highly generalized

paradigm for multilingual NLP, offering previously unattainable solutions with traditional task-specific models.

Although the primary focus of this thesis remains on enhancing multilingual representation for task-specific MSE and multilingual NMT models, it is anticipated that LLMs will progressively supersede the current state-of-the-art (SOTA) models in the realm of multilingual NLP. Furthermore, the techniques proposed in this thesis hold the potential to significantly contribute to the advancement of multilingual LLMs, potentially influencing their development and efficacy in this rapidly evolving field.

1.1.2 Training and Inference Paradigms

This section provides an overview of the key training and inference paradigms for MSE learning and NMT, which are essential in assessing the performance of MSE and multilingual NMT models. The discussion commences with introducing standard paradigms typically employed in supervised settings, followed by introducing crucial and unique inference paradigms in zero-shot settings for MSE and multilingual NMT models.

Supervised Scenarios

The efficacy of MSE models is determined by how accurately sentences from different languages are aligned in the shared embedding space. The alignment accuracy, typically measured using benchmark datasets comprising parallel sentences, reflects the model's ability to capture semantic similarities across languages. Thus, supervised scenarios for the MSE model focus on trained language pairs in the context of parallel sentence alignment tasks, which usually aim to retrieve parallel sentences from comparable corpora.[1] Although the training objectives for MSE continued to evolve, even in unsupervised ways, the SOTA MSE models were still constructed by aligning parallel sentences. Goswami et al. [61] proposed an unsupervised multi-task learning approach for training MSE, eliminating the reliance on parallel sentences. Despite this innovation, their results still fell short

[1] Comparable corpora refer to sets of documents in different languages, where many sentences within them are translations of each other.

of SOTA massively multilingual supervised models like LASER [12], LaBSE [54] and LEALLA [123] in cross-lingual sentence retrieval tasks. Recently, research has been delving into constructing MSE using LLMs through sentence-level contrastive objectives using parallel sentences, with promising results observed using BLOOM models [180].

Conversely, supervised scenarios of multilingual NMT models is relatively straightforward compared to MSE models. This is because the alignment nature of MSE models supports a broader range of cross-lingual tasks, whereas multilingual NMT models are typically confined to translation tasks. The effectiveness of a multilingual NMT model is primarily measured by its ability to accurately translate text between languages within the trained pairs and domains [48, 55]. Despite the promising performance of unsupervised multilingual NMT across various languages [97], largely due to the utilization of back-translation techniques in an iterative manner for improvement [189], these models generally fall short of their supervised counterparts that are trained on extensive parallel data. The studies on multilingual NMT discussed in this thesis primarily focus on the supervised paradigm, with evaluations conducted on trained language pairs [48, 55].

Zero-shot Scenarios

Zero-shot scenarios are a critical measure of a model's capacity to generalize to scenarios beyond its explicit training objectives, which serves as a robust indicator of its ability to comprehend and accurately represent linguistic semantics outside its trained scope. It is particularly vital for multilingual models, as it demonstrates the extent to which capabilities acquired from training in high-resource languages or domains can be effectively applied to previously unseen languages or domains. Successfully achieving this transferability significantly reduces the necessity for extensive training data across all targeted scenarios, thereby highlighting the efficiency and adaptability of multilingual models across diverse languages and domains.

The zero-shot capability of MSE models is typically assessed on unseen language pairs within parallel sentence alignment tasks, extending to various cross-lingual downstream tasks that depend on cross-lingual sentence alignment. This

includes zero-shot cross-lingual classification tasks, where a linear layer is fine-tuned to adapt the MSE model to one language and then directly tested on another. Such an approach evaluates the model's proficiency in transferring knowledge across languages by leveraging the well-aligned embedding space of pre-trained MSE models. In terms of enhancing the generalizability of MSE models in zero-shot scenarios, substantial research has been conducted. mUSE [245] was a forerunner in this field, employing multi-task learning across multiple natural language understanding (NLU) tasks. Simultaneously, LASER [12] adopted an LSTM framework for training MSE models with translation objectives, demonstrating effective generalization in zero-shot settings. Following this research, Reimers and Gurevych [173] introduced SBERT-distill, showing that fine-tuning from a monolingual English sentence encoder can yield a robust MSE model in zero-shot settings. LaBSE [54] further indicated that language model pre-training combined with AMS loss [244] can significantly enhance MSE models' zero-shot capabilities. This thesis includes evaluations in zero-shot scenarios to assess the generalizability capabilities of trained MSE models, but with a particular focus on addressing efficiency issues as outlined in Section 1.2.

For multilingual NMT, zero-shot inference involves testing the model's proficiency in translating between unseen language directions. This inference paradigm is critical in understanding how well the model can leverage its learned linguistic knowledge to translate between languages for which it has not been explicitly trained. The accuracy and fluency of these translations are key indicators of the model's adaptability and potential to bridge linguistic divides in real-world applications. Zero-shot translation is also a critical problem, as obtaining sufficient training data for all translation directions is often impractical. A multilingual NMT model's zero-shot translation performance usually benefits from the encoder-side representations being language-independent and decoder-side representations being language-specific. To achieve this, some studies have proposed removing encoder-side residual connections [108] or introducing language-independent constraints [4, 155, 7, 246, 120]. Other methods involve decoder pre-training and back-translation [64, 261], denoising autoencoder objectives [228], and encoder-side neural interlingua representations [113, 220, 270]. This thesis

not only incorporates zero-shot translation to assess the effectiveness of multilingual models but also introduces novel model architectures aiming at enhancing zero-shot translation capabilities, as detailed in Section 1.2.

1.1.3 Connections across Sentence Alignment and Translation Tasks

In the intricate landscape of NLP, the interconnectedness between sentence alignment and translation tasks plays a pivotal role. This section aims to elucidate the symbiotic relationships and mutual influences these tasks exert on each other, highlighting how advancements in one task can significantly benefit the other.

Firstly, the role of MSE models in facilitating translation corpus (i.e., parallel corpus) construction is noteworthy. MSE models, designed to encode sentences into meaningful, dense vectors, are instrumental in identifying parallel sentences across languages. This capability is particularly beneficial in constructing corpora for MT, especially in scenarios involving low-resource languages or domains where traditional methods of corpus construction are challenging [184, 181]. By efficiently aligning sentences from bilingual or multilingual corpora, these models not only enhance the quality of the MT training data but also expand the potential for more accurate and diverse translations [42, 53].

Secondly, the assessment of multilingual NMT models often involves a close examination of the quality of encoder-side sentence embeddings [236, 108, 114, 120]. The encoder in an NMT model is responsible for comprehending and encoding the source language into intermediate representations that the decoder can translate. The effectiveness of this encoding process, which is essentially the quality of the sentence embeddings, directly influences the translation's accuracy. Thus, evaluating the encoder's ability to generate language-agnostic sentence embeddings can provide crucial insights into the overall performance and efficiency of the multilingual NMT model.

Thirdly, the application of multilingual NMT models is not limited to direct translation tasks; conversely, they play a crucial role in the construction of MSE models. Multilingual NMT models, trained across a spectrum of languages, possess a unique ability to encode diverse linguistic inputs into a univer-

sal representation space. By harnessing this capability, they can be effectively employed to develop MSE models, which can be further employed for tasks requiring cross-lingual semantic understanding, such as multilingual document classification or cross-lingual information retrieval. It is noteworthy that the LASER series[2] [12, 70] of MSE models have capitalized on this interconnectedness.

1.2 Challenges and Our Proposals

In the preceding sections, we introduced the backgrounds of multilingual representation learning for MSE and multilingual NMT, delving into its foundational concepts, methodologies, training and inference paradigms, and the synergistic relationship between sentence alignment and translation tasks. Building upon this foundation, this section transitions to address the key challenges encountered in multilingual representation learning. Specifically, we identify three primary challenges: high computational demands, data scarcity, and limitations in Transformer architecture. As illustrated in Figure 1.3, we propose innovative methods to these challenges in subsequent chapters, with each challenge being the focus of a distinct chapter. These approaches aim to pave the way for more robust and efficient multilingual representation learning.

High Computational Demands

One of the most significant challenges is the high computational demands associated with expanding language support, a hurdle that becomes increasingly prominent as the number of languages in a model grows. This thesis specifically concentrates on addressing this challenge within the context of MSE models. In the realm of training MSE models for massively multilingual scenarios, mUSE [245] pioneered this effort by training MSE for 16 languages, adopting the training methodology of Chidambaram et al. [29]. Concurrently, LASER [12] utilized an LSTM framework to train MSE for 93 languages, expanding upon Schwenk and Douze [182]. Following this, Reimers and Gurevych [173] introduced SBERT-distill, leveraging parallel sentences to distill multilingual capabilities from pre-

[2] `https://github.com/facebookresearch/LASER`

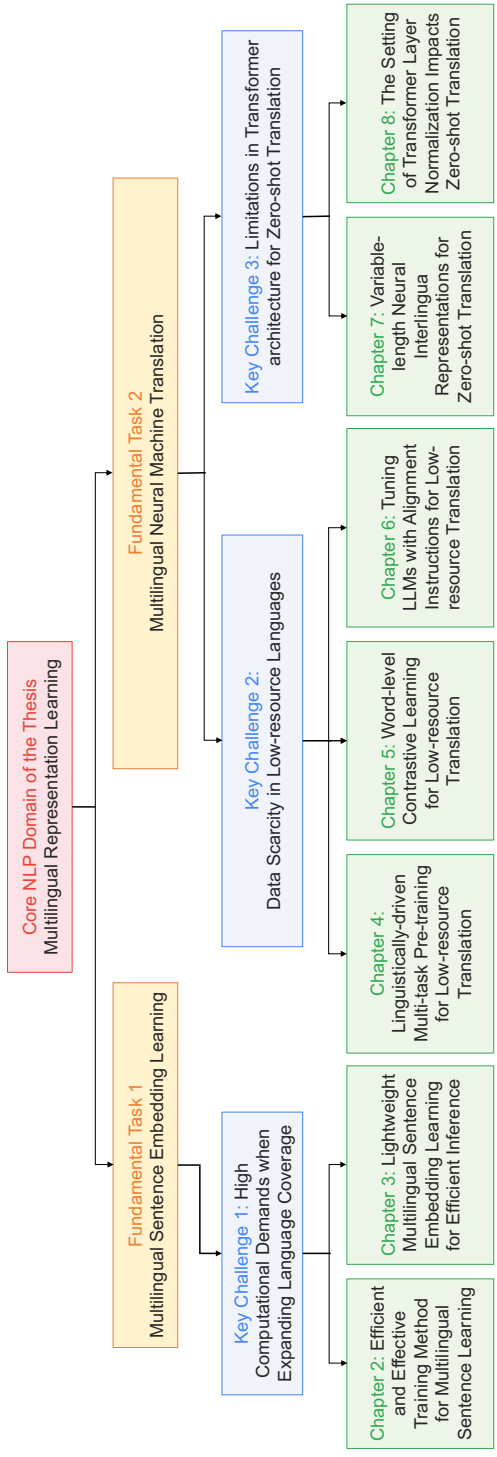

Figure 1.3: The main proposals of this thesis.

trained English sentence encoders for 50 languages. LaBSE [54] further advanced Yang et al. [244] by extending from bilingual scenarios to 109 languages using a pre-trained masked language model. However, the use of a large amount of data or inefficient model architectures results in heavy computation to train a new massively multilingual model according to our desired languages and domains. Concurrently, a model that supports a larger number of languages typically encompasses a greater number of parameters, leading to inefficient inference.

To tackle the challenge associated with expanding the language support in training MSE models, we present innovative efficient MSE training methods in Chapter 2. Specifically, we introduce efficient and effective massively multilingual sentence embedding (EMS), using cross-lingual token-level reconstruction (XTR) and sentence-level contrastive learning as training objectives. Compared with related studies, the proposed model can be efficiently trained using significantly fewer parallel sentences and GPU computation resources. To streamline the inference process for MSE models, we introduce a novel distilled MSE model, LEALLA, in Chapter 3. Specifically, we systematically explore learning language-agnostic sentence embeddings with lightweight models. We demonstrate that a thin-deep encoder can construct robust low-dimensional sentence embeddings for 109 languages. With our proposed distillation methods, we achieve further improvements by incorporating knowledge from a teacher model.

Data Scarcity

Data scarcity in low-resource languages also presents a critical obstacle, limiting the efficacy and accuracy of multilingual models in these languages. This thesis specifically focuses on this challenge in the context of multilingual NMT models. In addressing low-resource multilingual NMT, three main approaches have emerged: cross-lingual transfer, using data from different or multiple language pairs [273, 44, 48, 135]; data augmentation, notably back-translation to create synthetic bilingual data from monolingual sources [51, 72, 189, 268]; and monolingual pre-training, exemplified by the successes of GPT [166], BERT [46], and others [153, 206, 248], and tailored for NMT tasks by MASS [204] and mBERT [110], which jointly trains the encoder and decoder. However, the integration of mono-

lingual linguistic knowledge and cross-lingual alignment information, which are at finer granularities, remains underexplored. To address this gap, we propose innovative methods aimed at enhancing cross-lingual signals in low-resource multilingual NMT.

We first present innovative methods detailing the integration of linguistic knowledge to enhance multilingual representation in Chapter 4. Precisely, we propose novel sequence-to-sequence pre-training objectives for low-resource NMT: Japanese-specific sequence-to-sequence (JASS) for language pairs that involve Japanese as the source or target, and English-specific sequence-to-sequence (ENSS) for language pairs involving English. JASS focuses on masking and reordering Japanese linguistic units known as bunsetsu, whereas ENSS is proposed based on phrase structure masking and reordering tasks. Secondly, in Chapter 5, we delve into the utilization of cross-lingual word alignments to augment encoder-side multilingual representation, focusing on both from-scratch training and fine-tuning of multilingual NMT models. Specifically, we propose a word-level contrastive objective to leverage statistical word alignments for low-resource multilingual NMT, without the requirement to use high-quality bilingual dictionaries. Additionally, in Chapter 6, we extend our exploration to the application of word alignments in the fine-tuning of LLMs for low-resource multilingual NMT tasks. In particular, we introduce contrastive alignment instructions (AlignInstruct) to address the challenge of the lack of data in low-resource languages. AlignInstruct emphasizes cross-lingual supervision via a cross-lingual discriminator built using statistical word alignments, which is empirically demonstrated superior to NMT instruction tuning baseline methods.

Limitations in Transformer Architecture

Moreover, the issue of limitations in Transformer architecture for multilingual tasks is prevalent, given that the widely-used Transformer [219] architecture was originally designed for bilingual machine translation, specifically for English-French translation tasks. This underscores the need for more specialized and effective designs capable of addressing the complexities inherent in multiple languages. In this thesis, our focus is on investigating the optimal model architecture for zero-shot

NMT, a critical inference scenario in the realm of multilingual NMT models. Referring to the introduction of backgrounds about zero-shot NMT in Section 1.1.2, research focusing on exploring the optimal model architectures for zero-shot NMT almost remains blank with the exception that the setting of Transformer residual connections [108] was shown significantly to impact the performance of zero-shot NMT, and the fixed-length interlingua representation was shown beneficial for zero-shot NMT [270]. To this end, we delve into a closer look at the model architectures for zero-shot NMT in this thesis.

In Chapter 7, we unveil a novel Transformer architecture that constructs universal interlingua representations on top of Transformer encoder. This development significantly enhances the performance of zero-shot NMT. More precisely, we introduce a novel method to enhance neural interlingua representations by making their length variable, thereby overcoming the constraint of fixed-length neural interlingua representations introduced by previous work [270]. Moreover, in Chapter 8, we comprehensively explore the effects of layer normalization on zero-shot NMT. Our results demonstrate that post-layer normalization consistently outperforms pre-layer normalization for zero-shot NMT, regardless of the language tag and residual connection settings.

Chapter 2

EMS: Efficient and Effective Massively Multilingual Sentence Representation Learning

Cross-lingual sentence representation (CSR) models [182, 52, 255, 46, 29, 12, 177, 40, 244, 245, 173, 38, 122, 54, 123] prove to be essential for downstream NLP tasks like cross-lingual sentence retrieval and cross-lingual transfer without the need for initial training and monolingual model. Thus, CSR models benefit low-resource languages without sufficient training data.

A majority of the CSR training methods can be ascribed to one of the following two categories: *global fine-tuning* or *sentence embedding*. *global fine-tuning* methods indicate that for a specific downstream task, we conduct fine-tuning by updating pre-trained language models e.g., mBERT [46], XLM [40], and XLM-R [38]. The fine-tuning efficiency of this method group is determined by the scale of the pre-trained model. Thus, the update of the large-scale parameters of the pre-trained model tends to be the computation bottleneck for fine-tuning. The computationally lite *global fine-tuning* methods have been explored sufficiently either by compressing the model [98], training a student by knowl-

edge distillation [178, 207, 80]. On the other hand, *sentence embedding* methods, e.g., LASER [12], aim to train the CSR that aligns the embedding space across languages without further fine-tuning. For example, the English sentence "I am a student." should have an identical sentence embedding to its French translation, "Je suis un étudiant." As a result, this group of methods can be efficiently adapted to several cross-lingual downstream tasks by merely adding a multi-layer perceptron without the need for tuning parameters within the pre-trained CSR model. However, existing *massively multilingual sentence embedding (MSE)* models, LASER [12], SBERT-distill [173], and LaBSE [54], require a considerable amount of data or inefficient model architectures, for which the efficient training objectives have not been explored.

In this study, we present **E**fficient and effective massively **M**ultilingual **S**entence embedding (**EMS**), a computationally lite and effective architecture for training MSE, which ameliorates the data and computation efficiency to train an MSE model according to our preferred domains or language groups and may have a promising future for deploying MSE model training and adaptation on memory-limited devices. In particular, we propose cross-lingual token-level reconstruction (XTR) and sentence-level contrastive learning as training objectives. XTR captures the target token distribution information, whereas the contrastive objective serves to recognize translation pairs. We claim that these two objectives effectively construct the multilingual signals for learning MSE within the dual-encoder model architecture, which results in highly efficient model training. Compared with previous MSE models in the massively multilingual scenario, EMS can be trained using significantly fewer parallel data and less GPU consumption.

In contrast to our previous study [122], lightweight bilingual sentence representation learning, we focus on exploring how to train a model efficiently and effectively for a massively multilingual scenario in this work. To address this, we tailor the model capacity for a large number of languages and introduce a language embedding layer for the generative objective and a linear layer for the contrastive objective. Furthermore, our findings indicate that the combination of the XTR objective and the alignment-based sentence-level contrastive objective, as proposed in our previous study, is advantageous for massively multilingual training.

In contrast, the unified generative task (UGT) from our earlier work does not perform effectively in such a scenario. Notably, we discovered that the sentence-level contrastive objective consistently enhances performance in cross-lingual retrieval and classification tasks as jointly trained with XTR. This contrasts with our previous observation, where this objective was detrimental to classification tasks in bilingual sentence embedding models. In addition, regarding model performance, we validate the effectiveness of EMS with over 100 languages and more evaluation benchmarks in this study, along with the contribution of each model component via the ablation study.

Despite the small amount of training data and low-cost training, experimental results demonstrate that the proposed EMS learned a robustly aligned multilingual sentence embedding space. With regard to the Tatoeba [12] cross-lingual similarity benchmark, EMS significantly achieves better results than LASER and SBERT-distill and comparable results considering middle- and high-resource languages[1] compared with LaBSE. Based on the results on Flores [62, 42] cross-lingual similarity benchmark for non-English language pairs, we demonstrate that EMS is completely language-agnostic while LASER is an English-dependent model. Moreover, we evaluate the model performance for mining parallel sentences [199, 195] from larger comparable corpora, including ParaCrawl [17] and BUCC benchmarks [274, 275]. The experimental results show that EMS performs better than SBERT-distill and comparably with LASER. Furthermore, we evaluate the language-agnostic representation based on three classification tasks in a zero-shot manner, document genre classification based on MLDoc [183], and sentiment classification based on two Amazon review datasets [162, 84]. Empirical results show that EMS outperforms LASER and SBERT-distill on MLDoc and one of the Amazon review datasets and yields comparable performance with SBERT-distill and LaBSE on the other Amazon review dataset. In addition, upon integrating LaBSE's additive margin softmax (AMS) contrastive objective into the EMS framework, while maintaining identical training data and model architecture, we noted a decline in performance. This outcome suggests that the effectiveness of AMS's objective is heavily reliant on LaBSE's extensive batch

[1] languages for which we possess over 300k parallel sentences for training data.

size and training data. It also highlights the superior efficacy of the form of the contrastive objective proposed in our study and the complementary nature of the XTR generative objective.

The major contributions of this study are summarized as:

- The training architecture and objectives we developed were both efficient in terms of data and computation, and they achieved improved or competitive results in cross-lingual sentence retrieval and sentence classification tasks when compared to other MSE models.
- We identified effective forms of generative and contrastive objectives, and demonstrated that the proposed language embedding layers significantly enhance MSE performance in massively multilingual scenarios, marking a notable advancement from our previous study in bilingual settings.
- We revealed that incorporating temperature-based scaling and linear layers within the contrastive objective offers a more effective approach for MSE learning compared to the AMS-based contrastive objective used in LaBSE.
- We release the codes of the model training and the EMS model, which supports 62 languages.

2.1 Related Work

In this section, we revisit the literature on recent MSE models and training objectives for developing MSE.

2.1.1 Multilingual Sentence Embedding

The pursuit of dense text embeddings has evolved significantly, beginning with the advent of word vectors [128] and progressing to sentence embeddings. Initial approaches, such as those by Arora et al. [9], advocated for the weighted average of word embeddings to create sentence embeddings, establishing a robust baseline. Subsequent efforts shifted towards leveraging neural models [39, 25] and pre-trained Transformers [172, 144, 250, 226] as backbone architectures. Recent

studies have predominantly focused on refining the training objectives with contrastive loss [266, 60, 85, 241, 57, 28] and on the strategic use of various training datasets, often involving translation pairs [265]. More recently, the integration of large language models (LLMs) [78] and prompting methods [205] have further advanced the capabilities of sentence embedding models.

In the realm of multilingual contexts, Schwenk and Douze [182] pioneered the concept of MSE, leveraging intermediate representations from LSTM [73] encoder-decoder frameworks in neural machine translation (NMT). Concurrently, Grégoire and Langlais [63] devised MSE by aligning outputs from LSTM dual encoders (akin to Siamese networks [253]) into a unified representational space. Building on this, España-Bonet et al. [52] experimented with sum pooling of NMT encoder's top hidden states, diverging from the max pooling and last hidden state approach in Schwenk and Douze [182]. Yu et al. [255] introduced a training methodology for MSE that combines bidirectional NMT losses and minimizes the Euclidean distance between translation pair embeddings.

The transition to dual Transformer architectures replacing LSTM was initiated by Guo et al. [66], who first utilized Transformers for constructing MSE in bilingual settings. This was expanded by Chidambaram et al. [29], who incorporated multiple tasks such as conversational response [247], quick-thought [111], natural language inference [21], and translation into the training regimen. Building on these efforts, Yang et al. [244] further refined the training objectives by integrating an AMS loss, enhancing the approach proposed by Guo et al. [66].

Subsequently, research shifted towards massively multilingual contexts, aiming to develop universal sentence embedding models supporting a large number of languages, usually at least over 10 languages. mUSE [245] pioneered this effort by training MSE for 16 languages, adopting the training methodology of Chidambaram et al. [29]. Concurrently, LASER [12] utilized an LSTM framework to train MSE for 93 languages, expanding upon Schwenk and Douze [182]. Following this, Reimers and Gurevych [173] introduced SBERT-distill, leveraging parallel sentences to distill multilingual capabilities from pre-trained English sentence encoders for 50 languages. LaBSE [54] further advanced Yang et al. [244] by extending from bilingual scenarios to 109 languages using a pre-trained masked

language model.

Concurrently, the training objectives for MSE continued to evolve. Goswami et al. [61] proposed an unsupervised multi-task learning approach for training MSE, eliminating the reliance on parallel sentences. Despite this innovation, their results still fell short of massively multilingual supervised models like LASER and LaBSE in cross-lingual sentence retrieval tasks. mSimCSE [232] adapted the English monolingual SimCSE [57] to multilingual contexts, achieving performance comparable to LASER and marginally below LaBSE. LEALLA [123] introduced a method for distilling robust low-dimensional MSE from LaBSE using knowledge distillation. This technique could similarly be applied to distill efficient MSE from our EMS model. Recently, research has been delving into constructing MSE using LLMs through sentence-level contrastive objectives, with promising results observed using BLOOM models [180].

Recent research has also focused on incorporating word-level supervision in the training of MSE alongside traditional sentence-level contrastive objectives. Our previous work [122] introduced and validated the efficacy of a word-level XTR objective in bilingual settings, and this study extends that approach to a massively multilingual setting. Concurrently, Li et al. [102] developed a method for training MSE across 36 languages, introducing a representation translation learning task that utilizes contextualized token representations from one language to reconstruct their counterparts in another language. This method resonates with our focus on utilizing cross-lingual token-level signals to enhance MSE. Given the simultaneous development of these methods, a comparative analysis with their approach is reserved for future research.

In this study, we continue to focus on the exploration of effective objectives for training MSE in massively multilingual contexts. We introduce the token-level XTR and sentence-level contrastive objectives, ensuring enhanced training efficiency and effectiveness on downstream tasks. The subsequent subsection will detail the discussion of the training objectives for sentence embedding models.

2.1.2 Training Objectives for Sentence Embedding Learning

This section provides an in-depth survey of two training objective types usually used for constructing sentence embedding, followed by a comprehensive discussion on the current state of research regarding these objectives within the context of MSE.

Generative Objectives measure a generation probability of the token prediction, via training a language model, which primarily contributes to the performance of downstream tasks. BERT's masked language model (MLM) [46] and its variants [40, 174, 38] focused on optimizing the encoder-side token generation probability. Sequence-to-sequence learning used the encoder–decoder framework to train either a translation task [182, 52, 12] or a sentence reconstruction task [204, 167, 100] through optimizing the decoder-side token generation probability. Subsequently, sentence embedding could be constructed using the encoder-side output for both groups of generative objectives.

Contrastive Objectives aim to transform the representation space by adjusting the distance between the representations of tokens (or the sentences), which were initially used jointly with the generative objectives to improve sentence representation learning. Next sentence prediction (NSP) in BERT [46], token discrimination in ELECTRA [37], sentence discrimination in DeCLUTR [60], and hierarchical contrastive objective in HICTL [235] were the typical ones. Recent research, notably the SimCSE [57] study, has shown exceptional results by focusing solely on training with contrastive objectives.

Referring to the evolution of MSE training objectives discussed in Section 2.1.1, the contrastive objective has been widely adopted in MSE research, including our prior work [122]. Yet, in massively multilingual contexts, the optimal variant of the contrastive objective remains uncertain. This study evaluates the temperature-scaled contrastive objective inheriting our earlier work and contrasts it with the AMS loss used in LaBSE. While generative objectives have typically relied on translation tasks [12], these can be inefficient. Token-level generative tasks built upon a dual-encoder framework offer greater efficiency and have been less investigated. Therefore, we introduce the XTR objective into training for massive MSE, enhancing it with a suitable form of contrastive objective through

joint training, extending our previous research in the bilingual domain [122] to a massively multilingual scenario. The only other concurrent study employing a token-level generative objective is Li et al. [102], as mentioned in Section 2.1.1. Another efficient paradigm of the generative objective for MSE was by knowledge distillation introduced in SBERT-distill [173], which we treat as a baseline for comparison in this study.

2.2 Proposed Methods

We conduct massively MSE learning by employing the dual Transformer encoder as the backbone of the training framework. For the training objective, we propose a novel cross-lingual training method, which jointly optimizes generative and contrastive objectives. We introduce cross-lingual token-level reconstruction (XTR) as the generative objective and employ sentence-level self-supervised learning as the contrastive objective. The training framework and objectives that we propose are expected to learn a well-aligned representation space for multiple languages.

2.2.1 Architecture

We introduce the dual Transformer sharing parameters to encode parallel sentences along with several multi-layer perceptrons (MLP) to extract cross-lingual information and compute the generative and contrastive losses (Figure 2.1). We use parallel corpora as the training data. First, we build monolingual sentence representations $\mathbf{u}$ and $\mathbf{v}$ on top of a Transformer encoder. Two groups of the MLP are employed to construct two training objectives. After completing the model training, given a sentence in any language, we use the Transformer encoder to infer the language-agnostic sentence representation. We can implement cross-lingual downstream tasks in a zero-shot manner using $\mathbf{u}$ or $\mathbf{v}$, as they are representations independent of the specific language.

Specifically, as shown in Figure 2.1, assume that we have a parallel corpus $\mathbf{C}$ that includes multiple languages $\{l_1, l_2, ..., l_N\}$, and each sentence pair $S = (S_l, S_{l'})$ contains a sentence in language l and its translation in language l', where $l, l' \in \{l_1, l_2, ..., l_N\}$, as shown in the blue dashed box in Figure 2.1. We use the

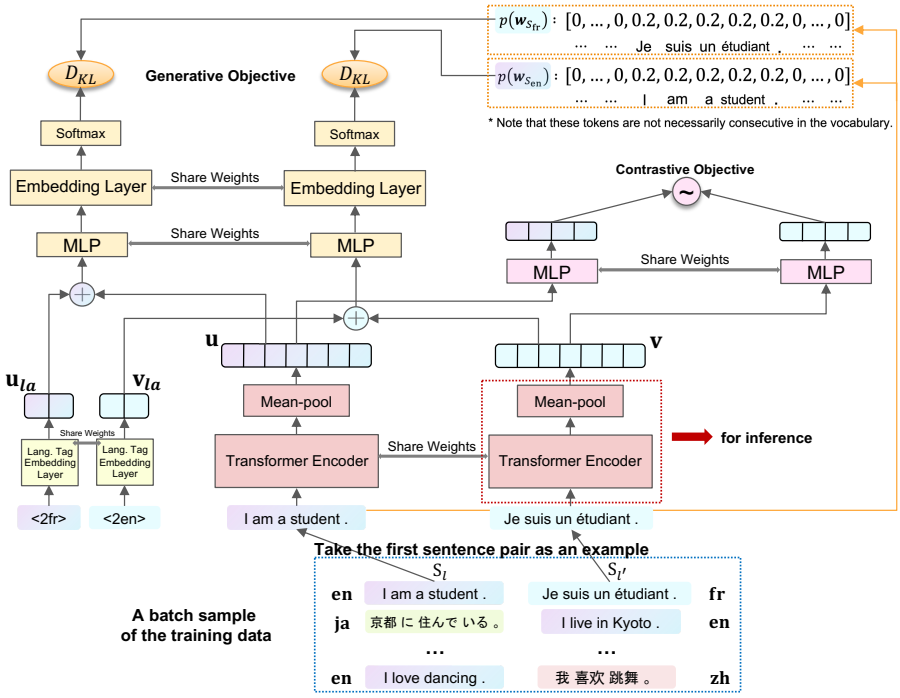

Figure 2.1: **Training architecture of EMS. u** and **v** are language-agnostic sentence representations for inference, and the model components in the red dashed rectangle are used for inference. $\mathbf{u}_{la}$ and $\mathbf{u}_{la}$ are the target language token representations. $\oplus$ denotes the hidden vector concatenation. A batch sample of the training data is given in the blue dashed box. Orange arrows and dashed box denote the gold token distributions within the generative objective. The part within the red dashed box indicates the pre-trained EMS model for downstream tasks.

dual Transformer encoder E sharing parameters to encode each sentence pair. Assume that the Transformer encoder outputs of S_l are $(\mathbf{h}_1^{\mathrm{T}}, \mathbf{h}_2^{\mathrm{T}}, ..., \mathbf{h}_{\|S_l\|}^{\mathrm{T}})$, where $\|S_l\|$ indicates the length of S_l. We use the mean-pooled hidden states as the language-agnostic sentence representation **u**:

$$\mathbf{u} = \frac{1}{\| S_l \|} \sum_i \mathbf{h}_i \tag{2.1}$$

Similarly, we can obtain **v** for $S_{l'}$.

2.2.2 Generative Objective

Generative objective plays an essential role in MSE learning. SBERT-distill and LaBSE use the pre-trained models as the model initialization; therefore, the pre-trained language models for each language serve as generative objectives. LASER finished the model training in one run without using any pre-training models, and the translation objective serves as a cross-lingual generative objective. Inspired by LASER, we include the generative objective for the one-run model training. However, the presence of the Transformer decoder in LASER increases the computational overhead. Instead, we propose a novel generative objective known as cross-lingual token-level reconstruction (XTR) to improve the training efficiency while retaining the quality of sentence representation, which circumvents using the Transformer decoder.[2]

As we expect the XTR objective to measure a cross-lingual reconstruction loss, it is necessary to notify the model what the target language is. Thus, we compute a target language representation for each sentence by employing a language embedding layer L_{la} to encode the target language token (e.g., $< 2en >$ if the target language is English). More precisely, for each sentence pair $S = (S_l, S_{l'})$,

$$\mathbf{u}_{la} = \mathbf{W}_{la}\mathbf{h}_{l'} \tag{2.2}$$

$$\mathbf{v}_{la} = \mathbf{W}_{la}\mathbf{h}_{l} \tag{2.3}$$

where $\mathbf{W}_{la} \in \mathbb{R}^{d_{la} \times d_{vcb}}$ denotes the parameters of L_{la}. $\mathbf{h}_l$ and $\mathbf{h}_{l'}$ respectively denote the one-hot embedding of $< 2l >$ and $< 2l' >$. d_{la} and d_{vcb} denote the dimension of the language embedding and the size of the vocabulary, respectively. The incorporation of language embeddings effectively clarifies the target of the XTR objective, particularly when transitioning from our previous bilingual-focused study to a massively multilingual scenario.

Subsequently, we concatenate the language representation with the sentence representation and use a fully connected layer L_{fc} to transform the concatenated representation for extracting the cross-lingual information. Finally, we use another

[2]It should be noted that both LASER and our model have the potential for further enhancement through language model pre-training. However, this aspect falls outside the scope of the current study and is left for future work.

linear embedding layer L_{emb} followed by Softmax to transform the representation to present two probability distributions, which are formulated as:

$$q_{S_l} = \text{softmax}(\mathbf{W}_{emb}\sigma_{xtr}(\mathbf{W}_{fc}(\mathbf{u}_{la} \oplus \mathbf{u}) + \mathbf{b}_{fc})) \tag{2.4}$$

$$q_{S_{l'}} = \text{softmax}(\mathbf{W}_{emb}\sigma_{xtr}(\mathbf{W}_{fc}(\mathbf{v}_{la} \oplus \mathbf{v}) + \mathbf{b}_{fc})) \tag{2.5}$$

where $\mathbf{W}_{emb} \in \mathbb{R}^{d_{vcb} \times (d_{la}+d)}$, $\mathbf{W}_{fc} \in \mathbb{R}^{(d_{la}+d)\times(d_{la}+d)}$, $\mathbf{b}_{fc} \in \mathbb{R}^{d_{la}+d}$, and d indicates the dimension of $\mathbf{u}$ (or $\mathbf{v}$). σ_{xtr} is the activation function in L_{fc}, for which we use swish [169]. $\oplus$ indicates the concatenation over the first dimension. In our previous study [122], we employed the identical parameters for $\mathbf{W}_{emb}$ as that in the Transformer encoder. We demonstrate in this work that use different parameters for $\mathbf{W}_{emb}$ would enhance further enhance the MSE in the massively multilingual scenario (see Section 2.4.7).

Assume that $\mathbf{B}_i$ is a batch sampled from the training corpus $\mathbf{C}$. Then, the training loss of the XTR objective for the $\mathbf{B}_i$ is formulated as follows:

$$\mathcal{L}^{(i)}_{XTR} = \sum_{S \in \mathbf{B}_i} \left(\mathcal{D}_{KL}\left(p_{S_{l'}}(\mathbb{W}) \parallel q_{S_l}\right) + \mathcal{D}_{KL}\left(p_{S_l}(\mathbb{W}) \parallel q_{S_{l'}}\right) \right) \tag{2.6}$$

where $\mathcal{D}_{KL}$ denotes KL-divergence and $\mathbb{W}$ indicates the vocabulary set. As illustrated in the orange dashed box in Figure 2.1, we use discrete uniform distribution for the tokens in S_l to define p_{S_l}. Specifically, for each $w \in \mathbb{W}$, $p_{S_l}(w)$ is defined as:

$$p_{S_l}(w) = \begin{cases} \dfrac{N_w}{\|S_l\|}, & w \in S_l \\ 0, & w \notin S_l \end{cases} \tag{2.7}$$

where N_w indicates the number of words w in sentence S_l, and N_w is 1 in most cases. $\|S_l\|$ indicates the length of S_l. In other words, $p_{S_l}(\mathbb{W})$ is approximately an average of one-hot embeddings of S_l's tokens. Similarly, we can obtain the definition of $p_{S_{l'}}(\mathbb{W})$.

Herein, we use the KL-divergence to measure the similarity between the token distribution of the sentence in the target language and the model output of the sentence in the source language and vice versa, which helps align the language-agnostic representation space. In Section 2.4.7, we will demonstrate that this objective also possesses good alignment abilities for non-English language pairs,

even when trained on English-centric data, thanks to the exposure to multiple languages during the training process.

Moreover, in our previous study [122], we introduced another generative objective known as the unified generative task (UGT) that combines XTR and single-word MLM [122]. We will provide empirical results and analyses to show that this objective is not relevant in the massively multilingual scenario and current model architecture (see Section 2.4.7).

2.2.3 Contrastive Objective

Based on our previous study [122], we employ a sentence-level contrastive objective as an assisting objective to force the model to grasp similar information of sentences across languages. We demonstrate that the sentence-level contrastive objective is a beneficial model component to jointly assist the generative objective. In Section 2.4.7, we provide empirical shreds of evidence that this objective plays a beneficial role in the generative objective introduced in Section 2.2.2.

Specifically, we employ in-batch sentence-level contrastive learning by discriminating between positive and negative samples for each sentence. Given a sentence, its translation (paired sentence in another language) is deemed as a positive sample, whereas other sentences within the batch are used as negative samples. Unlike our previous study, we employ temperature-based scaling and add two fully-connected layers to decrease the dimension of the sentence representation to compute the contrastive objective, following Chen et al. [27]. Assume that $\mathbf{B}_i$ is a batch sampled from the training corpus $\mathbf{C}$, and the j-th sentence pair of $\mathbf{B}_i$ is $S^{(ij)} = (S_l^{(ij)}, S_{l'}^{(ij)})$. Then the sentence-level contrastive objective for $\mathbf{B}_i$ is formulated as:

$$\begin{aligned}\mathcal{L}_{cntrs}^{(i)} = -\sum_{S^{(ij)} \in \mathbf{B}_i} \Big(& \log \frac{\exp\left(\text{sim}(S_l^{(ij)}, S_{l'}^{(ij)})/T\right)}{\sum_{S^{(ik)} \in \mathbf{B}_i} \exp\left(\text{sim}(S_l^{(ij)}, S_{l'}^{(ik)})/T\right)} \\ & + \log \frac{\exp\left(\text{sim}(S_l^{(ij)}, S_{l'}^{(ij)})/T\right)}{\sum_{S^{(ik)} \in \mathbf{B}_i} \exp\left(\text{sim}(S_l^{(ik)}, S_{l'}^{(ij)})/T\right)} \Big)\end{aligned} \tag{2.8}$$

where T denotes a temperature hyperparameter to scale the cosine similarity.

$\text{sim}(S_l, S_{l'})$ is defined as:

$$\text{sim}(S_l, S_{l'}) = \cos(\mathbf{h}(S_l), \mathbf{h}(S_{l'})) \tag{2.9}$$

$$\mathbf{h}(S_l) = \mathbf{W}_1\sigma_{cntrs}(\mathbf{W}_2\mathbf{u} + \mathbf{b}_2) + \mathbf{b}_1 \tag{2.10}$$

$$\mathbf{h}(S_{l'}) = \mathbf{W}_1\sigma_{cntrs}(\mathbf{W}_2\mathbf{v} + \mathbf{b}_2) + \mathbf{b}_1 \tag{2.11}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_{cntrs} \times d}$ and $\mathbf{W}_2 \in \mathbb{R}^{d \times d}$ mean the weights of two fully-connected layers, $\mathbf{b}_1 \in \mathbb{R}^{d_{cntrs}}$ and $\mathbf{b}_2 \in \mathbb{R}^d$ mean the biases of two fully-connected layers, and $d_{cntrs} < d$. According to Chen et al. [27], we use ReLU [137] for σ_{cntrs}. Our proposed objective diverges from LaBSE's AMS contrastive loss by omitting the additive margin and incorporating temperature-based scaling and linear layers prior to loss computation. We showcase the enhanced effectiveness of this approach for MSE learning in Section 2.4.

In Mao et al. [122], moreover, we introduced a sentence similarity-based contrastive task. We discard that objective in this study because we found that it has minimal impact on multilingual model training of EMS. This may be because it relies on high-dimensional sentence embeddings (e.g., 1,024) to determine similarities, while the sentence embedding dimension is reduced to a low-dimensional size in the current model architecture after adding two fully-connected layers.

2.2.4 Joint Training

We train the model by jointly optimizing the losses of the proposed generative and contrastive objectives. Specifically, we simultaneously train each batch with Eqs. (2.6) and (2.8):

$$\mathcal{L}^{(i)} = \frac{1}{\parallel \mathbf{B}_i \parallel}(\mathcal{L}^{(i)}_{XTR} + \mathcal{L}^{(i)}_{cntrs}) \tag{2.12}$$

where $\parallel \mathbf{B}_i \parallel$ denotes the number of sentence pairs within batch $\mathbf{B}_i$, namely, the batch size. Both $\mathcal{L}_{XTR}$ and $\mathcal{L}_{cntrs}$ play a dominant role in massively MSE training (details are given in Section 2.4.7).

2.3 Model Training

In this section, we introduce the parallel corpora that we used to train language-agnostic sentence representations and specific preprocessing and training details.

Model	af	ar	bg	bn	ca	cs	da	de	el	eo	es	et	eu	fa	fi	fr
LASER	67k	8.2M	4.9M	913k	**813k**	5.5M	7.9M	8.7M	6.5M	**397k**	**4.8M**	5.3M	1.2M	-	7.9M	8.8M
EMS (ours)	**50k**	**4.9M**	**2.8M**	**606k**	1.0M	**3.3M**	**4.3M**	**5.6M**	**3.9M**	683k	9.5M	**2.7M**	**818k**	5.1M	**4.2M**	**8.7M**
Model	gl	gu	he	hi	hr	hu	hy	id	it	ja	jv	ka	kk	ko	ku	lt
LASER	**349k**	-	4.1M	288k	4M	5.3M	**6k**	4.3M	8.3M	3.2M	-	296k	**4k**	**1.4M**	50k	3.2M
EMS (ours)	409k	0.3k	**2.7M**	**199k**	**2.3M**	**3.2M**	42k	**2.6M**	**6.1M**	**2.9M**	0.9k	**229k**	24k	1.9M	**0.3k**	**2.2M**
Model	lv	mk	ml	mn	mr	ms	my	nb	nl	pl	pt	ro	ru	sk	sl	sq
LASER	2M	4.2M	**373k**	-	**31k**	2.9M	**2k**	4.1M	8.4M	5.5M	8.3M	4.9M	9.3M	5.2M	5.2M	3.2M
EMS (ours)	**1.2M**	**2.4M**	402k	26k	126k	**1.9M**	3k	**46k**	**4.8M**	**3.2M**	**6.1M**	**3.0M**	**6.2M**	**2.8M**	**2.8M**	**2.1M**
Model	sr	sv	sw	ta	te	th	tl	tr	uk	ur	vi	yo	zh		Total	
LASER	4M	7.8M	173k	42k	33k	4.1M	**36k**	5.7M	**1.4M**	746k	4M	-	8.3M		204M	
EMS (ours)	**2.4M**	**4.2M**	**41k**	42k	**30k**	**2.2M**	45k	**3.8M**	1.5M	**50k**	**2.6M**	0.2k	**6.6M**		**143M**	

Table 2.1: **Number of parallel sentences in each language-used model for training. Bold** denotes fewer data used for training. Compared with LASER, we use 60% of the training data in total, and we use significantly fewer parallel sentences for 43 out of 61 language pairs. The total amount of the LASER training data is calculated in these 61 languages.

2.3.1 Training Data

We collected parallel corpora for 62 languages from OPUS[3] [214] (See Table 2.1).[4] The 62 languages that we selected cover all the languages in Schwenk et al. [181] and the languages of the cross-lingual generalization benchmark, XTREME [76]. While gathering each corpus, we used toolkits provided by Aulamo et al. [13][5] and Reimers and Gurevych [173].[6] Specifically, we used the following corpora for training:

Europarl is a parallel corpus extracted from the European Parliament website by Philipp Koehn [89]. We used the entire corpus for each language pair.

GlobalVoices is a parallel corpus of news stories from the website Global Voices compiled and provided by CASMACAT.[7] We used the entire corpus for each language pair.

NewsCommentary is a news commentary parallel corpus provided by WMT[8] for training statistical machine translation. We used the entire corpus for each

[3] `https://opus.nlpl.eu/`

[4] We do not distinguish between traditional and simplified Chinese.

[5] `https://github.com/Helsinki-NLP/OpusTools`

[6] `https://github.com/UKPLab/sentence-transformers`

[7] `http://casmacat.eu/corpus/global-voices.html`

[8] `https://statmt.org/`

language pair.

OpenSubtitles is a parallel corpus of movie subtitles collected from the website of `opensubtitles.org` [107]. Considering that the lengths of most sentences are short, we used at most 2M sentence pairs for each language pair to control the training data size.

Ted is a parallel corpus comprising TED talks. We used the 2020 version crawled by Reimers and Gurevych [173], which includes 4000 TED talks for each language pair available.

UNPC United nations parallel corpus of six languages [271]. We used 5M sentence pairs for en–ru and 2M sentence pairs for other language pairs.[9]

WikiMatrix is a parallel corpus crawled by Schwenk et al. [181]. We used the entire corpus for each language pair.

Tatoeba is a parallel corpus gathered from Tatoeba's website,[10] the language learning supporting website. As training on the Tatoeba benchmark will probably improve the evaluation performance on the Tatoeba benchmark [12], following Reimers and Gurevych [173], we excluded the training data of Tatoeba for most language pairs. Only for the language pairs that are not included in the aforementioned corpora, we used Tatoeba corpora.

The aforementioned training data leads to a 143M parallel corpus. As listed in Table 2.1, we used much fewer data for 43 languages than LASER. Moreover, we excluded the JW300 [1] corpus and pruned OpenSubtitles and UNPC corpora and included less training data than SBERT-distill.[11] In the next section (Section2.4), we will show that our model yields better or comparable sentence representation performance, compared with LASER and SBERT-distill. Considering our model's ability to deliver superior outcomes with reduced training data, it becomes feasible to extend our model to accommodate more low-resource languages, even with limited data availability.

[9] As the number of en–ru sentence pairs from other parallel corpora is relatively small, we used more for data for en–ru to balance the size for different language pairs.

[10] https://tatoeba.org/

[11] Reimers and Gurevych [173] used JW300 and all of the entire corpora we used.

Hyperparameters	Values
number of the Transformer layers	2, 4, **6**, 12
Transformer hidden dropout	0.0, **0.1**, 0.3
Transformer attention dropout	0.0, **0.1**
T	0.01, **0.1**, 0.2, 0.5, 1.0
learning rate	1e-4, **3e-4**, 5e-4, 1e-3
weight decay	0.0, **1e-5**, 1e-4, 1e-3
warm-up steps	0, 5,000, **10,000**, 20,000

Table 2.2: **Values of the hyperparameters tuned by grid search. Bold** denotes the best hyperparameter combination.

2.3.2 Preprocessing Details

For the parallel corpus containing 62 languages, we removed the sentences that appear in any evaluation dataset (see Section 2.4). We tokenized Chinese using jieba[12] and Japanese using Jumanpp[13] [130, 215], as the application of language-specific word segmentation for Japanese and Chinese has been shown to enhance performance across various tasks, including NMT [163, 119, 93, 118, 116] and MLM pre-training [40, 238]. We used Moses tokenizer for other languages.[14] We converted all the sentences to lowercase. Subsequently, we applied SentencePiece[15] [94] to convert words to subwords, which leads to a vocabulary with 60k tokens.[16] Finally, we add 62 language tokens (e.g., $< 2en >$, $< 2fr >$, ...) to the 60k vocabulary.

[12] `https://github.com/fxsjy/jieba`

[13] `https://github.com/ku-nlp/jumanpp`

[14] `https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl`

[15] `https://github.com/google/sentencepiece`

[16] SBERT-distill used a vocabulary of 250k tokens, which significantly improved the model parameters. In contrast, we used 60k, which is comparable with LASER's 50k vocabulary.

2.3.3 Training Details

We employed Transformer encoder [219] as the basic unit of the training architecture (Figure 2.1). We conducted a grid search for optimal hyperparameter combinations by observing the validation loss on the WikiMatrix validation datasets (Table 2.2).

As a result, the dual Transformer encoder sharing parameters has 6 layers, 16 attention heads, a hidden size of 1,024, and a feed-forward size of 4,096. The Transformer encoder can be substituted by encoders with other structures. d, d_{vcb}, d_{la}, and d_{cntrs} are 1,024, 60,000, 128, and 128, respectively. We set 0.1 for the temperature T of the contrastive objective.

For the model training, we fed the parallel sentences into the dual Transformer encoder and truncated the sentences up to 120 tokens.[17] We trained three epochs for the entire training corpus with the Adam optimizer [86], the learning rate of 0.0003 with the linear warm-up strategy of 10,000 steps, a weight decay of 0.00001, and a dropout[18] of 0.1 for the Transformer encoder. We used four V100 GPUs to conduct the model training with a batch size of 152 parallel sentences.

2.3.4 Efficiency Comparison with Competing Models

The superior efficiency of EMS stems from two key aspects: data efficiency and computation efficiency. In terms of data efficiency, as listed in Table 2.3, the proposed method includes 143M parallel sentences for model training, which is significantly less than those of other massive MSE models. Nevertheless, as demonstrated in Section 2.4, the reduced data for EMS results in a comparable or even improved model.

In terms of training efficiency, we employed the dual Transformer architecture as the basic model unit, whereas LASER required the encoder-decoder architecture to perform the translation task, where the presence of the decoder decreased

[17] Although LASER and SBERT-distill allowed much longer sentences during the training phase, we demonstrate that 120 tokens are sufficient for a single sentence with complete semantics. For the evaluation, documents longer than 120 tokens can be separated into several sentences, which would not limit the usage of our model.

[18] the hidden and attention dropouts

Model	#Langs	#Paral	Mono	Archit.	#Param.
LASER	93	223M		Seq2seq-LSTM	148M
SBERT-distill	50	>>223M	√	Dual-Trans	270M
LaBSE	109	6B	√	Dual-Trans	471M
EMS (ours)	62	143M		Dual-Trans	147M

Table 2.3: **Comparison between related studies and the proposed EMS.** "#Langs" and "#Paral" denote the number of languages the model supports and the number of parallel sentences used for training, respectively. "Mono" means whether the model incorporated monolingual data for training; "Archit." denotes the model architecture; "#Param." indicates the number of model parameters.

training efficiency. This indicates that the proposed model is an alternative to LASER, whereas SBERT-distill and LaBSE are complementary because the distillation from the English-SBERT and the use of the pre-trained model of LaBSE are feasible to be combined with the proposed training objectives. By leveraging a pre-trained model for initialization, EMS could potentially see further improvements. In this study, we do not use pre-trained models for comparison, as LaBSE's pre-trained model is not publicly accessible. Instead, we integrate LaBSE's AMS loss into our EMS framework, maintaining consistency in training data and model architecture, to determine the most efficient and effective training objective. For detailed configurations and results, please refer to Section 2.4.

With regard to the specific training time, the loss nearly converged after being trained for 0.5 epochs (122,196 steps) and converged completely after 3 epochs (733,176 steps), whereas LASER is trained for 17 epochs till convergence. Concerning the training time, SBERT-distill and LaBSE rely on large-scale pre-trained models; thus, their fine-tuning requires heavy computation as operating forwarding on large-scale models. LASER is trained with 80 V100 GPU×days, while our EMS requires 5 V100 GPU×days to nearly converge and 20 V100 GPU×days to converge fully, which indicates 4∼16 times speedup of EMS compared with LASER. In Section 2.4.7, we delve deeper into the training efficiency of each model component, demonstrating that the proposed generative and contrastive

objectives, along with the language embedding and linear layers for the contrastive objective, can be implemented efficiently without significantly increasing computational demands in a dual-encoder architecture.

In terms of inference efficiency, which can vary significantly depending on computational resources, we do not report the absolute time required for inference. As indicated in Table 2.3, our model has the fewest parameters, resulting in the quickest embedding inference time compared to LASER, SBERT-distill, and LaBSE. However, this does not lead to faster application in downstream tasks, as the embedding dimension remains at 1,024, identical to LASER and larger than 768 of both SBERT-distill and LaBSE. To enhance efficiency in downstream applications, the distillation technique from LEALLA [123] could be employed to create a lower-dimensional version of EMS with comparable performance.

In addition, other previous studies on learning language-agnostic sentence representation models [66, 29, 244], and mUSE [245], propose the training objective of distinguishing the positive translation from several *hard negative* samples. The heavy computation load of *hard negative* samples for each sentence limits the feasibility of their methods to a small number of languages, i.e., fewer than 16.

2.4 Evaluation

In this section, we evaluate the performance of the language-agnostic sentence representation on two groups of downstream tasks. On the one hand, without any further fine-tuning, we test the parallel sentence retrieval capability of the model using the cosine similarity between sentences. We evaluate this based on the following four tasks: Tatoeba benchmark [12], Flores benchmark [62, 42], BUCC benchmark [274, 275], and cross-lingual sentence retrieval on the ParaCrawl corpus [17]. On the other hand, by fine-tuning a simple multi-layer perceptron, we evaluate the model performance based on three cross-lingual sentence classification tasks in a zero-shot manner. Three evaluation tasks include the MLDoc benchmark [183] and cross-lingual sentiment classification on two versions of the multilingual Amazon review corpora [162, 84]. The former group of the evaluation measures the alignment performance of the language-agnostic sentence representa-

tion space, whereas the latter group evaluates the fundamental natural language classification ability of the model. More complicated cross-lingual natural language understanding (XNLU) tasks, e.g., XNLI [41] and XQuAD [38], have been comprehensively proven to perform better with cross-lingual language model pre-training and fine-tuning in XTREME [76], while *fixed representation* models are not competent to address such tasks [12]. Thus, we do not include the evaluation of XNLU in this study. Furthermore, we analyze the effectiveness of each component of the model structure based on an ablation study.

For all the evaluation tasks, we compare the following massively multilingual sentence representation models:

LASER [12] employed the BiLSTM encoder-decoder to train MSE for 93 languages by optimizing the translation task. 223M parallel sentences are used for training.

SBERT-distill [173] trained MSE for 50 languages by distilling the monolingual pre-trained encoder. Our training data are a subset of their data (Section 2.3.1). "paraphrase-xlm-r-multilingual-v1" is used for evaluation.[19]

LaBSE [54] trained MSE for 109 languages by fine-tuning the sentence-level contrastive task from mBERT. We *italicize* this model in the following tables (results) as the upper bound performance on downstream tasks because a large number of parallel sentences, 6B, are used for training. LaBSE continues to be the leading state-of-the-art model for parallel sentence retrieval in a massively multilingual scenario.

EMS (ours) We trained an MSE model for 62 languages. We used significantly less training data, thus less computation overhead, than those used in the previous study. The proposed model can be easily trained from scratch with competitive MSE performance.

LaBSE-EMS-vanilla In our EMS model architecture, we implement the AMS loss of LaBSE for learning MSE, a variant of the contrastive loss originally introduced in Yang et al. [244]. We substitute the standard contrastive loss in EMS with AMS, setting the margin to 0.3 without temperature-based scaling and linear

[19] `https://github.com/UKPLab/sentence-transformers/tree/master/examples/training/paraphrases`

layers upon sentence embedding.

LaBSE-EMS-scaled In this setting, based on LaBSE-EMS-vanilla, AMS is further applied with temperature-based scaling and linear layers on top of the sentence embeddings, as introduced in Section 2.2.3.

LaBSE-EMS-joint (ours) This setting evaluates the compatibility of the scaled AMS contrastive loss with our XTR generative loss within the EMS framework, combining LaBSE-EMS-scaled and XTR.

LaBSE-EMS-vanilla, LaBSE-EMS-scaled, and LaBSE-EMS-joint (ours) are for identifying the most effective contrastive loss for MSE learning and to assess the compatibility of LaBSE's AMS with EMS's XTR objectives. We also include the results of **LASER2** [70] on Tatoeba[20], **mUSE** [245] on Tatoeba and BUCC, and **LaBSE-bilingual** [244] on BUCC benchmarks for comparison.[21] LASER2 is a more recent version of the LASER model, incorporating SentencePiece [94] instead of BPE tokenization [190], trained with Transformer architecture, and can yield improved results for low-resource languages. On the other hand, mUSE is a universal sentence encoder model that is only compatible with 16 languages. In this study, we do not include LASER3-related models [70, 209] in our analysis, as their focus on adapting existing models to low-resource languages by language-specific parameters diverges from our objective of identifying efficient and effective objectives for one-for-all models.

2.4.1 Tatoeba Similarity Search

We use Tatoeba benchmark [12] to evaluate the cross-lingual alignment between English and other 58 languages.[22] Specifically, given a sentence in language l_1, we retrieve its translation from several sentences in language l_2. We use cosine similarity for retrieving sentences and report the average P@1 of $l_1 \rightarrow l_2$ and $l_2 \rightarrow l_1$ because both directions show similar precision considering a language pair.

[20] As LASER2 only outperforms LASER in a handful of low-resource languages (See Section 2.4.1), we solely present the findings on Tatoeba.

[21] mUSE serves as a crucial baseline on Tatoeba and BUCC for high-resource languages that have been utilized in previous studies [173, 54, 122].

[22] my, yo, and gu are not included in the Tatoeba benchmark.

Model	afr	ara	ben	bul	cat	ces	cmn	dan	deu	ell	epo	est	eus	fin	fra
LASER	89.5	92.0	**89.6**	95.0	95.9	96.5	95.4	96.0	99.0	95.0	97.2	96.7	94.6	96.3	95.6
LASER2	85.5	85.8	87.2	90.3	92.4	93.1	69.2	90.3	93.7	93.2	90.5	93.2	85.2	89.3	92.0
SBERT-distill	84.5	87.7	77.6	94.0	96.4	96.3	95.0	96.2	98.7	95.5	68.8	95.8	48.6	96.4	94.7
LaBSE-EMS-vanilla	27.1	25.5	18.1	45.7	32.1	38.8	31.5	41.9	54.2	41.1	37.8	32.8	21.1	33.9	40.1
LaBSE-EMS-scaled	90.3	89.7	88.0	91.4	96.0	96.3	94.3	95.1	99.4	95.4	98.0	96.6	93.6	96.1	95.6
LaBSE-EMS-joint (ours)	93.1	93.0	89.5	95.6	**97.2**	**97.5**	95.4	96.3	99.1	**96.5**	98.6	97.7	**95.1**	97.1	96.3
EMS (ours)	**94.0**	**93.9**	83.8	**95.8**	97.0	97.4	**95.9**	**97.0**	**99.3**	**96.5**	**98.9**	**97.8**	94.9	**98.0**	96.2
LaBSE	*97.4*	*91.0*	*91.3*	*95.7*	*96.5*	*97.5*	*96.2*	*96.4*	*99.4*	*96.6*	*98.4*	*97.7*	*95.8*	*97.0*	*96.0*

Model	glg	heb	hin	hrv	hun	hye	ind	ita	jav	jpn	kat	kaz	kor	kur	lit
LASER	95.5	92.2	94.7	97.2	96.0	36.1	94.5	95.3	22.9	90.7	35.9	18.6	88.9	17.2	96.2
LASER2	88.9	84.6	88.3	94.1	90.5	81.8	88.1	92.3	17.1	88.2	69.6	38.4	77.7	12.1	92.9
SBERT-distill	96.0	88.4	**96.4**	97.0	94.7	**91.3**	94.1	94.9	37.3	94.2	**91.4**	**73.7**	90.1	**43.7**	95.8
LaBSE-EMS-vanilla	30.2	30.2	19.5	39.1	32.3	10.6	40.5	44.9	11.5	25.1	8.1	8.3	17.4	11.2	31.8
LaBSE-EMS-scaled	95.6	89.6	92.5	96.8	96.6	81.2	94.4	93.5	43.9	93.7	64.1	55.7	88.4	22.3	96.1
LaBSE-EMS-joint (ours)	96.9	92.2	95.2	**97.5**	97.6	85.0	95.8	96.3	49.0	95.3	70.1	60.3	90.8	27.4	96.8
EMS (ours)	**97.1**	**92.5**	93.4	**97.5**	97.4	87.8	**95.8**	**96.7**	**55.6**	**95.8**	73.5	63.8	**92.3**	31.3	**97.3**
LaBSE	*97.2*	*93.0*	*97.7*	*97.8*	*97.2*	*95.0*	*95.3*	*94.6*	*84.4*	*96.4*	*95.9*	*90.5*	*93.5*	*87.1*	*97.3*

Model	lvs	mal	mar	max	mkd	mon	nld	nob	pes	pol	por	ron	rus	slk	slv
LASER	95.4	96.9	91.5	50.9	94.7	8.2	96.3	**98.8**	93.4	97.8	95.2	97.4	94.6	96.6	95.9
LASER2	92.2	95.1	89.5	30.3	89.2	2.8	92.4	86.9	84.1	91.2	92.4	93.0	91.2	93.9	91.1
SBERT-distill	96.4	94.0	91.0	58.5	92.2	**91.7**	96.0	98.0	94.8	97.0	94.8	96.4	93.5	96.2	95.5
LaBSE-EMS-vanilla	31.5	17.6	17.8	14.1	30.8	8.1	42.6	32.5	26.3	34.6	49.2	39.3	41.7	38.3	35.4
LaBSE-EMS-scaled	94.9	95.7	90.8	60.4	93.6	68.4	96.5	95.5	94.7	96.7	94.5	96.8	94.7	96.7	95.3
LaBSE-EMS-joint (ours)	96.7	**97.1**	**94.3**	66.2	96.4	72.0	**97.7**	97.4	95.6	98.1	**96.1**	**98.0**	**95.6**	**97.7**	96.7
EMS (ours)	**96.9**	77.8	88.2	**69.9**	**97.0**	73.9	**97.7**	97.5	**96.0**	**98.2**	95.9	97.9	95.2	97.5	**97.1**
LaBSE	*96.8*	*98.9*	*94.8*	*71.1*	*94.8*	*96.6*	*97.2*	*98.9*	*96.0*	*97.8*	*95.6*	*97.8*	*95.3*	*97.3*	*96.7*

Model	spa	sqi	srp	swe	swh	tam	tel	tgl	tha	tur	ukr	urd	vie	Avg.
LASER	98.0	98.0	95.3	96.6	**57.6**	69.4	79.7	50.6	95.4	97.5	94.5	81.9	96.8	84.7
LASER2	93.4	94.9	89.5	92.1	44.4	77.9	**93.6**	50.1	92.1	95.3	91.5	71.9	89.9	81.5
SBERT-distill	98.0	97.5	93.8	95.7	27.6	**85.7**	89.1	32.4	96.3	97.2	94.3	**92.2**	97.2	87.7
LaBSE-EMS-vanilla	47.2	38.3	33.9	36.5	4.1	9.4	7.5	8.9	32.9	36.4	37.4	9.9	35.8	29.0
LaBSE-EMS-scaled	98.5	97.9	95.8	95.4	37.1	69.7	64.1	76.6	95.0	97.8	94.7	76.3	96.5	87.6
LaBSE-EMS-joint (ours)	**98.7**	**98.4**	96.0	96.5	45.9	74.6	74.8	82.1	96.8	**99.0**	**96.0**	84.2	97.5	**90.0**
EMS (ours)	98.6	**98.4**	**96.4**	**97.0**	53.2	52.8	69.7	**84.8**	**97.4**	98.6	**96.0**	86.0	**97.7**	89.8
LaBSE	*98.4*	*97.6*	*96.2*	*96.5*	*88.6*	*90.7*	*98.3*	*97.4*	*97.1*	*98.4*	*95.2*	*95.3*	*97.8*	*95.3*

Table 2.4: **P@1 results on Tatoeba benchmark. Bold** fonts denote the best precisions among all the models except LaBSE. We report the average precision of the English→X and X→English for each language.

As shown in Table 2.4, in most languages, EMS achieves better retrieval precision than LASER, LASER2, SBERT-distill, LaBSE-EMS-vanilla, LaBSE-EMS-scaled, and performs comparably with LaBSE-EMS-joint. Observing the average score, 89.8, significantly outperforms LASER's 84.7, LASER2's 81.5, LaBSE-EMS-vanilla's 29.0, and is slightly higher than SBERT-distill and LaBSE-EMS-scaled.

Model	mUSE (15)	XTREME (38)	SBERT-distill (48)	<LASER (43)	>300k (42)	<300k (11)
mUSE	93.9	-	-	-	-	-
LASER	95.1	84.2	-	89.6	94.4	58.3
LASER2	89.0	80.6	-	85.0	88.9	66.4
SBERT-distill	94.9	85.5	94.8	-	92.1	**73.3**
LaBSE-EMS-vanilla	37.4	27.3	33.0	31.6	34.8	12.0
LaBSE-EMS-scaled	94.9	86.0	93.0	89.1	94.2	67.7
LaBSE-EMS-joint (ours)	96.3	**88.8**	94.8	91.4	**95.7**	73.0
EMS (ours)	**96.6**	88.2	**95.0**	**91.8**	95.4	72.0
LaBSE	*96.2*	*94.7*	-	-	*95.8*	*93.9*

Table 2.5: **Average P@1 results of different groups of the languages on Tatoeba benchmark. Bold** are the best precisions among all the models except LaBSE. "mUSE," "XTREME," and "SBERT-distill" denote the 15, 38, and 48 languages that the respective model or benchmark includes. "<LASER" denotes the 43 languages that use less training data than LASER. ">300k" and "<300k" indicate that LASER, LASER2, and EMS (the proposed model) include more than or less than 300k parallel sentences for training. Refer to Table 2.1; ">300k" and "<300k" contain 42 and 11 languages, respectively.

We further summarize the results of Table 2.4 in Table 2.5. First, with regard to 15 main languages that mUSE [245] supports, our model achieves the best retrieval prevision, even better than LaBSE, which leveraged 6B training data and used a large batch size of 4,096 sentences. Second, with regard to 38 languages that XTREME [76] supports, 48 languages that SBERT-distill supports, 43 languages for which we use less training data than LASER, and 42 languages for which all the models used training data over 300k, EMS consistently obtains higher retrieval precision than LASER, LASER2, SBERT-distill, LaBSE-EMS-vanilla, LaBSE-EMS-scaled, and performs on par with LaBSE-EMS-joint.

In addition, we observe similar results as compared with LaBSE for languages in which we used over 300k parallel sentences. This highlights the proposed model's efficiency in terms of data usage and computational resources for middle- and high-resource languages. Finally, with regard to 11 low-resource languages for which less than 300k training data are used in LASER, LASER2, LaBSE-EMS-vanilla, LaBSE-EMS-scaled, LaBSE-EMS-joint and EMS, EMS significantly outperforms than LASER, LASER2, LaBSE-EMS-vanilla, and LaBSE-EMS-scaled,

Model	amh	ang	arq	arz	ast	awa	aze	bel	ber	bos	bre	cbk	ceb	cha
LASER	42.0	37.7	39.5	68.9	86.2	36.1	66.0	69.6	68.2	96.5	15.8	77.0	15.7	29.2
LASER2	**69.4**	14.2	22.5	53.7	68.9	24.7	63.4	56.4	**72.0**	95.1	**23.4**	57.8	6.6	12.0
SBERT-distill	67.9	25.0	30.6	63.7	78.3	46.5	**85.0**	**86.9**	6.8	95.8	10.1	69.4	11.7	25.9
LaBSE-EMS-vanilla	1.5	17.9	4.7	13.9	36.6	5.6	8.2	14.5	1.9	46.8	3.1	11.7	6.4	13.9
LaBSE-EMS-scaled	2.1	32.5	33.9	66.4	82.7	48.3	49.1	51.5	4.8	95.9	6.5	69.8	22.6	29.9
LaBSE-EMS-joint (ours)	2.4	43.7	46.0	76.6	**89.0**	56.5	58.8	66.0	7.3	**96.6**	11.3	**80.6**	27.1	40.9
EMS (ours)	0.6	**47.4**	**48.7**	**77.7**	88.2	56.1	62.1	70.3	7.6	**96.6**	12.0	**80.6**	**30.1**	**46.4**
LaBSE	*94.0*	*64.2*	*46.2*	*78.4*	*90.6*	*73.2*	*96.1*	*96.2*	*10.4*	*96.2*	*17.3*	*82.5*	*70.9*	*39.8*

Model	cor	csb	cym	dsb	dtp	fao	fry	gla	gle	gsw	hsb	ido	ile	ina
LASER	7.5	43.3	8.6	48.0	7.2	71.6	51.7	3.7	5.2	44.4	54.5	83.7	86.2	**95.2**
LASER2	4.9	23.9	5.9	37.2	4.0	49.1	34.4	2.1	3.8	28.6	38.6	66.4	82.0	85.1
SBERT-distill	5.1	40.5	**34.9**	51.9	7.3	**50.8**	58.4	**7.5**	**18.6**	36.8	57.6	56.0	70.5	87.9
LaBSE-EMS-vanilla	2.1	14.8	3.0	12.8	2.0	9.4	17.3	1.3	2.5	20.1	15.2	16.1	23.9	26.6
LaBSE-EMS-scaled	4.5	50.8	9.5	50.3	6.2	30.3	55.8	3.4	4.2	39.3	58.5	77.5	76.6	88.2
LaBSE-EMS-joint (ours)	6.5	64.2	13.7	65.0	8.2	39.1	**65.6**	6.0	7.8	50.0	73.0	84.5	82.3	93.4
EMS (ours)	**7.8**	**69.2**	16.3	**69.7**	**9.5**	47.3	63.9	6.8	7.8	**54.7**	**79.0**	**88.1**	**86.4**	94.0
LaBSE	*12.8*	*56.1*	*93.6*	*69.3*	*13.3*	*90.6*	*89.9*	*88.8*	*95.0*	*52.1*	*71.2*	*90.9*	*87.1*	*95.8*

Model	isl	kab	khm	kzj	lat	lfn	mhr	nds	nno	nov	oci	orv	pam	pms
LASER	**95.6**	58.1	20.6	7.2	**58.5**	64.5	10.4	**82.9**	88.3	66.0	61.2	28.1	6.0	49.6
LASER2	90.8	**60.8**	**65.4**	2.9	46.5	44.5	5.4	64.7	55.5	53.7	45.4	19.2	2.8	28.7
SBERT-distill	75.8	2.7	64.8	8.0	28.0	57.7	11.9	50.7	**89.3**	58.8	52.4	33.4	7.0	44.3
LaBSE-EMS-vanilla	2.7	2.2	1.0	2.2	5.9	16.8	2.6	13.7	16.2	22.6	9.5	5.0	1.5	10.7
LaBSE-EMS-scaled	11.7	3.6	1.6	6.2	28.8	57.6	9.2	60.4	75.0	66.9	54.2	36.7	6.9	48.4
LaBSE-EMS-joint (ours)	17.9	4.2	1.1	9.2	43.6	68.1	14.2	70.7	81.5	74.3	64.6	47.1	12.7	60.0
EMS (ours)	22.2	4.7	1.2	**10.4**	46.4	**72.1**	**14.5**	73.7	84.9	**75.7**	**67.4**	**50.1**	**14.2**	**67.7**
LaBSE	*96.2*	*6.2*	*83.2*	*14.2*	*82.0*	*71.2*	*19.2*	*81.2*	*95.9*	*78.2*	*69.9*	*46.8*	*13.6*	*67.0*

Model	swg	tat	tuk	tzl	uig	uzb	war	wuu	xho	yid	yue	zsm	Avg.
LASER	46.0	**31.1**	20.7	44.7	45.2	18.7	13.6	**87.7**	8.5	5.7	**90.0**	96.4	**47.5**
LASER2	29.9	20.2	14.8	40.9	38.2	15.7	5.4	52.4	3.9	3.4	64.9	89.0	38.3
SBERT-distill	33.9	17.8	24.1	41.3	**65.5**	**32.6**	11.4	82.7	**11.6**	**52.7**	84.4	95.6	44.9
LaBSE-EMS-vanilla	15.6	3.6	9.6	11.5	1.4	6.5	2.1	8.2	7.0	0.8	9.0	40.5	10.8
LaBSE-EMS-scaled	47.3	16.2	26.1	52.4	4.3	17.8	19.3	72.6	7.4	4.4	67.5	96.1	38.0
LaBSE-EMS-joint (ours)	56.7	22.4	29.8	59.1	6.8	24.1	27.5	82.1	9.5	9.6	76.9	96.7	45.0
EMS (ours)	**58.9**	25.7	**30.5**	**61.1**	8.1	23.7	**28.7**	82.9	8.5	11.6	78.9	**97.0**	47.1
LaBSE	*65.2*	*87.9*	*80.0*	*63.0*	*93.7*	*86.8*	*65.3*	*90.3*	*91.9*	*91.0*	*92.1*	*96.9*	*70.2*

Table 2.6: **P@1 results of EMS's unseen languages on the Tatoeba benchmark.** Bold fonts denote the best precisions among all the models except LaBSE. We report the average of the English→X and X→English for each language.

whereas it is comparable with SBERT-distill and LaBSE-EMS-joint.[23]

Furthermore, we evaluate the other 54 unseen languages of our model (Table 2.6). Although few languages are trained in LASER and LASER2, we observe that EMS and LaBSE-EMS-joint still yield results comparable with LASER and significantly better than LASER2 for these 54 languages. This indicates that EMS has cross-lingual transferability for unseen languages to a certain extent with the joint vocabulary.

[23]LASER, LASER2, LaBSE-EMS-x and EMS utilized less than 300k training data in the "<300k(11)" setting, whereas SBERT-distill and LaBSE significantly used more training data.

Model	mUSE (182)	af-gu	hi-hy	jv-ka	kk-mr	my-nb	sw-ta	te-tl	ur-yo	<300k (240)
LASER	62.8	0.4	2.7	1.3	1.6	0.8	6.0	4.6	1.1	5.9
SBERT-distill	**99.7**	**84.7**	**99.1**	**46.6**	**82.5**	**97.3**	14.6	23.2	18.3	**60.6**
LaBSE-EMS-joint (ours)	**99.7**	13.6	89.6	39.3	72.7	3.5	32.7	38.0	20.1	46.6
EMS (ours)	**99.7**	8.9	89.6	38.7	68.6	6.5	**39.0**	**39.2**	**20.6**	47.0
LaBSE	*99.1*	*100.0*	*100.0*	*99.9*	*99.7*	*99.5*	*100.0*	*99.9*	*91.7*	*98.9*

Table 2.7: **Average P@1 results of non-English language pairs on Flores benchmark. Bold** are the best precisions among all the models except LaBSE. "mUSE" and "<300k" respectively denote 182 high-resource and 240 low-resource language pairs. We additionally report the specific results of 8 randomly selected low-resource language pairs.

In summary, the presented results on Tatoeba benchmarks highlight two key points: (1) EMS demonstrates superior data and computational efficiency, surpassing LASER, LASER2, and SBERT-distill; (2) AMS (i.e., LaBSE's contrastive loss) is not an appropriate form of contrastive loss in a dual-encoder framework, while temperature-based scaling and linear layers, as introduced in our contrastive loss (Section 2.2.3), facilitate the effectiveness of AMS, showing that AMS is compatible with our framework under certain changes. More precisely with the second point, the poor performance of LaBSE-EMS-vanilla demonstrates that LaBSE's AMS contrastive loss is dependent on LaBSE's extensive training data and large batch sizes. However, the incorporation of temperature-based scaling and linear layers (LaBSE-EMS-scaled) can enhance AMS's performance within EMS's efficient framework, and AMS's additive margin can enhance the performance in several low-resource languages (see results of LaBSE-EMS-joint).

2.4.2 Flores Similarity Search

In this section, we assess the model's ability to perform cross-lingual retrieval for non-English language pairs using the Flores multilingual benchmark [62, 42]. Flores is an N-way evaluation dataset that consists of 200 languages, with 1,012 sentences per language. We assess two distinct groups of languages: (1) 14 main languages, excluding English, which are supported by mUSE; (2) 16 low-resource languages, which are designated as "<300k" in Section 2.4.1. This results in a

Example 1	
en	The Declaration of Brussels (1874) stated that the "honours and rights of the family...should be respected."
zh	布鲁塞尔宣言（1874年）表示，"家庭荣誉和权利…应当受到尊重。"
Example 2	
en	In 2004. the E.U. undertook a major eastward enlargement, admitting ten new member states (eight of which were former communist states).
zh	2004年欧盟进行一次大规模东扩，接纳10个新成员国（其中的8个是前共产主义国家）。
Example 3	
en	In March 2013, Ban Ki-moon had also recommended to the Council that women raped in war have access to abortion services.
zh	2013年3月，潘基文同样建议安理会保证在战争中被强奸的妇女能享有堕胎服务。

Table 2.8: Extracted parallel sentence examples from BUCC that are not included in the official gold labels.

total of 182 high-resource and 240 low-resource language pairs. We compute the P@1 metric for each language pair, as described in Section 2.4.1.

Table 2.7 showcases the results for two groups of languages previously mentioned, along with eight randomly selected low-resource language pairs. Our analysis indicates that for 182 main non-English language pairs, SBERT-distill, LaBSE-EMS-joint, EMS, and LaBSE achieve nearly 100% precision, whereas LASER demonstrated a precision of 62.8. These results highlight that SBERT-distill, LaBSE-EMS-joint, EMS, and LaBSE are English-independent models. Secondly, for low-resource non-English language pairs, LASER performed poorly in retrieving the sentences accurately, whereas SBERT-distill, LaBSE-EMS-joint, and EMS delivered relatively good results. This demonstrates that the training objective of SBERT-distill, EMS, and LaBSE is conducive to generating language-agnostic embeddings. In contrast, the LASER model's translation objective still falls short of eliminating English as a pivot language for cross-lingual retrieval.

2.4.3 BUCC: Bi-text Mining

Moreover, we evaluate the model's cross-lingual retrieval performance on BUCC benchmark [274, 275] that contains the comparable corpora with the size of 150k∼1.2M for four language pairs: English–German, English–French, English–Russian, and English–Chinese. This task measures the model's ability to extract parallel sentences from comparable corpora. Following LASER and SBERT-distill, we use the margin-based scoring function [11] for mining parallel sentences. As the BUCC dataset mixes a significant number of monolingual sentences, we report F1 as the evaluation metric for this task following previous work [173, 54],

Model	en-de	en-fr	en-ru	en-zh	Avg.
mUSE	88.5	86.3	89.1	86.9	87.7
LASER	**95.4**	**92.4**	**92.3**	91.2	**92.8**
SBERT-distill	90.8	87.1	88.6	87.8	88.6
LaBSE-bilingual	92.6	90.0	90.1	**92.5**	91.3
LaBSE-EMS-joint (ours)	93.7	90.4	91.1	90.6	91.5
EMS (ours)	93.3	90.2	91.3	92.1	91.7
LaBSE	*95.9*	*92.5*	*92.4*	*93.0*	*93.5*

Table 2.9: **F1 Scores on the BUCC benchmark. Bold** fonts denote the best precisions among mUSE, LASER, SBERT-distill, LaBSE-bilingual, LaBSE-EMS-joint, and EMS.

which differs from the one employed for Tatoeba and Flores.

Results measured using F1 are listed in Table 2.9.[24] We observe that EMS exhibits significantly higher results than mUSE [245] and SBERT-distill, and comparable results with LaBSE-lingual and LaBSE-EMS-joint. However, compared with LASER and LaBSE, EMS exhibits slightly poor performance. Such performance deterioration is negligible because it can be attributed to incorrect gold labels within the BUCC dataset, which is also mentioned in Reimers and Gurevych [173]. For example, three extracted sentence pairs listed in Table 2.8 are translation pairs, whereas they are not contained in the official gold labels.

2.4.4 Cross-Lingual Sentence Retrieval

The Tatoeba benchmark supports the cross-lingual retrieval evaluation based on small-scale (1000 sentences for most language pairs) data, whereas the BUCC benchmark supports retrieval from large-scale data for four language pairs. Therefore, we conduct a cross-lingual sentence retrieval evaluation based on large-scale comparable data for 21 language pairs.[25] Based on our previous study [122], given

[24] We use the code from `https://github.com/UKPLab/sentence-transformers/blob/master/examples/applications/parallel-sentence-mining/bucc2018.py`

[25] In this study, we have chosen not to evaluate our model using the UN benchmark [271] as [54]. This decision is based on the fact that a portion of the UN benchmark data has been incorporated

Model	bg	cs	da	de	el	es	et	fi
LASER	90.5	**87.8**	**86.1**	89.4	**85.3**	89.4	87.6	**83.4**
SBERT-distill	83.3	73.6	78.6	81.4	72.2	82.5	75.1	73.7
LaBSE-EMS-joint (ours)	89.8	84.3	84.3	89.3	79.2	90.0	86.3	81.7
EMS (ours)	**90.9**	85.5	85.1	**90.1**	81.4	**90.9**	**87.7**	**83.4**
LaBSE	*91.2*	*87.8*	*88.9*	*90.4*	*85.3*	*89.8*	*88.3*	*82.8*
Model	fr	hr	hu	it	lt	lv	nl	pl
LASER	90.9	87.1	86.8	82.5	89.0	**84.8**	88.3	**81.9**
SBERT-distill	85.4	76.6	74.0	69.9	83.4	75.7	83.7	71.7
LaBSE-EMS-joint (ours)	90.8	85.7	82.6	82.6	87.9	82.5	89.1	79.2
EMS (ours)	**91.6**	**87.3**	**87.5**	**83.5**	**90.5**	84.0	**90.6**	81.5
LaBSE	*90.5*	*89.1*	*84.5*	*85.1*	*91.0*	*86.2*	*89.5*	*84.2*

Model	pt	ro	sk	sl	sv	Avg.
LASER	90.9	85.2	87.9	88.9	85.3	87.1
SBERT-distill	86.2	80.4	79.2	80.0	79.8	78.4
LaBSE-EMS-joint (ours)	90.3	86.0	86.9	87.7	86.0	85.8
EMS (ours)	**91.5**	**87.1**	**88.2**	**89.0**	**86.3**	**87.3**
LaBSE	*90.9*	*88.2*	*88.2*	*89.6*	*84.7*	*87.9*

Table 2.10: Cross-lingual sentence retrieval results on ParaCrawl. We report P@1 scores of 2,000 source queries while searching among 200k sentences in the target language. The best performance results among LASER, SBERT-distill, LaBSE-EMS-joint, and EMS are in **bold** font.

2000 sentences in language l_1, we conduct the translation retrieval from 200k candidate sentences in language l_2. Unlike our previous study, we used parallel sentences from ParaCrawl v5.0[26] [17] for evaluation because the previously used Europarl corpus is included in the training data in this study. We calculate P@1 for each language pair using margin-based scoring [11].

As reported in Table 2.10, EMS performs significantly better than SBERT-distill LaBSE-EMS-joint, and is comparable with LASER and LaBSE. The 21

into our model's training dataset, which could potentially bias the evaluation results.

[26] `https://opus.nlpl.eu/ParaCrawl-v5.php`

Model	en-de		en-es		en-fr		en-it		en-ja		en-ru		en-zh		Avg.
	→	←	→	←	→	←	→	←	→	←	→	←	→	←	
LASER	86.3	76.7	76.2	68.1	82.1	75.7	70.3	69.8	**71.5**	59.8	64.6	68.9	**77.7**	67.3	72.5
SBERT-distill	78.5	78.7	72.7	73.3	79.7	79.6	64.4	73.0	65.7	72.0	64.2	72.7	60.3	70.2	71.8
LaBSE-EMS-joint (ours)	86.1	81.0	78.1	**76.4**	**83.9**	80.5	71.1	70.1	64.3	**76.1**	67.3	76.0	65.8	**73.6**	75.0
EMS (ours)	**87.6**	**81.1**	**82.0**	75.5	82.9	**80.6**	70.4	**73.6**	67.0	72.3	**68.5**	**77.5**	68.6	69.1	**75.5**
LaBSE	*87.2*	*82.8*	*78.8*	*78.2*	*87.3*	*83.6*	*74.1*	*74.8*	*73.4*	*78.8*	*74.6*	*79.0*	*85.3*	*80.0*	*79.9*

Table 2.11: **MLDoc benchmark results (zero-shot scenario).** We report the mean accuracy of 5 runs. Best performance results of LASER, SBERT-distill, LaBSE-EMS-joint, and EMS are in **bold** font.

Model	en-de						en-fr						en-ja						Avg.
	books		dvd		music		books		dvd		music		books		dvd		music		
	→	←	→	←	→	←	→	←	→	←	→	←	→	←	→	←	→	←	
LASER	78.3	76.0	73.7	73.4	76.1	77.2	77.2	77.4	76.8	75.4	**75.8**	76.6	72.0	72.9	73.0	70.9	75.5	75.5	75.2
SBERT-distill	78.2	81.2	73.9	**77.1**	74.1	80.1	78.9	80.6	77.8	79.4	70.6	78.8	74.5	**81.9**	**76.5**	78.2	78.2	78.6	77.7
LaBSE-EMS-joint (ours)	**82.7**	82.8	**79.1**	73.2	77.4	**82.8**	81.8	84.0	**82.1**	78.5	74.9	80.6	**76.1**	78.8	75.6	75.2	77.5	80.3	79.1
EMS (ours)	82.3	**84.9**	77.0	76.7	**78.8**	81.9	80.4	**84.6**	78.0	**81.1**	74.7	**83.0**	75.6	79.5	75.4	**79.2**	**79.2**	**80.9**	**79.6**
LaBSE	*82.2*	*79.9*	*77.1*	*77.2*	*79.0*	*80.0*	*83.2*	*82.3*	*81.0*	*80.1*	*77.9*	*80.3*	*78.0*	*80.7*	*77.7*	*77.1*	*81.6*	*79.0*	*79.7*

Table 2.12: **Results of the cross-lingual sentiment classification of Amazon Review version-1.** We report the mean accuracy of 5 runs. The best performance results of LASER, SBERT-distill, LaBSE-EMS-joint, and EMS are in **bold** font.

languages evaluated herein are trained with more than 300k parallel sentences, for which we used approximately half of the LASER's training data and a tiny fraction of the LaBSE's training data. This suggests that our training architecture and objective are rather effective for languages where we used a certain number of parallel sentences. Furthermore, the superior performance relative to LaBSE-EMS-joint underscores the limitations in the effectiveness of the additive margin introduced by LaBSE's AMS contrastive objective, when handling large-scale cross-lingual retrieval tasks.

2.4.5 MLDoc: Multilingual Document Classification

Subsequently, we evaluate the model performance based on the MLDoc classification task. MLDoc[27] is a benchmark to evaluate cross-lingual sentence representations, which contain datasets for eight languages [99]. Following Artetxe and

[27] `https://github.com/facebookresearch/MLDoc`

Model	en-de		en-es		en-fr		en-ja		en-zh		Avg.
	→	←	→	←	→	←	→	←	→	←	
LASER	84.4	81.6	85.2	81.4	85.3	81.4	77.9	78.4	77.6	76.8	81.0
SBERT-distill	85.8	85.6	87.0	85.8	86.8	84.6	**81.7**	83.8	**81.6**	81.3	**84.4**
LaBSE-EMS-joint (ours)	**86.4**	83.5	85.8	**86.0**	86.0	85.1	79.4	80.8	78.0	81.8	83.3
EMS (ours)	85.7	**85.8**	**87.4**	84.9	**87.1**	**86.3**	79.0	**84.1**	78.5	**82.2**	84.1
LaBSE	*87.0*	*84.5*	*87.1*	*85.3*	*88.0*	*84.7*	*83.4*	*82.0*	*80.7*	*79.9*	*84.3*

Table 2.13: **Results of the cross-lingual sentiment classification of Amazon Review version-2.** We report the mean accuracy of 5 runs. The best performance results of LASER, SBERT-distill, LaBSE-EMS-joint, and EMS are in **bold** font.

Schwenk [12], we conduct the evaluation in a zero-shot manner using 1000 sentences in language l_1 for training, 1000 sentences in language l_1 for validation, and 4000 sentences in language l_2 for the test. Specifically, we train a multilayer perceptron classifier based on source language representations and test the classifier for the target language.

We list the average results of 5 runs for 7 language pairs and 14 directions in Table 2.11. We observe significantly higher accuracies of EMS in most directions than those of LASER and SBERT-distill, and comparable results with LaBSE-EMS-joint. These results demonstrate the effectiveness of the proposed training method. Although LASER yields better performance for English→Japanese and English→Chinese, it performs much worse in the reverse directions. We further calculate the average accuracy discrepancy between two directions for each language pair. LASER shows 7.3, whereas SBERT-distill is 3.5 and EMS is 4.7. This indicates that LASER is highly sensitive to the specific cross-lingual transfer direction, whereas SBERT-distill and EMS are much more robust.

2.4.6 CLS: Cross-Lingual Sentiment Classification

Moreover, we gauge the quality of language-agnostic sentence representation based on the sentiment classification task. We use the two versions of the Amazon review datasets for evaluation to conduct the zero-shot cross-lingual classification. The version-1 dataset [162] includes the data for English–German, English—French,

and English–Japanese on "books," "dvd," and "music" domains for each language pair. For each language pair and domain, we use 2000 sentences in language l_1 for training, 2000 sentences in language l_1 for validation, and 2000 sentences in language l_2 for testing. However, the version-2 dataset [84] includes five language pairs, whereas different genres of the reviews are mixed. For each language pair, we use 2000 sentences for training, 4000 sentences for validation, and 4000 sentences for the test. Same as on MLDoc, we train a multi-layer perceptron using the language-agnostic sentence representations in language l_1 and test the classifier for another language.

As listed in Tables 2.12 and 2.13, EMS significantly outperforms LASER and performs comparably to LaBSE on the two versions of the datasets, which proves the effectiveness of EMS. SBERT-distill achieves comparable results on the version-2 dataset, whereas its performance negligibly deteriorates on the version-1 dataset. This can be attributed to SBERT-distill's capability of clustering similar sentences (Section 4.1 in Reimers and Gurevych [173]). On the version-1 dataset, each genre of the reviews is evaluated; more similar sentences in each genre compared with version-2 lead to lower classification accuracy for version-1. Moreover, EMS marginally surpasses LaBSE-EMS-joint in two benchmarks, suggesting that the additive margin in LaBSE's AMS does not enhance EMS's contrastive loss in classification tasks, which aligns with the findings from the MLDoc benchmark.

2.4.7 Ablation Study and Training Efficiency

We conduct an ablation study to investigate the effectiveness of each model component and the computation resource. We report the results on the Tatoeba, Flores, and MLDoc benchmarks for cross-lingual sentence retrieval and classification tasks, respectively.

As listed in Table 2.14, we observe that the performance significantly decreases on Tatoeba, Flores, and MLDoc benchmarks by removing the language token, sentence-level contrastive objective, XTR objective, or the linear layer within the contrastive objective. Moreover, sharing the Transformer embedding layer parameters with the L_{emb} in the XTR objective and replacing XTR with UGT degrade the model performance. Among all these ablations, we observe a significant de-

Model	Tatoeba			Flores		MLDoc		Sec./1k Steps
	Avg. (58)	>300k (43)	<300k (15)	mUSE (182)	<300k (240)	en→ X Avg.	X→ en Avg.	
EMS	89.8	95.4	**73.7**	**99.7**	**47.0**	**75.3**	75.7	732
− *langs tok*	89.3	95.4	71.8	**99.7**	46.9	73.5	74.4	725
− $\mathcal{L}_{cntrs}$	84.3	93.9	56.6	99.5	33.3	71.0	72.1	725
− $\mathcal{L}_{XTR}$	85.5	92.9	64.5	97.1	33.5	68.8	69.1	696
− L_{cntrs_mlp}	86.3	93.8	64.6	99.1	39.0	69.8	73.8	727
share L_{emb} *params*	85.1	93.8	60.3	98.8	32.2	68.5	71.1	730
replace XTR with UGT	86.9	94.5	64.9	99.5	40.6	71.1	73.8	731
LaBSE-EMS-vanilla	29.0	34.6	13.0	19.0	4.6	42.5	43.3	709
LaBSE-EMS-joint (ours)	**90.0**	**95.7**	**73.7**	**99.7**	46.6	73.8	**76.2**	747
replace V100 with A100	89.8	95.5	73.3	**99.7**	46.6	75.1	75.4	-

Table 2.14: **Ablation study of each model component, the AMS objective of LaBSE, and the computation resource.** Best performances are in **bold** font. The training efficiency is measured in seconds per 1k steps, utilizing four V100 GPUs.

crease in low-resource languages for training data less than 300k on both Tatoeba and Flores benchmarks, which indicates that the performance is more sensitive to model components on low-resource languages. This motivates future exploration to improve the model performance more for low-resource languages.

By comparing "− $\mathcal{L}_{cntrs}$" with "− $\mathcal{L}_{XTR}$," we observe superior performances of "− $\mathcal{L}_{cntrs}$" on MLDoc and high-resource languages of Tatobea and Flores, and superior performances of "− $\mathcal{L}_{XTR}$" on Tatoeba. This demonstrates that the generative objective contributes more to the classification of downstream tasks and the detection of parallel sentences high-resource language pairs, whereas the contrastive objective is more beneficial for the detection of parallel sentences of low-resource language pairs.

Moreover, we observe a negligible decrease in "− *langs tok*" on the Tatoeba and Flores benchmark. As the ground-truth label we designed for the XTR objective includes information on tokens in specific languages, the effect of the language token gradually diminishes during the model training. In our prior research, we recommended using UGT for bilingual settings; however, the current XTR in EMS surpasses it in three benchmarks. This may be because UGT is a more challenging task than XTR, which concurrently predicts the token distribution of the target language and a masked token, and the current model architecture may not be capable of accommodating numerous languages for UGT. Moreover, in our

previous research [122], we found that contrastive objectives negatively impacted classification tasks like MLDoc. However, in the current study with EMS in a massively multilingual context, both generative and contrastive objectives consistently enhance performance in retrieval and classification tasks. This improvement is likely due to the enhanced general capabilities of sentence embeddings, a result of exposure to massively multilingual data.

In terms of training efficiency, as detailed in Table 2.14, the generative and contrastive objectives require extra 36 and 7 seconds per 1,000 steps, respectively. This is notably more efficient compared to using a Transformer decoder for the generative objective in a translation task like LASER and LASER2, where the process could potentially double the training time.[28] Our XTR approach enhances sentence embedding in a generative manner, bypassing the need for a Transformer decoder in a dual-encoder setup. Incorporating an additive margin into the contrastive objective leads to diminished performance and efficiency in LaBSE-EMS-vanilla compared to $-\mathcal{L}_{XTR}$, while LaBSE-EMS-joint shows only marginal improvements with a notable decrease in training efficiency relative to EMS. In addition, by replacing V100 GPUs with A100 and a larger batch size of 200 parallel sentences, only trivial performance fluctuation is observed, which suggests that EMS is robust to the computation resource.

2.4.8 Case Study for the XTR Objective

EMS utilizes an XTR objective that is independent of word order. To evaluate its robustness against sentences with identical word frequencies but differing semantics, we conducted a case study. This study compares EMS with LASER, SBERT-distill, and LaBSE using the following specific sentences designed for this purpose:

(1). *can you believe that what he actually did was steal the money she saved for the children?*

(2). *what can you actually believe she did was save the money for the children that he stole?*

[28] A 6-layer Transformer decoder demands more training time compared to a 6-layer Transformer encoder due to its auto-regressive token generation process.

We process the above sentences in their uncased form to guarantee an identical word bag for both sentences. Upon acquiring their sentence embeddings, we calculate the cosine similarity to determine the extent to which different MSE models perceived these two sentences as similar.

LASER, SBERT-distill, LaBSE, and EMS produce similarities of 0.90, 0.97, 0.95, and 0.97, respectively. This suggests that EMS struggles with such sentence pairs, despite the joint combined sentence-level contrastive objective should theoretically account for word order. Similarly, SBERT-distill and LaBSE, which do not incorporate word order-independent objectives like XTR, also fail to discern the semantic differences between the sentences. This indicates that sentence-level objectives within a dual-encoder architecture may not effectively address this issue. In contrast, LASER exhibits a lower similarity for this sentence pair, suggesting that its translation objective, which generates the target sentence word by word, might be more capable of resolving such issues. However, as these sentence pairs are relatively rare, further investigation into this limitation of the dual-encoder architecture is reserved for future research.

2.5 Summary of This Chapter

This study presented EMS, an efficient and effective method for MSE learning. To improve training efficiency in terms of data and computation while retaining the quality of MSE, we proposed a novel framework to train "XTR" generative and sentence-level contrastive objectives jointly. The empirical results based on four cross-lingual sentence retrieval tasks and three cross-lingual sentence classification tasks demonstrated the effectiveness of EMS. In future research, we aim to leverage LLMs for model initialization to further refine sentence embeddings. Additionally, we plan to streamline the model architecture through knowledge distillation, aiming for a more rapid inference experience.

Chapter 3

LEALLA: Learning Lightweight Language-agnostic Sentence Embeddings with Knowledge Distillation

Language-agnostic sentence embedding models [12, 245, 173, 121, 54, 115, 117] align multiple languages in a shared embedding space, facilitating parallel sentence alignment that extracts parallel sentences for training translation systems [181]. Among them, LaBSE [54] achieves state-of-the-art parallel sentence alignment accuracy over 109 languages. However, 471M parameters of LaBSE lead to the computationally heavy inference. The 768-dimensional sentence embeddings of LaBSE (LaBSE embeddings) make it suffer from computation overhead of downstream tasks (e.g., kNN search). This limits its application on resource-constrained devices. Therefore, we explore training a lightweight model to generate low-dimensional sentence embeddings while retaining the performance of LaBSE.

We first investigate the performance of dimension-reduced LaBSE embeddings and show that it performs comparably with LaBSE. Subsequently, we experiment with various architectures to see whether a lightweight encoder can obtain such

effective low-dimensional embeddings. We observe that the thin-deep [175] architecture is empirically superior for learning language-agnostic sentence embeddings. Diverging from previous work, we show that low-dimensional embeddings based on a lightweight model are effective for parallel sentence alignment of 109 languages.

LaBSE benefits from multilingual language model pre-training, but no multilingual pre-trained models are available for the lightweight architectures. Thus, we propose two knowledge distillation methods to further enhance the lightweight models by forcing the model to extract helpful information from LaBSE. We present three lightweight models improved with distillation: **LEALLA-small**, **LEALLA-base**, and **LEALLA-large**, with 69M, 107M, and 147M parameters, respectively. Fewer model parameters and their 128-d, 192-d, and 256-d sentence embeddings are expected to accelerate downstream tasks, while the performance drop of merely up to 3.0, 1.3, and 0.3 P@1 (or F1) points is observed on three benchmarks of parallel sentence alignment. In addition, we show the effectiveness of each loss function through an ablation study.

3.1 Background: LaBSE

LaBSE [54] fine-tunes dual encoder models [66, 244] to learn language-agnostic embeddings from a large-scale pre-trained language model [38]. LaBSE is trained with parallel sentences, and each sentence pair is encoded separately by a 12-layer Transformer encoder. The 768-d encoder outputs are used to compute the training loss and serve as sentence embeddings for downstream tasks. Expressly, assume that the sentence embeddings for parallel sentences in a batch are $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ where N denotes the number of the sentence pairs within a batch. LaBSE trains the bidirectional additive margin softmax (AMS) loss:

$$\mathcal{L}_{ams} = \frac{1}{N}\sum_{i=1}^{N}(\mathcal{L}(\mathbf{x}_i, \mathbf{y}_i) + \mathcal{L}(\mathbf{y}_i, \mathbf{x}_i)), \tag{3.1}$$

where the loss for a specific sentence pair in a single direction is defined as:

$$\mathcal{L}(\mathbf{x}_i, \mathbf{y}_i) = -\log \frac{e^{\phi(\mathbf{x}_i, \mathbf{y}_i) - m}}{e^{\phi(\mathbf{x}_i, \mathbf{y}_i) - m} + \sum_{n \neq i} e^{\phi(\mathbf{x}_i, \mathbf{y}_n)}}. \tag{3.2}$$

m is a margin for optimizing the separation between correct and incorrect translation pairs. $\phi(\mathbf{x}_i, \mathbf{y}_i)$ is defined as Cosine Similarity between $\mathbf{x}_i$ and $\mathbf{y}_i$.

3.2 Lightweight Language-agnostic Embeddings

To address the efficiency issue of LaBSE, we probe the lightweight model for learning language-agnostic embeddings with the following experiments: (1) We directly reduce the dimension of LaBSE embeddings to explore the optimal embedding dimension; (2) We shrink the model size with various ways to explore the optimal architecture.

3.2.1 Evaluation Settings

We employ Tatoeba [12], United Nations (UN) [271], and BUCC [157] benchmarks for evaluation, which assess the model performance for parallel sentence alignment. Following LaBSE [54] and LASER [12], we report the average P@1 of bidirectional retrievals for all the languages of Tatoeba, the average P@1 for four languages of UN, and the average F1 of bidirectional retrievals for four languages of BUCC.[1] Tatoeba [12] supports the evaluation across 112 languages and contains up to 1,000 sentence pairs for each language and English. The languages of Tatoeba that are not included in the training data of LaBSE and LEALLA serve as the evaluation for unseen languages. UN [271] is composed of 86,000 aligned bilingual documents for en-ar, en-es, en-fr, en-ru, and en-zh. Following LaBSE [54], we evaluate the model performance for es, fr, ru, and zh on the UN task. There are about 9.5M sentence pairs for each language with English after deduping. BUCC shared task [157] is a benchmark to mine parallel sentences from comparable corpora. We conduct the evaluation using BUCC2018 tasks for en-de, en-fr, en-ru, and en-zh, following the setting of Reimers and Gurevych [173].[2] For the results of LaBSE reported in Table 3.3, we re-conduct the evaluation experiments using the open-sourced model of LaBSE.[3]

[1] For BUCC, we use margin-based scoring [11] for filtering translation pairs.

[2] `https://github.com/UKPLab/sentence-transformers/blob/master/examples/applications/parallel-sentence-mining/bucc2018.py`

[3] `https://tfhub.dev/google/LaBSE`

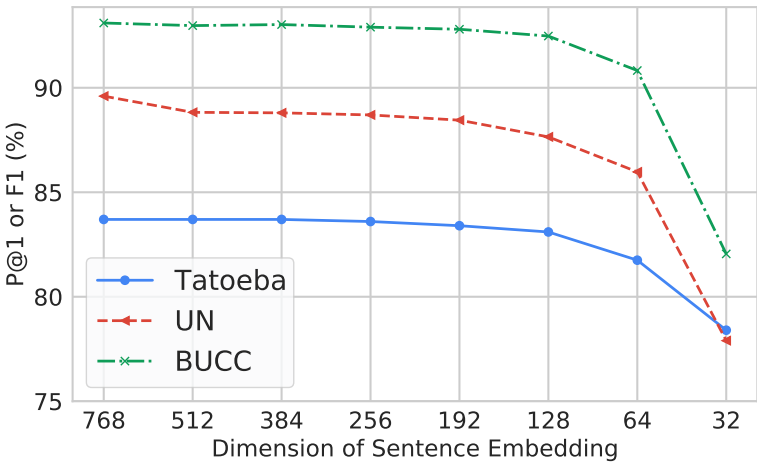

Figure 3.1: Dimension reduction for LaBSE.

3.2.2 Exploring the Optimal Dimension of Language-agnostic Sentence Embeddings

Mao et al. [122] showed that a 256-d bilingual embedding space could achieve an accuracy of about 90% for parallel sentence alignment. However, existing multilingual sentence embedding models such as LASER [12], SBERT [173], EMS [115], and LaBSE generate 768-d or 1024-d sentence embeddings, and whether a low-dimensional space can align parallel sentences over tens of languages with a solid accuracy (>80%) remains unknown. Thus, we start with the dimension reduction experiments for LaBSE to explore the optimal dimension of language-agnostic sentence embeddings.

We add an extra dense layer on top of LaBSE to transform the dimension of LaBSE embeddings from 768 to lower values. We experiment with seven lower dimensions ranging from 512 to 32. We fine-tune 5k steps for fitting the newly added dense layer, whereas other parameters of LaBSE are fixed. Refer to Section 3.3.2 for training details.

As shown in Figure 3.1, the performance drops more than 5 points when the dimension is 32 on Tatoeba, UN, and BUCC. Meanwhile, given sentence embeddings with a dimension over 128, they performs slightly worse than 768-d LaBSE embeddings with a performance drop of fewer than 2 points, showing

#	L	d_h	H	P	P_E	Tatoeba	UN	BUCC
LaBSE								
0	12	768	12	471M	85M	83.7	89.6	93.1
Fewer Layers								
1	6	768	12	428M	42M	82.9	88.6	91.9
2	3	768	12	407M	21M	82.2	87.5	91.2
Smaller Hidden Size								
3	12	384	12	214M	21M	82.6	88.4	92.1
4	12	192	12	102M	6M	81.0	87.0	91.3
Thin-deep Architecture								
5	24	384	12	235M	42M	83.2	88.6	92.4
6	24	256	8	147M	19M	82.9	88.5	92.2
7	24	192	12	107M	11M	81.7	87.4	91.9
8	24	128	8	69M	5M	80.3	86.3	90.4

Table 3.1: Results of LaBSE variants. **L**, $\mathbf{d_h}$, **H**, **P**, and $\mathbf{P_E}$ denote the number of layers, dimension of hidden states, number of attention heads, number of parameters, and number of encoder parameters (except for the word embedding layer). Refer to Appendix A.3 for detailed results.

that low-dimensional sentence embeddings can align parallel sentences in multiple languages. Refer to Appendix A.2 for detailed results.

3.2.3 Exploring the Optimal Architecture

Although we revealed the effectiveness of the low-dimensional embeddings above, it is generated from LaBSE with 471M parameters. Thus, we explore whether such low-dimensional sentence embeddings can be obtained from an encoder with less parameters. We first reduce the number of layers (#1 and #2 in Table 3.1) and the size of hidden states (#3 and #4) to observe the performance. Subsequently, inspired by the effectiveness of FitNet [175] and MobileBERT [207] and taking advantage of the low-dimensional sentence embeddings shown above, we

experiment with thin-deep architectures with 24 layers (#5 - #8), leading to fewer encoder parameters.[4] Refer to Section 3.3.2 for training details.

We report the results in Table 3.1. First, architectures with fewer layers (#1 and #2) perform worse than LaBSE on all three tasks and can only decrease parameters by less than 15%. Second, increasing the number of layers (#5 and #7) improves the performance of 12-layer models (#3 and #4) with a limited parameter increase of less than 10%. Referring to LaBSE (#0), low-dimensional embeddings from thin-deep architectures (#5 - #8) obtain solid results on three benchmarks with performance drops of only 3.4 points at most. Until this point, we showed that thin-deep architecture is effective for learning language-agnostic sentence embeddings.

3.3 Knowledge Distillation from LaBSE

Besides the large model capacity, multilingual language model pre-training benefits LaBSE for parallel sentence alignment. As no multilingual pre-trained language models are available for lightweight models we investigated in Section 3.2.3, we instead explore extracting helpful knowledge from LaBSE.

3.3.1 Methodology

Feature distillation and logit distillation have been proven to be effective paradigms for knowledge distillation [71, 175, 252, 211]. In this section, we propose methods applying both paradigms to language-agnostic sentence embedding distillation. We use LaBSE as a teacher to train students with thin-deep architectures which were discussed in Section 3.2.3.

Feature Distillation

We propose applying feature distillation to language-agnostic sentence embedding distillation, which enables lightweight sentence embeddings to approximate the LaBSE embeddings via an extra dense layer. We employ an extra trainable dense

[4]Following MobileBERT, we attempted architectures that have an identical size for hidden state and feed-forward hidden state, but it works poorly than #5 - #8. (Refer to Appendix A.3)

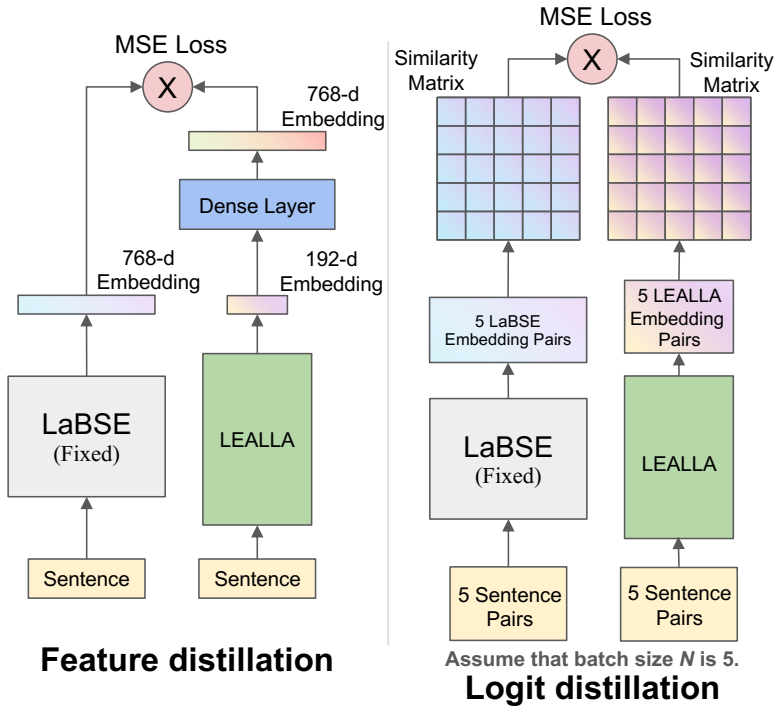

Figure 3.2: Feature and logit distillation from LaBSE.

layer on top of the lightweight models to unify the embedding dimension of LaBSE and lightweight models to be 768-d, as illustrated in Figure 3.2.[5][6] The loss function is defined as follows:

$$\mathcal{L}_{fd} = \frac{1}{N}\sum_{i=1}^{N}(\| \mathbf{x}_i^t - f(\mathbf{x}_i^s) \|_2^2 + \| \mathbf{y}_i^t - f(\mathbf{y}_i^s) \|_2^2), \tag{3.3}$$

where $\mathbf{x}^t$ (or $\mathbf{y}^t$) and $\mathbf{x}^s$ (or $\mathbf{y}^s$) are the embeddings by LaBSE and the lightweight model, respectively. $f(\cdot)$ is a trainable dense layer transforming the dimension from d ($d < 768$) to 768.

[5]SBERT [173] used feature distillation to make monolingual sentence embeddings multilingual, but distillation between different embedding dimensions has not been studied.

[6]We investigated another two patterns to unify the embedding dimensions in Appendix A.1, but they performed worse.

Logit Distillation

We also propose applying logit distillation to language-agnostic sentence embedding distillation to extract knowledge from the sentence similarity matrix as shown in Figure 3.2. Logit distillation forces the student to establish similar similarity relationships between the given sentence pairs as the teacher does. We propose the following mean squared error (MSE) loss:

$$\mathcal{L}_{ld} = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\left(\phi \left(\mathbf{x}_i^t, \mathbf{y}_j^t \right) - \phi \left(\mathbf{x}_i^s, \mathbf{y}_j^s \right) \right) / T \right)^2, \tag{3.4}$$

where T is a distillation temperature, and other notations follow those in Equations 3.2 and 3.3.

Combined Loss

Finally, we combine two knowledge distillation loss functions with the AMS loss (Equation 3.1) to jointly train the lightweight model:

$$\mathcal{L}_{lealla} = \alpha \mathcal{L}_{ams} + \beta \mathcal{L}_{fd} + \gamma \mathcal{L}_{ld}. \tag{3.5}$$

Here α, β, and γ are weight hyperparameters, which are tuned with the development data.

3.3.2 Experiments

Training

We train three models, **LEALLA-small**, **LEALLA-base**, and **LEALLA-large**, using the thin-deep architectures of #8, #7, and #6 in Table 3.1 and the training loss of Equation 3.5. All of the models in this work are trained with the same training data and development data as LaBSE [54]. Refer to Section 3.1 and Appendix C of Feng et al. [54] for dataset and supported language details. We train models on Google Cloud TPU V3 with 32-cores with a global batch size of 8,192 sentences and a maximum sequence length of 128. For a fair comparison with LaBSE for more than 109 languages, we use the 501k vocabulary of LaBSE (trained with BPE [190]) and do not consider modifying its size in this work. We

Hyperparameter	Bound
α	1
β	1e02, 1e03, 1e04, 1e05
γ	1e-01, 1e-02, 1e-03
batch size	2,048, 4,096, 8,192
learning rate	1e-4, 5e-4, 1e-3

Table 3.2: Hyperparameter bounds.

Model	La.	d	P	Ttb.	UN					BUCC				
					es	fr	ru	zh	avg.	de	fr	ru	zh	avg.
LASER [12]	93	1024	154M	65.5	-	-	-	-	-	**95.4**	**92.4**	**92.3**	91.7	**93.0**
m-USE [245]	16	512	**85M**	-	86.1	83.3	88.9	78.8	84.3	88.5	86.3	89.1	86.9	87.7
SBERT [173]	50	768	270M	67.1	-	-	-	-	-	90.8	87.1	88.6	87.8	88.6
EMS [115]	62	1024	148M	69.2	-	-	-	-	-	93.3	90.2	91.3	**92.1**	91.7
LaBSE [54]	**109**	768	471M	**83.7**	**90.8**	**89.0**	**90.4**	**88.3**	**89.6**	**95.5**	**92.3**	**92.2**	**92.5**	**93.1**
LEALLA-small	**109**	**128**	**69M**	80.7	89.4	86.0	88.7	84.9	87.3	94.0	90.6	91.2	90.3	91.5
LEALLA-base	**109**	**192**	**107M**	**82.4**	**90.3**	**87.4**	**89.8**	**87.2**	**88.7**	94.9	91.4	91.8	91.4	92.4
LEALLA-large	**109**	**256**	147M	**83.5**	**90.8**	**88.5**	**89.9**	**87.9**	**89.3**	**95.3**	**92.0**	**92.1**	**91.9**	**92.8**

Table 3.3: Results of LEALLA. We mark the best 3 scores in **bold**. **La.**, **d**, **P**, and **Ttb.** indicate the number of languages, dimension of sentence embeddings, number of parameters, and Tatoeba.

employ AdamW [112] for optimizing the model using the initial learning rate of 1e-03 for models with a hidden state size larger than 384 and 5e-04 for models with a hidden state size smaller than 256. For LEALLA-small and LEALLA-base, α, β, and γ are set as 1, 1e03 and 1e-02. For LEALLA-large, they are set as 1, 1e04, and 1e-02, respectively. T in Equation 3.4 is set to 100. All the models in Section 3.2.2 are trained for 5k steps. Models in Section 3.2.3 and Section 3.3 with a hidden state size over 256 are trained for 200k steps, and those with a hidden state size below 192 are trained for 100k steps. It costs around 24 hours, 36 hours, and 48 hours to train LEALLA-small, LEALLA-base, and LEALLA-large, respectively. Hyperparameters are tuned using a held-out development dataset following Feng et al. [54] with a grid search. The bounds tuned for each hyperparameter are shown in Table 3.2.

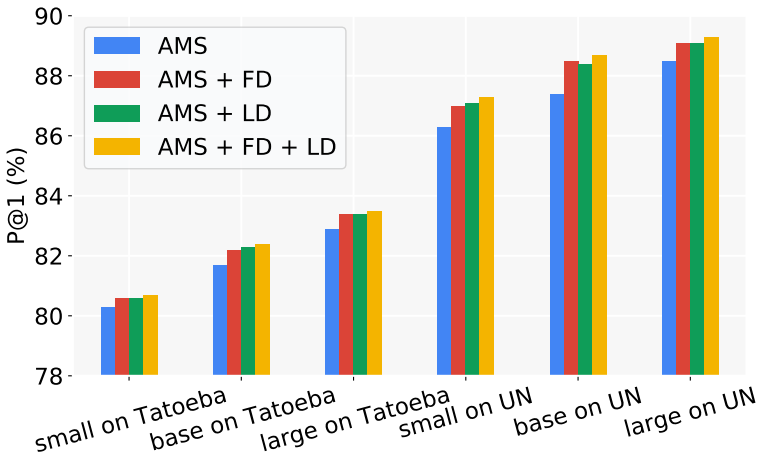

Figure 3.3: LEALLA with different loss combinations. AMS, FD, and LD mean $\mathcal{L}_{ams}$, $\mathcal{L}_{fd}$, and $\mathcal{L}_{ld}$.

Results

The results of LEALLA on Tatoeba, UN, and BUCC benchmarks are presented in Table 3.3. Overall, LEALLA can yield competitive performance compared with previous work. LEALLA-large performs comparably with LaBSE, where the average performance difference on three tasks is below 0.3 points. LEALLA-base and LEALLA-small obtain strong performance for high-resource languages on UN and BUCC, with a performance decrease less than 0.9 and 2.3 points, respectively. They also achieve solid results on Tatoeba with 1.3 and 3 points downgrades compared with LaBSE. The solid performance of LEALLA on Tatoeba demonstrates that it is effective for aligning parallel sentences for more than 109 languages. Moreover, all the LEALLA models perform better or comparably with previous studies other than LaBSE.

Ablation Study

We inspect the effectiveness of each loss component in an ablative manner. First, we compare settings with and without distillation loss functions. As shown in Figure 3.3, by adding $\mathcal{L}_{fd}$ or $\mathcal{L}_{ld}$, LEALLA trained only with $\mathcal{L}_{ams}$ is improved on Tatoeba and UN tasks. By further combining $\mathcal{L}_{fd}$ and $\mathcal{L}_{ld}$, LEALLA consistently

Loss	LEALLA-small		LEALLA-base		LEALLA-large	
	Tatoeba	UN	Tatoeba	UN	Tatoeba	UN
all	*80.7*	*87.3*	*82.4*	*88.7*	*83.5*	*89.3*
$\mathcal{L}_{ams}$	**80.3**	**86.3**	**81.7**	87.4	**82.9**	**88.5**
$\mathcal{L}_{fd}$	78.2	85.2	81.1	**88.1**	82.4	88.1
$\mathcal{L}_{ld}$	75.1	2.3	80.6	63.1	82.3	84.1

Table 3.4: Results of LEALLA with each loss function. "*all*" denotes LEALLA without ablation (with all the loss functions).

achieves superior performance. Second, we separately train LEALLA with each loss. Referring to the results reported in Table 3.4, LEALLA trained only with $\mathcal{L}_{fd}$ yields solid performance in the "small" and "base" models compared with $\mathcal{L}_{ams}$, showing that distillation loss benefits parallel sentence alignment. $\mathcal{L}_{fd}$ and $\mathcal{L}_{ld}$ perform much worse in the "small" model, which may be attributed to the discrepancy in the capacity gaps between the teacher model (LaBSE) and the student model ("small" or "base").[7] Refer to Appendix A.4 for all detailed results in this section.

3.4 Summary of This Chapter

We presented LEALLA, a lightweight model for generating low-dimensional multilingual sentence embeddings. Experimental results showed that LEALLA could yield solid performance for 109 languages after distilling knowledge from LaBSE. Future work can focus on reducing the vocabulary size of LaBSE to shrink the model further and exploring the effectiveness of lightweight model pre-training for parallel sentence alignment.

[7] $\mathcal{L}_{ld}$ can hardly work for UN and BUCC as they contain hundreds of thousands of candidates for the model to score, which is more complicated than the 1,000 candidates of Tatoeba.

Chapter 4

Linguistically-driven Multi-task Pre-training for Low-resource Neural Machine Translation

Neural machine translation (NMT) [16, 208] can achieve state-of-the-art performance when large parallel corpora are available for training. However, this prerequisite for parallel corpora limits its usefulness for several language pairs, such as Japanese, Chinese, and Korean, along with domains (history and COVID) for which such large corpora do not exist. Often, these resource-poor language pairs consist of languages that have resource-rich monolingual corpora. Therefore, it is possible to compensate for the lack of parallel corpora by leveraging large monolingual corpora. One popular approach for this is data augmentation, for instance, through back-translation [189, 72]. Another approach involves pre-training the NMT model on tasks that only require monolingual corpora [165, 204].

As a promising technique for leveraging monolingual corpora, pre-training has experienced a surge in popularity in NLP ever since models such as BERT [46] achieved state-of-the-art results in text understanding. However, BERT-like models were not designed to be used for NMT in the sense that they are essentially techniques for pre-training encoders, but not sequence-to-sequence models. To address this, Song et al. [204], Lewis et al. [100] and Liu et al. [110] recently pro-

posed self-supervised language-agnostic pre-training methods, which are sequence-to-sequence pre-training tasks for NMT, have achieved new state-of-the-art results in low-resource scenarios.

Languages that are sufficiently "rich" to have large monolingual corpora often have available tools for linguistic analysis. Meanwhile, usually a low-resource language pair is composed by a resource-rich language and a low-resource language and the linguistic knowledge of the resource-rich language can be easily extracted. In addition, studies such as Sennrich and Haddow [187] and Murthy et al. [135] have demonstrated that linguistic knowledge can improve NMT without using additional corpora. Therefore, it is natural to use monolingual corpora and linguistic tools in bilingual low-resource scenarios. However, the manner in which linguistic knowledge should be provided is not always clear, because NMT models are implemented in an end-to-end scheme. From a technical perspective, it is practical to extract linguistic features on the monolingual side. Therefore, monolingual pre-training provides an ideal framework for leveraging monolingual corpora and injecting linguistic information.

In Mao et al. [118], we proposed a linguistically motivated pre-training approach known as Japanese-specific sequence-to-sequence (JASS), which was inspired by masked sequence-to-sequence pre-training (MASS), but focused on syntactic analysis obtained by using a parser. Particularly, we added syntactic constraints to the sentence-masking process of the MASS to obtain the bunsetsu-based MASS task (BMASS).[1] We also proposed the bunsetsu reordering-based sequence-to-sequence (BRSS), which is a linguistically motivated reordering task. Several previous studies [100, 167] have provided evidence that "multi-task" pre-training that combines various styles of self-supervised training tasks results in significantly superior results for NMT. We proposed JASS based on a combination of the above-mentioned two tasks and it is tailored for NMT involving Japanese.

In contrast, in this study, we also propose linguistically-driven pre-training methods for English to leverage linguistic-specific information in the pre-training phase.[2] They are referred to as phrase structure-based MASS (PMASS) & head

[1]For BMASS, bunsetsus are used as syntactic spans, which is the elementary syntactic component of Japanese. It can be extracted using the KNP. [95, 130]

[2]Although some language pairs involving English are middle- or high-resource scenarios (par-

finalization-based sequence-to-sequence (HFSS), and their combination is denoted as English-specific sequence-to-sequence (ENSS).[3] Moreover, unlike the proposed methods for Japanese, the proposed methods for English can be transplanted onto any SVO language. Thus, our proposed ENSS and JASS can be applied to any translation pair involving English or Japanese.[4]

We experimented with ASPEC Japanese–English & Japanese–Chinese [140], Wikipedia Japanese–Chinese [34, 35], and News English–Korean [151] in various pre-training settings for JASS and ENSS.[5] Our results indicate that BMASS, BRSS, and HFSS significantly outperform the state-of-the-art MASS pre-training, whereas PMASS yields marginal improvements. Furthermore, we demonstrate that linguistically-driven multi-task pre-training methods (JASS & ENSS) lead to further improvements of up to +2.9 BLEU points for Japanese to English, +2.7 BLEU points for English to Japanese, +4.3 BLEU points for Japanese to Chinese , +7.0 BLEU points for Chinese to Japanese, +0.5 BLEU points for English to Korean, and +1.3 BLEU points for Korean to English in low-resource scenarios, respectively.

Unlike in our previous study [118], we provide substantial analyses for evaluating the translations generated by JASS and ENSS, which focus on the relationship between different pre-training tasks, and the specific adequacy and fluency of corresponding translations. Specifically, we validate the superior translation adequacy improvement of linguistically-driven methods by implementing automatic adequacy evaluation using LASER, human evaluation, and case study. To confirm the complementary nature between the masked language model and reordering the pre-training task, we performed an evaluation of the pre-training accuracy.

We expect this study to extend the usefulness of linguistically-driven pre-

allel corpora size over 100k), we deem that it is worth proposing methods for English because a large number of low-resource language pairs involving English are still present.

[3]Head finalization [77] is the technique used to reorder sentences in SVO language to be SOV-like sentences.

[4]According to the reordering task we proposed (specifically, BRSS and HFSS), more significant improvements are expected to observe on English–SOV language or Japanese–SVO language.

[5]In Mao et al. [118], we only conducted experiments on ASPEC Japanese–English and JaRuNC Japanese–Russian for JASS (BLEU results on Japanese–Russian were excessively low for comparison).

training methods for more low-resource language pairs and compensate for the defects of Mao et al. [118] in terms of the empirical evaluation. The contributions of this study can be summarized as follows.

1. **BMASS and BRSS:** Linguistically-driven novel pre-training methods for NMT involving Japanese.
2. **PMASS and HFSS:** Linguistically-driven novel pre-training methods for NMT involving English (can be theoretically implemented on any SVO language).
3. **Multi-task pre-training (JASS and ENSS)**: We demonstrate that multi-task training through the combination of the masked language model and reordering task (BMASS+BRSS & MASS+HFSS) leads to better performance. Particularly, BMASS and BRSS can complement each other more if they are performed based on analogous syntactic units.
4. **Empirical evaluation:** Comparisons among MASS, BART, JASS, ENSS and newly added baseline methods (MultiMASS and Deshuffling) for 6 translation directions and 3 different domains in several data size settings to identify situations in which each technique can be the most effective compared to other techniques.
5. **Analyses:** Linguistic and statistical analyses of pre-training methods, their inter-relationships, and corresponding translations.

4.1 Related Work

4.1.1 Low-resource Neural Machine Translation

There are mainly three lines of work related to improving NMT in low-resource situations: cross-lingual transfer, data augmentation, and monolingual pre-training. These approaches are potentially complementary. Our work belongs to the monolingual pre-training category.

Cross-lingual transfer addresses the low-resource issue by using data from different language pairs. One can use a richer language pair [273] or several language

pairs simultaneously [44, 48]. Murthy et al. [135] also proposed reordering the assisting languages to be similar to a low-resource language.

Data augmentation involves the creation of synthetic bilingual data from monolingual data. In the popular back-translation approach [51, 72, 189], the source side of the data is synthesized using an MT system to back-translate the target side data. Recently, Zhou et al. [268] proposed the creation of this source side through rule-based reordering via word-to-word translation.

In monolingual pre-training approaches, all or part of a model is first trained on tasks that require monolingual data.[6] Pre-training has enjoyed significant success in other NLP tasks with the development of GPT [166], BERT [46], and several others [153, 206, 248, 197, 202, 203, 198].

Pre-training schemes such as BERT were designed for natural language understanding (NLU) tasks and they are not directly suitable for NMT. Conneau and Lample [40] and Ren et al. [174] proposed multilingual variants. However, they trained the encoder and decoder independently. To address this, Song et al. [204] recently proposed MASS, a new state-of-the-art NMT pre-training task that jointly trains the encoder and decoder. Our approach develops on the initial idea of MASS, but adds more diverse and linguistically-motivated training objectives.

Linguistic information is known to be useful for NMT [187], especially in low-resource scenarios. Outside of pre-training, studies [135, 262, 268] have successfully used a linguistically-motivated reordering similar to that of our BRSS task. Sun et al. [206] used linguistically-motivated pre-training tasks for text understanding. To the best of our knowledge, there are no studies on linguistically-motivated pre-training tasks for NMT.

4.1.2 Pre-training Tasks for Neural Machine Translation

After the appearance of BERT [46], several pre-training methods have been proposed to enhance NMT [40, 100, 105, 110, 167, 174, 194, 201, 204, 227, 233, 249]. Particularly, Song et al. [204] proposed a random span reconstruction task to

[6] This is an instance of "transfer learning," similar to cross-lingual transfer. "Pre-training" often implies that the training task differs from the target task.

pre-train a sequence-to-sequence framework for NMT; Wang et al. [227] first proposed using shuffling, deleting, and replacing operations to implement the denoising pre-training for the NMT system; thereafter, Lewis et al. [100] combined the denoising methods with the masked language model pre-training of Song et al. [204], and provided detailed empirical results for a large number of language pairs; mBART [110] is a multilingual sequence-to-sequence denoising pre-training that is pre-trained through denoising tasks on 25 languages including Japanese, English, Chinese, Russian, and others, and it can be deemed as an extension of Lewis et al. [100]; other studies focus on leveraging the cross-lingual supervision between languages through word alignment [105], phrase alignment [174], sentence-level alignment [40], code-switching technique [249], or assisting languages (shared scripts) [201].

Among the above-mentioned pre-training techniques for NMT, we observe that no study has focused on leveraging specific linguistic features for NMT. Syntactic span-masking [269] and semantic-aware BERT [267] have been proposed using linguistically-driven supervision for language understanding tasks. However, linguistically-driven methods for sequence-to-sequence pre-training should be considered and explored.

Studies have also focused on improving MASS. Siddhant et al. [194] adapted MASS in multilingual scenarios; Qi et al. [164] proposed using an n-stream self-attention mechanism to enhance MASS for language generation tasks. No previous study has attempted to enhance MASS from a linguistic perspective, which will be explored in our study.

Moreover, Wang et al. [233] highlighted that multitask learning can significantly benefit multilingual NMT. In addition to the MT task, the essential jointly-learned tasks should be masked langauge model task and denoising (reconstruction) task, which are two basic pre-training styles based on which we propose our linguistically-driven methods.

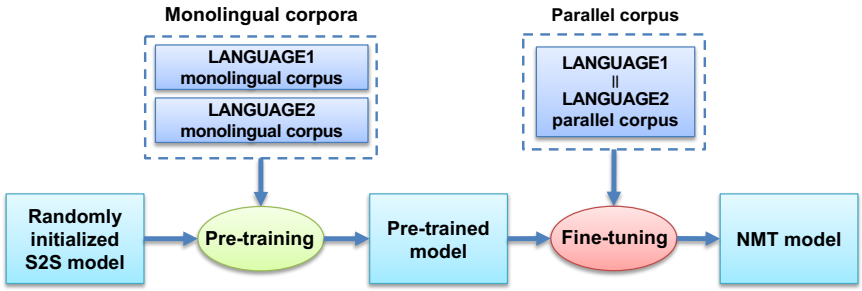

Figure 4.1: **Pre-training and fine-tuning for NMT.** "S2S" denotes sequence-to-sequence.

4.2 Preliminary Backgrounds

In this section, we introduce the preliminary backgrounds of pre-training and fine-tuning for NMT and MASS, which serve as the backbone for this study.

4.2.1 Pre-training and Fine-tuning for NMT

We first introduce the pre-training and fine-tuning pipelines for the NMT. As shown in Figure 4.1 below, we first utilize monolingual corpora to pre-train the initialized sequence-to-sequence model. Subsequently, we use a parallel corpus of languages of interest to fine-tune the pre-trained models. The fine-tuned model was the final NMT model. All the experiments in this study will be conducted on the basis of this pre-training and fine-tuning pipeline for NMT.

4.2.2 MASS

MASS is a pre-training method for NMT proposed by Song et al. [204]. As shown in Figure 4.2, in MASS pre-training, the input is a sequence of tokens where a part of the sequence is masked and the output is a sequence where the masking is inverted.

We consider $x \in \mathcal{X}$, which is a sequence of tokens where $\mathcal{X}$ is a monolingual corpus. Additionally, we consider the token span $C = [p_i, p_j]$, where $0 < p_i \leq p_j \leq len(x)$ and $len(x)$ are the number of tokens in sentence x. We denote the

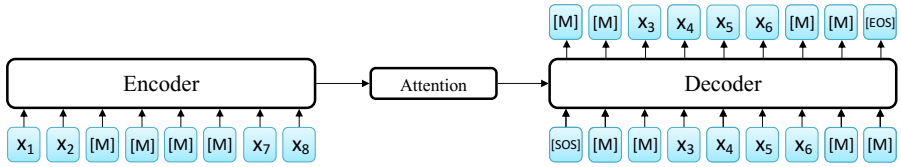

Figure 4.2: **Sequence-to-sequence structure for MASS.** x_i represents a token and x_3 to x_6 are consecutive tokens to be masked/predicted.

masked sequence by x^C, where tokens in positions from p_i to p_j in x are replaced by a mask token $[M]$. $x^{!C}$ is the sequence with an inverted mask, that is, where tokens in positions other than the aforementioned fragments are replaced by the mask token $[M]$. In MASS, the pre-training objective is to predict the masked fragments in x using an encoder-decoder model, where x^C is the input to the encoder and $x^{!C}$ is the target output of the decoder. The log-likelihood objective function is

$$\mathcal{L}_{mass}(\mathcal{X}) = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \log P\left(x^{!C} | x^C, \boldsymbol{\theta}\right). \tag{4.1}$$

where $\boldsymbol{\theta}$ denotes the model parameters. The number of tokens to be masked is a hyperparameter of the MASS. The NMT model is jointly pre-trained with the MASS task for both the source and target languages.

4.3 Proposed Methods

In this section, we describe JASS and ENSS, which are our proposed pre-training techniques.

4.3.1 Proposed Methods for Japanese

Our methods are based on the ideas of the original MASS and are improved by jointly learning multiple linguistics-aware tasks. For Japanese, we propose a bunsetsu-based MASS (BMASS) pre-training and bunsetsu reordering-based sequence-to-sequence (BRSS) pre-training. Their combination, Japanese-specific sequence-to-sequence (JASS) pre-training, is introduced in the following section.

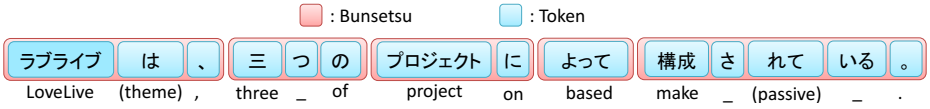

Figure 4.3: Word and bunsetsu segmentations for a Japanese sentence with meaning "LoveLive is made of three projects." In word for word English translations, "_" represents words with no meaningful translations.

Bunsetsu

Bunsetsu is the syntactic component of Japanese sentences [95, 130]. It is equivalent to the concepts of noun phrases or verb phrases in English syntax and it constitutes a minimal unit of meaning. The concept of "word" is ambiguous for writing systems such as Japanese where word-separators are not applicable, and Japanese segmenters [95, 130] can segment Japanese sentences either in words or bunsetsus. Therefore, bunsetsu is also more likely to correspond to a well-defined entity or concept than words. Figure 4.3 illustrates the difference between the word- and bunsetsu-level segmentation. Each bunsetsu contains self-contained information and case markers, which indicate its relation with other bunsetsus. Based on the bunsetsu, we introduce our proposed pre-training techniques for the Japanese.

BMASS

We propose BMASS, which leverages syntactically parsed Japanese monolingual data for sequence-to-sequence pre-training. MASS pre-trains an NMT model by making it predict random parts of a sentence given their context, whereas BMASS involves making the model predict a set of bunsetsus given the contextual bunsetsus. We expect this will allow the model to learn about bunsetsus and thereby focus on predicting meaningful subsequences instead of random, albeit fluent subsequences.

To perform BMASS, we modify the definition of mask C in Equation 4.1: $C = [[p_{i_1}, p_{j_1}], [p_{i_2}, p_{j_2}], ... [p_{i_n}, p_{j_n}]]$, where $0 < p_{i_1} \leq p_{j_1} \leq p_{i_2} \leq p_{j_2} \leq ...p_{i_n} \leq p_{j_n} \leq len(x)$. Term $len(x)$ denotes the number of tokens in sentence x. Subsequently, the $k-th$ position span from p_{i_k} to p_{j_k} corresponds to the start and end of a

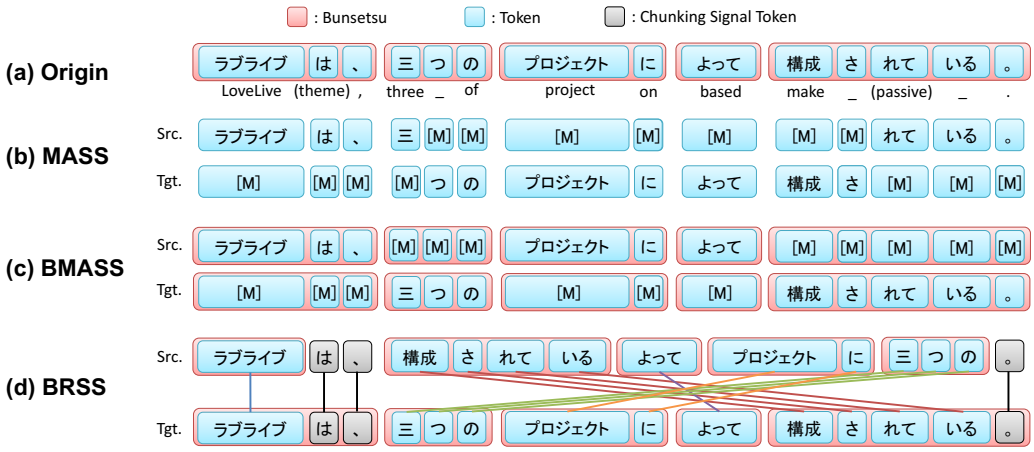

Figure 4.4: Example of source and target for MASS, BMASS, and BRSS with the meaning "LoveLive is made of three projects."

specific bunsetsu in a Japanese sentence. Consequently, we denote the BMASS loss as $\mathcal{L}_{bmass}$. The main difference between MASS and BMASS is that in MASS, we mask random token spans, whereas in BMASS, we only mask tokens spans that are complete bunsetsus. The number of bunsetsus to be masked constitutes a hyperparameter for BMASS. Figures 4.4-b and 4.4-c provide training pairs for MASS and BMASS.

Note that our BMASS pre-training task differs from the entity masking task of ERNIE [206] and random span masking of SpanBERT [82]. ERNIE and SpanBERT have been proposed without using syntactic units and they are employed in natural language understanding downstream tasks.

BRSS

Japanese sentences are typically in an SOV word order that can be reordered to SVO to reduce the difficulty of translation to languages with SVO order. We first define a simple process for reordering a (typically SOV) Japanese sentence into a "SVO Japanese" pseudo-sentence that will be used in BRSS. There are several previous studies on reordering a SOV-ordered sentence to a SVO-ordered sentence [74, 91]. In our case, to consistently leverage bunsetsu units in Japanese

with BMASS, we propose bunsetsu-based reordering, which is able to generate an SVO-ordered Japanese sentence while retaining syntactic information at the bunsetsu-level. We first define "chunking signal words" as any punctuation mark or the topic marker "は." The reordering process is as follows:

1. split the sentence into bunsetsus
2. select sequences of bunsetsus bounded by chunking signal words
3. simply reverse the order of the bunsetsus in these sequences without using rules

We can now propose BRSS, which involves a Japanese sentence and its reordered version obtained using the aforementioned procedure. Refer to Figure 4.4-d as an example of a bunsetsu-reordered sentence. The pre-training objective was a reordering task. We expect that this will allow the system to learn the structure of the Japanese language, and prepare it for the reordering operation it will have to perform when translating to a language with different grammar. Although BRSS task is constructed by simple rules, the predictions for the bunsetsu boundaries and orders are expected to equip the model with abundant linguistic knowledge. We have two choices from which we can make the NMT system predict the original sentence given the reordered sentence (BRSS.F) or vice-versa (BRSS.R). We will experiment with both options.

4.3.2 Proposed Methods for English

Similar to the proposed methods for Japanese, we propose two linguistically-driven methods for English that are based on the MASS language model and reordering sequence-to-sequence language model, respectively. One is phrase structure-based MASS (PMASS), and the other method is head finalization-based sequence-to-sequence pre-training (HFSS). The combination of PMASS, HFSS, and ENSS is introduced in the next section. Before introducing our proposed methods for English, we first provide background information on head-driven phrase structure grammar and head finalization, which forms our linguistically-driven methods.

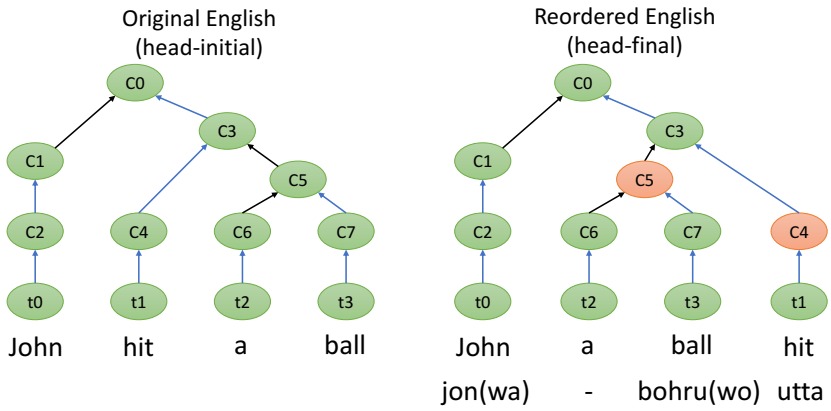

Figure 4.5: Example of HPSG parsing result and head finalization. Head finalization [77] reorders an English sentence into a Japanese-like sentence. Blue arrows denote the "head."

Head-driven Phrase Structure Grammar

As opposed to dependency-based grammar, head-driven phrase structure grammar (HPSG) [158, 159] is lexicalism-based grammar that focuses on generalizing phrase structures. HPSG primarily handles word and phrase signs in a sentence in terms of their syntactic and semantic roles. Thus, HPSG should be an appropriate parsing rule for extracting phrase structures in sentences and applying the following proposed pre-training techniques. Figure 4.5 (left) shows an instance of parsing an English sentence using HPSG grammar.

Head Finalization

Using the above-mentioned HPSG, sentences in any language can be characterized using phrase structures. From the definition of a phrase, the "head" of a phrase is subsequently defined as the syntactically determinant part in a phrase. In other words, "head" determines the syntactic category of the phrase and its "dependents." Particularly, English is referred to as a "head-initial" language because "head" appears before its "dependents," whereas Japanese is referred to as a "head-final" language because "head" usually appears after "dependents" in

a phrase.

The deliberate phrase structures provided by the HPSG parser are utilized in several scenarios in the NLP. Particularly, Isozaki et al. [77] proposed a simple reordering rule for the SVO language (head-initial languages) by using the phrase structure information provided by the HPSG parser. Figure 4.5 shows an example of reordering an English sentence to be an SOV-like sentence on the basis of the result of HPSG parsing. By reordering sentences in SVO languages such as English to be SOV-like sentences, the performance of statistical machine translation (SMT) is improved. Particularly, Isozaki et al. [77] first proposed head finalization and applied it to English-to-Japanese SMT; Han et al. [68] applied it to Chinese-to-Japanese SMT and obtained significant improvements; more recently, Zhou et al. [268] utilized this reordering technique to generate synthetic parallel sentences in the back-translation phase when translating SOV and SVO languages. In this study, we utilize this reordering rule in the pre-training phase for NMT (see Section 4.3.2).

PMASS

We propose PMASS by leveraging phrase-span information in an English sentence. In general, we perform PMASS pre-training by limiting the masked tokens in MASS to be an entire phrase span. Thus, for masking plural phrase spans, we denote it as PMASS.P. For masking only a single phrase span, we denote it as PMASS.S. Particularly, the source and target for PMASS.P and PMASS.S pre-training can be generated using our proposed phrase-masking algorithms described in Appendix B.1. Inspired by MASS, we force the number of masked tokens to be approximately half of the length of the sentence to guarantee the effectiveness of the sequence-to-sequence masked language model. Examples of PMASS.P and PMASS.S are presented in Figure 4.6-c. We observe that several phrase spans in PMASS.P and a single long phrase span in PMASS.S are masked. We expect such special masking patterns to force the NMT system to extract more phrase-level syntactic information in the pre-training phase.

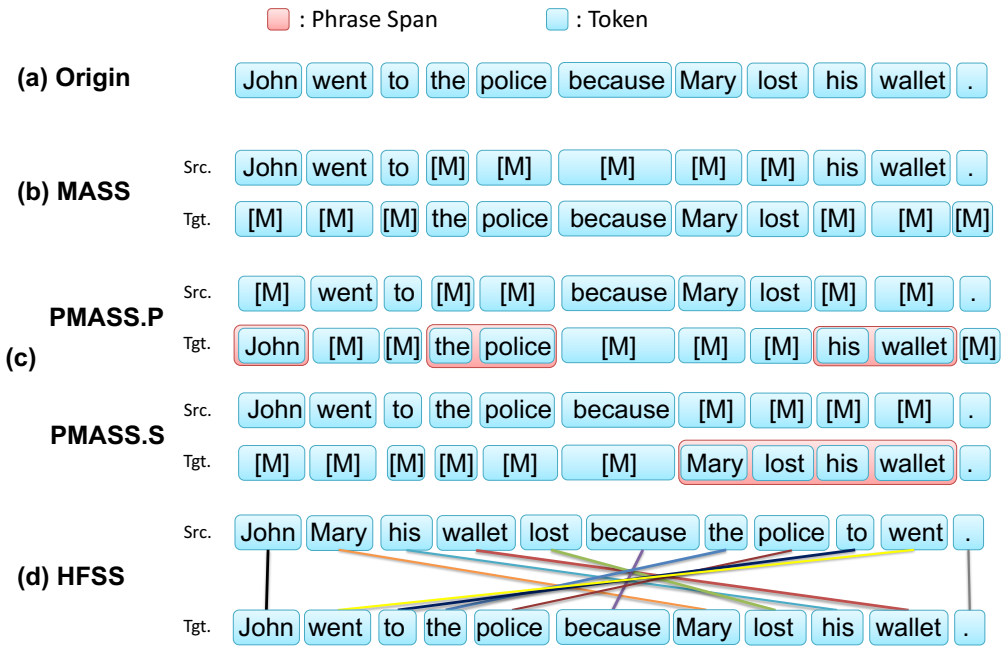

Figure 4.6: Example of source and target for MASS, PMASS, and HFSS of a sentence in English.

HFSS

We propose HFSS using the head finalization technique [77] for pre-training English. As shown in Figure 4.6-d, the pre-training task is also a reordering task that simulates the translation from SOV languages to English. More precisely, the source sentence for sequence-to-sequence pre-training is the reordered (SOV-like or head-finalized) English sentence, and the target sentence is the original English monolingual sentence. We expect HFSS to help the system learn the word reordering pattern of the translation between head-initial (SVO) and head-final (SOV) languages in advance.

According to the prior experiments for Japanese (see [118] and 4.5.1), BRSS.F consistently outperforms BRSS.R. In addition, BART [100] also claims that reconstructing the original sentence benefits the language generation tasks. Therefore, we do not distinguish HFSS with HFSS.F and HFSS.R (HFSS.F performs

pre-training with the SOV-SVO pattern, whereas HFSS.R performs the reverse pattern).[7] Instead, we directly defined HFSS using the pre-training pattern of HFSS.F. Moreover, HFSS is performed on the basis of head finalization, which utilizes the results from HPSG parsers. This is consistent with PMASS in which we extract phrases using HPSG-parsing results.

We develop our proposal on English through head finalization, whereas for SOV languages such as Japanese, it is unmanageable to reorder SOV sentences to SVO-like sentences [77]. Furthermore, HFSS can be used for all head-initial languages apart from English, as well-developed reordering rules have been proposed and demonstrated to be effective for NMT. However, BRSS can only be implemented for Japanese-involved translation pairs because bunsetsu information is required to establish the source and target sentences for sequence-to-sequence pre-training.

4.3.3 Multi-task Pre-training

Multi-task pre-training objectives lead to a robust initial state for NMT systems [100, 167]. Because our proposed methods can also be categorized into two groups of pre-training tasks, we propose a multi-task pre-training task for both Japanese and English.

We define JASS pre-training, which is a combination of the two previous procedures: BMASS and BRSS. Our actual pre-training will consist of the joint execution of these two pre-training sessions. Therefore, the pre-training objective for JASS is

$$\mathcal{L}_{jass}(\mathcal{X}_{ja}) = \mathcal{L}_{bmass}(\mathcal{X}_{ja}) + \mathcal{L}_{brss}(\mathcal{X}_{ja}) \tag{4.2}$$

where $\mathcal{X}_{ja}$ represents the monolingual corpus of Japanese, and $\mathcal{L}_{brss}$ denotes the reordering loss using the forward or reverse variants mentioned in Section 4.3.1. We expect BMASS & BRSS to jointly learn syntactic knowledge and BRSS to learn word ordering knowledge.

[7] More precisely, HFSS.F denotes the source sentence of the head-finalized English sentence and the target sentence of the original English sentence. HFSS.R indicates the source sentence of the original English sentence and the target sentence of the head-finalized English sentence.

For English, we similarly define ENSS pre-training, which combines PMASS and HFSS. More precisely, the training objective is:

$$\mathcal{L}_{enss}(\mathcal{X}_{en}) = \mathcal{L}_{pmass}(\mathcal{X}_{en}) + \mathcal{L}_{hfss}(\mathcal{X}_{en}) \tag{4.3}$$

where $\mathcal{X}_{en}$ denotes the monolingual corpus of English, $\mathcal{L}_{pmass}$ the PMASS.P or PMASS.S loss, and $\mathcal{L}_{hfss}$ the reordering loss of HFSS.

JASS is specifically designed for Japanese, whereas theoretically, ENSS can be transplanted onto any SVO language as long as we can extract the phrase structure information of the corresponding language from a HPSG parser.

We also mixed JASS pre-training for Japanese with MASS pre-training for the other languages involved in the translation. In practice, we therefore designated using JASS pre-training for Japanese monolingual data with BMASS and BRSS objectives, along with "other languages" monolingual data with the MASS objective. Similarly, for English, ENSS pre-training consists of PMASS & HFSS for English and MASS for "other languages" involved in fine-tuning translation pair.

We also consider attempting the combination of our proposed linguistically-driven methods with a strong baseline pre-training objective, MASS, which we refer to as MASS + JASS (or ENSS) in the subsequent sections. To allow the pre-training model to determine the language and sub-task (MASS, BMASS, BRSS, PMASS, and HFSS) that it should perform, we prepend tags to inputs similar to those used in [81] (see Section 4.4.2 for details).

4.4 Experimental Settings

In this section, we evaluate our pre-training methods on simulated low-resource scenarios for ASPEC Japanese–English [140], Japanese–Chinese translations [138, 124], and realistic low-resource scenarios for Wikipedia Japanese–Chinese [34, 35] and News English–Korean [151] translations.

4.4.1 Datasets

We used monolingual data for pre-training and parallel data for fine-tuning. Refer to Table 4.1 for an overview.

	Language	Dataset	Size
Monolingual	Ja	Common Crawl	22M
	Zh	Common Crawl	22M
	En	Common Crawl	22M
	Ko	Common Crawl	22M
Parallel	Ja-En	ASPEC-JE	1M
	Ja-Zh	ASPEC-JC	670k
	Ja-Zh	Wikipedia	258k
	En-Ko	News	94k

Table 4.1: Overview of training data. "Size" denotes the number of the monolingual sentences or parallel sentences.

Monolingual data: For pre-training, we use monolingual data of 22M lines each for Japanese, English, Chinese, and Korean, randomly sub-sampled from Common Crawl mentioned in the official WMT monolingual training data.[8] [9] For pre-training in Japanese–English and English–Korean, given that these two languages have different scripts and thus have few common words, the pre-training objectives for each language will work separately, even though they are performed jointly for two languages. However, for pre-training in Japanese and Chinese, they share more characters, which indicates that the monolingual pre-training tasks will be run in a pseudo-cross-lingual manner. Thus, we also expect to see whether such pre-training will benefit from more fine-tuning.

Parallel Data: We use scientific abstracts domain ASPEC parallel corpus for training Japanese–English and Japanese–Chinese models. For Japanese–Chinese fine-tuning, we also utilize the Wikipedia parallel corpus, which is a real low-resource scenario. We use News parallel corpus for English–Korean, which is a low-resource dataset.

[8] `http://www.statmt.org/wmt19/translation-task.html`

[9] Different from Mao et al. [118]. Currently, we unify the monolingual corpus domains for all the languages for fairer comparisons.

For ASPEC, we used the official training, development, and test splits provided by WAT 2019.[10] [11] For Wikipedia, we used the dataset released by Kyoto University.[12] For News, we use dataset provided by Park et al. [151].[13]

4.4.2 Pre-processing

We tokenize the monolingual data by using the Moses tokenizer for English and Korean,[14] Jumanpp for Japanese,[15] and jieba for Chinese.[16] We obtain the bunsetsu information by using KNP[17] and obtain the HPSG parsing results using enju.[18] Sentences with more than 175 tokens were removed. For each language pair, we constructed a joint vocabulary with 60,000 sub-word units through byte-pair encoding (BPE) [190] on the concatenated monolingual corpora involved during pre-training.[19], whereas 40,000 BPE merge operations is set for Japanese-English. In the multi-task pre-training, each sentence is prepended with a task token $[MASS]$, $[BMASS]$, $[BRSS]$, $[PMASS]$, or $[HFSS]$, and a language token $[Ja]$, $[En]$, $[Zh]$, or $[Ko]$.[20] This ensures that the model learns to distinguish between different pre-training objectives and languages. This token can be used when monolingual pre-training is conducted jointly by multiple languages and multiple tasks.

[10] `http://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2019/index.html\#task.html`

[11] For ASPEC Japanese–English, we use the first 1M parallel sentences. Parallel sentences for different fine-tuning size settings were randomly sampled from the selected 1M dataset.

[12] `http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?Wikipedia\%20Chinese-Japanese\%20Parallel\%20Corpus`

[13] `https://sites.google.com/site/koreanparalleldata`

[14] `https://github.com/moses-smt/mosesdecoder`

[15] `https://github.com/ku-nlp/jumanpp`

[16] `https://github.com/fxsjy/jieba`

[17] `https://github.com/ku-nlp/pyknp`

[18] `https://mynlp.is.s.u-tokyo.ac.jp/enju/`

[19] Particularly, 30,000 BPE merging operations will lead to a joint vocabulary with a size of approximately 60,000 for Japanese–Chinese and English-Korean

[20] As an implementation trick, we recommend to unify the task tag for the same group of tasks, e.g. use the same tag for $[BRSS]$ and $[HFSS]$.

4.4.3 Training and Evaluation Details

In our experiments, we used the open-source OpenNMT [87] implementation of the Transformer [219] NMT model.[21] The hyperparameters are set to the Transformer-big setting in OpenNMT. Particularly, our model has a 6-layer encoder and decoder, a hidden size of 1024, feed-forward hidden layer size of 4096, batch size of 4096, dropout rate of 0.3, and 16 attention heads. An ADAM optimizer with a learning rate of 10^{-4} was used for both pre-training and fine-tuning. All the pre-training tasks are run until convergence on four TITAN V100 GPU cards occurs, and fine-tuning uses only one GPU. It took approximately two days for each pre-training run. Mixed precision training [127] was used for both pre-training and fine-tuning. For multi-task pre-training, data are randomly shuffled such that even in each mini-batch, different pre-training objectives appear, corresponding to a real joint pre-training. Our proposed pre-training methods converge within the similar training time as compared to that of MASS.

Pre-training tasks are evaluated using perplexity, and the checkpoint with the lowest pre-training perplexity was selected for fine-tuning. We used BLEU [150] for automatic evaluation, adequacy, and fluency for human evaluation. We performed early stopping using 1-gram accuracy and perplexity on the development set. We evaluated the statistical significance of our BLEU scores through bootstrap resampling [88].

4.4.4 Baselines

In addition to MASS, we employ the "text infilling" in BART as another main baseline.[22] We also define two pre-training baselines for comparison with our proposed methods. They are named multi-span-based MASS (MultiMASS) and deshuffling. Moreover, the joint training with MASS and deshuffling was set as the multi-task pre-training baseline. All of the baselines are as follows:

Baselines without pre-training. First, we employ the vanilla Transformer big as the baseline without pre-training because all of the pre-training methods are

[21] `https://github.com/OpenNMT/OpenNMT-py`

[22] Text infilling has been demonstrated as the most effective pre-training objective for NMT among several objectives in BART [100].

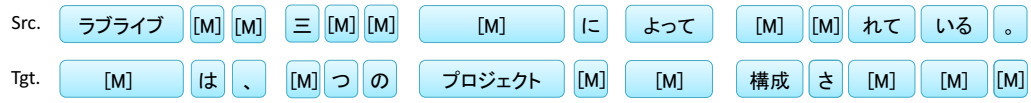

Figure 4.7: Example of source and target for MultiMASS with the meaning "Love-Live is made of three projects."

based on this model structure. Moreover, following Araabi and Monz [6], we also present the best performance for low-resource NMT by using Transformer model. Hyperparameter details are shown in Appendix B.2.

MASS. Using the same settings as in Song et al. [204].

BART (text infilling). Different from MASS, BART (text infilling) masks several token spans within a sentence by a single $[M]$ where span lengths are samples from Poisson distribution and the model is also required to predict the lengths of the masked spans. We use the same settings as in Lewis et al. [100].[23]

MultiMASS. MultiMASS is a baseline method added to help demonstrate the effectiveness of masking specific syntactic units such as bunsetsu or phrase spans in a sentence that we propose as BMASS and PMASS.

As shown in Figure 4.7, MultiMASS predicts several randomly masked tokens in a sentence, which differs from the single masked span in MASS, masked busetsu spans in BMASS, several phrase spans in PMASS.P, and a single phrase span in PMASS.S.

Deshuffling. Deshuffling denotes the pre-training task of random shuffling-based sentence reconstruction, which is also a crucial pre-training task. We perform this pre-training task as another baseline to confirm the effectiveness of reordering syntactic units in BRSS and the reordering driven by head finalization of HFSS. A pre-training example is presented in Figure 4.8.

[23]In order to conduct fair comparisons for our proposed methods, we only present the most effective sub-task, text infilling, within BART. The combination of text-infilling and sentence permutation is proven to be the best practice of BART. With regard to sentence permutation, we do not consider it in this study because it is mainly designed for document NMT. When it comes to multi-sentence pre-training, sentence permutation and other possible patterns of multi-sentence linguistically-driven pre-training tasks should be explored and compared in future work.

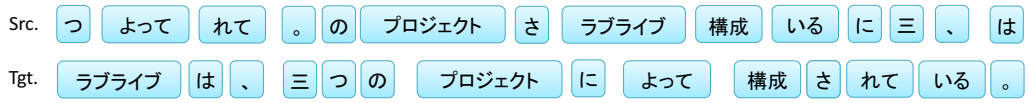

Figure 4.8: Example of source and target for deshuffling with the meaning "Love-Live is made of three projects."

Multi-task Baseline. The multi-task baseline is the combination of the respective best baseline methods from the masked language model and reordering pre-training. Thus, the multi-task baseline consists of MASS,[24] and deshuffling. The baseline is formulated as follows:

$$\mathcal{L}(\mathcal{X}) = \mathcal{L}_{mass}(\mathcal{X}) + \mathcal{L}_{deshuffling}(\mathcal{X}) \tag{4.4}$$

where $\mathcal{X}$ represents the monolingual corpora.

4.4.5 Pre-trained Models

We pre-trained our NMT models by leveraging the monolingual data of the source and target languages. For Japanese, we can use MASS, BMASS, or BRSS, whereas for English, we can use MASS, PMASS, or HFSS. For Chinese and Korean, we use only the MASS. Particularly, we pre-trained different types of models in Table 4.2. Note that we use MASS for ENSS because PMASS underperforms MASS by a significant margin (see 4.5.1).

4.4.6 Fine-tuned NMT Models

We fine-tuned to improve Japanese-English, English-Japanese, Japanese-Chinese, Chinese-Japanese, English–Korean and Korean–English translations. We trained the following NMT models:

1. **Ja–En and En–Ja:** Japanese to English and English to Japanese models using from 3k to 50k parallel sentences randomly sampled from **ASPEC** for fine-tuning.

[24] MASS outperforms MultiMASS, we therefore use MASS rather than MultiMASS. (See 4.5.1)

#	Pre-trained Model	Details
Main baseline		
1	MASS	Using the same settings as in Song et al. [204].
1*	BART (text infilling)	Using the same settings as in Lewis et al. [100].
Proposed methods for Japanese		
2	BMASS	Similar to MASS, we mask half the number of bunsetsus during pre-training.
3	BRSS	We separately pre-trained on the SVO–SOV (BRSS.F) as well as SOV–SVO (BRSS.R) models.
4	JASS	Multi-task training of BMASS and BRSS.
Combinations of proposed methods with MASS		
5	MASS+BMASS	Multi-task training of MASS and BMASS.
6	MASS+BRSS	Multi-task training of MASS and BRSS.
7	MASS+BMASS+BRSS	Multi-task training of BMASS, BRSS and MASS.
Other baselines for Japanese		
8	MultiMASS (Ja)	Based on MASS pre-training, several random tokens are masked rather than one consecutive span.
9	Deshuffling (Ja)	Random shuffling-based original sentence reconstruction.
10	MASS+Deshuffling (Ja)	Multi-task pre-training baseline for Japanese.
Proposed methods for English		
11	PMASS	Similar to MASS, we mask an entire phrase span based on the head-driven phrase structure grammar. We performed the experiments for PMASS.P and PMASS.S, respectively.
12	HFSS	We train SOV (head finalized)—SVO (original) models for English.
13	ENSS	Multi-task training of MASS and HFSS.
Other baselines for English		
14	MultiMASS (En)	Based on the MASS, several random tokens are masked rather than one consecutive span.
15	Deshuffling (En)	Random shuffling-based original sentence reconstruction.
16	MASS+Deshuffling (En)	Multi-task pre-training baseline for English.
Combination of the proposed methods for English and Japanese		
17	JASS+ENSS	Multi-task training of JASS and ENSS.
Baseline for #17		
18	MASS+Deshuffling	Multi-task pre-training baseline for JASS+ENSS.

Table 4.2: Settings of pre-trained models.

2. **Ja–Zh and Zh–Ja:** Japanese to Chinese and Chinese to Japanese models using from 3k to 50k parallel sentences randomly sampled from **ASPEC**

and **Wikipedia**, respectively, for fine-tuning.

3. **En-Ko and Ko-En:** English to Korean and Korean to English models using 20k (randomly sampled) and 94k (full dataset) parallel sentences from **News** for fine-tuning.

We compared these models with pre-trained model baselines and vanilla baselines, which are fully-supervised NMT models on the same data settings, but without pre-training. In addition, fine-tuning results under the high-resource scenarios (with more than 50k parallel sentences) are provided and discussed in 4.5.6.

4.5 Results and Analyses

Tables 4.3, 4.4, 4.5, and 4.6 contain the NMT BLEU results of our proposed methods for Japanese–English, Japanese–Chinese and English–Korean translation on various translation domains, respectively. Subsequently, we provide in-depth analysis for translation quality in terms of adequacy by using LASER [12], human evaluation scores, specific cases for the real low-resource scenario of Wikipedia Ja-Zh. Finally, we conduct an investigation on the pre-training accuracy to analyze the difference between the pre-trained models and their complementation of each other, and present the results in middle/high-resource scenarios.

4.5.1 NMT Results

In Tables 4.3 and 4.4, where we simulate several low-resource settings for Japanese–English and Japanese–Chinese translations on ASPEC with different pre-training datasets; in Table 4.5 and 4.6, where we use realistic low-resource settings for Wikipedia Japanese–Chinese translation and News English–Korean translation, we observe that all settings using pre-training outperform those without pre-training (#0 & #0*), which indicates the importance of pre-training. The results also indicate that JASS (#4) and ENSS (#13) are generally better than MASS (#1). With regard to two main baselines with pre-training, MASS and BART (text infilling), we observe that MASS outperforms BART (text infilling) in most cases as shown in Table 4.3, 4.4, 4.5. So we focus on the comparisons with MASS

#	Model	Ja-En				En-Ja			
		3k	10k	20k	50k	3k	10k	20k	50k
Main baselines									
0	w/o pre, vanilla	0.8	2.1	3.5	16.1	1.1	2.7	5.1	19.4
0*	w/o pre, optimized	2.2	6.8	10.7	19.8	3.3	6.5	13.6	23.7
1	MASS	8.8	13.8	17.2	21.2	9.1	16.0	20.6	25.0
1*	BART (text infilling)	3.1	11.1	15.5	20.7	5.6	14.9	19.8	25.6†
Proposed methods for Japanese									
2	BMASS	8.9	13.9	17.4	21.8	8.7	15.9	20.1	25.4
3	BRSS	8.8	14.9†	18.1†	22.0†	10.0†	17.3†	21.0	26.0†
3 (R)	BRSS.R	8.2	14.3†	17.7†	21.7†	10.0†	17.2†	20.5	25.7†
4	JASS	10.6†	15.7†	**18.9**†	**22.3**†	**11.5**†	17.7†	21.6†	26.5†
Combinations of proposed methods with MASS									
5	1 + 2	9.2	14.8†	17.7†	21.7†	9.7†	16.6†	20.9	25.9†
6	1 + 3	**10.9**†	15.9†	18.3†	22.2†	11.0†	17.7†	**21.7**†	**26.8**†
7	1 + 4	10.5†	15.5†	18.5†	22.0†	**11.5**†	**17.9**†	**21.7**†	26.4†
Other Baselines for Japanese									
8	MultiMASS (Ja)	7.1	12.1	15.1	20.5	6.9	13.0	17.7	24.1
9	Deshuffling (Ja)	6.8	12.7	16.6	21.0	7.8	14.7	19.3	24.9
10	1 + 9	8.2	13.3	17.0	21.4	8.3	15.5	19.5	25.4
Proposed methods for English									
11	PMASS.P	6.8	12.1	15.9	20.7	5.5	13.5	17.8	24.5
11*	PMASS.S	6.5	12.3	16.2	21.2	6.2	13.5	18.2	24.6
12	HFSS	10.5†	**16.3**†	**18.9**†	**22.6**†	9.8†	17.8†	**21.7**†	**26.8**†
13	ENSS	**11.2**†	**16.7**†	**19.0**†	22.1†	**11.7**†	**18.7**†	**22.5**†	**27.0**†
Other baselines for English									
14	MultiMASS (En)	6.9	12.0	15.2	20.1	7.0	12.8	17.5	23.8
15	Deshuffling (En)	6.6	12.5	15.9	20.9	6.8	14.1	19.2	24.7
16	1 + 15	7.7	13.2	16.7	21.0	8.6	15.7	20.4	25.6
Combination of methods for Japanese and English									
17	4 + 13	**10.9**†	**16.4**†	18.7†	**22.3**†	**11.9**†	**18.4**†	**22.0**†	26.5†
18	10 + 16 (baseline)	7.2	12.6	16.4	20.9	8.4	14.8	19.1	25.5

Table 4.3: BLEU scores for simulated low/high-resource settings for Japanese–English ASPEC translation using from 3k to 50k parallel sentences for fine-tuning. Pre-trained models used for fine-tuning are numbered according to their description in Section 4.4.5. Results better than MASS with statistical significance $p < 0.05$ are marked in †. Bold denotes the three top scores.

in the following analyses. We also present the results by combing BART (text in-

#	Model	Ja-Zh				Zh-Ja			
		3k	10k	20k	50k	3k	10k	20k	50k
Main baselines									
0	w/o pre, vanilla	0.7	3.4	11.5	21.0	1.9	4.5	16.0	28.2
0*	w/o pre, optimized	3.7	12.0	19.5	23.3	6.7	15.8	24.8	31.2
1	MASS	15.7	20.3	22.4	24.7	19.4	25.9	29.4	32.9
1*	BART (text infilling)	13.5	19.0	21.3	24.4	20.3†	25.8	29.1	33.0
Proposed methods									
2	BMASS	16.7†	21.1†	23.0†	25.3†	20.9†	27.2†	30.2†	33.7†
3	BRSS	15.6	21.1†	22.6	24.9	20.7†	26.8†	30.0†	33.3†
4	JASS	**17.1**†	**22.2**†	**23.2**†	25.2†	21.6†	27.5†	**30.4**†	**33.6**†
Combinations of proposed methods with MASS									
7	1 + 4	17.0†	21.7†	23.1†	**25.4**†	**21.8**†	**27.6**†	30.2†	33.4†
Other baselines									
8	MultiMASS	14.5	20.5	22.3	24.7	19.6	25.7	29.8	33.2
9	Deshuffling	14.1	19.5	21.6	24.3	18.4	25.0	28.7	32.8
10	1 + 9	15.0	20.2	22.1	25.0	18.9	25.9	29.3	33.1

Table 4.4: BLEU scores for simulated low-resource settings for Japanese–Chinese ASPEC translation using 3k to 50k parallel sentences for fine-tuning. Results better than MASS with statistical significance $p < 0.05$ are marked in †.

filling) with ours in Appendix B.3.[25] Without pre-training, we observe that using optimized Transformer (#0*) benefits the low-resource setting, which has been proven by previous work [6, 191]. However, pre-training can further improve the optimized baselines without pre-training.

Particularly, for the Japanese–English translation, BMASS (#2) is comparable to MASS; BRSS (#3 & #3(R)) and their combination, along with JASS (#5) are significantly better than MASS. However, as summarized in Tables 4.4 and 4.5, the results for two parallel corpora on different domains for Japanese–Chinese

[25]Note that better results from BART than MASS in Lewis et al. [100] are based on the multi-task objectives while we are comparing with the most effective single task within BART here.

#	Model	Ja-Zh				Zh-Ja			
		3k	10k	20k	50k	3k	10k	20k	50k
Main baselines									
0	w/o pre, vanilla	0.9	2.9	2.9	6.0	1.6	2.9	3.9	6.5
0*	w/o pre, optimized	3.3	7.8	11.7	21.9	6.7	12.0	16.2	24.2
1	MASS	7.7	15.4	18.3	23.4	9.6	17.6	23.3	27.1
1*	BART (text infilling)	5.9	14.0	18.0	21.8	8.7	17.8	24.2†	28.5†
Proposed methods									
2	BMASS	10.8†	15.7	**20.1**†	24.5†	16.2†	19.4†	25.4†	**30.0**†
3	BRSS	11.6†	16.2†	20.0†	24.6†	15.7†	21.6†	25.0†	28.3†
4	JASS	**12.0**†	**17.0**†	**20.1**†	**25.0**†	**16.6**†	21.2†	**26.5**†	29.2†
Combinations of proposed methods with MASS									
7	1 + 4	11.8†	16.8†	**20.1**†	24.6†	**16.6**†	**22.3**†	25.5†	29.6†
Other baselines									
8	MultiMASS	8.2	13.8	18.6	21.5	10.7	17.3	22.0	26.4
9	Deshuffling	9.3	14.2	18.7	22.7	12.4	18.4	23.2	27.4
10	1 + 9	8.7	13.8	19.4	23.2	14.3	18.8	24.8	27.8

Table 4.5: BLEU scores for simulated low-resource settings for Japanese–Chinese Wikipedia translation using from 3k to 50k parallel sentences for fine-tuning. Results better than MASS with statistical significance $p < 0.05$ are marked in †.

yield significantly better results when using our proposed BMASS and BRSS. We observe that only a few settings on Japanese-to-Chinese BRSS yield lower BLEU results than MASS, whereas other settings using the proposed methods yield better results than MASS by significant margins. Although MASS is better than BMASS for Japanese–English translation, the reverse can be observed for the Japanese–Chinese translation. This indicates that the effects of the proposed span-masking techniques might correlate with specific translation directions and domains. We suppose it is worth exploring the span-masking tricks that are non-sensitive to language pairs and domains in the future.

As summarized in Table 4.3 and 4.6, our proposed methods of leveraging linguistic knowledge for English yield significantly higher BLEU results when

#	Model	En-Ko		Ko-En	
		20k	94k	20k	94k
Main baselines					
0	w/o pre-training	1.3	2.1	2.9	4.5
0*	optimized Transformer	2.1	3.7	3.9	8.3
1	MASS	2.9	4.5	5.6	9.6
Proposed methods					
2	PMASS	2.4	4.3	5.2	9.4
3	HFSS	3.1	4.8	**7.7**†	10.3†
4	ENSS (1 + 3)	**3.2**	**5.0**†	6.8†	**10.9**†
Other combinations					
	2 + 3	3.0	4.7	7.0†	10.6†

Table 4.6: BLEU scores for simulated low-resource settings for English–Korean News translation using 20k and 94k parallel sentences for fine-tuning. Results better than MASS with statistical significance $p < 0.05$ are marked in †. The BLEU scores are relatively low because English–Korean is a dissimilar language pair. Previous work [191, 151] reported similar BLEU results.

we perform the reordering pre-training task, HFSS (#12). However, the proposed linguistically-driven masked language modeling tasks PMASS.P (#11) and PMASS.S (#11*) yielded comparable results to several other baseline methods such as MultiMASS (#14) and deshuffling (#15). This demonstrates that the syntactical span-based masked language model may merely work on head-final languages such as Japanese.[26] Considering the weak performance of the PMASS, we combined HFSS with MASS for ENSS. The multi-task pre-trained ENSS yielded the highest results on almost all the low-resource settings. We will explore proper

[26]The weak performance of PMASS can also be attributed to the discrete nature of the remaining tokens (tokens that are not masked) without constituting complete semantic spans. We will attempt chunking-based masking for PMASS in future work to allow PMASS to be performed in a manner similar to BMASS.

chunking techniques for linguistically-driven span-masking pre-training for languages like English in the future.

However, in Table 4.3, when performing a universal linguistically-driven pre-training simultaneously for Japanese and English (#17), we did not achieve further significant BLEU improvements. This can be attributed to the increased dependence of NMT on specific linguistic information on a single language side, and the joint pre-training does not allow linguistic knowledge transfer across languages and between dissimilar languages.

In addition to the main baseline MASS, we also performed several other sequence-to-sequence pre-training baselines: MultiMASS (#8 & #14) and deshuffling (#9 & #15) along with their multi-task combinations (#10, #16 & #18) for Japanese and English. As summarized in Tables 4.3, 4.4, 4.5, and 4.6, we observe that the proposed masked style pre-training task, BMASS, and reordering pre-training tasks, BRSS & HFSS, outperform these baselines by significant margins, thereby indicating that linguistically-driven methods should be superior to self-supervised pre-training without leveraging linguistic features. Moreover, we investigated the percentages of the words of which the position changed. For Japanese pre-training, the percentages are 79.58% for BRSS and 94.72% for deshuffling. For English pre-training, the percentages are 91.97% for HFSS and 95.22% for deshuffling. Although there exists a gap for the percentages between BRSS and deshuffling, we can see that the percentages of deshuffling and HFSS are similar, which demonstrates that the quality of the linguistically generated reordered sentence is much more important than the percentage.

As summarized in Table 4.3, BRSS-F (English-order to Japanese-order) yielded slightly better results than BRSS-R (vice-versa); thus, we only experimented with BRSS-F for the remaining experiments. We suppose that the reason is that training the decoder with the original sentence is more important than training the encoder with it, which is also the reason why BART pre-training [100] treats the original sentence as the target sentence to be predicted from the decoder.[27] In other words, forcing the decoder to generate a natural sentence leads to a better initialized decoder for NMT. Meanwhile, HFSS pre-training is performed in an

[27]In BART, the original sentences without any noise are treated as target sentences.

#	Model	ASPEC		ASPEC		Wikipedia		News	
		Ja-En	En-Ja	Ja-Zh	Zh-Ja	Ja-Zh	Zh-Ja	En-Ko	Ko-En
*	Reference	80.78		86.10		87.26		73.93	
0	w/o pre-training	52.59	45.89	69.54	67.08	57.55	56.48	59.68	65.38
1	MASS	75.63	76.09	85.52	86.32	81.08	78.52	72.54	73.30
2	BMASS	75.75	76.68	85.42	86.49	80.91	81.36	-	-
3	BRSS	78.34	76.66	85.87	86.54	81.71	**84.29**	-	-
4	JASS	**80.00**	77.63	**85.96**	**86.58**	**85.39**	83.08	-	-
11	PMASS	76.08	73.67	-	-	-	-	71.90	74.14
12	HFSS	79.38	79.13	-	-	-	-	73.60	75.59
13	ENSS	79.79	**79.64**	-	-	-	-	**74.13**	**75.66**

Table 4.7: Adequacy scores evaluated by LASER embedding-based cosine similarity for ASPEC Japanese–English, Japanese–Chinese, Wikipedia Japanese–Chinese and News English–Korean translations, respectively, using 10k sentences for fine-tuning (using 94k sentences for English–Korean). Reference (*) is the cosine similarity between test sets in two languages.

analogous manner for the same reason.

As mentioned above, JASS yields the best results when we consider only linguistically driven methods for Japanese. After combining the proposed methods for Japanese with MASS (#5∼#7 in Table 4.3), we observe comparable results as compared to JASS by combing MASS and BRSS. This indicates the effects of combining masked style methods and reordering style methods. In Table 4.4 and 4.5, we believe that BMASS is better than MASS for combining with BRSS because of the significant improvements yielded by BMASS.

Moreover, as summarized in Tables 4.4 and 4.5, we observe that on the ASPEC domain, JASS improves up to 2.2 BLEU scores, whereas on the Wikipedia domain, JASS achieves up to 7.0 BLEU improvements. This demonstrates the promising performance of the proposed methods. Meanwhile, this indicates that the overlapping of pre-training domain with the fine-tuning domain is directly proportional to the realization of improvements by linguistically-driven pre-training

#	Model	BLEU		Adequacy		Fluency	
		Ja-Zh	Zh-Ja	Ja-Zh	Zh-Ja	Ja-Zh	Zh-Ja
0	w/o pre-training	2.9	2.9	1.22	1.05	3.90	3.99
1	MASS	15.4	17.6	2.72	2.33	4.11	4.09
2	BMASS	15.7	19.4	3.12	2.88	4.34	4.32
3	BRSS	16.2	**21.6**	3.30	3.35	4.30	**4.40**
4	JASS	**17.0**	21.2	**3.79**	**3.44**	**4.47**	4.36

Table 4.8: Adequacy and fluency of Wikipedia Japanese–Chinese translations using 10k sentences for fine-tuning.

methods.

Finally, by comparing the BLEU results in Table 4.3 with those reported by Mao et al. [118], we find that the BLEU scores of models pre-trained with News Crawl are better than those pre-trained with the Common Crawl monolingual corpus, which shows that pre-training with a high-quality monolingual dataset leads to superior fine-tuning results.

4.5.2 Adequacy Evaluation

Reference-free MT evaluation evaluates the translation system without using the target reference. Such an evaluation can help circumvent the noise existing in the references of translation targets. After the emergence of multilingual sentence encoders [12], Yankovskaya et al. [251] proposed the use of multilingual sentence embeddings encoded by LASER to implement the reference-free MT evaluation. More precisely, we first apply LASER to encode the source sentence and the translated sentence, respectively. Thereafter, the cosine value of those two embeddings is used to evaluate the similarity between the source and translation. This cosine value is thus the metric used to evaluate translation adequacy. This approach has two advantages. The first advantage is that target references are not required, as mentioned above. The other advantage is that every two translation directions

#	Model	BLEU		Adequacy		Fluency	
		Ja-En	En-Ja	Ja-En	En-Ja	Ja-En	En-Ja
0	w/o pre-training	2.1	2.7	1.08	1.08	2.56	3.60
1	MASS	13.8	16.0	2.61	3.03	3.40	4.17
2	PMASS	12.1	13.5	2.40	2.76	3.24	4.07
3	HFSS	16.3	17.8	3.24	4.00	3.60	4.31
4	ENSS	**16.7**	**18.7**	**3.72**	**4.11**	**3.76**	**4.42**

Table 4.9: Adequacy and fluency of ASPEC Japanese–English translations using 10k sentences for fine-tuning.

can be compared with each other because language-agnostic embedding is used for evaluation.

We report the adequacies in Table 4.7. First, we observe that methods with pre-training can yield more semantically correct translations than those without pre-training. Second, our proposed methods can significantly obtain higher LASER similarity scores than the MASS baseline, particularly the results on ASPEC Japanese–English, Wikipedia Chinese–Japanese and News English–Korean translations. Moreover, we can observe that the adequacy results obtained from the LASER embedding-based cosine similarity scores are consistent with the BLEU results.

4.5.3 Human Evaluation

Following Nakazawa et al. [139], we performed adequacy and fluency evaluations for the Japanese–Chinese and Japanese–English translations when 10k Wikipedia parallel sentences and 10k ASPEC parallel sentences were used for fine-tuning the pre-trained models. We randomly sampled 100 test-set English sentences and blindly evaluated their translations across various models. Each sentence was scored on a scale of 1 to 5, with 1 representing the worst score. The higher the score, the more adequate (meaningful) or fluent (well-formed) the sentence is. The

#		
	Reference–Ja	水の性質の多様性について, まず, 水分子同士の間に働く力である水素結合と, そのネットワーク構造について解説した。
	Reference–En	Various properties of water were explained on hydrogen bonds in which the force works among the water molecules and the network structure.
0	w/o pre-training	This study introduces the outline of the development of the system, and it is described.
1	MASS	The network structure of the water, hydrogen combination as the power of the water, and the network structure are explained.
2	BMASS	On the basis of the water properties, hydrogen coupling and the network structure are explained in the first stage of water.
3	BRSS	On the formation of the water, this study explains hydrogen bond and hydrogen bond, which is connected between the water vapor man fellows.
4	JASS	This study explains the development of the properties of water and hydrogen combination, which is the power between the moisture man fellows and the network structure.
8	MultiMASS (Ja)	This study explains the rich characteristics of the water and also explains the network structure of the hydrogen joining with the network structure.
9	Deshuffling (Ja)	This study explains the potential of the water in the water, and the network structure that is connected between the water and hydrogen joining.
10	Multi-task baseline (Ja)	The active properties of water are explained, and hydrogen combination that is connected to the network structure and the power of the water are explained.
11	PMASS	The importance of the properties of water and the network structure, which is the active component of the water, are explained.
12	HFSS	The formation of the properties of water is first explained, then hydrogen combination and the network structure between the moisture man.
13	ENSS	The importance of the property of the water is first explained: hydrogen combination and the network structure, which is the power for the entire body of the water.
14	MultiMASS (En)	The growth of the water is explained, and the network structure and structure are explained through the hydrogen combination and network structure.
15	Deshuffling (En)	The network structure of the water properties is explained, and the network structure with hydrogen in the water is described.
16	Multi-task baseline (En)	This study explains the growth of the water properties, and it also explains hydrogen bonding and its network structure with the ability to develop between the water molecules.

Table 4.10: Japanese–English translation examples fine-tuned using 10k ASPEC parallel sentences.

final score was the average of the scores of 100 sentences. We did not consider the references, but only considered the sources for our evaluation.

In Table 4.8 and 4.9, we can observe that NMT models, even without pre-training, are capable of generating rather fluent sentences, and the lack of parallel sentences (low-resource scenario) will mainly influence the translation adequacy (refer to the extremely low adequacy of models without pre-training). Meanwhile, we can observe that our proposed BMASS, BRSS, JASS, HFSS, and ENSS result in large improvements in adequacy and moderate improvements in fluency, for both translation directions, whereas PMASS yielded marginal improvements. The

#		
	Reference–Ja	翌嘉永２年（１８４９年）１１月２２日に婚姻し、御廉中様（将軍世子の正室）と称された。
	Reference–Zh	翌嘉永２年（１８４９年）１１月２２日成婚，被称作御廉中 样（将 军世子的正室）。 (He married on November 22nd in the second year of Kaei (1849), and was called Gorenju (Seishitsu of the Shogun's son).)
0	w/o pre-training	慶長２年（１５８３年)、劉宮将軍となり、幕府の命を率いた。 (He became 劉宮 Shogun in the second year of Keicho, and lived the life of Shogun.)
1	MASS	翌嘉永２年（１８４９年）５月２７日に成婚。 (He married on May 27th in the second year of Kaei (1849).)
2	BMASS	翌嘉永２年（１８４９年）１１月２３日に結婚。 (He married on November 23rd in the second year of Kaei (1849).)
3	BRSS	翌嘉永２年（１８４９年）１１月２３日に成婚し、御廉の中 样（将軍世子の正室）と呼ばれる。 (He married on November 23rd in the second year of Kaei (1849), and was called Naka of Goren (Seishitsu of the Shogun's son).)
4	JASS	翌嘉永２年（１８４９年）１１月２２日成婚し、御廉中様（将軍世子の正室）と称した。 (He married on November 22nd in the second year of Kaei (1849), and was called Gorenju (sSeishitsu of the Shogun's son).)
8	MultiMASS	翌嘉永２年（１８４９年）１月２９日に成婚した。 (He married on January 29th in the second year of Kaei (1849).)
9	Deshuffling	翌嘉永２年（１８４９年）１１月２３日、御廉中の正室（将軍世子の正室）と称された。 (He was called Seishitsu of Gorenju (Seishitsu of the Shogun's son) on November 22nd in the second year of Kaei (1849).)
10	Multi-task baseline	翌嘉永２年（１８４９年）１１月２２日成婚し、御廉の代わりに義子と呼ばれる。 (He married on November 22nd in the second year of Kaei (1849), and was called Gishi instead of Goren.)

Table 4.11: Japanese–Chinese translation examples fine-tuned using 10k Wikipedia parallel sentences. Sentences in brackets correspond to English sentences of the above Japanese translations.

improved performance of adequacy compared with that of MASS demonstrates the effectiveness of linguistically-driven pre-training methods. Moreover, we can observe that the results of human evaluation are almost consistent with those of BLEU.

4.5.4 Case Study

We conducted case studies on Japanese-to-English translation fine-tuned using 10k ASPEC parallel sentences and Chinese-to-Japanese translation fine-tuned using 10k Wikipedia parallel sentences to make improvements shown by BLEU score evaluations visible. As summarized in Tables 4.10 and 4.11, we find that the vanilla NMT system trained using 10k parallel sentences without pre-training can hardly implement the translation. With regard to models with pre-training, we observed that MASS and other baseline models generated several incorrect tokens in terms of semantics, whereas the entire sentence seemed fluent. However, our proposed methods can generate sentences with superior adequacy and fluency, where fewer missing keywords are observed.

#	Model	Overall	MASS	BMASS	BRSS
1	MASS	69.75	69.75	-	-
2	BMASS	77.32	-	77.32	-
3	BRSS	87.90	-	-	95.90
4	JASS	85.15	-	77.34	**97.89**
5	1 + 2	74.53	**70.17**	78.59	-
6	1 + 3	81.58	69.72	-	94.43
7	1 + 4	80.81	**70.22**	77.90	**97.73**

Table 4.12: Component-wise and overall pre-training accuracies on ASPEC Japanese development sentences. Column names "MASS," "BMASS," and "BRSS" denote the pre-training components in the respective model. Note the boost of BRSS accuracy in multitask settings, although the opposite could have been expected.

4.5.5 Pre-training Accuracy

Pre-training accuracy is the accuracy of the monolingual pre-training tasks, and it can be an indicator of task complexity and pre-training objective performance. Tables 4.12 and 4.13 summarize the component-wise and overall pre-training accuracies for various models, respectively, on the ASPEC Japanese and English development set sentences. Regarding individual component methods, it can be observed that MASS and PMASS are the harder tasks, given their low accuracy, whereas BRSS and HFSS are the easier tasks. Moreover, for Japanese, the accuracy of MASS and BRSS improves when coupled with BMASS, whereas for English, the accuracy of HFSS and MASS improves when they are combined with each other. Cross-referencing these accuracies with the BLEU scores in Table 4.3, we observe that an increase in BLEU scores has no significant relationship with the pre-training accuracy. However, masked language model-based pre-training methods (MASS & BMASS) seem to act as an accuracy improving catalyst for BRSS and HFSS, and this in turn has a positive impact on the translation quality.

#	Model	Overall	MASS	PMASS	HFSS
1	MASS	70.97	70.97	-	-
2	PMASS	71.04	-	71.04	-
3	HFSS	96.48	-	-	96.48
4	ENSS	84.97	**71.24**	-	**98.05**

Table 4.13: Component-wise and overall pre-training accuracies on ASPEC English development sentences. Column names "MASS," "PMASS," and "HFSS" denote the pre-training components in the respective model. Note the boost of the HFSS accuracy in multitask settings, although the opposite could have been expected.

One possible reason for this is that multi-task training of different pre-training methods helps boost the performance of individual methods. This is in accordance with several previous studies on multi-task training for NMT [48, 100, 110, 167]. Therefore, we recommend that such an analysis of multi-objective pre-training methods can help isolate the importance of individual pre-training objectives. Nevertheless, our analyses reveal that the components of JASS, BMASS, and BRSS, and the components of ENSS, MASS, and HFSS are certainly responsible for improving translation quality for Japanese-involved or English-language pairs.

4.5.6 Results in Middle/High-resource Scenarios

As summarized in Table 4.14, we report that BLEU leads to middle/high-resource scenarios. The fine-tuning is performed by more than 200k parallel sentences on the respective language pair and domain. By comparing with models without pre-training, we find that pre-training can still lead to some improvements, but much less than those in low-resource scenarios. Second, we observe that most pre-training methods obtained comparable BLEU results regardless of whether they were linguistically-driven methods or not. This indicates that in middle/high-resource scenarios, our proposed methods might be limited, which also shows

#	Model	Ja-En		En-Ja		Ja-Zh			Zh-Ja		
		ASP 200k	ASP 1M	ASP 200k	ASP 1M	ASP 200k	ASP 672k	Wiki 258k	ASP 200k	ASP 672k	Wiki 258k
Main baselines											
0	w/o pre-training	26.1	27.5	33.4	35.8	27.0	31.2	24.6	36.7	42.4	30.4
1	MASS	26.5	28.8	33.7	37.6	27.2	**33.0**	29.4	36.7	**44.8**	34.6
Proposed methods for Japanese											
2	BMASS	26.5	**28.9**	33.8	37.8	**27.8**	32.6	29.8	36.4	44.7	**35.6**
3	BRSS	**26.8**	28.4	**34.0**	37.4	27.2	32.7	30.8	36.8	44.5	35.0
4	JASS	26.7	28.8	33.2	37.5	27.2	32.7	**31.1**	**37.4**	**44.8**	35.4
Proposed methods for English											
11	PMASS.P	26.3	28.0	33.5	37.0	-	-	-	-	-	-
11*	PMASS.S	26.2	28.9	33.2	37.8	-	-	-	-	-	-
12	HFSS	26.5	28.6	33.9	37.7	-	-	-	-	-	-
13	ENSS	26.3	28.8	33.9	**37.9**	-	-	-	-	-	-

Table 4.14: BLEU scores in middle/high-resource scenarios. "ASP" and "Wiki" denote ASPEC and Wikipedia parallel corpus, respectively.

that linguistically-driven supervision can be utilized to compensate for the lack of parallel sentences.

4.6 Summary of This Chapter

In this study, we proposed JASS and ENSS pre-training methods that leverage information from syntactic structures of sentences on the basis of language-agnostic pre-training schemes such as MASS for NMT. Our work leveraged abundant monolingual data and syntactic analysis such that the pre-training phase became aware of specific language structures. Our experiments on ASPEC Japanese–English, Japanese–Chinese, Wikipedia Japanese–Chinese, and News English–Korean translations demonstrated that JASS and ENSS outperform MASS and other language-agnostic pre-training methods in most low-resource settings. This demonstrates the importance of injecting language-specific information into the pre-training objective, as well as the benefit of multi-task pre-training with masked style and reordering objectives. Our adequacy evaluation through LASER, human evaluation, and case study also demonstrated that our methods resulted in a significant

improvement in terms of the adequacy and fluency of translations. The analyses of pre-training accuracy reveal the complementary nature of individual tasks within JASS and ENSS.

Our future work will focus on implementing linguistic-aware multilingual pre-training using more languages for more robust pre-trained models. We also note that Raffel et al. [167] demonstrated that several NLP tasks such as text understanding can be reformulated as text-to-text tasks. This broadens the domain of usefulness of sequence-to-sequence pre-training tasks including ours, and we will be interested in evaluating our approach on various NLP tasks.

Chapter 5

When do Contrastive Word Alignments Improve Many-to-many Neural Machine Translation?

Many-to-many neural machine translation (NMT) [55, 81, 3, 186, 8, 105, 149] jointly trains a translation system for multiple language pairs and obtain significant gains consistently across many translation directions. Previous work [105] shows that word alignment information helps improve pre-training for many-to-many NMT. However, manually cleaned high-quality ground-truth bilingual dictionaries are used to pre-edit the source sentences, which are unavailable for most language pairs.

Recently, contrastive objectives [37, 65, 60, 235, 122, 224, 193, 223] have been shown to be superior at leveraging alignment knowledge in various NLP tasks by contrasting the representations of positive and negative samples in a discriminative manner. This objective, which should be able to utilize word alignment learned by any toolkit, which in turn will remove the constraints of using manually constructed dictionaries, has not been explored in the context of leveraging word alignment for many-to-many NMT.

An existing contrastive method [149] for multilingual NMT relies on sentence-level alignments. Given that the incorporation of word alignments has led to improvements in previous work, we believe that fine-grained contrastive objectives focusing on word alignments should help improve translation. Therefore, this study proposes word-level contrastive learning for many-to-many NMT using the word alignment extracted by automatic aligners. We conduct experiments on three many-to-many NMT systems covering general and spoken language domains. Results show that our proposed method achieves significant gains of 0.8 BLEU in the general domain compared to previous word alignment based methods and the sentence-level contrastive method.

We then analyze how the word-level contrastive objective affects NMT training. Inspired by previous work [12] that train sentence retrieval models using many-to-many NMT, we speculate that our contrastive objectives affect the sentence retrieval performance and subsequently impact the translation quality. Further investigation reveals that in many-to-many NMT, the sentence retrieval precision of the multilingual encoder for a language pair strongly correlates with its translation quality (BLEU), which provides insight about when contrastive alignment improves translation. This revelation emphasizes the importance of improving the retrieval performance of the encoder for many-to-many NMT.

5.1 Word-level Contrastive Learning for NMT

Inspired by the contrastive learning framework [27] and the sentence-level contrastive learning objective [149], we propose a word-level contrastive learning objective to explicitly guide the training of the multilingual encoder to obtain well-aligned cross-lingual representations. Specifically, we use word alignments, obtained using automatic word aligners, to supervise the training of the multilingual encoder by a contrastive objective alongside the NMT objective.

5.1.1 Alignment Extraction

Two main approaches for automatically extracting aligned words from a sentence pair are: using a bilingual dictionary and using unsupervised word aligners. The

former extracts fewer but precise alignments, whereas the latter extracts more but noisy alignments. We extract word-level alignments by both methods and explore how they impact NMT training. For the former approach, we use word2word [30] to construct bilingual lexicons and then extract word pairs from parallel sentences. The extracted word pairs are combined to form a phrase if words are consecutive in the source and target sentence. For the latter approach, we use FastAlign [49] and use only 1-to-1 mappings for training.

5.1.2 Word-level Contrastive Learning

With the extracted alignments, we propose a word-level contrastive learning objective for the multilingual encoder by the motivation that the aligned words within a sentence pair should have a similar contextual representation. We expect the supervision of the contrastive objective on the corresponding contextual word representation leads to a robust multilingual encoder. Assume that the tokenized source and target parallel sentences in the $i-th$ batch are $\mathcal{D}_i = \{src_{ij}, tgt_{ij}\}_{j=1}^{B}$, and the extracted alignments from all the sentence pairs in each batch are $\mathcal{A}_i = \{s_{ik}, t_{ik}\}_{k=1}^{N}$, where B and N denote the batch-size and the number of alignments, respectively. Note that s_{ik} and t_{ik} may contain several tokens after the word combination for word2word or subword tokenization for NMT. Then the word-level contrastive loss in a batch is:

$$\begin{aligned}\mathcal{L}_{align}^{(i)} = -\sum_{k=1}^{N}(&\log \frac{\exp\left(sim(s_{ik}, t_{ik})/\mathcal{T}\right)}{\sum_{m=1}^{N}\exp\left(sim(s_{ik}, t_{im})/\mathcal{T}\right)} \\ &+ \log \frac{\exp\left(sim(s_{ik}, t_{ik})/\mathcal{T}\right)}{\sum_{m=1}^{N}\exp\left(sim(s_{im}, t_{ik})/\mathcal{T}\right)})\end{aligned} \tag{5.1}$$

where $\mathcal{T}$ denotes a similarity scaling temperature. The similarity between two words is measured by:

$$sim(word_x, word_y) = cos(g(\bar{\mathbf{x}}), g(\bar{\mathbf{y}})) \tag{5.2}$$

where $g(\mathbf{x}) = \mathbf{W}_2\sigma(\mathbf{W}_1\mathbf{x})$ and $\bar{\mathbf{x}}$ denotes the average of contextual hidden states of the corresponding subword positions on top of the multilingual encoder. Following Chen et al. [27], we use an MLP between contrastive loss and the contextual

La. pair	Train	Valid	Test	Size	OD Size	N (w2w)	N (FA)
en-et	WMT18	WMT18	WMT18	1.9M	10.7M	5,762,977	38,454,477
en-it	IWSLT17	IWSLT15	IWSLT16	231k	13.6M	603,032	3,000,011
en-ja	IWSLT17	IWSLT15	IWSLT16	223k	10.7M	684,583	2,797,882
en-kk	WMT19	WMT19	WMT19	124k	851k	124,511	279,429
en-my	ALT	ALT	ALT	18k	446k	75,383	377,392
en-nl	IWSLT17	IWSLT15	IWSLT16	237k	12.7M	564,697	2,836,873
en-ro	WMT16	WWT16	WMT16	612k	11.0M	3,271,848	13,092,240
en-tr	WMT17	WWT16	WMT16	207k	11.1M	770,873	2,885,102
en-vi	IWSLT15	IWSLT13	IWSLT14	133k	11.9M	354,167	2,120,755

Table 5.1: **Data Source and number of the extracted word pairs.** La. pair, N (w2w) and N (FA) denote the language pair, the number of the word pairs extracted by word2word and FastAlign, respectively. "Size" denotes the size of training data and "OD Size" denotes the number of the out-of-domain sentence pairs used for training FastAlign.

representation for NMT loss. ReLU activation is used for σ, $\mathbf{W}_1$ is $d \times d$ and $\mathbf{W}_2$ is $d \times d'$, where d is the encoder's hidden dimension and $d' < d$.

Finally, to jointly train with the NMT loss, we use the following equation to combine our proposed word-level contrastive loss for a batch:

$$\mathcal{L}^{(i)} = \frac{1}{B}(\mathcal{L}^{(i)}_{NMT} + w\frac{N_T}{2N}\mathcal{L}^{(i)}_{align}) \tag{5.3}$$

where N_T is the number of the tokens within a batch, $\frac{N_T}{2N}$ is a multiplier that scales the contrastive loss to be consistent with NMT loss, and w is a weight to balance the joint training.

5.2 Experimental Settings

5.2.1 Datasets and Preprocessing

We selected ten languages, including English (en), Estonian (et), Italian (it), Japanese (ja), Kazakh (kk), Burmese (my), Dutch (nl), Romanian (ro), Turkish (tr), Vietnamese (vi) from different language families to train the NMT systems.

We used the parallel datasets from different domains for the selected nine language pairs, including IWSLT, WMT, and ALT. We followed mBART [110] for tokenization. For Japanese, we use Jumanpp [130, 215] for segmentation, and we follow the same settings as in mBART [110] for other languages: `myseg.py` [47] is used for Burmese, Moses tokenization and special normalization is used for Romanian following Sennrich et al. [188],[1] and Moses tokenization for other languages.[2] Following mBART, we apply SentencePiece [94] to further segment sentences into subwords.[3]

The datasets used for NMT training, validation and test are shown in Table 5.1. For each parallel dataset, we implemented two approaches as stated in Section 5.1.1 to extract word pairs for the contrastive training objective. Data source and the number of the extracted word pairs are shown in Table 5.1. For the word alignment extraction using FastAlign, we also use out-of-domain parallel corpora to train the FastAlign jointly, aiming to obtain word alignments with less noise. The out-of-domain corpora for all the language pairs contain Tatoeba, Europarl, GlobalVoices, NewsCommentary, OpenSubtitles, TED, WikiMatrix, QED, GNOME, bible-uedin, and ASPEC [140]. We collect them from the OPUS project [32] and WAT.[4] The number of the out-of-domain parallel sentences for each language pair is shown in Table 5.1.

Many-to-many NMT systems We established three many-to-many NMT systems as follows:

- **222_en-ja**: Bidirectional en-ja NMT model using en-ja parallel corpus.
- **626_en-it-ja-nl-tr-vi**: 6-to-6 multilingual NMT model using spoken language domain corpora for en-it, en-ja, en-nl, en-tr and en-vi.
- **626_en-tr-ro-et-my-kk**: 6-to-6 multilingual NMT model using general domain corpora for en-tr, en-ro, en-et, en-my and en-kk.

[1] `https://github.com/rsennrich/wmt16-scripts`

[2] `https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl`

[3] `https://github.com/google/sentencepiece`

[4] `https://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2021/index.html`

5.2.2 Baselines and Ours

For each language group setting above, we conducted NMT experiments on both the multilingual training from scratch (**MLSC**) [81, 3] and the mBART multilingual fine-tuning (**mBART FT**) [212] as baselines. We applied our proposed word-level contrastive learning in both MLSC and mBART FT, and compared with another strong baseline, word alignment based joint NMT training (+**align**) [58]. For applying our method, we investigated the performance of joint training with word pairs extracted by both word2word (+**w2w**) and FastAlign (+**FA**). We omitted Lin et al. [105] as a baseline because their method can not be applied to mBART fine-tuning, and they used high-quality ground-truth dictionaries, which are unavailable for most languages pairs.

5.2.3 Implementation

We used `mBART-large` (mBART-25) for mBART FT and `Transformer-base` [219] for MLSC. Following Tang et al. [212], we set the oversampling temperature of 1.5 for all the settings. For MLSC, we set the dropout of 0.3 to avoid overfitting on small-scale training data. We used the batch size of 1,024 tokens for all the settings. For our word-level contrastive learning, we set the weight of 0.1, the temperature of 0.2, d' of 128, and a smaller dropout of 0.2 because our proposed objective serves as a regularization part. We followed the hyperparameter setting of Garg et al. [58] for word alignment-based joint NMT training. We used 8 NVIDIA A100 for mBART FT and 8 TITAN Xp for MLSC model training. The model is validated every 1000 steps for 222_en-ja and 2000 steps for both two 626 settings. We do the early stopping if no improvement of the validation loss is observed for 8 checkpoints. The model with the best validation loss was used for evaluation.

Methods	222_en-ja	626_I	626_II
MLSC	13.90	23.76	13.55
+align	13.90	23.67	13.39
+w2w (ours)	13.85	23.44	**13.69**
+FA (ours)	13.30	23.68	13.48
mBART FT	18.90	29.11	20.64
+align	18.55	28.87	20.42
+w2w (ours)	18.80	29.08	**20.89**
+FA (ours)	18.65	29.01	**20.87**

Table 5.2: **Overall average BLEU of all the systems.** 626_I and 626_II denote 626_en-it-ja-nl-tr-vi and 626_en-tr-ro-et-my-kk, respectively. Results better than MLSC or mBART FT are marked **bold**. Refer to Appendix C.1 for the detailed scores of all the systems.

5.3 Results and Analyses

5.3.1 BLEU Results

We report case-sensitive tokenized BLEU [150] results in Table 5.2 and 5.3. In Table 5.2, we observe that with our proposed training objectives, BLEU scores are comparable in 222_en-ja and 626_en-it-ja-nl-tr-vi while they are slightly improved in 626_en-tr-ro-et-my-kk. However, "+align" performs comparable or even worse compared with the baseline. Referring to Table 5.3 for specific BLEUs on each language pair, we find that with our methods, translation performances are significantly improved for mBART FT while nontrivial improvements can merely be observed on en-ro and en-kk direction for MLSC. This indicates that NMT fine-tuning on monolingual pre-trained models (mBART) may benefit more from our proposed methods. Note that the BLEU improvements for MLSC are not significant, and we explain why this happens in the "Word Retrieval P@1 is improved" part.

Methods	en-tr		en-ro		en-et		en-kk		en-my	
	→	←	→	←	→	←	→	←	→	←
MLSC	9.3	12.6	25.0	26.2	10.8	15.1	0.5	5.3	15.1	15.6
+align	9.0	12.4	24.6	26.5	10.7	14.6	0.4	5.4	15.0	15.3
+w2w (ours)	9.4	12.6	24.8	26.8	10.8	15.1	0.5	5.8	15.2	15.9
+FA (ours)	9.1	12.2	24.8	26.7	10.7	14.8	0.3	5.6	15.0	15.6
mBART FT	17.7	22.2	33.8	37.1	14.5	24.3	1.8	14.1	17.8	23.1
+align	17.5	21.9	33.8	36.7	15.2	24.3	1.8	14.0	16.9	22.1
+w2w (ours)	17.6	22.2	34.2	37.5	15.0	25.0	1.2	14.1	18.3	23.8
+FA (ours)	17.5	22.2	34.3	37.5	14.9	25.1	1.3	14.4	17.9	23.6

Table 5.3: **BLEU scores of 626_en-tr-ro-et-my-kk system.** Significantly better scores [88] are in cyan, and marginal improvements are in lightcyan.

5.3.2 Latent Encoder Alignment Property

We now inspect which aspect of alignment-based methods impacts the translation performance. Previous work [12] showed that the encoder of a strong multilingual NMT system is an ideal model for the bilingual sentence retrieval task. In addition, Arivazhagan et al. [7] introduced the correlation between the encoder-side sentence representation[5] and the translation quality. Inspired by these, we speculate that alignment-based objectives affect sentence retrieval performance, which further impacts the translation quality. We train MLSC and mBART FT and report the sentence retrieval precision and NMT loss during the training. Results are reported in Figure 5.1. We observe that the validation retrieval precision show similar trends as the NMT loss. This indicates that during many-to-many NMT training from scratch, encoder-side sentence-level retrieval precision is optimized along with the NMT loss.

[5]Usually a pooled encoder output.

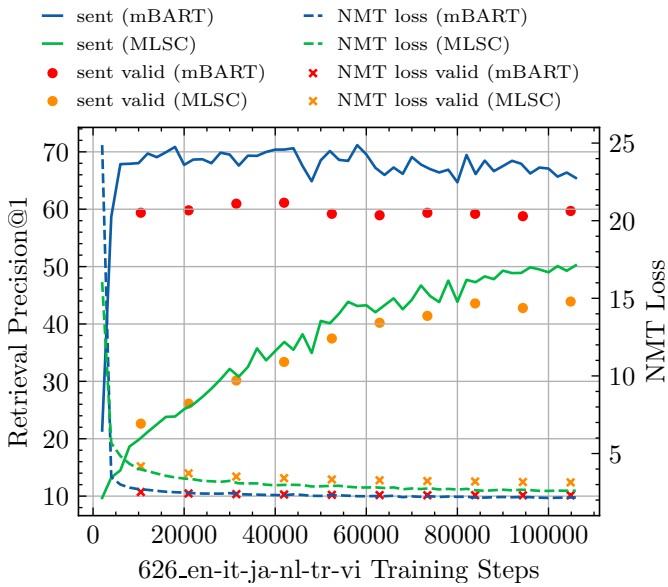

Figure 5.1: **NMT loss, sentence retrieval P@1 of the encoder in MLSC and mBART FT.** The average of the contextual embeddings on top of the encoder is used as the sentence embedding. We report the average in-batch retrieval precision of both directions of each language pair.

5.3.3 Sentence Retrieval P@1

According to the investigation of the encoder alignment property above, we verify the relationship between BLEU score and sentence retrieval precision on the validation set for each language pair. Results are shown in Figure 5.2. Cross-referencing the BLEU score in Table 5.3, we found that BLEU scores are improved when the encoder achieves gains on the sentence retrieval precision.[6] For example, we see increases of the retrieval P@1 on en-ro, en-et, and en-my on mBART FT (the middle of Figure 5.2) while BLEU scores are significantly improved on these three language pairs (Table 5.3). We further calculate the Pearson correlation coefficient between the BLEU changes and sentence retrieval P@1 changes for mBART+align, mBART+w2w, and mBART+FA in the 626_en-tr-ro-et-my-kk

[6]222_en-ja MLSC setting can hardly learn a well-aligned encoder while our methods improve the encoder sentence-level alignment quality without sacrificing BLEU scores.

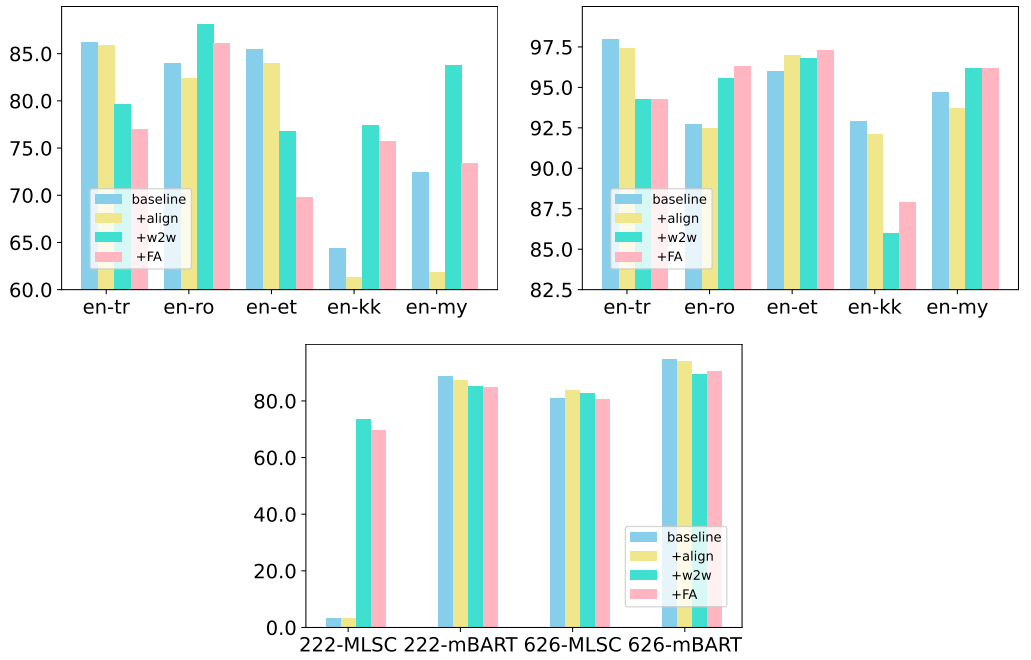

Figure 5.2: **Sentence retrieval P@1 on the validation set for each language pair.** *Top Left* and *Top Right* are the results on 626_en-tr-ro-et-my-kk MLSC and mBART FT, respectively. "626" in *Bottom* subfigure denote 626_en-it-ja-nl-tr-vi. Refer to Appendix C.2 for setup and results in details.

setting. Results are 0.79, 0.93, 0.90, respectively, demonstrating a strong correlation between translation quality and sentence retrieval precision.

5.3.4 Word Retrieval P@1

We probe the trained contextualized word representations on top of the encoder. As shown in Figure 5.3, we observe that the word retrieval precision is improved in all the settings. This demonstrates that the encoder parameters of the NMT system trained with our proposed objective are of a rather different distribution. By just changing the random seed, we can expect similar BLEU results, but we cannot obtain a better aligned encoder. However, the improvement of the word retrieval precision does not directly contribute to the translation quality, which we explain next.

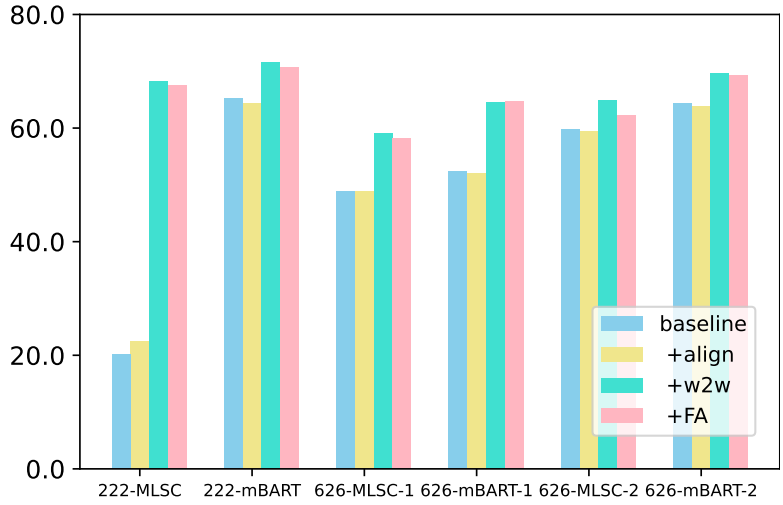

Figure 5.3: **Average Word retrieval P@1 on the validation set for each language pair.** "626-*-1" and "626-*-2" indicate 626_en-it-ja-nl-tr-vi and 626_en-tr-ro-et-my-kk, respectively. Refer to Appendix C.3 for setup and results in details.

5.3.5 Word-level Contrastive Objective and Sentence Retrieval P@1

With the word-level contrastive objective, we observed significant BLEU score improvements on language pairs such as en-ro, en-et and en-my for mBART FT as presented in Table 5.3. However, noisy word pairs [148] extracted via word alignment toolkits leads to poor supervision signals for improving sentence retrieval P@1, which in turn prevents some language pairs such as en-kk from exhibiting BLEU improvements. We found that for en-kk, the numbers of extracted word pairs per sentence by word2word and FastAlign are 1.0 and 2.2, respectively. In contrast, these numbers are 4.2 and 20.7 for improved language pairs, calculated from Table 5.1. Although better extracted word alignments for the word-level contrastive objective leads to BLEU improvements, its contribution towards improvements varies for MLSC and mBART FT, as shown in Table 5.3. We expect these findings to provide new perspectives for improving many-to-many NMT.

5.3.6 Sentence-level Contrastive Objective

We conducted the experiments for the sentence-level contrastive objective [149] on all two six-to-six settings and compared it against our proposed approach. The average BLEUs of our methods significantly outperform those of sentence-level contrastive objectives (see Table C.2 and C.3), clearly showing the sentence-level objective's limitation. Moreover, we checked the sentence retrieval P@1 for Pan et al. [149] (Table C.5 and C.6) and found that it correlates with BLEU changes, indicating that sentence-level contrastive objective is suboptimal for language pairs with decreased retrieval precision.[7]

5.4 Summary of This Chapter

We proposed a word-level contrastive learning objective for many-to-many NMT. Experimental results showed that our proposed method leads to significantly better translation for several language pairs, which is then explained by analyses showing the relationship between BLEU scores and sentence retrieval performance of the NMT encoder. Future work can focus on: (1) further improving the encoder's retrieval performance in many-to-many NMT; (2) contrastive objective's feasibility in a massively multilingual scenario.

[7]Note that the sentence-level contrastive objective incorporates sentences in multiple languages for contrastive loss. It does not necessarily improve the pair-wise retrieval precision.

Chapter 6

Tuning LLMs with Contrastive Alignment Instructions for Machine Translation in Unseen, Low-resource Languages

Large language models (LLMs) achieved good performance for a wide range of NLP tasks for prevalent languages [23, 31, 180, 216, 132, 145, 5, 217, 225]. However, insufficient coverage for low-resource languages remains to be one significant limitation. Low-resource languages are either not present, or orders of magnitude smaller in size than dominant languages in the pre-training dataset. This limitation is in part due to the prohibitive cost incurred by curating good quality and adequately sized datasets for pre-training. Incrementally adapting existing multilingual LLMs to incorporate an unseen, low-resource language thus becomes a cost-effective priority to address this limitation. Previous study [45, 134, 254] explored extending language support using either continual pre-training [141, 10, 133, 50], or parameter efficient fine-tuning (PEFT) methods [154, 75, 109] on monolingual tasks. Extending language support for cross-lingual tasks remains underexplored due to the challenge of incrementally inducing cross-lingual understanding and generation abilities in LLMs [254].

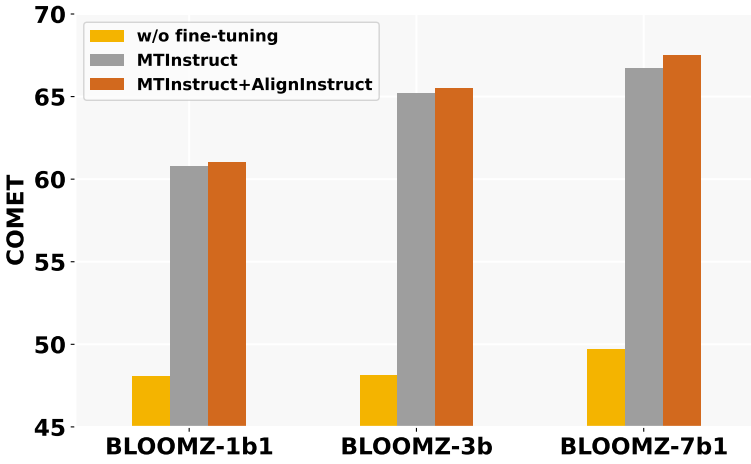

Figure 6.1: **Average COMET scores of BLOOMZ models across 24 unseen languages**, comparing settings of without fine-tuning, fine-tuning with MTInstruct, and fine-tuning that combines MTInstruct and AlignInstruct.

This study focused on machine translation (MT) to highlight the cross-lingual LLM adaptation challenge. The challenge lies in enabling translation for low-resource languages that often lack robust cross-lingual signals. We first explored the efficacy of fine-tuning LLMs with MT instructions (MTInstruct) in unseen, low-resource languages. MTInstruct is a method previously shown to bolster the translation proficiency of LLMs for supported languages [101]. Subsequently, given that cross-lingual alignments are suboptimal in LLMs as a result of data scarcity of low-resource languages, we proposed contrastive alignment instructions (AlignInstruct) to explicitly provide cross-lingual supervision during MT fine-tuning. AlignInstruct is a cross-lingual discriminator formulated using statistical word alignments. Our approach was inspired by prior studies [96, 174, 105, 114], which indicated the utility of word alignments in enhancing MT. In addition to AlignInstruct, we discussed two word-level cross-lingual instruction alternatives cast as generative tasks, for comparison with AlignInstruct.

Our experiments fine-tuned the BLOOMZ models [132] of varying sizes (1b1, 3b, and 7b1) for 24 unseen, low-resource languages, and evaluated translation on OPUS-100 [261] and Flores-200 [42]. We first showed that MTInstruct effectively

induced the translation capabilities of LLMs for these languages. Building on the MTInstruct baseline, the multi-task learning combining AlignInstruct and MTInstruct resulted in stronger translation performance without the need for additional training corpora. The performance improved with larger BLOOMZ models, as illustrated in Figure 6.1, indicating that AlignInstruct is particularly beneficial for larger LLMs during MT fine-tuning. When compared with the generative variants of AlignInstruct, our results indicated that discriminator-style instructions better complemented MTInstruct. Furthermore, merging AlignInstruct with its generative counterparts did not further improve translation quality, underscoring the efficacy and sufficiency of AlignInstruct in leveraging word alignments for MT.

In zero-shot translation evaluations on the OPUS benchmark, AlignInstruct exhibited improvements over the MTInstruct baseline in 30 zero-shot directions not involving English, when exclusively fine-tuned with three unseen languages (German, Dutch, and Russian). However, when the fine-tuning data incorporated supported languages (Arabic, French, and Chinese), the benefits of AlignInstruct were only evident in zero-shot translations where the target language was a supported language.

To interpret the inherent modifications within the BLOOMZ models after applying MTInstruct or AlignInstruct, we conducted a visualization of the layer-wise cross-lingual alignment capabilities of the model representations. In addition, we discussed the effect of monolingual instructions in the resource-constrained scenario.

6.1 Related Work

6.1.1 Prompting LLMs for MT

LLMs have shown good performance for multilingual MT through few-shot in-context learning (ICL) [79]. Vilar et al. [222] showed that high-quality examples can improve MT based on PaLM [31]. Agrawal et al. [2] and Zhang et al. [259] explored strategies to compose better examples for few-shot prompting for XGLM-7.5B [104] and GLM-130B [257]. Ghazvininejad et al. [59], Peng et al. [152], and Moslem et al. [131] claimed that dictionary-based hints and domain-specific style

information can improve prompting OPT [264], GPT-3.5 [23], and BLOOM [180] for MT. He et al. [69] used LLMs to mine useful knowledge for prompting GPT-3.5 for MT.

6.1.2 Fine-tuning LLMs for MT

ICL-based methods do not support languages unseen during pre-training. Current approaches address this issue via fine-tuning. Zhang et al. [263] explored adding new languages to LLaMA [216] with interactive translation task for unseen high-resource languages. However, similar task datasets are usually not available for most unseen, low-resource languages. Li et al. [101] and Xu et al. [239] showed multilingual fine-tuning with translation instructions can improve the translation ability in supported languages. Our study extended their finding to apply in the context of unseen, low-resource languages. In parallel research, Yang et al. [243] undertook MT instruction fine-tuning in a massively multilingual context for unseen languages. However, their emphasis was on fine-tuning curriculum based on resource availability of languages, whereas we exclusively centered on low-resource languages and instruction tuning tasks.

6.2 Methodology

This section presents MTInstruct as the baseline, and AlignInstruct. The MTInstruct baseline involved fine-tuning LLMs using MT instructions. AlignInstruct dealt with the lack of cross-lingual signals stemming from the limited parallel training data in low-resource languages. The expectation was enhanced cross-lingual supervision cast as a discriminative task without extra training corpora. Following this, we introduced two generative variants of AlignInstruct for comparison and discussed monolingual instructions for MT fine-tuning.

6.2.1 Baseline: MTInstruct

Instruction tuning [231, 129, 36, 147, 179, 234] has been shown to generalize LLMs' ability to perform various downstream tasks, including MT [101].

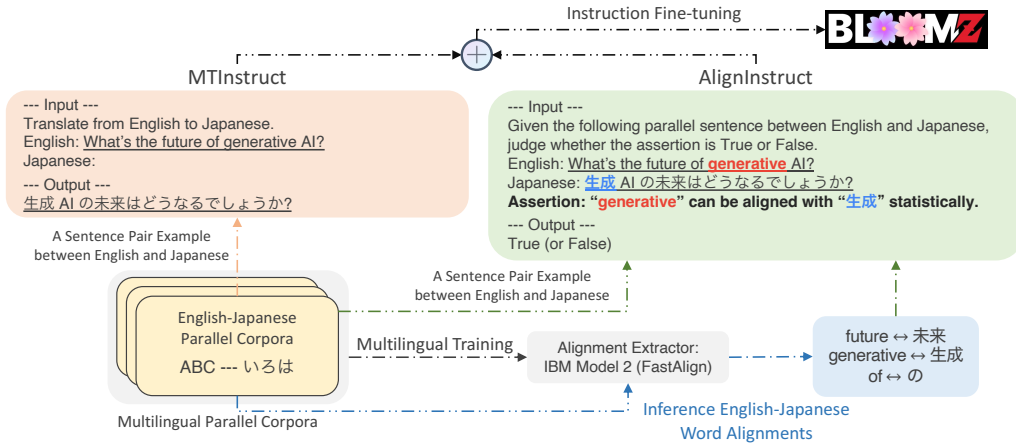

Figure 6.2: **Proposed instruction tuning methods combining MTInstruct (Section 6.2.1) and AlignInstruct (Section 6.2.2) for LLMs in MT tasks.** $\oplus$ denotes combining multiple instruction patters with a specific fine-tuning curriculum (Section 6.3.2). IBM Model 2 indicates word alignment model of statistical machine translation [22].

Given a pair of the parallel sentences, $\left((x_i)_1^N, (y_j)_1^M\right)$, where $(x_i)_1^N := x_1 x_2 \dots x_N$, $(y_i)_1^N := y_1 y_2 \dots y_N$. $x_i, y_j \in \mathcal{V}$ are members of the vocabulary $\mathcal{V}$ containing unique tokens that accommodate languages X and Y. Li et al. [101] showed that the following MT instructions (MTInstruct) can improve the translation ability in an LLM with a limited number of parallel sentences:

- **Input:** "Translate from Y to X.
  Y: $y_1 y_2 \dots y_M$.
  X: "
- **Output**: "$x_1 x_2 \dots x_N$."

Note that Li et al. [101] demonstrated the utility of MTInstruct solely within the context of fine-tuning for languages acquired at pre-training phase. This study called for an assessment of MTInstruct on its efficacy for adapting to previously unsupported languages, denoted as X, accompanied by the parallel data in a supported language Y.

6.2.2 AlignInstruct

Word alignments have been demonstrated to enhance MT performance [96, 174, 105, 114], both in the fields of statistical machine translation (SMT) [22] and neural machine translation (NMT) [208, 16]. Ren et al. [174] and Mao et al. [114] reported the utility of SMT-derived contrastive word alignments in guiding encoder-decoder NMT model training. Built upon their findings, we introduced AlignInstruct for bolstering cross-lingual alignments in LLMs. We expected AlignInstruct to enhancing translation performance particularly for languages with no pre-training data and limited fine-tuning data.

As shown in Figure 6.2, we employed FastAlign [49] to extract statistical word alignments from parallel corpora. Our approach depended on a trained FastAlign model, IBM Model 2 [22], to ensure the quality of the extracted word pairs. These high-quality word alignment pairs were regarded as "gold" word pairs for constructing AlignInstruct instructions.[1] Assuming one gold word pair ($x_k x_{k+1}, y_l y_{l+1}$ y_{l+2}) was provided for the sentence pair $\left((x_i)_1^N, (y_j)_1^M\right)$, the AlignInstruct instruction reads:

- **Input:** "Given the following parallel sentence between Y and X, judge whether the assertion is True or False.
  Y: $y_1 y_2 \dots y_M$.
  X: $x_1 x_2 \dots x_N$.
  Assertion: "$y_l y_{l+1} y_{l+2}$" can be aligned with "$x_k x_{k+1}$" statistically."
- **Output**: "True" (or "False")

Instructions with the "False" output were constructed by uniformly swapping out part of the word pair to create misalignment. We anticipated that this treatment forced the model to learn to infer the output by recognizing true alignment-enriched instructions. This would require the model to encode word-level cross-lingual representation, a crucial characteristic for MT tasks.

[1] Note that these word pairs may not necessarily represent direct translations of each other; instead, they are word pairs identified based on their co-occurrence probability within the similar context. Refer to IBM model 2 in SMT.

6.2.3 Generative Counterparts of AlignInstruct

Previous studies [103, 256] have suggested the importance of both discriminative and generative tasks in fine-tuning LLMs. We accordingly considered two generative variants of AlignInstruct. We then compared them with AlignInstruct to determine the most effective training task. As detailed in Section 6.4, our results indicated that these variants underperformed AlignInstruct when applied to unseen, low-resource languages.

HintInstruct

HintInstruct as a generative variant of AlignInstruct was instructions containing word alignment hints. It was inspired by Ghazvininejad et al. [59], where dictionary hints were shown to improve few-shot in-context leaning. Instead of relying on additional dictionaries, we used the same word alignments described in Section 6.2.2, which were motivated by the common unavailability of high-quality dictionaries for unseen, low-resource languages. Let $\{(x_{k_s}x_{k_s+1}\dots x_{k_s+n_s}, y_{l_s}y_{l_s+1}\dots y_{l_s+m_s})\}_{s=1}^{S}$ be S word pairs extracted from the sentence pair $\left((x_i)_1^N, (y_j)_1^M\right)$. HintInstruct follows the instruction pattern:

- **Input**: "Use the following alignment hints and translate from Y to X.
  Alignments between X and Y:
  – $(x_{k_1}x_{k_1+1}\dots x_{k_1+n_1}, y_{l_1}y_{l_1+1}\dots y_{l_1+m_1})$,
  – $(x_{k_2}x_{k_2+1}\dots x_{k_1+n_1}, y_{l_2}y_{l_2+1}\dots y_{l_2+m_2})$,
  …,
  – $(x_{k_S}x_{k_S+1}\dots x_{k_S+n_S}, y_{l_S}y_{l_S+1}\dots y_{l_S+m_S})$,
  Y: $y_1y_2\dots y_M$.
  X: "
- **Output**: "$x_1x_2\dots x_N$."

where S denotes the number of the word alignment pairs used to compose the instructions. Different from AlignInstruct, HintInstruct expects the translation targets to be generated.

ReviseInstruct

ReviseInstruct was inspired by Ren et al. [174] and Liu et al. [110] for the notion of generating parallel words or phrases, thereby encouraging a model to encode cross-lingual alignments. A ReviseInstruct instruction contained a partially corrupted translation target, as well as a directive to identify and revise these erroneous tokens. Tokens are intentionally corrupted at the granularity of individual words, aligning with the word-level granularity in AlignInstruct and HintInstruct. ReviseInstruct follows the instruction pattern:

- **Input**: "Given the following translation of X from Y, output the incorrectly translated word and correct it.
  Y: $y_1 y_2 \dots y_M$.
  X: $x_1 x_2 \dots x_k x_{k+1} \dots x_{k+n} \dots x_N$."
- **Output**: "The incorrectly translated word is "$x_k x_{k+1} \dots x_{k+n}$". It should be "$x_j x_{j+1} \dots x_{j+m}$"."

6.2.4 Monolingual Instructions

New language capabilities may be induced through continual pre-training on monolingual next-word prediction tasks [254]. The coherence of the generated sentences is crucial in MT [233, 110], especially when the target languages are unseen and low-resource. We examined the significance of this approach in fostering the translation quality. We reused the same parallel corpora to avoid introducing additional monolingual datasets.

Given a monolingual sentence, $(x_i)_1^N$, with length N in an unseen language X. The LLM is incrementally trained on the following task:

- **Input**: "Given the context, complete the following sentence: $x_1 x_2 \dots x_{l<N}$,"
- **Output**: "$x_{l+1} x_{l+2} \dots x_N$."

Language	ISO 639-1	Language Family	Subgrouping	Script	Seen Script	#sent.
Afrikaans	af	Indo-European	Germanic	Latin	✓	275,512
Amharic	am	Afro-Asiatic	Semitic	Ge'ez	✗	89,027
Belarusian	be	Indo-European	Balto-Slavic	Cyrillic	✗	67,312
Welsh	cy	Indo-European	Celtic	Latin	✓	289,521
Irish	ga	Indo-European	Celtic	Latin	✓	289,524
Scottish Gaelic	gd	Indo-European	Celtic	Latin	✓	16,316
Galician	gl	Indo-European	Italic	Latin	✓	515,344
Hausa	ha	Afro-Asiatic	Chadic	Latin	✓	97,983
Georgian	ka	Kartvelian	Georgian-Zan	Georgian	✗	377,306
Kazakh	kk	Turkic	Common Turkic	Cyrillic	✗	79,927
Khmer	km	Austroasiatic	Khmeric	Khmer	✗	111,483
Kyrgyz	ky	Turkic	Common Turkic	Cyrillic	✗	27,215
Limburgish	li	Indo-European	Germanic	Latin	✓	25,535
Burmese	my	Sino-Tibetan	Burmo-Qiangic	Myanmar	✗	24,594
Norwegian Bokmål	nb	Indo-European	Germanic	Latin	✓	142,906
Norwegian Nynorsk	nn	Indo-European	Germanic	Latin	✓	486,055
Occitan	oc	Indo-European	Italic	Latin	✓	35,791
Sinhala	si	Indo-European	Indo-Aryan	Sinhala	✗	979,109
Tajik	tg	Indo-European	Iranian	Cyrillic	✗	193,882
Turkmen	tk	Turkic	Common Turkic	Latin	✓	13,110
Tatar	tt	Turkic	Common Turkic	Cyrillic	✗	100,843
Uyghur	ug	Turkic	Common Turkic	Arabic	✓	72,170
Northern Uzbek	uz	Turkic	Common Turkic	Latin	✓	173,157
Eastern Yiddish	yi	Indo-European	Germanic	Hebrew	✗	15,010
Total						4,498,632

Table 6.1: **Statistics of training data for BLOOMZ+24**: 24 unseen, low-resource languages for BLOOMZ. ✓and ✗indicate whether script is seen or unseen.

6.3 Experimental Settings

6.3.1 Backbone Models and Unseen Languages

Our experiments fine-tuned the BLOOMZ models [132] for MT in unseen, low-resource languages. BLOOMZ is an instruction fine-tuned multilingual LLM from BLOOM [180] that supports translation across 46 languages. Two lines of experiments evaluated the effectiveness of the MTInstruct baseline and AlignInstruct:

BLOOMZ+24 Tuning BLOOMZ-7b1, BLOOMZ-3b, and BLOOMZ-1b1[2] for 24 unseen, low-resource languages. These experiments aimed to: (1) assess the effectiveness of AlignInstruct in multilingual, low-resource scenarios; (2) offer comparison across various model sizes. We used the OPUS-100 [261][3] datasets as training data. OPUS-100 is an English-centric parallel corpora, with around 4.5M parallel sentences in total for 24 selected languages, averaging 187k sentence pairs for each language and English. Training data statistics of BLOOMZ+24 are shown in Table 6.1. Several selected languages involved previously unseen scripts by BLOOMZ, but such fine-tuning is practical as BLOOMZ is a byte-level model with the potential to adapt to any language. Note that our proposed methods can be applied to any byte-level generative LLMs.

We used OPUS-100 and Flores-200 [42][4] for evaluating translation between English and 24 unseen languages (48 directions in total) on in-domain and out-of-domain test sets, respectively. The identical prompt as introduced in Section 6.2.1 was employed for inference. Inferences using alternative MT prompts are discussed in Appendix D.3.

BLOOMZ+3 Tuning BLOOMZ-7b1 with three unseen languages, German (de), Dutch (nl), and Russian (ru), or a combination of these three unseen languages and another three seen (Arabic (ar), French (fr), and Chinese (zh)). We denote the respective setting as **de-nl-ru** and **ar-de-fr-nl-ru-zh**. These experiments assessed the efficacy of AlignInstruct in zero-shot translation scenarios, where translation directions were not presented during fine-tuning, as well as the translation performance when incorporating supported languages as either source or target languages. To simulate the low-resource fine-tuning scenario, we randomly sampled 200k parallel sentences for each language. For evaluation, we used the OPUS-100 supervised and zero-shot test sets, comprising 12 supervised directions involving English and 30 zero-shot directions without English among six languages.

[2] https://huggingface.co/bigscience/bloomz

[3] https://opus.nlpl.eu/opus-100.php

[4] https://github.com/facebookresearch/flores/blob/main/flores200/README.md

6.3.2 Training Details and Curricula

The PEFT method, LoRA [75], was chosen to satisfy the parameter efficiency requirement for low-resource languages, as full-parameter fine-tuning would likely under-specify the models. We employed 128 V100 GPUs for the BLOOMZ+24 and 32 V100 GPUs for the BLOOMZ+3 experiments. The batch sizes were configured at 4 sentences for BLOOMZ-7b1 and 8 sentences for both BLOOMZ-3b and BLOOMZ-1b1, per GPU device. We configured LoRA with a rank of 8, an alpha of 32, and a dropout of 0.1. Consequently, the BLOOMZ-7b1, BLOOMZ-3b, and BLOOMZ-1b1 models had 3.9M, 2.5M, and 1.2M trainable parameters, respectively, constituting approximately 0.05 - 0.10% of the parameters in the original models. We conducted training for 5 epochs, ensuring a stable convergence is achieved. To facilitate this stability, we introduced a warm-up ratio of 0.03 into our training process. Maximum input and output length were set as 384. S for HintInstruct was set as 5 at most. Additionally, we used mixed precision training [127] to expedite computation using DeepSpeed [170]. We tuned the optimal learning rate for each individual experiment according to validation loss. We conducted all experiments once due to computational resource constraints and reported the average scores across all languages.

How AlignInstruct and MTInstruct are integrated into training remained undetermined. To that end, we investigated three training curricula:

Multi-task Fine-tuning combined multiple tasks in a single training session [24]. This was realized by joining MTInstruct and AlignInstruct training data, denoted as **MT+Align**.[5]

Pre-fine-tuning & Fine-tuning arranges AlignInstruct and MTInstruct into two stages; namely, curriculum learning [19].[6] This configuration, denoted as **Align→MT**, validates whether AlignInstruct should precede MTInstruct.

Mixed Fine-tuning [33, 200] arranged the two aforementioned curricula to start with MT+Align, followed by MTInstruct, denoted as **MT+Align→MT**.

[5]Note that AlignInstruct and MTInstruct were derived from the same parallel corpora.

[6]An effective curriculum often starts with a simple and general task, followed by a task-specific task.

BLOOMZ model	Objective	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
		BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
BLOOMZ-7b1	w/o fine-tuning	3.61	8.82	47.94	6.70	18.49	51.49	2.00	9.35	37.04	9.95	24.47	52.18
	Individual objectives												
	MTInstruct	11.54	25.33	64.68	18.59	33.25	68.75	3.30	17.10	42.62	11.37	27.14	55.82
	AlignInstruct	4.73	9.23	49.11	5.32	12.90	53.05	1.97	8.90	40.64	3.47	11.93	39.20
	Multiple objectives with different curricula												
	MT+Align	**12.28**	**26.17**	**65.28**	**18.72**	**34.02**	**69.75**	3.26	**17.20**	**43.05**	**11.60**	**27.38**	**56.28**
	Align→MT	**11.73**	**25.48**	64.64	17.54	32.62	68.70	**3.35**	**17.21**	**42.76**	11.32	**27.21**	55.81
	MT+Align→MT	**12.10**	**26.16**	**65.14**	18.23	**33.54**	**69.56**	3.28	**17.26**	**43.13**	**11.48**	**27.34**	**56.12**
BLOOMZ-3b	w/o fine-tuning	4.63	9.93	48.38	5.90	16.38	47.88	2.00	9.09	38.88	5.86	18.56	46.47
	Individual objectives												
	MTInstruct	10.40	23.08	62.66	16.10	31.15	67.67	2.85	16.23	41.30	8.92	24.57	52.77
	AlignInstruct	1.70	4.05	44.10	0.87	3.20	42.32	0.16	3.09	31.10	0.10	1.80	29.27
	Multiple objectives with different curricula												
	MT+Align	**10.61**	**23.64**	**63.03**	**16.73**	**31.51**	**67.94**	**2.95**	**16.62**	**41.86**	**9.50**	**25.16**	**53.63**
	Align→MT	10.22	22.53	62.22	15.90	30.31	66.79	**3.02**	**16.43**	**41.67**	**9.07**	**24.70**	**53.11**
	MT+Align→MT	**10.60**	**23.35**	**62.69**	**16.58**	**31.64**	**68.29**	**2.93**	**16.57**	**41.74**	**9.41**	**25.08**	**53.44**
BLOOMZ-1b1	w/o fine-tuning	3.76	7.57	46.81	4.78	14.11	49.27	1.24	6.93	37.38	3.49	14.56	43.05
	Individual objectives												
	MTInstruct	7.42	17.85	58.05	11.99	25.59	63.50	2.11	14.40	38.90	5.33	20.65	48.42
	AlignInstruct	2.51	5.29	45.56	3.13	8.92	48.73	0.35	3.79	31.21	1.35	6.43	33.24
	Multiple objectives with different curricula												
	MT+Align	**7.80**	**18.48**	**58.58**	**12.57**	**25.92**	63.49	**2.16**	**14.54**	**39.36**	**5.46**	**20.90**	**48.81**
	Align→MT	**7.49**	**18.09**	**58.38**	11.80	24.70	62.58	2.08	14.28	**39.04**	5.24	20.53	48.37
	MT+Align→MT	**7.98**	**18.61**	**58.74**	**12.43**	**25.78**	63.30	**2.16**	**14.46**	**39.25**	**5.37**	**20.67**	**48.57**

Table 6.2: **Results of BLOOMZ+24 fine-tuned with MTInstruct and AlignInstruct on different curricula** as described in 6.3.2. Scores that surpass the MTInstruct baseline are marked in **bold**.

6.4 Evaluation and Analysis

This section reports BLEU [150, 161], chrF++ [160], and COMET [171] scores for respective experimental configurations. We further characterized of the degree to which intermediate embeddings were language-agnostic after fine-tuning.

6.4.1 BLOOMZ+24 Results

Table 6.2 shows the scores for the unmodified BLOOMZ models, as well as the models of BLOOMZ+24 under MTInstruct, AlignInstruct, and the three distinct curricula. Non-trivial improvements in all metrics were evident for BLOOMZ+24 under MTInstruct. This suggests that MTInstruct can induce translation capabilities in unseen languages. Applying AlignInstruct and MTInstruct via the curricula further showed better scores than the baselines, suggesting the role of AlignInstruct as complementing MTInstruct. Align→MT was an exception, performing similarly to MTInstruct. This may indicate AlignInstruct's complementarity depends on its cadence relative to MTInstruct in a curriculum.

Superior OPUS and Flores scores under the xx→en direction were evident, compared to the reverse direction, en→xx. This suggests that our treatments induced understanding capabilities more than generative ones. This may be attributed to the fact that BLOOMZ had significant exposure to English, and that we used English-centric corpora. Finally, we noted the inferior performance of Flores than OPUS. This speaks to the challenge of instilling translation abilities in unseen languages when dealing with the out-of-domain MT task.

6.4.2 Assessing AlignInstruct Variants

From the results reported in Table 6.3, we observed the objectives with AlignInstruct consistently outperformed those with HintInstruct or ReviseInstruct across metrics and model sizes. Namely, easy, discriminative instructions, rather than hard, generative ones, may be preferred for experiments under similar data constraints. The low-resource constraint likely made MTInstruct more sensitive to the difficulty of its accompanying tasks.

Further, combining more than two instruction tuning tasks simultaneously

BLOOMZ model	Objective	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
		BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
BLOOMZ-7b1	MTInstruct	11.54	25.33	64.68	18.59	33.25	68.75	3.30	17.10	42.62	11.37	27.14	55.82
	MT+Align	**12.28**	**26.17**	**65.28**	**18.72**	**34.02**	**69.75**	3.26	**17.20**	**43.05**	**11.60**	**27.38**	**56.28**
	MT+Hint	**12.12**	**25.92**	**64.82**	18.25	33.18	**69.21**	**3.34**	**17.13**	**42.95**	**11.45**	**27.37**	**56.21**
	MT+Revise	**11.96**	**25.73**	**64.99**	**18.69**	**33.74**	**69.30**	**3.34**	17.10	**43.01**	**11.44**	**27.37**	**56.08**
BLOOMZ-3b	MTInstruct	10.40	23.08	62.66	16.10	31.15	67.67	2.85	16.23	41.30	8.92	24.57	52.77
	MT+Align	**10.61**	**23.64**	**63.03**	**16.73**	**31.51**	**67.94**	**2.95**	**16.62**	**41.86**	**9.50**	**25.16**	**53.63**
	MT+Hint	**10.49**	**23.34**	62.66	**16.29**	**31.43**	**68.16**	**3.11**	**16.95**	**42.17**	**9.52**	**25.25**	**53.72**
	MT+Revise	**10.52**	23.03	62.38	**16.22**	30.98	67.27	**2.99**	**16.83**	**41.84**	**9.47**	**25.21**	**53.29**
BLOOMZ-1b1	MTInstruct	7.42	17.85	58.05	11.99	25.59	63.50	2.11	14.40	38.90	5.33	20.65	48.42
	MT+Align	**7.80**	**18.48**	**58.58**	**12.57**	**25.92**	63.49	**2.16**	**14.54**	**39.36**	**5.46**	**20.90**	**48.81**
	MT+Hint	**7.71**	**18.15**	**58.26**	11.52	24.88	62.98	**2.21**	**14.61**	**39.59**	**5.47**	**20.78**	**48.56**
	MT+Revise	7.31	**17.99**	**58.18**	**12.00**	25.33	63.11	2.07	14.32	**38.97**	**5.41**	**20.91**	**48.67**

Table 6.3: **Results of BLOOMZ+24 fine-tuned combining MTInstruct with AlignInstruct (or its generative variants).** Scores that surpass the MTInstruct baseline are marked in **bold**.

Objective	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
MTInstruct	11.54	25.33	64.68	18.59	33.25	68.75	3.30	17.10	42.62	11.37	27.14	55.82
MT+Align	**12.28**	**26.17**	**65.28**	**18.72**	**34.02**	**69.75**	3.26	**17.20**	**43.05**	**11.60**	**27.38**	**56.28**
MT+Align+Revise	**12.08**	**25.73**	64.67	**19.23**	**34.32**	**69.65**	**3.33**	**17.25**	**43.05**	**11.60**	**27.61**	**56.51**
MT+Align+Hint	**12.02**	**25.51**	64.68	**19.40**	**34.44**	**69.54**	3.25	16.87	**42.85**	**11.58**	**27.48**	**56.31**
MT+Hint+Revise	**12.10**	**25.69**	**64.71**	**19.58**	**34.49**	**69.46**	**3.34**	**17.24**	**43.07**	**11.70**	**27.62**	**56.48**
MT+Align+Hint+Revise	**12.00**	**25.39**	64.35	**19.68**	**34.48**	**69.58**	**3.40**	**17.17**	**43.09**	**11.67**	**27.54**	**56.44**

Table 6.4: **Results of BLOOMZ+24 combining MTInstruct with multiple objectives among AlignInstruct, HintInstruct, and ReviseInstruct on BLOOMZ-7b1.** Scores that surpass MTInstruct are marked in **bold**.

did not guarantee consistent improvements, see Table 6.4. Notably, MT+Align either outperformed or matched the performance of other objective configurations. While merging multiple instruction tuning tasks occasionally resulted in superior BLEU and chrF++ scores for OPUS xx→en, it fell short in COMET scores compared to MT+Align. This indicated that while such configurations might enhance word-level translation quality, as reflected by BLEU and chrF++ scores, due to increased exposure to cross-lingual word alignments, MT+Align better captured the context of the source sentence as reflected by COMET scores. Overall, these instruction tuning tasks did not demonstrate significant synergistic effects for fine-tuning for unseen languages.

Objective	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
MTInstruct	11.54	25.33	64.68	18.59	33.25	68.75	3.30	17.10	42.62	11.37	27.14	55.82
MT+Mono-full	9.89	22.42	62.56	15.43	29.04	65.45	3.00	16.68	42.34	10.26	25.15	53.67
MT+Mono-half	10.23	22.45	62.59	15.51	29.65	66.18	3.18	16.91	**42.69**	10.66	26.15	54.41
MT+Mono-full+Align	10.15	22.35	62.39	15.72	29.86	66.54	3.07	16.59	42.54	10.61	25.58	54.59
MT+Mono-half+Align	10.09	22.61	63.01	16.00	30.34	67.15	3.10	16.75	**42.63**	10.79	26.27	54.87
MT+Mono-full+Align+Hint+Revise	10.33	23.04	63.06	17.16	31.61	67.40	3.23	16.70	**42.74**	10.98	26.18	54.97
MT+Mono-half+Align+Hint+Revise	10.62	23.10	63.07	17.32	31.80	67.43	3.20	16.93	**42.97**	11.09	26.77	55.41

Table 6.5: **Results of BLOOMZ+24 fine-tuned incorporating monolingual instructions on BLOOMZ-7b1.** Scores that surpass the MTInstruct baseline are marked in **bold**.

6.4.3 Assessing Monolingual Instructions

We conducted experiments with two MonoInstruct settings: **MonoInstruct-full**, an objective to generate the entire sentence, and **MonoInstruct-half** for generating the latter half of the sentence given the first half, inspired by GPT [166] and MASS [204], respectively. We reported the MonoInstruct results in Table 6.5. Firstly, we observed that fine-tuning MTInstruct jointly with either MonoInstruct-full or MonoInstruct-half harms the MT performance, which could be attributed to the inherent difficulty of monolingual instruction tasks and the limited amount of monolingual data. We found that the simpler MT+Mono-half yielded better results than MT+Mono-full as richer contexts were provided. However, MonoInstruct still did not improve the MTInstruct baseline. Secondly, further combining MonoInstrcut with AlignInstruct variants yielded improvements compared with MT+Mono-full (or half), but underperformed the MTInstruct baseline. This suggested that improving MT performance with monolingual instructions is challenging without access to additional monolingual data.

6.4.4 BLOOMZ+3 Zero-shot Evaluation

Table 6.6 reports the results of the two settings, de-nl-ru and ar-de-fr-nl-ru-zh. Results of MT+Align+Hint+Revise and pivot-based translation are reported in Appendices D.1 and D.4. In the de-nl-ru setting, where BLOOMZ was fine-tuned with the three unseen languages, we noticed MT+Align consistently outperformed the MTInstruct baseline across all evaluated zero-shot directions. Notably,

Fine-tuned Languages	Objective	Zero-shot Directions				Supervised Directions			
		Directions	BLEU	chrF++	COMET	Directions	BLEU	chrF++	COMET
-	w/o fine-tuning	overall	6.89	19.14	57.95	en→xx	13.38	26.65	64.28
						xx→en	21.70	42.05	72.72
		seen→seen	16.95	30.78	74.58	en→seen	20.13	32.87	76.99
		seen→unseen	2.30	13.31	49.98	en→unseen	6.63	20.43	51.56
		unseen→seen	7.78	20.07	62.74	seen→en	26.30	48.70	78.22
		unseen→unseen	2.37	14.83	46.06	unseen→en	17.10	35.40	67.23
de-nl-ru	MTInstruct	overall	8.38	22.75	59.93	en→xx	17.05	32.02	69.26
						xx→en	25.13	45.02	76.29
		seen→seen	14.52	27.25	70.48	en→seen	17.60	29.87	73.81
		seen→unseen	6.14	22.82	54.75	en→unseen	16.50	34.17	64.70
		unseen→seen	7.56	19.22	61.99	seen→en	25.73	47.07	77.52
		unseen→unseen	6.85	23.45	54.07	unseen→en	24.53	42.97	75.06
	MT+Align	overall	**8.86**	**23.30**	**60.70**	en→xx	16.63	31.73	68.79
						xx→en	**25.62**	**45.37**	**76.45**
		seen→seen	**14.77**	**27.80**	**71.07**	en→seen	15.80	28.47	72.35
		seen→unseen	**6.31**	**23.08**	**54.81**	en→unseen	**17.47**	**35.00**	**65.24**
		unseen→seen	**8.61**	**20.24**	**63.81**	seen→en	**25.90**	**47.13**	77.47
		unseen→unseen	**7.15**	**23.70**	**54.51**	unseen→en	**25.33**	**43.60**	**75.43**
ar-de-fr-nl-ru-zh	MTInstruct	overall	11.79	26.36	63.22	en→xx	21.18	35.52	70.86
						xx→en	28.35	48.00	77.30
		seen→seen	22.68	35.32	76.39	en→seen	26.20	37.77	78.22
		seen→unseen	7.10	24.50	55.18	en→unseen	16.17	33.27	63.50
		unseen→seen	12.56	24.74	68.83	seen→en	31.97	52.93	79.72
		unseen→unseen	6.78	22.62	53.69	unseen→en	24.73	43.07	74.88
	MT+Align	overall	**12.13**	**26.65**	**63.23**	en→xx	**21.33**	**35.65**	**70.99**
						xx→en	**28.60**	**48.27**	**77.49**
		seen→seen	**23.67**	**36.53**	**76.89**	en→seen	**26.30**	37.63	**78.25**
		seen→unseen	**7.27**	24.32	54.96	en→unseen	**16.37**	**33.67**	**63.73**
		unseen→seen	**12.92**	**25.29**	**69.10**	seen→en	**32.03**	**53.07**	**79.93**
		unseen→unseen	6.68	22.30	53.19	unseen→en	**25.17**	**43.47**	**75.05**

Table 6.6: **Results of BLOOMZ+3 without fine-tuning or fine-tuned with MTInstruct, or MT+Align.** Scores that surpass the MTInstruct baseline are marked in **bold**. xx includes seen and unseen languages.

MT+Align enhanced the translation quality for unseen→seen and seen→unseen directions compared to w/o fine-tuning and MTInstruct, given that the model was solely fine-tuned on de, nl, and ru data. This suggested AlignInstruct not only benefits the languages supplied in the data but also has a positive impact on other languages through cross-lingual alignment supervision. In terms of supervised directions involving English, we noticed performance improvements associated with unseen languages, and regression in seen ones. The regression may be attributed

to forgetting for the absence of seen languages in fine-tuning data. Indeed, continuous exposure to English maintained the translation quality for seen→en. As LoRA is modular, the regression can be mitigated by detaching the LoRA parameters for seen languages.

The ar-de-fr-nl-ru-zh setting yielded a consistently higher translation quality across all directions when compared with the de-nl-ru setting. This improvement was expected, as all the six languages were included. Translation quality improved for when generating seen languages under the zero-shot scenario. However, the same observation cannot be made for unseen languages. This phenomenon underscored the effectiveness of AlignInstruct in enhancing translation quality for BLOOMZ's supported languages, but suggested limitations for unseen languages when mixed with supported languages in zero-shot scenarios. In the supervised directions, we found all translation directions surpassed the performance of the MTInstruct baseline. This highlighted the overall effectiveness of AlignInstruct in enhancing translation quality across a range of supervised directions.

6.4.5 How did MTInstruct and AlignInstruct Impact BLOOMZ's Representations?

This section analyzed the layer-wise cosine similarities between the embeddings of parallel sentences to understand the changes in internal representations after fine-tuning. The parallel sentences were prepared from the English-centric validation datasets. We then mean-pool the outputs at each layer as sentence embeddings and compute the cosine similarities, as illustrated in Figure 6.3. Results for BLOOMZ+3 are discussed in Appendix D.2.

We observed that, after MTInstruct fine-tuning, the cosine similarities rose in nearly all layers (Δ1, Figure 6.3). This may be interpreted as enhanced cross-lingual alignment, and as indicating the acquisition of translation capabilities. Upon further combination with AlignInstruct (Δ2, Figure 6.3), the degree of cross-lingual alignment rose in the early layers (layers 4 - 7) then diminished in the final layers (layers 29 & 30). This pattern aligned with the characteristics of encoder-decoder multilingual NMT models, where language-agnostic encoder representations with language-specific decoder representations improve multilin-

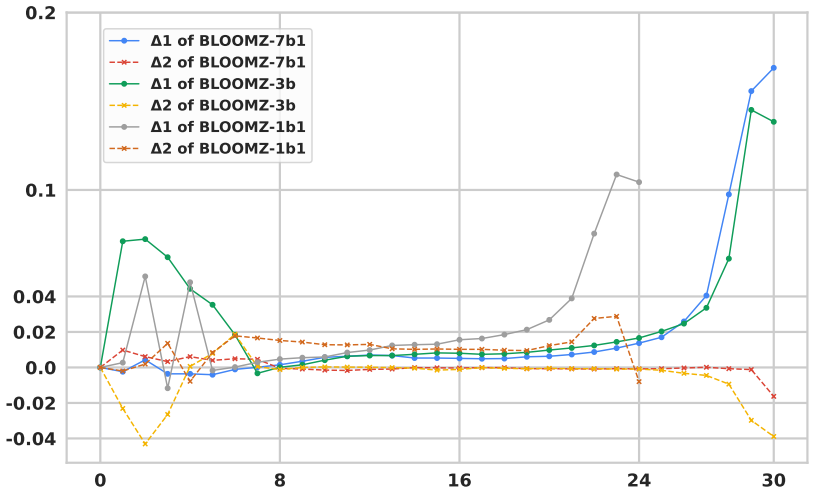

Figure 6.3: **Differences in cosine similarity of layer-wise embeddings for BLOOMZ+24.** $\Delta 1$ represents the changes from the unmodified BLOOMZ to the one on MTInstruct, and $\Delta 2$ from MTInstruct to MT+Align.

gual NMT performance [108, 236, 120]. This highlights the beneficial impact of AlignInstruct.

6.5 Summary of This Chapter

In this study, we introduced AlignInstruct for enhancing the fine-tuning of LLMs for MT in unseen, low-resource languages while limiting the use of additional training corpora. Our multilingual and zero-shot findings demonstrated the strength of AlignInstruct over the MTInstruct baseline and other instruction variants. Our future work pertains to exploring using large monolingual corpora of unseen languages for MT and refining the model capability to generalize across diverse MT prompts.

Chapter 7

Variable-length Neural Interlingua Representations for Zero-shot Neural Machine Translation

Multilingual neural machine translation (MNMT) [48, 55, 67, 81, 43] systems enable translation between multiple language pairs within a single model by learning shared representations across different languages. One of the key challenges in building effective MNMT systems is zero-shot translation performance involving unseen language pairs.

Previous work reveals that improving the language-independency of encoded representations is critical for zero-shot translation performance, with neural interlingua representations [113, 220, 270] being proposed as an effective method for achieving this. Neural interlingua representations are shared, language-agnostic representations that behave as a neural pivot between different natural languages. As shown in Figure 7.1 (a), it enables sentences in different languages with the same meaning to have the same interlingua representations. Previous work has shown the effectiveness of fixed-length neural interlingua representations for zero-shot translation. However, a fixed length can limit neural interlingua representa-

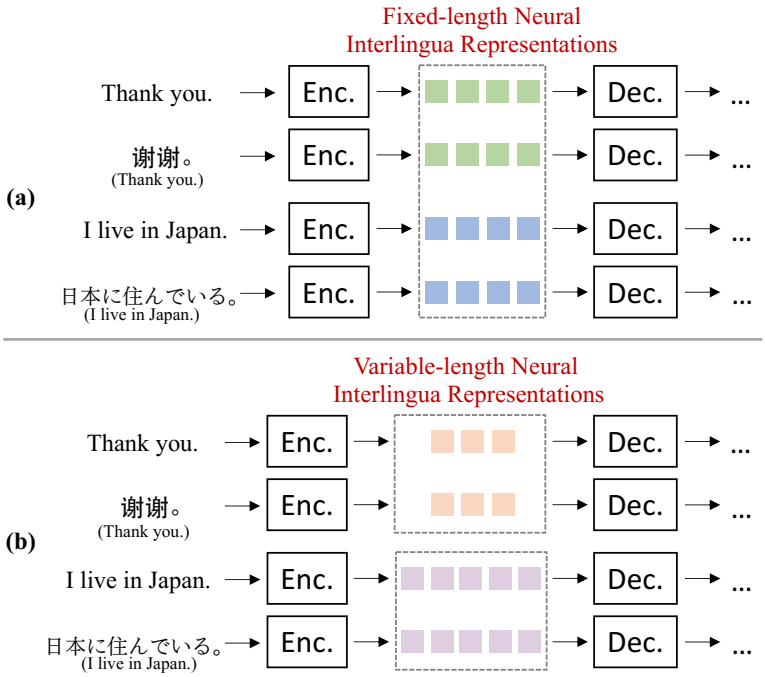

Figure 7.1: **(a) Previous fixed-length neural interlingua representations; (b) Our proposed variable-length neural interlingua representations.** Each colored box denotes the representation ($\mathbb{R}^{d\times 1}$) on the corresponding position. "Enc.", "Dec.", and "d" are encoder, decoder, and dimension of model hidden states.

tions' flexibility and representation ability. It is highly model size and training data size-sensitive according to our experimental results for different settings of model and training data size.

This study proposes a novel method for improving neural interlingua representations by making their length variable. As shown in Figure 7.1 (b), our method enables the length of the interlingua representations to vary according to different lengths of source sentences, which may provide more flexible neural interlingua representations. Specifically, we utilize the sentence length in the centric language[1] (e.g., English) as the length of neural interlingua representations. We pro-

[1]In this work, we consider using an x-centric parallel corpus, wherein all sentence pairs within the corpus consist of sentences in language x paired with another language. It is noteworthy that

pose a variable-length interlingua module to project sentences in different source languages with the same meaning into an identical neural interlingua representation sequence. To enable translating from non-centric language source sentences during inference, we also introduce a length predictor within the variable-length interlingua module. Moreover, as for the initialization of the interlingua module, we propose a novel method that facilitates knowledge sharing between different interlingua lengths, which can avoid introducing redundant model parameters. We expect that variable-length interlingua representations provide enhanced representations according to different source sentence lengths, which mitigates the model size and training data size-sensitive problem of previous work in low-resource scenarios and improves performance for zero-shot translation.

We conduct experiments on three MNMT datasets, OPUS [261], IWSLT [26], and Europarl [89] with different settings of training data size and model size. Results demonstrate that our proposed method yields superior results for zero-shot translation compared to previous work. Our method exhibits stable convergence in different settings while previous work [270] is highly sensitive to different model and training data sizes. However, we also observe the inferior performance of our method for translation from non-centric language source languages. We attribute it to the accuracy of the interlingua length predictor and point out the possible directions of this research line.

7.1 Related Work

Constructing neural interlingua representations is a powerful method to improve shared encoder representations across various source languages and enhance zero-shot translation. Lu et al. [113] first proposed the concept of neural interlingua representations for MNMT, intending to bridge multiple language-specific encoders and decoders using an intermediate interlingua attention module, which has a fixed sequence length. Vázquez et al. [220] extended this approach with a universal encoder and decoder architecture for MNMT and introduced a regular-

the English-centric corpus is the most prevalent setting. We denote a language distinct from x as a "non-centric language" in the subsequent text.

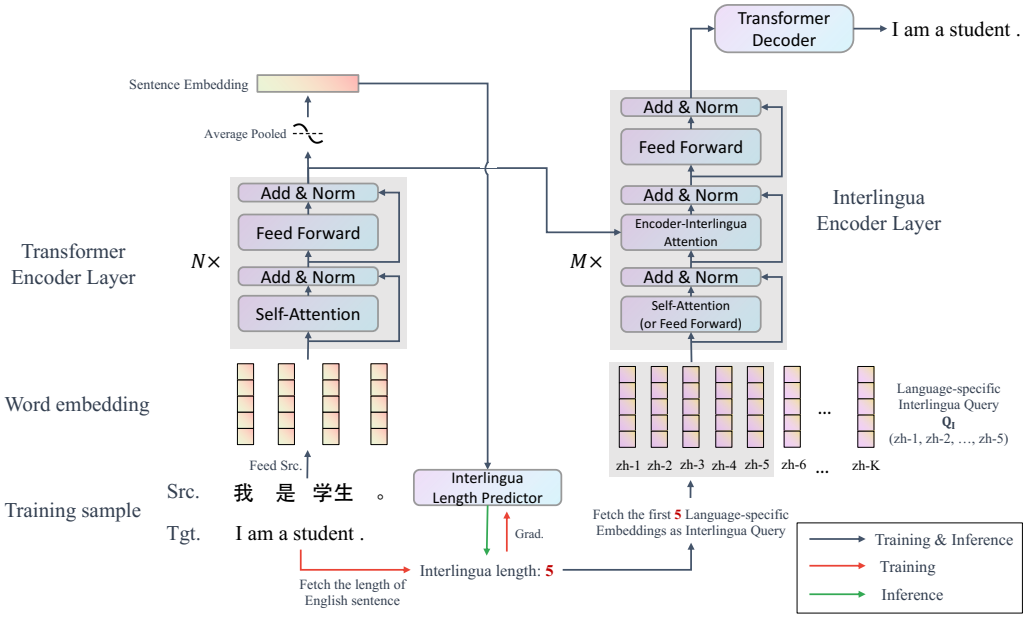

Figure 7.2: **Variable-length interlingua module.** "zh-x" denotes the x-th embedding of a Chinese-specific interlingua query.

ization objective for the interlingua attention similarity matrix. More recently, Zhu et al. [270] applied the neural interlingua approach in the Transformer [219] model architecture and proposed a position-wise alignment objective to ensure consistent neural interlingua representations across different languages. However, these methods utilized fixed-length neural interlingua representations, which may reduce the model's representation ability for source sentences with different lengths. This study focuses on revisiting and improving neural interlingua approaches.

7.2 Variable-length Interlingua Representations

We present an MNMT model that comprises three distinct components: a source language encoder, a neural interlingua module, and a decoder. The source language encoder converts source sentences to language-specific representations, the neural interlingua module generates language-agnostic representations, and the decoder converts these representations into the target language translation. In

this section, we introduce a novel neural interlingua module.

Specifically, we propose variable-length neural interlingua representations surpassing prior work's fixed-length constraint. To achieve this breakthrough, we have developed a module that includes interlingua encoder layers, an interlingua length predictor, and a language-specific interlingua query. Our module uses an embedding sharing mechanism, as shown in Figure 7.2. Moreover, we introduce the objectives that guide the training of variable-length neural interlingua representations.

7.2.1 Variable-length Interlingua Module

Interlingua Encoder Layers In accordance with Zhu et al. [270], we construct a variable-length interlingua module within a Transformer model architecture. Our model utilizes N Transformer encoder layers and 6 Transformer decoder layers, with M interlingua encoder layers introduced between them. To maintain consistency with a standard 6-layer Transformer encoder, we set $M + N = 6$, ensuring that the number of model parameters remains almost the same. Each interlingua encoder layer consists of a sequential series of operations, including self-attention mechanisms (or feed-forward networks),[2] encoder-interlingua attention, and feed-forward networks, as illustrated in Figure 7.2.

The input representations for interlingua encoder layers are denoted as $\mathbf{Q}_{\mathrm{I}} \in \mathbb{R}^{d \times \mathrm{len_I}(X)}$, where d and $\mathrm{len_I}(X)$ respectively indicates the dimension of hidden representations and the length of the neural interlingua representations given a source sentence $X = x_1, x_2, ..., x_k$. Specifically, we define $\mathrm{len_I}(X)$ as follows:

$$\mathrm{len_I}(X) = \begin{cases} \mathrm{len}(X), & X \text{is in centric} \\ \mathrm{len}(\mathrm{CT}(X)), & X \text{is in non-centric} \end{cases}, \tag{7.1}$$

where $\mathrm{CT}(X)$ denotes the translation of X in the centric language. We use teacher forcing to generate interlingua length during training. For instance, if we use

[2]We utilize feed-forward networks for the first interlingua encoder layer and employ a self-attention mechanism for subsequent layers. This is because the interlingua query is initially weak and unable to capture similarities through a self-attention mechanism. This design choice is similar to that of Zhu et al. [270].

English-centric parallel sentences as training data, $\text{len}_\text{I}(X)$ for each sentence pair will be the length of English sentences. Thus, sentences that convey the same semantic meaning can have the same interlingua length, and interlingua length is variable according to different sentences. For the initialization of $\mathbf{Q}_\text{I}$, we will provide a detailed explanation of how to generate it later in this section.

Subsequently, $\mathbf{Q}_\text{I}$ undergoes self-attention (or feed-forward networks), and we obtain the output $\mathbf{Q}_\text{I}^{'}$. Assume that the contextualized representations on top of N Transformer encoder layers are $\mathbf{H}_\text{S} \in \mathbb{R}^{d \times k}$. Then we establish an encoder-interlingua attention mechanism:

$$\mathbf{H}_\text{EI} = \text{Attn}(\mathbf{Q}_\text{I}^{'}, \mathbf{H}_\text{S}, \mathbf{H}_\text{S}), \tag{7.2}$$

where $\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ indicates the multi-head attention mechanism [219]. This encoder-interlingua attention inherits the design in previous studies of neural interlingua representations [113, 220, 270].

Finally, we pass $\mathbf{H}_\text{EI}$ through position-wise feed-forward networks to obtain $\mathbf{H}_\text{I}$, the output of the interlingua encoder layers. $\mathbf{H}_\text{I}$ serves as a language-agnostic neural interlingua and can vary in length depending on the source sentence. Once we have $\mathbf{H}_\text{I}$, we feed it into a standard Transformer decoder to generate the translation.

Interlingua Length Predictor Length of interlingua representations is not readily available during inference when translating from non-centric source sentences (e.g., non-English source sentences) using Equation (7.1). To address this, we propose using an interlingua length predictor to obtain $\text{len}_\text{I}(X)$ for inference. Specifically, we treat the length prediction of translation in the centric language as a classification task, addressed utilizing mean pooled contextualized representations atop the Transformer encoder.[3] More precisely, we predict X's interlingua length as:

$$\text{len}_\text{I}(X) = \underset{i}{\text{argmax}} \ \text{softmax}(\frac{\mathbf{1}\ \mathbf{H}_\text{S}^\text{T}}{k}\mathbf{W} + \mathbf{b})_i, \tag{7.3}$$

where k is the length of X, $\mathbf{1} \in \mathbb{R}^{1 \times k}$ denotes a vector with all the elements of 1, $\mathbf{W} \in \mathbb{R}^{d \times K}$ and $\mathbf{b} \in \mathbb{R}^{1 \times K}$ indicates the weight and bias of a linear layer, and K

[3] We attempted to treat it as a regression task, but the performance of the regression model was notably inferior to that of the classifier-based predictor.

is the maximum sequence length allowed in the model.

Language-specific Interlingua Query Here, we present the method for obtaining input representations $\mathbf{Q}_{\mathrm{I}}$ for the interlingua encoder layers. Initially, we randomly initialize an embedding matrix $\mathbf{E}_l \in \mathbb{R}^{d \times K}$ containing K embeddings for the source language l. Next, we extract the first $\mathrm{len}_{\mathrm{I}}(X)$ embeddings from $\mathbf{E}_l$ to obtain $\mathbf{Q}_{\mathrm{I}}$.

$$\mathbf{Q}_{\mathrm{I}} = \mathbf{E}_l \mathbf{I}_{\mathrm{S}}, \tag{7.4}$$

where $\mathbf{I}_{\mathrm{S}} \in \mathbb{R}^{K \times \mathrm{len}_{\mathrm{I}}(X)}$ has 1s as main diagonal elements and 0s for other elements. Note that the language-specific nature of $\mathbf{E}_l$ allows the model to learn a unique mapping from each language to the neural interlingua representations. Zhu et al. [270] used the technique of language-aware positional embedding [229] for both the neural interlingua representations and the source and target sentences, resulting in ambiguity regarding whether the improvements were from the neural interlingua representations or not. In contrast, our proposed language-specific interlingua query clarifies whether a language-specific mapping to neural interlingua representations benefits zero-shot translation.

7.2.2 Training Objectives

Given a training sample sentence pair (X, Y), we introduce the following training objective, combining an NMT loss, an interlingua alignment loss, and a length prediction loss. The interlingua alignment loss is utilized to guarantee the consistency of the neural interlingua representations for each training sentence pair sample. In contrast, the length prediction loss ensures the generation of variable interlingua length during inference. Specifically, the training objective is defined as follows:

$$\mathcal{L}(X, Y) = \alpha \mathcal{L}_{\mathrm{NMT}} + \beta \mathcal{L}_{\mathrm{IA}} + \gamma \mathcal{L}_{\mathrm{LP}}, \tag{7.5}$$

where α, β, and γ are weight hyperparameters for each loss, $\mathcal{L}_{\mathrm{LP}}$ is a cross-entropy loss computed from the softmax outputs from Equation (7.3), and $\mathcal{L}_{\mathrm{IA}}$ is a position-wise alignment loss using cosine similarity following Zhu et al. [270]:

$$\mathcal{L}_{\mathrm{IA}} = 1 - \frac{1}{\mathrm{len}_{\mathrm{I}}(X)} \sum_i \cos < \mathbf{H}_{\mathrm{I}}(X)_i, \mathbf{H}_{\mathrm{I}}(Y)_i > . \tag{7.6}$$

Datasets	Languages	# Sup.	# Zero.	# Train	# Valid	# Test
OPUS	ar, de, en, fr, nl, ru, zh	12	30	12,000,000	2,000	2,000
IWSLT	en, it, nl, ro	6	6	1,378,794	2,562	1,147
Europarl	de, en, es, fr, nl	8	12	15,782,882	2,000	2,000

Table 7.1: **Statistics of the training data**. "# Sup." and "# Zero." indicate the respective number of language pairs for supervised and zero-shot translation. "# Train" denotes the total number of the training parallel sentences while "# Valid" and "# Test" showcase the number per language pair.

Here $\mathbf{H}_{\mathrm{I}}(\cdot)_i$ denotes the i-th column of $\mathbf{H}_{\mathrm{I}}(\cdot)$.[4] Please note that during training, we always have $\mathrm{len}_{\mathrm{I}}(X) = \mathrm{len}_{\mathrm{I}}(Y)$ because we apply teacher forcing to generate the interlingua length for the sentence pair (X, Y). With $\mathcal{L}_{\mathrm{IA}}$, different sentence pairs with varying lengths of translation in centric language can be represented using variable-length neural interlingua representations. This can enhance the bridging ability for zero-shot translation.

7.3 Experimental Settings

7.3.1 Datasets

Our study involves conducting experiments on zero-shot translation using three distinct datasets, OPUS [261], IWSLT [26], and Europarl [89], each comprising 7, 4, and 5 languages, respectively. For each dataset, we adopt the train, valid, and test splits following Zhang et al. [261], Wu et al. [236], and Liu et al. [108]. Table 7.1 presents each dataset's overall statistics. The training and validation data exclusively contains English-centric sentence pairs, indicating the centric language is English in all the experiments, leading to 12, 6, and 8 supervised directions, and 30, 6, and 12 zero-shot directions for each dataset. Jieba[5] is used

[4] To derive $\mathbf{H}_{\mathrm{I}}(Y)$, it is necessary to feed the target sentence to both the encoder and interlingua encoder layers, which can potentially result in increased computational requirements.

[5] `https://github.com/fxsjy/jieba`

to segment Chinese while Moses[6] [90] is utilized to tokenize other languages. We employ BPE [190] with 50,000, 40,000, and 50,000 merge operations to create a joint vocabulary for each dataset, resulting in the vocabulary sizes of 66,158, 40,100, and 50,363, respectively.

7.3.2 Overall Training and Evaluation Details

For the OPUS and IWSLT datasets, we utilize a `Transformer-base` model, while for Europarl, we employ a `Transformer-big` model, to evaluate the performance of Transformer with both sufficient and insufficient training data. Regarding language tag strategies to indicate the source and target languages to the model, we adopt the method of appending the source language tag to the encoder input and the target language tag to the decoder input [110]. This approach allows for the creation of fully language-agnostic neural interlingua representations in between.[7] The maximum sentence length is set as 256, which indicates that $K = 256$ (Section 7.2.1). Our models are trained using Fairseq.[8] As the data size for each language pair is relatively similar, oversampling is not implemented for MNMT. The dropout rate was set to 0.1, 0.4, and 0.3 for each dataset, and we use the Adam optimizer [86] with a learning rate of 5e-4, 1e-3, and 5e-4, respectively, employing 4,000 warm-up steps. The `Transformer-base` model was trained using four 32 GB V100 GPUs, and the `Transformer-big` model was trained using eight 32 GB V100 GPUs, with a batch size of 4,096 tokens. To speed up training, mixed precision training [127] is also employed. Each dataset is trained for 500, 200, and 500 epochs.

For evaluation, we choose the evaluation checkpoint based on the validation $\mathcal{L}_{\text{NMT}}$ with the lowest value. We use a beam size of 5 during inference on the trained models to conduct inference. We report SacreBLEU [161].[9]

[6] `https://github.com/moses-smt/mosesdecoder`

[7] We do not consider employing target language tag appending on the encoder-side [81] in this work because it would require removing both the source and target language information after feeding the source sentence to obtain the neural interlingua representations.

[8] `https://github.com/facebookresearch/fairseq`

[9] We utilize the "zh" tokenization mode for Chinese, and the "13a" tokenization mode for other languages.

7.3.3 Baselines and Respective Training Details

To compare our variable-length neural interlingua method with previous fixed-length neural interlingua methods, we trained the following settings:

MNMT [81] denotes the system trained with standard `Transformer-base` or `Transformer-big` for multiple language pairs. We applied the language tag strategy of source language tag for encoder input and target language tag for decoder input.

Pivot translation [272] involves translating a source language into a pivot language, usually English, and then translating the pivot language into the target language. This system constitutes a robust baseline for zero-shot translation, which we include for reference. We implement this setting by feeding the pivot language output of the MNMT model to itself to generate the target language.

Len-fix. Uni. Intl. We follow the setting described by Zhu et al. [270], but we remove its language-aware positional embedding to test whether a single interlingua module can improve zero-shot translation. Compared to our variable-length interlingua representations presented in Section 7.2.1, these fixed interlingua representations have a universal len_I (Equation (7.1)) for different source sentences and a universal $\mathbf{E} \in \mathbb{R}^{d \times \text{len}_\text{I}}$ for different languages and without a $\mathbf{Q}_\text{I}$ (Equation (7.4)). The fixed interlingua length is set to 17, 21, and 30, which are the average lengths of each dataset following Zhu et al. [270] and Vázquez et al. [220].

Len-fix. LS. Intl. The only difference between this system and the "Len-fix. Uni. Intl." system mentioned above is the initialization of the interlingua query. We use a language-specific $\mathbf{E}_l \in \mathbb{R}^{d \times \text{len}_\text{I}}$ for each source language l without a $\mathbf{Q}_\text{I}$ (Equation (7.4)).

Len-vari. Intl. (ours) This refers to variable-length neural interlingua representations proposed in Section 7.2.

For the last three neural interlingua settings, we set M and N to 3 for both the Transformer encoder and interlingua encoder layers. The values of α, β, and γ (Equation (7.5)) are set as 1.0, 1.0, and 0.1, respectively. We remove the first residual connection within the first interlingua encoder layer to improve the language-independency of the interlingua representations, inspired by Liu et al. [108].

Methods	Zero-shot			Supervised: From en			Supervised: To en		
	OPUS	IWSLT	Europarl	OPUS	IWSLT	Europarl	OPUS	IWSLT	Europarl
Pivot	*22.0*	*19.9*	*29.5*	-	-	-	-	-	-
MNMT	16.5	13.1	29.0	**31.2**	**29.6**	**32.9**	**36.8**	**33.5**	**36.1**
Len-fix. Uni. Intl.	18.2	12.7	17.4	29.6	19.6	20.1	35.3	22.2	21.8
Len-fix. LS. Intl.	18.4	4.7	5.8	30.1	7.3	6.7	35.7	12.9	7.1
Len-vari. Intl. (ours)	**18.9**†	**14.8**	**29.6**	30.2†	26.2	32.6	34.0	27.1	33.8

Table 7.2: **Overall BLEU results on OPUS, IWSLT, and Europarl**. The best result among all the settings except *Pivot* is in **bold**. We mark the results significantly [88] better than "Len-fix. Uni. Intl." with † for OPUS dataset.

Methods	de–fr		ru–fr		nl–de		zh–ru		zh–ar		nl–ar		Zero-shot
	→	←	→	←	→	←	→	←	→	←	→	←	Avg.
Pivot	*23.4*	*21.2*	*31.0*	*26.0*	*21.8*	*23.6*	*24.8*	*37.9*	*24.0*	*38.9*	*7.4*	*17.4*	*22.0*
MNMT	17.6	15.0	21.5	17.7	17.9	21.4	15.3	27.6	18.0	28.6	5.3	13.3	16.5
Len-fix. Uni. Intl.	20.1	17.0	25.0	22.4	19.5	21.3	20.3	30.9	19.6	30.4	6.1	14.4	18.2
Len-fix. LS. Intl.	**20.7**	17.7	25.7	21.7	19.8	21.6	19.9	31.5	**20.1**	31.6	**6.5**	**14.5**	18.4
Len-vari. Intl. (ours)	20.6†	**18.3**†	**26.0**†	**23.4**†	**20.2**†	**22.1**†	**20.8**	**31.8**†	20.0	**31.9**†	6.3	**14.5**	**18.9**†

Table 7.3: **BLEU results of zero-shot translation on OPUS**. We randomly select six zero-shot language pairs and report the results. The best result among all the settings except "*Pivot*" is in **bold**. We mark the results significantly [88] better than "Len-fix. Uni. Intl." with †.

7.4 Results and Analysis

We now present in tables 7.2, 7.3, and 7.4 the results of our variable-length interlingua approach and compare them against several baselines.

7.4.1 Main results

Firstly, Tables 7.2 and 7.3 indicate that our proposed variable-length interlingua representations outperform previous work in zero-shot directions. The severe overfitting issue of "Len-fix. Uni. Intl." and "Len-fix. LS. Intl." on IWSLT and Europarl suggests that they are limited to model size and training data size settings, while our proposed method can converge stably on all three settings. These results demonstrate that our flexible interlingua length can benefit zero-

Methods	en–ar		en–de		en–fr		en–nl		en–ru		en–zh		Supervised Avg.	
	→	←	→	←	→	←	→	←	→	←	→	←	From en	To en
MNMT	**23.9**	**37.8**	**30.8**	**34.6**	**33.9**	**35.5**	**27.8**	**31.5**	29.4	**35.1**	**41.2**	**46.4**	**31.2**	**36.8**
Len-fix. Uni. Intl.	22.6	36.6	28.9	33.0	31.7	33.5	27.4	30.1	28.4	34.0	38.8	44.6	29.6	35.3
Len-fix. LS. Intl.	22.9	36.8	29.0	33.8	32.3	33.9	27.7	30.6	28.9	34.3	39.5	44.8	30.1	35.7
Len-vari. Intl. (ours)	23.3†	33.8	30.1†	32.3	32.9†	32.6	27.3	27.9	**29.5**†	32.2	38.0	45.3†	30.2†	34.0

Table 7.4: **BLEU results of supervised translation on OPUS**. The best result among all the settings is in **bold**. We mark the results significantly [88] better than "Len-fix. Uni. Intl." with †.

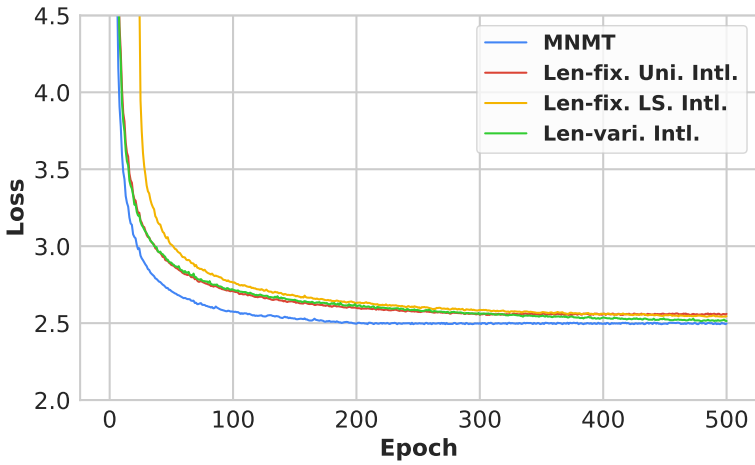

Figure 7.3: Validation NMT loss curve on OPUS.

shot translation more effectively. Secondly, our proposed method performs better than previous work in "from en" supervised directions as shown in Tables 7.2 and 7.4, but still falls short of the MNMT baseline. This may be attributed to the interlingua module's weak source-target awareness. Thirdly, our variable-length neural interlingua representations perform significantly worse on "to en" directions than "Len-fix." methods on OPUS and MNMT on all datasets. We provide analysis of this phenomenon next.

7.4.2 Validation NMT Loss

We investigate why variable-length neural interlingua representations perform poorly in "to en" supervised directions by analyzing the validation NMT loss,

	ar	de	fr	nl	ru	zh	Avg.
Acc. of Len. Pre.	20.6	26.5	17.6	19.3	21.1	13.8	19.8
Avg. of | Len. Pre. − *gold* |	2.4	3.4	3.8	3.1	3.3	3.9	3.3
BLEU w/ Len. Pre.	33.8	32.3	32.6	27.9	32.2	45.3	34.0
BLEU w/ *gold*	35.5†	33.4†	33.3†	29.4†	33.4†	46.0†	35.2†

Table 7.5: **Accuracy of the interlingua length predictor, averaged absolute difference between predicted length and *gold* length, and "to en" BLEU scores of each non-English source language on OPUS.** "w/ Len. Pre." and "w/ *gold*" indicate using the predicted interlingua length and the correct interlingua length (length of the English translation), respectively. Accuracy of the length predictor and average abosulute difference are evaluated using OPUS's test set. We mark the results significantly [88] better than "BLEU w/ Len. Pre." with †.

an approximate measure of NMT performance on the validation set. Figure 7.3 displays the validation NMT loss for all settings on OPUS. We observe that variable-length interlingua representations can converge well, even smaller than the validation loss of "Len-fix. Uni. Intl." and "Len-fix. LS. Intl." However, the interlingua length predictor was teacher-forced during training, indicating the validation NMT loss was calculated with a 100% accurate interlingua length predictor. As a result, the inaccurate interlingua length predictor is likely the primary cause of our method's inferior performance in "to en" directions, despite its well-converged validation NMT loss.

7.4.3 Impact of the Interlingua Length Predictor

We analyze the interlingua length predictor and identify the reason for the subpar performance in "to en" translations. We input the source sentences of the test set in non-English languages into the model and check whether the predicted length in interlingua is identical to the length of its English reference. We present the accuracy on the OPUS dataset in Table 7.5. The results show that the accuracy for each language is approximately 20.0%, which can result in error propagation when

translating from those languages. To further understand the impact of the length predictor quality on translation performance, we attempt to provide the model with the correct interlingua length instead of relying on the length predictor. As shown in Table 7.5, the results reveal significant BLEU improvements when the correct interlingua length is applied. This suggests that the performance issue encountered when translating from a non-centric source language can be addressed by upgrading the interlingua length predictor's accuracy. Furthermore, we can also enhance zero-shot translation performance if we have a better length predictor. Nevertheless, we observe that even with a low length prediction accuracy of approximately 20.0%, we can still achieve solid BLEU performance, averaging 34.0 BLEU points. This indicates that an incorrectly predicted length with just a trivial difference, as shown in Table 7.5, will not result in the enormous information loss required for translation.

7.5 Summary of This Chapter

This study introduced a novel variable-length neural interlingua approach that improved zero-shot translation results while providing a more stable model than previous fixed-length interlingua methods. Although our analysis revealed a performance downgrade in "to en" directions, we have identified the problematic model component and plan to address it in future studies.

Chapter 8

Exploring the Impact of Layer Normalization for Zero-shot Neural Machine Translation

Multilingual neural machine translation (MNMT) enables translation between unseen language pairs, i.e., zero-shot translation (ZST) [81, 56]. Prior studies have explored techniques such as language tags [236], residual connections [108], and novel training objectives [4, 155, 7, 64, 270, 261, 228, 246, 125] for improving ZST. They primarily used the Transformer architecture [219], which has two variations depending on the position of layer normalization (LayerNorm) [14], namely, PreNorm (applied at the input of layers) [15] and PostNorm (applied after residual connections), as shown in Figure 8.1. As previous studies showed that PreNorm can result in more stable training and faster convergence compared to PostNorm for MNMT [237], most ZST works [155, 236, 108] use PreNorm as the default setting following those MNMT studies. However, Xu et al. [240] revealed that PreNorm carries the risk of overfitting the training data. We thus hypothesize that in a multilingual scenario, PreNorm may overfit supervised directions and have poor ZST generalizability. We systematically explore PreNorm and PostNorm's effect on ZST to verify this.

Using the OPUS, IWSLT, and Europarl datasets and a total of 54 ZST di-

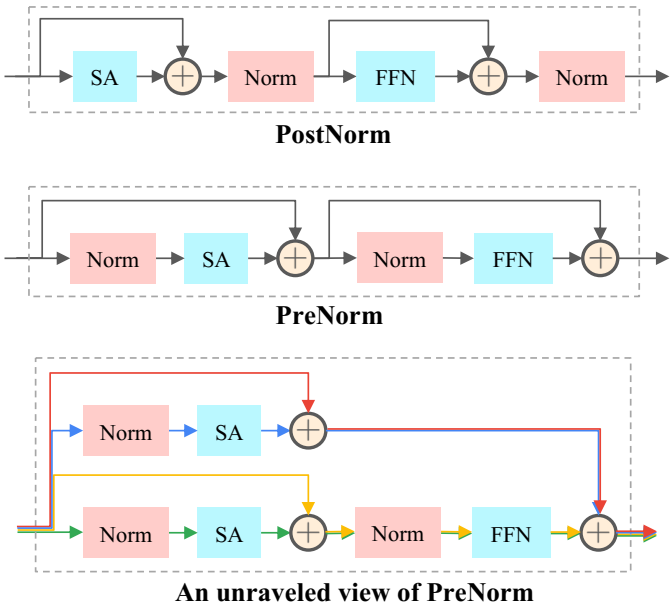

Figure 8.1: **PostNorm, PreNorm, and an unraveled view of PreNorm in a Transformer encoder layer**. "Norm," "SA," and "FFN" denote LayerNorm, self-attention, and feed-forward network. $\oplus$ is residual connection. Paths with different colors in the unraveled view of PreNorm indicate respective sub-networks.

rections, we show that PostNorm consistently outperforms PreNorm by up to 12.3 BLEU points. Following previous work, we also evaluate different language tag [236] and residual connection [108] settings, as they have been shown to impact ZST but we observe that PostNorm continues to be superior thereby lending credibility to our hypothesis.

To better understand the performance differences, we introduce a novel analysis approach called **layer-wise language recognition** (**LLR**), which tracks the off-target rates for each encoder and decoder layer by training token-level classifiers to recognize the source or target language. This analysis shows that PreNorm is more sensitive to language tag settings than PostNorm, negatively impacting ZST performance. Additionally, by examining the unraveled view of PreNorm (Figure 8.1) inspired by Veit et al. [221], we reveal structural flaws in PreNorm for ZST. Our analysis demonstrates that the order of LayerNorm and self-attention/feed-forward network in PreNorm is the main factor affecting its

ZST performance.

Given the prevalent use of PreNorm as the default setting in ZST baselines and frameworks such as Fairseq [146][1] and Tensor2Tensor [218], our study emphasizes the importance of careful consideration in the LayerNorm setting for ZST.

8.1 Background: LayerNorm

LayerNorm [14] normalizes the input $\mathbf{x}$ by zero-centering and scaling to have a unit standard deviation, followed by an additional trainable transformation, including a gain and bias adjustment. Specifically, it is formulated as:

$$\text{LayerNorm}(\mathbf{x}) = \frac{\mathbf{x} - \mathbf{E}(\mathbf{x})}{\sqrt{\mathbf{V}(\mathbf{x})}} \cdot \mathbf{g} + \mathbf{b}, \tag{8.1}$$

where $\mathbf{g}$ and $\mathbf{b}$ are trainable gain and bias. $\mathbf{E}$ and $\mathbf{V}$ indicate expectation and variance. LayerNorm is commonly used in two positions in the Transformer, as shown in Figure 8.1. PostNorm, which is the originally proposed setting of the Transformer [219], involves applying LayerNorm after each sub-module (i.e., self-attention or feed-forward network) and residual connections. PreNorm [15], on the other hand, involves applying LayerNorm directly before each sub-module and is known to stabilize Transformer training. While variants of Transformer LayerNorm like RMSNorm [260] have been proposed, the vanilla PreNorm and PostNorm are still the most widely adopted settings in current multilingual NMT literature. Therefore, we only focus on PreNorm and PostNorm in this work.

Nguyen and Salazar [143] have explored the impacts of normalization and initialization choices on supervised low-resource NMT settings, however, we delve deeper and focus on the significance of the positioning of LayerNorm for zero-shot NMT. We expect this to complete the understanding of LayerNorm's role in multilingualism, particularly in the context of zero-shot translation.

[1] `https://github.com/facebookresearch/fairseq/tree/main/examples/multilingual`

Datasets	Languages	$\mathbf{N}_{zero}$	$\mathbf{S}_{train}$	Arch.
OPUS	ar, de, en, fr, nl, ru, zh	30	12.00M	`base`
IWSLT	en, it, nl, ro	6	1.38M	`base`
Europarl	de, en, es, fr, nl	12	15.78M	`big`

Table 8.1: **Statistics of the training data**. $\mathbf{N}_{zero}$ and $\mathbf{S}_{train}$ denote number of the ZST directions and size of the training data, respectively. `base` and `big` indicate `Transformer-base` and `Transformer-big`.

8.2 Experiments and Results

We evaluate the performance of PreNorm and PostNorm for ZST on various datasets and language pairs. We then analyze the off-target rates and structural discrepancies between PreNorm and PostNorm to understand performance differences.

8.2.1 Experimental Settings

Datasets We perform ZST experiments on three datasets, respectively. They are OPUS [261], IWSLT [26], and Europarl [89]. The statistics of the datasets are summarized in Table 8.1. We include 7, 4, and 5 languages for each dataset. The training data consists of only English-centric sentence pairs, resulting in 30, 6, and 12 ZST directions for each dataset. The total number of parallel sentences for each dataset is 12.00M, 1.38M, and 15.78M, respectively. We apply BPE [190] with merge operations of 50k, 40k, and 50k to create a joint vocabulary for each dataset.

Training We employ `Transformer-base` model for OPUS and IWSLT, and `Transformer-big` for Europarl, in accordance with the distinct sizes of training data. We consider the following settings:

(1) PreNorm or PostNorm: PreNorm involves LayerNorm directly before each sub-module (i.e., self-attention or feed-forward network), while PostNorm applies LayerNorm after each sub-module and residual connections, as shown in

Figure 8.1.[2]

(2) S-ENC-T-DEC or T-ENC: Source language tag on the encoder-side and target language tag on the decoder-side; or only target language tag on the encoder-side. Wu et al. [236] showed that this setting impacts ZST for Transformer with PreNorm.

(3) w/ or w/o Res.: With the residual connection for self-attention in the middle (4^{th}) encoder layer or not. Liu et al. [108] revealed that "w/o Res." improves ZST for the model trained with PreNorm. We experiment this with different LayerNorm settings as this may reduce the potential of overfitting on supervised directions, then further impacts ZST, which aligns with our hypothesis.

The settings above lead to eight different combinations, shown in Table 8.2 (#1 - #8). For data preprocessing, we utilize jieba[3] for Chinese segmentation and Moses[4] [90] for tokenization of other languages. After applying BPE, we obtain vocabularies with sizes of $66,158$, $40,100$, and $50,363$ for OPUS, IWSLT, and Europarl, respectively. For multilingual training, we do not apply oversampling as the data size for each language pair is comparable. The maximum sentence length is set to 256. We train Transformer models using Fairseq[5] and set the dropout rate to 0.1, 0.4, and 0.3 for each dataset. Adam [86] is used as the optimizer with a learning rate of 5e-4, 1e-3, and 5e-4 for each dataset, and $4,000$ warm-up steps are employed. We train the Transformer-base model using 4 32G V100 GPUs and the Transformer-big model using 8 32G V100 GPUs with the batch size of $4,096$ tokens. Additionally, we employ mixed precision training [127] to accelerate the training process. We train each dataset for 200, 100, and 400 epochs, respectively.

8.2.2 Main Results

We evaluate ZST systems using SacreBLEU [161] and off-target rates. We report in Table 8.2 BLEU scores for both zero-shot and supervised directions. For ZST, we also present pivot-based translation results as a reference. For OPUS, we use

[2]We also experiment with the setting of LayerNorm without trainable parameters [240] in Appendix E.3.

[3]`https://github.com/fxsjy/jieba`

[4]`https://github.com/moses-smt/mosesdecoder`

[5]`https://github.com/facebookresearch/fairseq`

#	Layer Norm	Language Tag	Res.	Zero-shot OPUS	Zero-shot IWSLT	Zero-shot Europarl	Supervised OPUS	Supervised IWSLT	Supervised Europarl
0		*Pivot*		21.8	20.0	29.5	-	-	-
1	**PreNorm**	S-ENC-T-DEC	w/	10.1 (42.19%)	4.9 (64.84%)	24.9 (7.73%)	33.7	31.5	34.3
2	**PostNorm**	S-ENC-T-DEC	w/	**16.8** (**8.59%**)	**12.4** (**10.61%**)	**29.2** (**0.34%**)	33.9	31.5	34.5
3	**PreNorm**	T-ENC	w/	13.3 (22.99%)	13.7 (**3.98%**)	29.5 (0.23%)	33.7	31.6	34.4
4	**PostNorm**	T-ENC	w/	**14.0** (**22.86%**)	**15.5** (4.59%)	**30.8** (**0.11%**)	34.1	31.5	34.5
5	**PreNorm**	S-ENC-T-DEC	w/o	14.3 (20.67%)	8.0 (50.16%)	16.7 (41.87%)	33.6	30.9	34.3
6	**PostNorm**	S-ENC-T-DEC	w/o	**16.0** (**15.27%**)	**17.4** (**1.83%**)	**29.0** (**0.41%**)	33.8	30.7	34.4
7	**PreNorm**	T-ENC	w/o	13.4 (27.15%)	16.2 (1.54%)	29.9 (2.15%)	33.5	30.9	34.3
8	**PostNorm**	T-ENC	w/o	**13.9** (**26.68%**)	**17.8** (**1.50%**)	**30.8** (**0.13%**)	33.9	30.6	34.4

Table 8.2: **BLEU scores and off-target rates (shown in brackets)**. We report the average score of three seeds; refer to Appendix E.5 for BLEU score of each translation direction and seed. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer. We mark lower off-target rates and significantly higher BLEU scores [88] between PreNorm and PostNorm in **bold** for ZST.

the test sets following Zhang et al. [261], while for IWSLT and Europarl, we choose the test sets following Wu et al. [236]. We select the checkpoint with the lowest validation loss for evaluation. The inference is performed on the trained models using a beam size of 5. For calculating SacreBLEU,[6] we utilize the "zh" tokenization mode for Chinese, and the "13a" tokenization mode for other languages. We use the model of setting #4[7] (Table 8.2) for pivot-based translation. To calculate the off-target rates, we utilize the language identification tool provided by FastText [83].[8] Our experiment has revealed that this tool is slightly more accurate than another tool called "langdetect,"[9] as it can achieve an accuracy of 98% when decoding reference English sentences in the test set, whereas "langdetect" only achieves accuracy of around 92%. Our findings are as follows:

PreNorm vs. PostNorm: We find that PostNorm consistently yields better BLEU scores than PreNorm for ZST across various language tag and residual con-

[6] `https://github.com/mjpost/sacrebleu`

[7] We use this setting as it achieves the best performance for supervised directions, as shown in Table 8.2.

[8] `https://fasttext.cc/docs/en/language-identification.html`

[9] `https://github.com/Mimino666/langdetect`

nection settings, while their performance is comparable for supervised directions.

Impact of Language Tag and Residual Connection: We observe that using the "T-ENC" language tag and "w/ Res." improves ZST performance for IWSLT, which aligns with the findings of Wu et al. [236] and Liu et al. [108]. Nevertheless, the best performance is achieved using "w/ Res." for PostNorm with "S-ENC-T-DEC" and "T-ENC" tags for OPUS and Europarl, respectively (#2 and #4). Given that Wu et al. [236] and Liu et al. [108] used PreNorm as the default setting (#2, #4, #6 and #8 are unreported results in their work), our results emphasize the need to consider PostNorm as the default setting for ZST, while the language tag and residual connection settings have less impact.

Off-target Rates: Off-target rates help understand the different BLEU score gaps between PreNorm and PostNorm, which ranges from 0.5 to 12.3 BLEU points. For PreNorm and PostNorm with the "T-ENC" language tag (#3, #4, #7, and #8), they have similar off-target rates, with a discrepancy ranging from -0.61% to 2.02%, which results in narrow BLEU score gaps, ranging from 0.5 to 1.8 points. However, for PreNorm and PostNorm with the "S-ENC-T-DEC" language tag (#1, #2, #5, and #6), the off-target rates show a more considerable discrepancy, ranging from 5.40% to 54.23%, resulting in BLEU score gaps from 1.7 to 12.3 points. Further analysis of the nature of Transformer hidden states in the next section explores the reason for these different off-target rates in translations.

8.2.3 Tracking Off-targets within Transformer

We probe the language independence of hidden states to track off-targets within Transformer and reveal the differences between PreNorm and PostNorm. In previous work, language independence was primarily analyzed using either SVCCA [168] or language classification accuracy (LCA) [108]. However, we provide evidence in Appendix E.1 that SVCCA, which measures the cosine similarity between hidden states, are not suitable for ZST systems. Instead, LCA trains a classifier to inspect the hidden states on top of the encoder, but it does not simulate the training of a ZST system, which may introduce bias in the analysis for ZST.[10] In this work,

[10]Liu et al. [108] regulate the output language via a decoder-side language tag, hence analyzing only the encoder states poses no issues as the target language tag does not impact them.

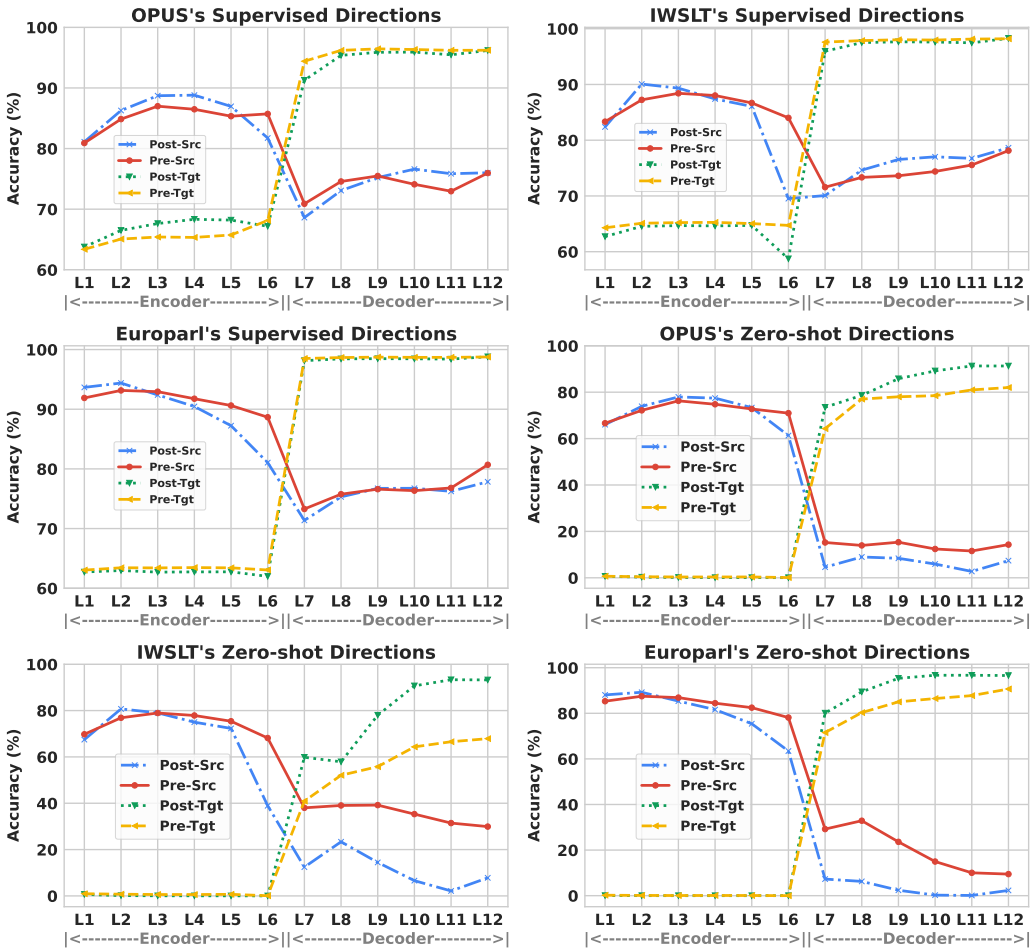

Figure 8.2: **The LLR results of #1 and #2 (Table 8.2) for both ZST and supervised directions for each dataset**. We report the average accuracy of three seeds and all the supervised or zero-shot directions. "Pre-Src" and "Pre-Tgt" indicate the layer-wise source and target language recognition for a PreNorm system (#1), while "Post-Src" and "Post-Tgt" denote similary for a PostNorm system (#2). "L1" to "L6" are 6 encoder layers and "L7" to "L12" are 6 decoder layers. We present the figures of other systems (#3 - #8) in Appendix E.4.

Nevertheless, with other language tag settings such as S-ENC-T-DEC and T-ENC, employed in this study, we require a method to obtain hidden states properly, given their impact on hidden states.

we propose a novel approach for ZST based on LCA:

LLR tailors classifiers for each layer to recognize the source or target language. We train a token-level linear classifier for each layer to utilize hidden states in each layer as features to identify the source or target language. We use hidden states obtained by feeding sentence pairs in supervised directions to simulate the training of ZST. We then test each layer's classifer's ability to recognize the source or target language for supervised or zero-shot directions. This approach enables the trained classifier to best represent the language recognition ability of hidden states in a ZST system.

We train two types of linear classifiers for each encoder and decoder layer. One is for recognizing the source language, and the other is for the target language. Each linear classifier is a linear transformation from the dimension of the hidden states (512 or $1,024$) to the number of source or target languages (e.g., 7 for OPUS). We use the validation set of all supervised directions to obtain the hidden state of each token in each layer and set their source language tag or target language tag as the gold labels. Note that the decoder hidden state of each token in each layer is obtained auto-regressively without teacher-forcing. We train each classifier for 3 epochs[11] with a learning rate of 1e-3 and a batch size of 64 sentences. For inference, we utilize the test sets of all supervised or zero-shot directions for computing the LLR results for corresponding directions, respectively.

The LLR results for #1 and #2 in Table 8.2 are presented in Figure 8.2. First, we find that the encoder and decoder hidden states are highly correlated with the target and source languages, respectively, for supervised directions (L1 to L6 of Pre/Post-Tgt and L7 to L12 of Pre/Post-Src of 3 upper sub-figures), which may impact the generalizability for ZST. Second, we see that the encoder hidden states of PostNorm are less dependent on the source language than PreNorm (L6 of Pre/Post-Src of 3 lower sub-figures). Third, we observe that the hidden states in all the decoder layers of PostNorm are more dependent on the target language and less on the source language than PreNorm (L7 to L12 of 3 lower sub-figures). The latter two points contribute to the observed gaps in off-target rates between

[11]The classifier can fully converge within 3 epochs as the classifier is lightweight that only contains a small number of parameters.

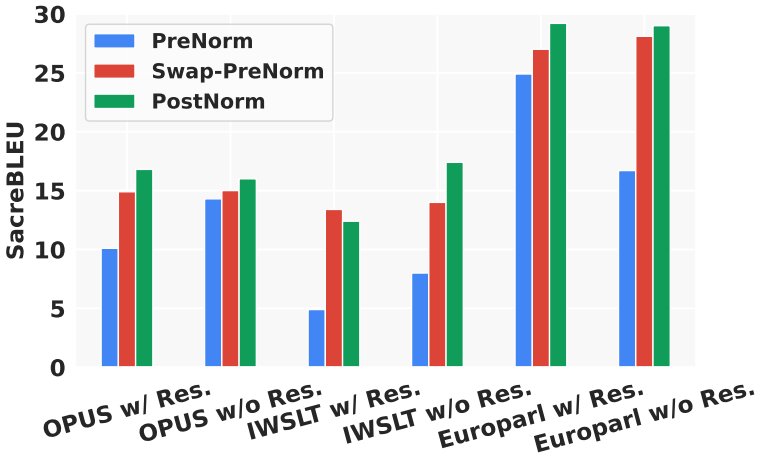

Figure 8.3: **BLEU scores of systems with "S-ENC-T-DEC" for ZST**. We report the mean of three seeds.

PreNorm and PostNorm. Conclusions for #5 and #6 with the "S-ENC-T-DEC" tag are identical (Appendix E.5).

For systems using "T-ENC," we find that the LLR are similar between PreNorm and PostNorm (Appendix E.5) and attribute the BLEU score gaps to translation quality (i.e., adequacy and fluency).

8.2.4 Unraveling Structural Flaws of PreNorm

We investigate the structural differences between PreNorm and PostNorm to explain the observed differences in hidden states for models trained with the "S-ENC-T-DEC" tag. Inspired by Veit et al. [221], we present an "unraveled view" for PreNorm, which decomposes the residual connections by the summation of several sub-networks, as shown in Figure 8.1 (paths with different colors indicate sub-networks). However, this is not applicable to PostNorm, as LayerNorm is located after residual connections. Based on this analysis, the structural characteristic of PreNorm is:

(1) Shallow Sub-network Nature: PreNorm includes shallow sub-networks, such as the embedding layer output fed through encoder layers without any operation except for the final LayerNorm (red path in Figure 8.1), but PostNorm

does not.

(2) LayerNorm Before SA/FFN: In PreNorm, LayerNorm is placed directly before the self-attention (SA) or feed-forward module (FFN) within the residual connection module.

To analyze the impact of these structural characteristics on the generalizability of PreNorm in ZST, we swap the order of LayerNorm and SA/FFN within the residual connection module (**Swap-PreNorm**), while keeping the shallow sub-network nature of PreNorm. Refer to Appendix E.2 for specific illustrations of Swap-PreNorm. The results, presented in Fig 8.3, show that PreNorm can be significantly improved through Swap-PreNorm, with Swap-PreNorm approaching the performance of PostNorm. This demonstrates that ZST is more sensitive to the position of LayerNorm in PreNorm than its shallow sub-network nature.

8.3 Summary of This Chapter

In this study, we comprehensively explored the effects of LayerNorm on ZST performance. Our results demonstrate that PostNorm consistently outperforms PreNorm for ZST, regardless of the language tag and residual connection settings used. Through in-depth analysis of off-target rates and structural flaws in the PreNorm model, we were able to identify the underlying factors that contribute to the performance discrepancy. Our study suggests that care should be taken when selecting the LayerNorm setting for ZST in future research.

Chapter 9

Conclusion

9.1 Summary

This thesis has embarked on a comprehensive exploration of multilingual representation learning, addressing the three identified challenges and contributing novel solutions within this domain, with a specific focus on sentence alignment and translation tasks. These tasks are essential in the broader context of multilingual NLP, enabling machines to understand and translate across diverse human languages with increased proficiency and efficiency.

Chapter 2 introduced EMS, a method for MSE learning that efficiently addresses high computational demands. To balance training efficiency against data and computational needs while maintaining MSE quality, we developed an innovative approach to concurrently train "XTR" generative and sentence-level contrastive objectives. The effectiveness of EMS was validated through empirical evaluations on four cross-lingual sentence retrieval tasks and three cross-lingual sentence classification tasks. Future research directions include utilizing LLMs for initial model training to enhance sentence embeddings and refining the model structure via knowledge distillation for faster inference.

Chapter 3 introduced LEALLA, a streamlined model designed to produce compact MSE, addressing the issue of computational intensity during inference. The experimental outcomes indicated that LEALLA, after distilling knowledge from LaBSE, achieved robust performance across 109 languages. Future research

could concentrate on diminishing LaBSE's vocabulary size to further compress the model and investigate the potential of lightweight model pre-training in parallel sentence alignment tasks.

Chapter 4 introduced JASS and ENSS, novel pre-training methods that incorporate syntactic structures of sentences, based on language-agnostic schemes like MASS, to address data scarcity in low-resource languages for NMT. Utilizing abundant monolingual data and syntactic analysis, these methods enhance language-specific structure awareness during pre-training. Experiments on various language pairs showed that JASS and ENSS surpass MASS and similar methods in low-resource contexts, highlighting the value of language-specific inputs and multi-task pre-training. They significantly improved translation adequacy and fluency, as confirmed by LASER metrics, human evaluations, and case studies. Future work will extend linguistically-aware pre-training to more languages while exploring the applicability of these sequence-to-sequence tasks to a broader range of NLP tasks.

In Chapter 5, we introduced a word-level contrastive learning approach for multilingual NMT to tackle the challenge of data scarcity in low-resource languages. Our experiments demonstrated notable improvements in translation quality across various language pairs, further elucidated by analysis linking BLEU scores to the sentence retrieval capabilities of the NMT encoder. Future research directions include: (1) enhancing the retrieval performance of the encoder in many-to-many NMT setups, and (2) assessing the viability of the contrastive objective in massively multilingual contexts.

Chapter 6 presented AlignInstruct, a method aimed at improving the fine-tuning of LLMs for NMT in low-resource, previously unseen languages, with a focus on minimizing the need for extra training corpora to address the data scarcity issue. The results from our multilingual and zero-shot experiments highlighted AlignInstruct's superiority compared to the MTInstruct baseline and other instruction tuning approaches. Future efforts will be directed toward leveraging large monolingual datasets in new languages for MT and enhancing the model's ability to adapt to a variety of MT prompts.

In Chapter 7, we unveiled an innovative variable-length neural interlingua

method, which not only enhanced zero-shot translation outcomes but also yielded a more reliable model compared to earlier fixed-length interlingua techniques, addressing the issue of suboptimal model architecture for zero-shot NMT. Despite observing a decline in performance in translations to English, our analysis pinpointed the specific model component responsible, setting the stage for targeted improvements in future research.

Chapter 8 thoroughly examined the impact of LayerNorm on the performance of zero-shot NMT, aiming to overcome issues related to suboptimal model architecture for zero-shot NMT. The findings revealed that PostNorm has a consistent edge over PreNorm in zero-shot NMT scenarios, independent of language tag and residual connection configurations. By analyzing off-target rates and identifying structural weaknesses in the PreNorm model, we uncovered the reasons behind this performance gap. The insights from our study emphasize the importance of careful LayerNorm configuration choices in future zero-shot NMT research.

In conclusion, the research presented in this thesis marks a significant stride in the field of multilingual NLP. It has not only provided a deeper understanding of the challenges inherent in multilingual representation learning but also offered innovative and practical solutions to overcome these obstacles. As the world becomes increasingly interconnected, the importance of effective multilingual communication grows. The contributions of this thesis thus hold considerable promise for future applications in global communication, information access, and beyond, fostering a world where language barriers continue to diminish.

9.2 Future Prospects

Firstly, the techniques presented in this thesis hold the potential for integration into a singular, comprehensive multilingual model, an endeavor we aim to pursue in future research. Initially, this integration would involve combining the MSE and multilingual NMT models within a unified Transformer encoder-decoder framework. Here, sentence embeddings would be generated by the encoder, while the decoder would produce translations. Following this, the proposed methods for enhancing multilingual representation, particularly those aimed at increasing ef-

ficiency and boosting performance in low-resource languages, could be combined into a single, cohesive training phase. Additionally, the novel model architectures and configurations developed for enhancing Transformer models in zero-shot translation scenarios warrant empirical exploration to assess their compatibility with MSE learning and low-resource translation.

Secondly, the insights obtained from this thesis could significantly contribute to the development of robust multilingual LLMs. Our proposed training objectives, which focus on word alignment and linguistic features, could effectively facilitate better language alignment in multilingual LLMs. Moreover, the efficient MSE models we introduced could enhance the retrieval-based applications of LLMs, such as retrieval-based few-shot in-context learning. Furthermore, our findings regarding the application of Transformer architectures in multilingual contexts offer valuable guidance for future research into the Transformer architectures of multilingual LLMs. This knowledge could be instrumental in further advancing the field and unlocking new possibilities in multilingual LLMs.

Last but not least, looking beyond the conclusion of this thesis, several promising avenues for future research in multilingual NLP emerge. These prospects not only aim to broaden the scope of current methodologies but also seek to deepen the understanding and application of multilingual representation learning.

Expanding Language Coverage

A key direction for future research is the further expansion of language coverage. This includes a focus on low-resource languages and regional dialects, which are often underrepresented in current NLP models. Addressing these gaps can significantly enhance communication inclusiveness and preserve linguistic diversity. Efforts here may involve developing more sophisticated models and algorithms capable of learning from limited data and adapting to linguistic variations.

Integration with Multimodal Data

Secondly, integrating multilingual NLP with multimodal data presents an exciting frontier. Using images and videos as universal pivots can offer innovative ways to bridge language barriers. This approach can leverage the universal nature of

visual and auditory information to complement and enhance language alignment and translation. Future research in this area could explore the development of integrated models that process and interpret multimodal data, which benefits the alignment across languages.

Cross-Cultural Understanding

Finally, it is critical to enhance cross-cultural understanding. This involves detecting and addressing cultural nuances in language use, which is crucial for accurate and sensitive communication across different societies. Future research could develop models that are not only linguistically adept but also culturally aware, capable of interpreting and respecting the subtle cultural contexts embedded in languages.

In summary, the future of multilingual NLP holds immense potential for further exploration and development. By integrating current multilingual techniques on LLMs, expanding language coverage, deepening cross-cultural understanding, and integrating with multimodal data, we can look forward to more inclusive, accurate, and diverse language technologies. These advancements will not only push the boundaries of NLP but also play a vital role in fostering global communication and understanding.

Appendix A

Supplementary Materials of LEALLA

A.1 Discussion about Feature Distillation

We additionally investigate another two patterns for feature distillation. As illustrated in Figure A.1, "*Distillation-first*" modifies the position for computing the MSE loss compared with $\mathcal{L}_{fd}$ of Equation 3.3. The `[CLS]` pooler within the LEALLA encoder is used to generate 768-d embeddings first. A dense layer is employed to transform the 768-d embeddings to low-dimension after calculating the MSE loss. "*Synchronized*" transforms the LaBSE embeddings to low-dimension, then the MSE loss is constructed between two low-dimensional embeddings. As the MSE loss is computed simultaneously with the AMS loss, it is denoted as "*Synchronized*". For "*Synchronized*", it requires a fixed dense layer to conduct the dimension reduction for the LaBSE embeddings, for which we utilize the pretrained model introduced in Section 3.2.2. We denote these two patterns of feature distillation as $\mathcal{L}_{df}$ and $\mathcal{L}_{syn}$.

As reported in Table A.1, $\mathcal{L}_{ams} + \mathcal{L}_{fd}$ ($\mathcal{L}_{fd}$ is feature distillation introduced in the main text) consistently outperforms $\mathcal{L}_{ams} + \mathcal{L}_{df}$ and $\mathcal{L}_{ams} + \mathcal{L}_{syn}$ in all the three LEALLA models. $\mathcal{L}_{ams} + \mathcal{L}_{df}$ and $\mathcal{L}_{ams} + \mathcal{L}_{syn}$ perform comparably on Tatoeba with the models trained without distillation loss. $\mathcal{L}_{ams} + \mathcal{L}_{df}$ obtains performance gains for high-resource languages on UN and BUCC compared with

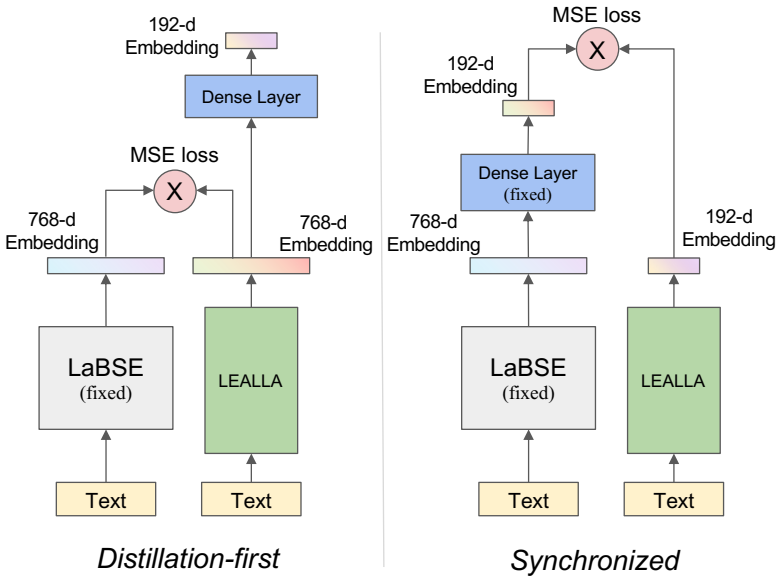

Figure A.1: Another two patterns of feature distillation.

$\mathcal{L}_{ams}$, but still underperforms $\mathcal{L}_{ams} + \mathcal{L}_{fd}$.

$\mathcal{L}_{df}$ forces the lightweight model to approximate the teacher embeddings first in the intermediate part of the model, on top of which the low-dimensional sentence embeddings are generated for computing the AMS loss, while $\mathcal{L}_{fd}$ (Equation 3.3) is calculated after computing the AMS loss. As the AMS loss directly indicates the evaluation tasks, we suppose $\mathcal{L}_{fd}$ is a more flexible objective for feature distillation. In addition, $\mathcal{L}_{syn}$ is not beneficial because it depends on a dimension-reduced LaBSE, which is a less robust teacher compared with LaBSE.

A.2 Results of Dimension-reduction Experiments

We report all the results of Section 3.2.2 in Table A.2.

A.3 Results of All Thin-deep Architectures

Table A.3 presents the detailed results of each architecture we explored in Section 3.2.3. Besides showing the results for each language on UN and BUCC for

Model	Tatoeba	UN					BUCC				
		es	fr	ru	zh	avg.	de	fr	ru	zh	avg.
LEALLA-small											
$\mathcal{L}_{ams}$	80.3	88.1	85.2	88.0	83.9	86.3	93.0	89.7	90.6	88.3	90.4
$\mathcal{L}_{ams} + \mathcal{L}_{fd}$	**80.6**	89.3	**86.8**	88.0	**84.0**	**87.0**	**93.9**	**90.6**	**91.4**	**89.7**	**91.4**
$\mathcal{L}_{ams} + \mathcal{L}_{df}$	80.0	**89.4**	86.3	**88.1**	83.9	86.9	93.8	90.1	91.1	88.9	91.0
$\mathcal{L}_{ams} + \mathcal{L}_{syn}$	80.2	88.5	85.0	87.1	82.8	85.9	93.6	89.9	90.9	88.7	90.8
LEALLA-base											
$\mathcal{L}_{ams}$	81.7	89.8	85.9	88.6	85.4	87.4	94.2	91.0	91.3	**91.1**	91.9
$\mathcal{L}_{ams} + \mathcal{L}_{fd}$	**82.2**	**90.2**	**87.5**	**89.4**	**86.8**	**88.5**	**95.0**	**91.6**	**91.7**	91.0	**92.3**
$\mathcal{L}_{ams} + \mathcal{L}_{df}$	81.8	90.0	87.3	89.2	86.3	88.2	94.7	91.4	**91.7**	90.9	92.2
$\mathcal{L}_{ams} + \mathcal{L}_{syn}$	81.9	89.7	86.7	88.8	85.9	87.8	94.5	91.1	**91.7**	90.3	91.9
LEALLA-large											
$\mathcal{L}_{ams}$	82.9	90.1	87.1	89.3	87.4	88.5	94.6	91.2	91.5	91.4	92.2
$\mathcal{L}_{ams} + \mathcal{L}_{fd}$	**83.4**	**90.6**	**88.4**	**89.8**	**87.7**	**89.1**	**95.3**	**92.0**	**92.0**	**92.0**	**92.8**
$\mathcal{L}_{ams} + \mathcal{L}_{df}$	83.0	90.3	87.6	89.7	87.2	88.7	**95.3**	91.9	**92.0**	91.7	92.7
$\mathcal{L}_{ams} + \mathcal{L}_{syn}$	83.0	90.0	87.4	89.7	86.8	88.5	94.9	91.7	91.8	91.4	92.5

Table A.1: Results of comparisons among three feature distillation objectives. $\mathcal{L}_{df}$ and $\mathcal{L}_{syn}$ indicate "*Distillation-first*" and "*Synchronized*" objectives in Figure A.1.

models #0 - #8, we provide the results of a further smaller thin-deep architecture (#9) and MobileBERT-like [207] thin-deep architectures (#10 - #12). The 64-d thin-deep architecture contains only 33M parameters. However, its performance on three evaluation benchmarks downgrades by up to 7.4 points compared with #5 - #8, which demonstrates that 128-d may be a lower bound as universal sentence embeddings for aligning parallel sentences for 109 languages. Moreover, #10 - #12 show the results of MobileBERT-like architectures whose feed-forward hidden size is identical to hidden size. They have fewer parameters than #5 - #8, but they perform worse than #5 - #8, respectively (e.g., compare #10 with #6). Therefore, we did not employ MobileBERT-like architectures for LEALLA.

A.4 Results of Ablation Study

We report all the results of the ablation study (Section 3.3.2) in Table A.4.

Dimension	Tatoeba	UN					BUCC				
		es	fr	ru	zh	avg.	de	fr	ru	zh	avg.
768 (LaBSE)	83.7	90.8	89.0	90.4	88.3	89.6	95.5	92.3	92.2	92.5	93.1
512	83.7	90.1	88.1	89.7	87.4	88.8	95.4	92.1	92.0	92.4	93.0
384	83.7	90.1	88.1	89.6	87.4	88.8	95.5	92.0	92.0	92.6	93.0
256	83.6	90.3	87.9	89.2	87.4	88.7	95.3	92.0	92.1	92.2	92.9
192	83.4	89.8	87.5	89.5	87.0	88.5	95.2	91.9	91.9	92.2	92.8
128	83.1	89.2	86.9	88.6	85.9	87.7	95.1	91.4	91.8	91.6	92.5
64	81.8	88.4	84.4	87.3	83.8	86.0	93.9	89.8	90.7	88.9	90.8
32	78.4	82.7	74.8	80.4	73.7	77.9	87.1	81.5	84.1	75.5	82.1

Table A.2: Results of the dimension-reduced LaBSE embeddings.

#	L	d_h	d_{ff}	H	P	P_E	Tatoeba	UN					BUCC				
								es	fr	ru	zh	avg.	de	fr	ru	zh	avg.
LaBSE																	
0	12	768	3072	12	471M	85M	83.7	90.8	89.0	90.4	88.3	89.6	95.5	92.3	92.2	92.5	93.1
Fewer Layers																	
1	6	768	3072	12	428M	42M	82.9	90.2	87.4	89.2	87.4	88.6	94.3	90.9	91.2	91.1	91.9
2	3	768	3072	12	407M	21M	82.2	89.4	86.1	88.0	86.5	87.5	93.7	90.1	90.8	90.1	91.2
Smaller Hidden Size																	
3	12	384	1536	12	214M	21M	82.6	90.1	86.9	89.6	87.0	88.4	94.4	91.2	91.4	91.3	92.1
4	12	192	768	12	102M	6M	81.0	89.4	85.6	88.1	85.0	87.0	93.6	90.4	91.1	89.9	91.3
Thin-deep Architecture																	
5	24	384	1536	12	235M	42M	83.2	90.6	87.3	89.2	87.4	88.6	94.7	91.5	91.6	91.9	92.4
6	24	256	1024	8	147M	19M	82.9	90.1	87.1	89.3	87.4	88.5	94.6	91.2	91.5	91.4	92.2
7	24	192	768	12	107M	11M	81.7	89.8	85.9	88.6	85.4	87.4	94.2	91.0	91.3	91.1	91.9
8	24	128	512	8	69M	5M	80.3	88.1	85.2	88.0	83.9	86.3	93.0	89.7	90.6	88.3	90.4
9	24	64	256	8	33M	1M	75.2	83.7	78.6	83.0	72.1	79.4	87.9	83.0	86.0	75.1	83.0
MobileBERT-like Thin-deep Architecture																	
10	24	256	256	4	138M	10M	82.1	89.4	86.5	88.4	86.5	87.7	94.1	91.0	91.0	91.7	92.0
11	24	192	192	4	102M	6M	81.0	89.0	85.4	88.5	85.3	87.1	93.8	90.3	91.0	89.9	91.3
12	24	128	128	4	66M	2M	79.7	88.1	84.1	87.6	83.3	85.8	92.6	88.8	90.4	87.6	89.9

Table A.3: Results of thin-deep and MobileBERT-like architectures. $\mathbf{L}$, $\mathbf{d_h}$, $\mathbf{d_{ff}}$, $\mathbf{H}$, $\mathbf{P}$, and $\mathbf{P_E}$ indicate the number of layers, dimension of hidden states, dimension of feed-forward hidden states, number of attention heads, number of model parameters, and number of encoder parameters (except for the word embedding layer).

Model	Tatoeba	UN					BUCC				
		es	fr	ru	zh	avg.	de	fr	ru	zh	avg.
LEALLA-small											
$\mathcal{L}_{ams}$	80.3	88.1	85.2	88.0	83.9	86.3	93.0	89.7	90.6	88.3	90.4
$\mathcal{L}_{fd}$	78.2	89.0	84.6	87.5	79.6	85.2	**94.2**	90.5	91.2	88.9	91.2
$\mathcal{L}_{ld}$	75.1	1.5	1.1	0.9	5.6	2.3	0.1	0.0	0.1	0.0	0.1
$\mathcal{L}_{ams} + \mathcal{L}_{fd}$	80.6	89.3	**86.8**	88.0	84.0	87.0	93.9	**90.6**	**91.4**	89.7	91.4
$\mathcal{L}_{ams} + \mathcal{L}_{ld}$	80.6	**89.6**	85.8	88.6	84.4	87.1	94.1	90.3	91.2	90.0	91.4
$\mathcal{L}_{ams} + \mathcal{L}_{fd} + \mathcal{L}_{ld}$	**80.7**	89.4	86.0	**88.7**	**84.9**	**87.3**	94.0	**90.6**	91.2	**90.3**	**91.5**
LEALLA-base											
$\mathcal{L}_{ams}$	81.7	89.8	85.9	88.6	85.4	87.4	94.2	91.0	91.3	91.1	91.9
$\mathcal{L}_{fd}$	81.1	90.2	87.3	89.4	85.5	88.1	**95.0**	**91.6**	**91.8**	91.3	**92.4**
$\mathcal{L}_{ld}$	80.6	66.3	49.4	51.0	85.7	63.1	57.5	80.1	60.6	88.6	71.7
$\mathcal{L}_{ams} + \mathcal{L}_{fd}$	82.2	90.2	**87.5**	89.4	86.8	88.5	**95.0**	**91.6**	91.7	91.0	92.3
$\mathcal{L}_{ams} + \mathcal{L}_{ld}$	82.3	90.0	**87.5**	89.2	86.8	88.4	94.8	91.3	91.6	**91.4**	92.3
$\mathcal{L}_{ams} + \mathcal{L}_{fd} + \mathcal{L}_{ld}$	**82.4**	**90.3**	87.4	**89.8**	**87.2**	**88.7**	94.9	91.4	**91.8**	**91.4**	**92.4**
LEALLA-large											
$\mathcal{L}_{ams}$	82.9	90.1	87.1	89.3	87.4	88.5	94.6	91.2	91.5	91.4	92.2
$\mathcal{L}_{fd}$	82.4	89.8	87.2	89.4	86.1	88.1	**95.3**	91.8	92.0	**92.2**	**92.8**
$\mathcal{L}_{ld}$	82.3	87.2	78.8	83.3	86.9	84.1	88.4	87.4	86.9	91.8	88.6
$\mathcal{L}_{ams} + \mathcal{L}_{fd}$	83.4	90.6	88.4	89.8	87.7	89.1	**95.3**	**92.0**	92.0	92.0	**92.8**
$\mathcal{L}_{ams} + \mathcal{L}_{ld}$	83.4	90.6	87.9	**90.0**	87.7	89.1	**95.3**	91.8	91.7	92.4	**92.8**
$\mathcal{L}_{ams} + \mathcal{L}_{fd} + \mathcal{L}_{ld}$	**83.5**	**90.8**	**88.5**	89.9	**87.9**	**89.3**	**95.3**	**92.0**	**92.1**	91.9	**92.8**

Table A.4: Results of LEALLA with different loss functions and loss combinations.

Appendix B

Supplementary Materials of JASS+ENSS

B.1 Algorithms for PMASS

In this section, we introduce Algorithms 1 and 2 for PMASS.S and PMASS.P respectively. We utilize the HPSG parsing result (Figure 4.5 (left)) to detect phrase spans to be masked. For PMASS.S, we can rapidly detect an entire phrase span to be masked. For PMASS.P, we start from the root of the HPSG parsing tree and stochastically mask the left child or the right child; then shift to the unmasked child node to find the next masking candidate. We implement this in a recursive manner.

B.2 Hyperparameters for Optimized Transformer

Following Araabi and Monz [6], we use the hyperparameter settings shown in Table B.1 for training optimized Transformer on different parallel data settings. Although optimized hyperparameter settings can significantly improve low-resource NMT, they require laborious grid search for the optimal setting while fine-tuning NMT based on pre-trained models do not.

Algorithm 1: Algorithm for determining masked phrase spans for PMASS.S.

Input: Length of the sentence L, tree of HPSG parsing result for the sentence T.

Output: Token List M consisting of all the tokens on N. (to be masked)

```
Initialize Current Node N by ROOT of T;
while number of tokens on N > int(L/2) do
    if number of tokens on left child of N > number of tokens on right
     child of N then
        N ← left child of N;
    else
        N ← right child of N;
    end
end
```

B.3 Results of Combining BART with Ours

In Table B.2, B.3 and B.4, we report the results of combining BART and our proposed methods for Japanese–English and Japanese–Chinese translations. We observe that BART (text infilling) can not further improve our proposed methods, which indicates that BART (text infilling) does not have a complement nature with our linguistically-driven multi-task pre-training methods.

Algorithm 2: Algorithm for determining masked phrase spans for PMASS.P.

Input: Length of the sentence L, tree of HPSG parsing result for the sentence T.

Output: `Pmass(`*N=ROOT of T, L, l=0, M=empty list*`)` (tokens to be masked)

```
Function Pmass(N, L, l, M):
    if tag of N is sentence then
        return Pmass(child of N, L, l, M)
    else if tag of N is tok then
        if int(L/2) − l > 0 then
            M.append(token on N)
        return M
    else if N only has one child and N.tag is cons then
        return Pmass(child of N, L, l, M)
    else
        ll ← number of tokens on the left child of N;
        lr ← number of tokens on the right child of N;
        if ll is 1 and lr is 1 then
            if int(L/2) − l > 1 then
                M.append(token on N)
            return M
        else if int(L/2) <= l then
            return M
        else if ll <= int(L/2) − l and lr > int(L/2) − l then
            if random p < 0.5 then
                M ← M+ tokens on the left child of N;
                l ← l + ll;
                return Pmass(right child of N, L, l, M)
            else
                return Pmass(right child of N, L, l, M)
            end
```

```
27
28
29      else if lr <= int(L/2) − l and ll > int(L/2) − l then
30          if random p < 0.5 then
31              M ← M+ tokens on the right child of N;
32              l ← l + lr;
33              return Pmass(left child of N, L, l, M)
34          else
35              return Pmass(left child of N, L, l, M)
36          end
37      else if ll > int(L/2) − l and lr > int(L/2) − l then
38          if random p < 0.5 then
39              return Pmass(left child of N, L, l, M)
40          else
41              return Pmass(right child of N, L, l, M)
42          end
43      else
44          M ← M+ tokens on the left child of N;
45          l ← l + ll;
46          return Pmass(right child of N, L, l, M)
47      end
48  Initialize Current Node N by ROOT of T, Empty Token List M;
49  l ← 0 ;
50  Pmass(N, L, l, M)
```

Hyperparameters	Default	3k	10k	20k	50k	94k
BPE operations	30k	5k	10k	10k	12k	15k
Encoder/decoder layers	6	2	2	2	2	2
Attention heads	16	2	2	2	2	2
Embedding dimension	1024	512	512	512	512	512
Feed forward dimension	4096	512	1024	1024	2048	2048
Dropout	0.3	0.3	0.3	0.3	0.3	0.3
Label smoothing	0.1	0.6	0.5	0.5	0.5	0.4
Batch size	4096	4096	4096	4096	4096	8192

Table B.1: Hyperparameters for optimized Transformer. "Default" denotes the setting of Transformer-big. For English-Japanese, BPE operations for "Vanilla Transformer-big" is 40k.

Model	Ja-En				En-Ja			
	3k	10k	20k	50k	3k	10k	20k	50k
MASS	8.8	13.8	17.2	21.2	9.1	16.0	20.6	25.0
ENSS	**11.2**†	**16.7**†	**19.0**†	22.1†	**11.7**†	**18.7**†	**22.5**†	27.0†
BART (text infilling)	3.1	11.1	15.5	20.7	5.6	14.9	19.8	25.6†
BART + ENSS	10.7†	15.8†	18.9†	**22.4**†	10.6†	17.6†	21.2†	**27.2**†

Table B.2: BLEU scores compared with BART for simulated low/high-resource settings for Japanese–English ASPEC translation using from 3k to 50k parallel sentences for fine-tuning. Results better than MASS with statistical significance $p < 0.05$ are marked in †.

Model	Ja-Zh				Zh-Ja			
	3k	10k	20k	50k	3k	10k	20k	50k
MASS	15.7	20.3	22.4	24.7	19.4	25.9	29.4	32.9
JASS	**17.1**†	**22.2**†	**23.2**†	**25.2**†	21.6†	**27.5**†	**30.4**†	**33.6**†
BART (text infilling)	13.5	19.0	21.3	24.4	20.3†	25.8	29.1	33.0
BART + JASS	**17.1**†	21.3†	23.1†	25.0	**21.9**†	**27.5**†	**30.4**†	**33.6**†

Table B.3: BLEU scores compared with BART for simulated low-resource settings for Japanese–Chinese ASPEC translation using 3k to 50k parallel sentences for fine-tuning. Results better than MASS with statistical significance $p < 0.05$ are marked in †.

Model	Ja-Zh				Zh-Ja			
	3k	10k	20k	50k	3k	10k	20k	50k
MASS	7.7	15.4	18.3	23.4	9.6	17.6	23.3	27.1
JASS	**12.0**†	**17.0**†	**20.1**†	**25.0**†	**16.6**†	21.2†	**26.5**†	29.2†
BART (text infilling)	5.9	14.0	18.0	21.8	8.7	17.8	24.2†	28.5†
BART + JASS	11.4†	16.5†	19.4†	24.3†	16.2†	**22.5**†	26.2†	**30.0**†

Table B.4: BLEU scores compared with BART for simulated low-resource settings for Japanese–Chinese Wikipedia translation using from 3k to 50k parallel sentences for fine-tuning. Results better than MASS with statistical significance $p < 0.05$ are marked in †.

Appendix C

Supplementary Materials of WCL

C.1 BLEU Scores

We report all the BLEU results of 222_en-ja, 626_en-it-ja-nl-tr-vi, and 626_en-tr-ro-et-my-kk in Table C.1, C.2 and C.3, respectively.

C.2 Sentence Retrieval Precision

We report the sentence retrieval precision for all the systems in Tables C.4, C.5 and C.6. The sentence retrieval previsions are evaluated by using the validation dataset of each language pair. The mean pooled encoder output is used as the sentence embedding. We use cosine similarity to conduct the retrieval task, and report the average retrieval precision of both directions of each language pair.

C.3 Word Retrieval Precision

We report the word retrieval precision for all the systems in Tables C.7, C.8, and C.9. The word retrieval precision are computed by using the validation dataset and the word2word alignments on it. The mean pooled encoder output on corresponding positions is used as the contextualized word embedding.

Methods	en-ja	ja-en
MLSC	15.9	11.9
+align	16.3	11.5
+w2w (ours)	16.0	11.7
+FA (ours)	15.6	11.0
mBART FT	19.8	18.0
+align	19.6	17.5
+w2w (ours)	19.4	18.2
+FA (ours)	19.5	17.8

Table C.1: **BLEU scores of 222_en-ja system.** Significantly better scores are in cyan, and marginal improvements are in lightcyan. The significance test is done with Koehn [88].

Methods	en-ja		en-vi		en-it		en-nl		en-tr		Avg.
	→	←	→	←	→	←	→	←	→	←	
MLSC	15.4	11.8	29.6	28.6	27.5	32.7	29.1	36.4	11.6	14.9	23.76
+align	15.1	11.4	29.4	28.3	27.7	33.0	28.9	36.0	11.8	15.1	23.67
+w2w (ours)	15.3	11.6	29.7	28.2	27.6	32.4	28.6	35.8	10.8	14.4	23.44
+FA (ours)	15.5	11.6	29.6	28.0	27.8	33.2	29.1	35.9	11.2	14.9	23.68
+sent	15.1	11.6	29.6	28.3	27.3	32.7	28.1	36.6	11.3	14.7	23.53
mBART FT	17.8	17.0	34.1	35.7	32.5	38.0	32.6	41.6	18.7	23.1	29.11
+align	17.6	16.7	33.7	35.6	32.0	37.7	32.5	41.3	18.7	22.9	28.87
+w2w (ours)	17.6	17.2	34.2	35.7	32.5	38.2	32.1	41.7	18.7	22.9	29.08
+FA (ours)	17.5	17.7	34.0	35.2	32.4	37.9	32.3	41.4	18.6	23.1	29.01
+sent	17.8	16.5	33.7	35.6	32.2	38.1	32.5	41.2	18.1	22.9	28.86

Table C.2: **BLEU scores of 626_en-it-ja-nl-tr-vi system.** Significantly better scores are in cyan, and marginal improvements are in lightcyan. The significance test is done with Koehn [88].

We use cosine similarity to implement the retrieval for word pairs in a batch, and present the average in-batch retrieval precision of both directions of each language

Methods	en-tr		en-ro		en-et		en-kk		en-my		Avg.
	→	←	→	←	→	←	→	←	→	←	
MLSC	9.3	12.6	25.0	26.2	10.8	15.1	0.5	5.3	15.1	15.6	13.55
+align	9.0	12.4	24.6	26.5	10.7	14.6	0.4	5.4	15.0	15.3	13.39
+w2w (ours)	9.4	12.6	24.8	26.8	10.8	15.1	0.5	5.8	15.2	15.9	13.69
+FA (ours)	9.1	12.2	24.8	26.7	10.7	14.8	0.3	5.6	15.0	15.6	13.48
+sent	8.7	12.1	24.5	26.0	10.4	14.5	0.4	5.3	13.8	14.6	13.03
mBART FT	17.7	22.2	33.8	37.1	14.5	24.3	1.8	14.1	17.8	23.1	20.64
+align	17.5	21.9	33.8	36.7	15.2	24.3	1.8	14.0	16.9	22.1	20.42
+w2w (ours)	17.6	22.2	34.2	37.5	15.0	25.0	1.2	14.1	18.3	23.8	20.89
+FA (ours)	17.5	22.2	34.3	37.5	14.9	25.1	1.3	14.4	17.9	23.6	20.87
+sent	17.2	22.1	34.2	37.0	14.2	24.1	1.6	14.0	17.7	23.4	20.55

Table C.3: **BLEU scores of 626_en-tr-ro-et-my-kk system.** Significantly better scores are in cyan, and marginal improvements are in lightcyan. The significance test is done with Koehn [88].

pair. Batch size is set as 512 tokens.

Methods	en-ja
MLSC	3.3
+align	**3.5**
+w2w (ours)	**73.5**
+FA (ours)	**69.6**
mBART FT	88.9
+align	87.4
+w2w (ours)	85.2
+FA (ours)	84.8

Table C.4: **Sentence retrieval P@1 on the validation set for 222_en-ja.**

Methods	en-ja	en-vi	en-it	en-nl	en-tr	Avg.
MLSC	52.7	84.6	91.0	85.7	89.7	80.9
+align	**53.5**	82.8	**91.2**	**86.4**	88.9	80.6
+w2w (ours)	**73.4**	**85.7**	**91.4**	84.7	83.1	**83.7**
+FA (ours)	**71.3**	**84.9**	**91.3**	83.8	82.0	**82.7**
+sent	87.2	84.7	91.1	87.7	86.6	87.5
mBART FT	87.1	96.2	97.3	94.6	98.5	94.7
+align	85.1	95.8	97.3	94.2	98.5	94.2
+w2w (ours)	81.6	91.4	94.7	90.8	89.6	89.6
+FA (ours)	82.6	92.3	95.0	91.7	90.4	90.4
+sent	76.2	88.3	93.6	88.7	89.8	87.3

Table C.5: **Sentence retrieval P@1 on the validation set for 626_en-it-ja-nl-tr-vi.**

Methods	en-tr	en-ro	en-et	en-kk	en-my	Avg.
MLSC	86.2	84.0	85.4	64.4	72.4	78.5
+align	85.9	82.4	84.0	61.3	61.8	75.1
+w2w (ours)	79.6	**88.1**	76.8	**77.4**	**83.7**	**81.1**
+FA (ours)	77.0	**86.1**	69.8	**75.7**	**73.4**	76.4
+sent	76.3	77.6	55.2	63.8	71.4	68.9
mBART FT	98.0	92.7	96.0	92.9	94.7	94.9
+align	97.4	92.5	**97.0**	92.1	93.7	94.5
+w2w (ours)	94.3	**95.6**	**96.8**	86.0	**96.2**	93.8
+FA (ours)	94.3	**96.3**	**97.3**	87.9	**96.2**	94.4
+sent	94.6	**97.3**	95.4	**93.1**	**95.7**	**95.2**

Table C.6: **Sentence retrieval P@1 on the validation set for 626_en-tr-ro-et-my-kk.**

Methods	en-ja
MLSC	20.1
+align	**22.5**
+w2w (ours)	**68.3**
+FA (ours)	**67.6**
mBART FT	65.2
+align	64.3
+w2w (ours)	**71.5**
+FA (ours)	**70.7**

Table C.7: **Word retrieval P@1 on the validation set for 222_en-ja.**

Methods	en-ja	en-vi	en-it	en-nl	en-tr	Avg.
MLSC	61.8	54.6	42.8	42.1	42.7	48.8
+align	**61.9**	54.1	**43.7**	42.0	42.3	48.8
+w2w (ours)	**64.0**	**64.7**	**55.8**	**57.7**	**52.8**	**59.0**
+FA (ours)	58.2	**65.2**	**59.2**	**60.1**	**48.1**	**58.2**
mBART FT	64.5	57.2	47.4	45.9	47.2	52.4
+align	64.0	56.8	47.3	45.7	46.8	52.1
+w2w (ours)	**71.3**	**70.1**	**60.6**	**62.9**	**57.8**	**64.5**
+FA (ours)	**68.6**	**69.4**	**63.2**	**64.7**	**57.4**	**64.7**

Table C.8: **Word retrieval P@1 on the validation set for 626_en-it-ja-nl-tr-vi.**

Methods	en-tr	en-ro	en-et	en-kk	en-my	Avg.
MLSC	41.9	63.2	64.4	63.4	65.8	59.7
+align	40.9	63.2	63.9	63.4	**66.2**	59.5
+w2w (ours)	**50.1**	**66.5**	**67.6**	**68.8**	**71.3**	**64.9**
+FA (ours)	**47.2**	**66.7**	**65.7**	**65.4**	**66.3**	**62.3**
mBART FT	46.8	66.1	68.0	68.7	71.7	64.3
+align	46.4	65.9	67.8	68.5	71.1	63.9
+w2w (ours)	**55.6**	**70.3**	**72.8**	**74.7**	**74.4**	**69.6**
+FA (ours)	**55.3**	**70.1**	**73.0**	**74.0**	**74.0**	**69.3**

Table C.9: **Word retrieval P@1 on the validation set for 626_en-tr-ro-et-my-kk.**

Appendix D

Supplementary Materials of AlignInstruct

D.1 Results of MT+Align+Hint+Revise for models of BLOOMZ+3

We present the results in Table D.1. Co-referencing the results in Table 6.6, compared with MT+Align, a clear advantage for the MT+Align+Hint+Revise setting in supervised directions involving English (en→seen and seen→en) in the ar-fr-de-nl-ru-zh setting was observed. This result suggested that AlignInstruct's variants played a crucial role in preserving the BLOOMZ's capabilities for supported languages. However, in all other scenarios, AlignInstruct alone proved sufficient to enhance the performance beyond the MTInstruct baseline, but hard to achieve further improvements with additional instructions.

D.2 Representation Change of BLOOMZ+3

The representation change observed in de-nl-ru was consistent with the findings presented in Section 6.4.5, which highlighted an initial increase in cross-lingual alignment in the early layers, followed by a decrease in the final layers. When mixing fine-tuning data with supported languages, the changes exhibited more intricate patterns. As illustrated by ar-fr-zh in ar-de-fr-nl-ru-zh in Figure D.1,

Languages	Zero-shot Directions				Supervised Directions			
	Directions	BLEU	chrF++	COMET	Directions	BLEU	chrF++	COMET
de-nl-ru	overall	**8.94**	**23.53**	**60.67**	en→xx	16.70	31.83	68.98
					xx→en	**25.18**	45.00	**76.45**
	seen→seen	14.00	**27.58**	**70.59**	en→seen	15.97	28.53	**72.69**
	seen→unseen	**6.49**	**23.01**	**54.92**	en→unseen	**17.43**	**35.13**	**65.27**
	unseen→seen	**9.50**	**21.90**	**64.69**	seen→en	25.33	46.70	77.51
	unseen→unseen	6.73	22.70	53.34	unseen→en	**25.03**	**43.30**	**75.39**
ar-de-fr-nl-ru-zh	overall	**12.07**	**26.67**	63.13	en→xx	**21.62**	**36.12**	**70.94**
					xx→en	**28.92**	**48.60**	**77.50**
	seen→seen	**23.52**	**36.13**	**76.62**	en→seen	**26.87**	**38.40**	**78.40**
	seen→unseen	**7.16**	24.48	55.02	en→unseen	**16.37**	**33.83**	63.49
	unseen→seen	**12.91**	**25.23**	**68.91**	seen→en	**32.57**	**53.70**	**80.06**
	unseen→unseen	6.73	**22.65**	53.12	unseen→en	**25.27**	**43.50**	**74.93**

Table D.1: **Results of BLOOMZ+3 with MT+Align+Hint+Revise.** Co-referencing Table 6.6, scores that surpass the MTInstruct baseline are marked in **bold**.

sentence alignment declined after MTInstruct fine-tuning but elevated after further combining with AlignInstruct. We leave the interpretation of this nuanced behavior in future work.

D.3 Inference using Different MT Prompts

We investigated the performance of fine-tuned models when using various MT prompts during inference, aiming to understand models' generalization capabilities with different test prompts. We examined five MT prompts for the fine-tuned models of BLOOMZ-7b1, following Zhang et al. [259], which are presented in Table D.2. The results, showcased in Table D.3, revealed that in comparison to the default prompt used during fine-tuning, the translation performance tended to decline when using other MT prompts. We observed that MT+Align consistently surpasses MTInstruct for xx→en translations, though the results were mixed for en→xx directions. Certain prompts, such as PROMPT-3 and PROMPT-4, exhibited a minor performance drop, while others significantly impacted translation quality. These findings underscored the need for enhancing the models' ability

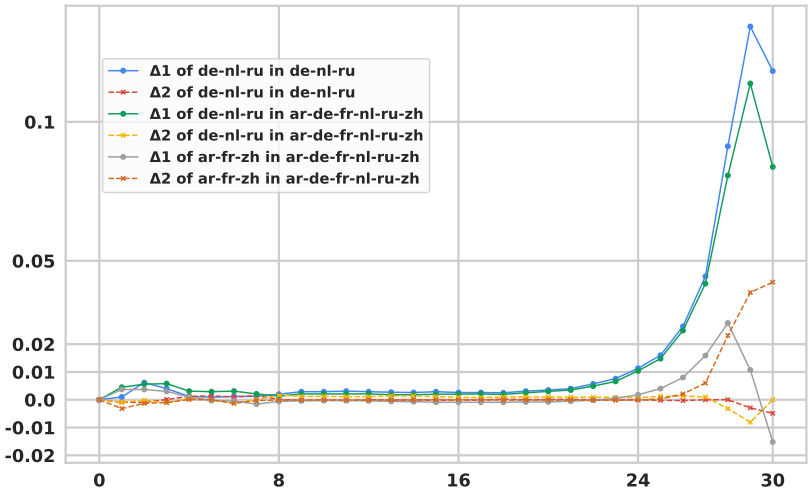

Figure D.1: **Differences in cosine similarity of layer-wise embeddings for BLOOMZ+3.** $\Delta 1$ represents the changes from the unmodified BLOOMZ to the one on MTInstruct, and $\Delta 2$ from MTInstruct to MT+Align.

to generalize across diverse MT prompts, potentially by incorporating a range of MT prompt templates during the fine-tuning process, as stated in the Limitations section.

D.4 Zero-shot Translation using English as Pivot

Pivot translation serves as a robust technique for zero-shot translation, especially given that we used English-centric data during fine-tuning. In Table D.4, we present results that utilize English as an intermediary pivot for translations between non-English language pairs. Our findings indicated that employing the English pivot typically yielded an enhancement of approximately 1.1 - 1.2 BLEU points compared to direct translations in zero-shot directions when fine-tuning BLOOMZ. When contrasting the MTInstruct baseline with our proposed method, MT+Align, we observed that combining AlignInstruct consistently boosted performance in pivot translation scenarios.

Prompt	Definition
PROMPT-default	Translate from Y to X. Y: $y_1 y_2 \dots y_M$. X:
PROMPT-1	Y: $y_1 y_2 \dots y_M$. X:
PROMPT-2	$y_1 y_2 \dots y_M$. X:
PROMPT-3	Translate to X. Y: $y_1 y_2 \dots y_M$. X:
PROMPT-4	Translate from Y to X. $y_1 y_2 \dots y_M$. X:
PROMPT-5	Translate to X. $y_1 y_2 \dots y_M$. X:

Table D.2: **MT prompt variants investigated for fine-tuned models.** These MT prompts are following the design in Zhang et al. [259].

D.5 Result Details of BLOOMZ+24 and BLOOMZ+3

We present per language detailed results of original BLOOMZ-7b1 and fine-tuned BLOOMZ-7b1 models in Tables D.5, D.6, D.7, D.8, D.9, D.10, D.11, D.12, respectively for the BLOOMZ+24 and BLOOMZ+3 settings.

Prompt	Objective	en→xx			xx→en		
		BLEU	chrF++	COMET	BLEU	chrF++	COMET
PROMPT-default	MTInstruct	11.54	25.33	64.68	18.59	33.25	68.75
	MT+Align	**12.28**	**26.17**	**65.28**	**18.72**	**34.02**	**69.75**
PROMPT-1	MTInstruct	5.29	11.31	50.74	7.87	20.08	57.10
	MT+Align	**5.30**	**11.38**	**51.29**	**8.93**	**20.77**	**58.01**
PROMPT-2	MTInstruct	2.20	6.68	45.78	7.15	19.08	57.03
	MT+Align	1.91	5.35	43.92	**7.61**	18.80	56.40
PROMPT-3	MTInstruct	10.59	22.69	62.77	15.85	29.93	66.64
	MT+Align	9.20	20.80	61.45	**16.17**	**30.58**	**67.75**
PROMPT-4	MTInstruct	8.67	20.73	61.32	15.20	28.95	65.51
	MT+Align	**8.91**	20.53	**61.55**	**16.25**	**30.67**	**67.06**
PROMPT-5	MTInstruct	6.61	14.55	55.93	10.88	22.41	60.48
	MT+Align	6.02	12.28	52.72	**11.83**	**23.85**	**61.28**

Table D.3: **Results of using different MT prompts for BLOOMZ-7b1 fine-tuned models during inference.** Refer to Table D.2 for details about definitions of different MT prompts. We report the average results for the BLOOMZ+24 setting. Results better than the MTInstruct baseline are marked in **bold**.

MTInstruct	BLEU	chrF++	COMET	**MT+Align**	BLEU	chrF++	COMET
overall	11.79	26.36	63.22	overall	**12.13**	**26.65**	**63.23**
seen→seen	22.68	35.32	76.39	seen→seen	**23.67**	**36.53**	**76.89**
seen→unseen	7.10	24.50	55.18	seen→unseen	**7.27**	24.32	54.96
unseen→seen	12.56	24.74	68.83	unseen→seen	**12.92**	**25.29**	**69.10**
unseen→unseen	6.78	22.62	53.69	unseen→unseen	6.68	22.30	53.19
MTInstruct with English pivot	BLEU	chrF++	COMET	**MT+Align with English pivot**	BLEU	chrF++	COMET
overall	12.99	28.01	65.38	overall	**13.25**	**28.30**	**65.57**
seen→seen	23.10	35.30	76.30	seen→seen	**23.48**	**35.57**	**76.43**
seen→unseen	9.00	27.67	59.54	seen→unseen	**9.28**	**28.03**	**59.73**
unseen→seen	13.18	24.98	68.77	unseen→seen	**13.36**	**25.22**	**68.94**
unseen→unseen	8.57	25.77	58.17	unseen→unseen	**8.83**	**26.07**	**58.42**

Table D.4: **Results of BLOOMZ+3 using English as a pivot language for zero-shot translation evaluation.** Results of MT+Align surpassing corresponding those of MTInstruct are marked in **bold**.

Language	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
af	3.8	13.2	56.38	7.6	22.0	59.14	2.6	14.9	33.60	20.1	38.0	65.61
am	0.1	0.3	33.17	0.5	8.3	43.57	0.3	0.6	30.65	1.9	12.6	46.24
be	4.2	5.1	47.26	7.3	17.5	48.57	0.4	3.3	31.58	4.2	22.3	49.27
cy	2.7	10.5	53.21	6.2	16.0	53.25	1.2	11.2	34.17	6.0	20.3	53.45
ga	1.2	10.6	42.85	4.0	16.4	46.05	1.2	11.6	33.94	5.5	19.6	46.97
gd	9.3	16.0	51.40	47.6	55.9	59.30	1.2	11.2	36.28	4.2	18.8	43.73
gl	4.5	25.6	64.93	17.2	36.7	66.07	13.4	38.5	74.77	51.0	67.8	85.77
ha	0.1	5.4	38.42	0.3	11.2	42.58	1.5	10.2	35.77	6.9	18.9	47.37
ka	0.3	1.9	31.97	0.6	9.2	44.48	0.4	1.4	28.81	2.4	17.0	47.57
kk	4.3	4.9	50.51	5.1	14.2	51.51	0.5	1.6	33.66	5.1	19.8	51.40
km	2.8	4.5	51.68	3.9	11.1	50.40	0.8	2.9	39.56	5.6	16.2	50.42
ky	10.0	10.6	54.23	10.3	24.0	55.99	0.6	1.6	30.19	3.8	17.9	48.05
li	6.6	16.2	61.39	5.9	24.8	61.65	2.0	14.9	41.01	9.8	29.8	46.92
my	1.8	2.4	45.44	3.0	5.0	48.33	0.4	0.8	29.58	1.0	3.7	44.15
nb	5.8	18.2	57.01	13.9	33.0	56.37	3.9	19.3	46.74	19.8	40.3	63.56
nn	6.3	18.6	62.33	8.9	25.3	56.28	3.7	19.7	41.75	16.9	37.5	62.37
oc	6.0	13.6	60.16	5.1	18.6	58.51	9.6	33.6	67.22	53.0	68.5	79.57
si	0.6	2.0	41.84	1.6	9.4	48.58	0.5	1.4	28.08	1.6	9.1	42.67
tg	0.4	1.4	36.26	1.1	11.8	43.54	0.4	1.5	26.63	3.3	18.0	43.79
tk	7.9	10.6	55.34	5.3	13.0	47.33	0.7	8.7	31.94	4.2	20.1	45.05
tt	0.0	1.0	28.98	0.2	13.3	42.85	0.3	1.4	27.86	4.2	20.2	48.15
ug	0.0	0.4	32.44	0.3	11.2	45.69	0.3	0.9	31.34	3.0	16.5	48.99
uz	0.7	2.1	35.94	1.0	12.8	41.86	1.5	11.5	40.65	3.1	18.7	49.43
yi	7.3	16.5	57.47	4.0	23.0	63.91	0.7	1.7	33.22	2.1	15.6	41.87
avg.	3.61	8.82	47.94	6.70	18.49	51.49	2.00	9.35	37.04	9.95	24.47	52.18

Table D.5: Detailed results of BLOOMZ-7b1 without fine-tuning.

Language	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
af	25.0	41.4	71.05	38.5	52.3	78.94	10.1	31.0	45.42	33.9	51.1	72.66
am	3.0	12.8	59.55	3.4	19.8	59.71	0.2	5.2	42.97	1.4	16.0	49.47
be	8.9	14.9	55.16	14.0	24.9	62.37	0.7	12.3	30.90	3.7	21.0	49.99
cy	20.2	38.0	71.55	33.2	49.3	77.72	5.0	20.3	38.38	13.1	30.2	57.47
ga	15.6	37.1	63.87	29.2	49.1	75.94	3.7	21.2	39.17	12.5	30.3	57.53
gd	13.1	24.7	62.14	66.0	69.6	77.70	2.2	19.6	40.75	7.1	22.3	50.05
gl	16.9	37.6	70.62	24.7	43.6	75.62	21.9	45.2	77.26	46.6	64.5	86.86
ha	12.3	32.7	71.75	10.0	29.8	64.51	1.9	17.1	49.24	6.8	22.1	48.81
ka	4.6	18.1	67.39	10.0	24.3	60.50	0.3	6.8	27.46	1.5	14.9	46.10
kk	12.6	19.5	66.07	14.6	28.2	71.80	0.8	13.0	35.76	3.9	19.7	52.24
km	19.7	25.2	63.24	13.9	32.1	75.02	0.5	12.3	35.60	6.2	22.4	56.45
ky	16.0	20.5	66.27	21.1	33.8	73.06	0.9	12.7	36.10	3.0	17.5	50.40
li	13.5	32.8	70.97	21.3	35.7	67.20	3.3	19.9	42.21	14.6	31.4	55.94
my	6.2	14.3	58.04	5.2	15.6	63.65	0.2	12.9	40.37	1.3	12.7	48.38
nb	12.7	30.4	63.27	22.2	42.1	76.74	7.9	28.4	44.15	25.6	44.3	72.56
nn	18.3	38.0	77.18	27.1	47.7	81.80	7.3	25.7	45.35	24.3	42.9	70.06
oc	10.0	20.0	63.31	13.4	27.1	69.89	8.0	27.5	51.48	46.9	63.5	79.64
si	5.2	21.4	68.16	11.5	26.4	70.79	0.9	12.9	41.73	3.7	19.2	57.41
tg	5.5	22.0	66.08	8.0	25.9	60.54	1.1	15.8	65.14	3.1	19.6	45.06
tk	24.4	26.7	65.53	30.4	37.8	70.39	0.7	10.8	42.36	3.9	18.8	46.23
tt	1.9	17.6	60.01	3.6	19.6	54.99	0.4	13.7	50.78	1.6	14.3	42.58
ug	1.2	19.7	49.76	4.2	21.2	61.34	0.4	12.9	35.88	1.7	16.7	50.29
uz	3.1	18.2	62.12	5.7	22.0	61.12	0.5	3.6	34.67	3.9	18.8	50.32
yi	7.1	24.3	59.13	14.9	20.2	58.66	0.3	9.5	29.77	2.5	17.2	43.27
avg.	11.54	25.33	64.68	18.6	33.25	68.75	3.30	17.10	42.62	11.37	27.14	55.82

Table D.6: Detailed results of BLOOMZ-7b1 fine-tuned with MTInstruct for BLOOMZ+24.

Language	OPUS en→xx			OPUS xx→en			Flores en→xx			Flores xx→en		
	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET	BLEU	chrF++	COMET
af	25.0	41.9	70.72	36.9	52.2	78.68	10.6	31.9	45.84	33.5	51.1	72.84
am	3.4	13.2	60.62	4.9	22.8	62.43	0.3	5.4	44.20	1.4	16.4	51.05
be	8.3	14.5	55.23	13.9	25.1	62.72	0.8	12.5	30.93	3.6	20.6	49.14
cy	20.6	39.0	71.73	33.8	49.4	77.55	4.7	20.3	38.70	14.6	31.5	58.34
ga	17.6	39.3	65.76	32.6	52.7	77.49	3.4	21.4	39.99	13.6	31.6	58.73
gd	15.6	27.2	62.09	48.1	55.4	75.90	2.3	20.3	40.81	7.4	22.0	49.99
gl	17.1	37.2	70.85	24.4	43.3	75.90	21.7	44.9	77.09	45.6	63.5	86.60
ha	14.6	35.0	73.34	11.4	31.3	65.69	1.9	17.3	50.88	7.4	22.5	49.57
ka	4.9	18.9	67.54	10.5	25.3	61.27	0.3	6.9	27.61	2.1	16.0	47.04
kk	12.3	19.3	65.73	15.6	28.0	71.01	0.9	13.0	35.86	4.1	19.8	52.43
km	20.4	26.5	63.38	14.4	35.2	75.62	0.6	12.5	35.44	7.1	22.9	57.81
ky	15.8	19.6	64.74	23.3	35.8	74.70	0.9	13.3	36.71	2.9	17.4	50.06
li	13.2	29.4	65.18	22.3	38.2	71.93	3.1	19.7	42.58	12.5	28.7	54.60
my	7.6	15.4	58.84	6.3	18.0	66.45	0.3	13.3	40.97	1.2	14.4	50.79
nb	13.5	31.4	64.08	24.2	44.2	77.58	7.9	28.7	44.12	25.5	44.9	72.72
nn	19.0	38.0	77.61	28.5	47.7	81.68	7.0	26.7	46.14	25.8	44.1	70.55
oc	9.1	19.3	63.25	13.5	27.5	70.13	7.5	25.9	50.48	47.3	63.8	79.39
si	5.1	22.1	69.60	13.9	29.1	72.51	1.1	13.1	43.01	5.6	22.7	61.89
tg	6.6	23.7	66.31	8.8	27.2	61.52	0.9	15.6	65.51	3.4	19.9	45.45
tk	27.2	26.2	66.11	31.2	38.7	70.47	0.7	11.4	43.64	3.8	18.2	45.87
tt	2.1	18.6	60.75	5.0	21.5	56.95	0.4	13.3	50.64	1.5	13.7	42.76
ug	1.1	20.7	51.12	5.5	23.4	63.42	0.4	13.8	37.51	2.1	16.3	50.45
uz	3.5	18.6	62.09	7.4	23.3	62.01	0.2	1.9	34.50	3.7	18.2	50.09
yi	11.1	33.1	70.13	12.8	21.2	60.47	0.4	9.8	30.08	2.6	17.0	42.57
avg.	12.28	26.17	65.28	18.72	34.02	69.75	3.26	17.20	43.05	11.60	27.38	56.28

Table D.7: Detailed results of BLOOMZ-7b1 fine-tuned with MT+Align for BLOOMZ+24.

Zero-shot	BLEU	chrF++	COMET	Supervised	BLEU	chrF++	COMET
ar-de	1.4	14.8	56.19	en-ar	11.1	32.4	75.66
ar-fr	21.9	46.1	74.19	en-de	12.2	29.2	59.16
ar-nl	0.6	11.2	56.59	en-fr	26.8	49.2	77.42
ar-ru	3.1	6.2	48.41	en-nl	2.0	16.0	46.52
ar-zh	18.4	14.4	73.65	en-ru	5.7	16.1	49.00
de-ar	2.0	17.8	64.91	en-zh	22.5	17.0	77.90
de-fr	12.0	33.4	63.45	avg.	13.38	26.65	64.28
de-nl	3.7	17.9	47.30				
de-ru	1.3	11.8	45.53				
de-zh	8.9	7.6	61.52				
fr-ar	11.2	33.4	74.20		BLEU	chrF++	COMET
fr-de	4.6	23.4	48.83	ar-en	26.7	48.4	78.12
fr-nl	2.8	17.2	52.14	de-en	21.1	38.5	71.99
fr-ru	3.1	10.4	45.12	fr-en	27.7	49.8	79.46
fr-zh	20.9	17.0	76.20	nl-en	12.3	31.1	61.29
nl-ar	1.3	13.2	59.46	ru-en	17.9	36.6	68.40
nl-de	5.9	22.8	46.49	zh-en	24.5	47.9	77.08
nl-fr	9.6	29.6	58.30	avg.	21.70	42.05	72.72
nl-ru	0.8	9.0	42.83				
nl-zh	3.3	3.7	53.96				
ru-ar	6.5	25.3	68.38				
ru-de	2.0	17.0	48.06				
ru-fr	15.7	38.7	67.54				
ru-nl	0.5	10.5	46.14				
ru-zh	10.7	11.3	67.18				
zh-ar	8.6	29.7	73.47				
zh-de	1.6	17.6	49.90				
zh-fr	20.7	44.1	75.79				
zh-nl	0.6	10.4	48.53				
zh-ru	2.9	8.6	44.13				
avg.	6.89	19.14	57.95				
seen→seen	16.95	30.78	74.58	en→seen	20.13	32.87	76.99
seen→unseen	2.30	13.31	49.98	en→unseen	6.63	20.43	51.56
unseen→seen	7.78	20.07	62.74	seen→en	26.30	48.70	78.22
unseen→unseen	2.37	14.83	46.06	unseen→en	17.10	35.40	67.23

Table D.8: Detailed results of BLOOMZ-7b1 without fine-tuning.

Zero-shot	BLEU	chrF++	COMET	Supervised	BLEU	chrF++	COMET
ar-de	4.7	20.9	56.43	en-ar	9.1	27.2	71.47
ar-fr	20.8	42.5	71.47	en-de	19.8	36.1	66.53
ar-nl	7.2	22.9	58.29	en-fr	23.0	44.5	74.98
ar-ru	5.0	21.0	54.73	en-nl	15.5	36.1	64.76
ar-zh	14.0	12.4	67.94	en-ru	14.2	30.3	62.82
de-ar	2.4	16.2	64.53	en-zh	20.7	17.9	74.97
de-fr	11.9	31.2	64.44	avg.	17.05	32.02	69.26
de-nl	9.4	28.1	54.22				
de-ru	5.1	19.6	55.41				
de-zh	4.2	5.8	55.26				
fr-ar	10.1	29.1	70.72		BLEU	chrF++	COMET
fr-de	8.6	27.7	53.77	ar-en	26.5	46.9	76.92
fr-nl	10.3	30.1	57.55	de-en	27.0	44.0	76.97
fr-ru	7.9	26.0	56.82	fr-en	27.5	49.0	78.80
fr-zh	18.1	18.5	72.24	nl-en	21.8	41.3	73.99
nl-ar	2.0	15.1	63.73	ru-en	24.8	43.6	74.23
nl-de	9.7	28.1	52.58	zh-en	23.2	45.3	76.83
nl-fr	13.2	32.3	65.17	avg.	25.13	45.02	76.29
nl-ru	5.1	18.6	55.13				
nl-zh	3.0	5.4	54.34				
ru-ar	5.9	15.0	60.36				
ru-de	5.6	23.8	52.66				
ru-fr	17.9	38.4	68.66				
ru-nl	6.2	22.5	54.41				
ru-zh	7.5	13.6	61.40				
zh-ar	6.7	22.1	67.48				
zh-de	3.3	19.6	51.75				
zh-fr	17.4	38.9	73.00				
zh-nl	4.8	19.3	54.41				
zh-ru	3.5	17.9	49.02				
avg.	8.38	22.75	59.93				
seen→seen	14.52	27.25	70.48	en→seen	17.60	29.87	73.81
seen→unseen	6.14	22.82	54.75	en→unseen	16.50	34.17	64.70
unseen→seen	7.56	19.22	61.99	seen→en	25.73	47.07	77.52
unseen→unseen	6.85	23.45	54.07	unseen→en	24.53	42.97	75.06

Table D.9: Detailed results of BLOOMZ-7b1 fine-tuned with MTInstruct for BLOOMZ+3 de-nl-ru.

Zero-shot	BLEU	chrF++	COMET	Supervised	BLEU	chrF++	COMET
ar-de	5.1	20.8	55.25	en-ar	8.4	26.0	70.45
ar-fr	20.3	42.5	71.78	en-de	21.1	36.7	67.15
ar-nl	6.4	21.6	57.48	en-fr	22.9	44.4	74.67
ar-ru	5.2	21.5	55.51	en-nl	16.1	36.8	65.26
ar-zh	16.0	14.1	69.55	en-ru	15.2	31.5	63.30
de-ar	2.4	16.3	64.01	en-zh	16.1	15.0	71.93
de-fr	13.5	34.3	66.25	avg.	16.63	31.73	68.79
de-nl	9.7	28.0	55.00				
de-ru	5.3	19.6	55.61				
de-zh	7.2	7.3	60.64				
fr-ar	10.0	28.2	69.86		BLEU	chrF++	COMET
fr-de	9.2	27.8	54.03	ar-en	27.1	47.0	76.54
fr-nl	10.8	31.0	58.50	de-en	27.8	44.4	77.57
fr-ru	8.6	26.7	57.07	fr-en	27.1	48.7	78.82
fr-zh	15.9	15.8	70.78	nl-en	22.6	42.2	74.25
nl-ar	2.2	15.4	63.47	ru-en	25.6	44.2	74.46
nl-de	10.2	28.5	53.65	zh-en	23.5	45.7	77.04
nl-fr	14.4	34.4	66.55	avg.	25.62	45.37	76.45
nl-ru	5.3	19.3	55.53				
nl-zh	5.5	6.2	58.77				
ru-ar	6.5	16.0	62.69				
ru-de	6.1	24.3	52.89				
ru-fr	18.2	39.0	69.95				
ru-nl	6.3	22.5	54.36				
ru-zh	7.6	13.3	61.94				
zh-ar	8.7	26.5	70.88				
zh-de	3.0	19.5	50.82				
zh-fr	17.7	39.7	73.56				
zh-nl	4.4	19.3	54.20				
zh-ru	4.1	19.5	50.47				
avg.	8.86	23.30	60.70				
seen→seen	14.77	27.80	71.07	en→seen	15.80	28.47	72.35
seen→unseen	6.31	23.08	54.81	en→unseen	17.47	35.00	65.24
unseen→seen	8.61	20.24	63.81	seen→en	25.90	47.13	77.47
unseen→unseen	7.15	23.70	54.51	unseen→en	25.33	43.60	75.43

Table D.10: Detailed results of BLOOMZ-7b1 fine-tuned with MT+Align for BLOOMZ+3 de-nl-ru.

Zero-shot	BLEU	chrF++	COMET	Supervised	BLEU	chrF++	COMET
ar-de	6.9	24.7	58.10	en-ar	14.6	35.6	76.70
ar-fr	26.2	48.2	74.96	en-de	20.4	36.0	65.96
ar-nl	8.8	24.7	59.53	en-fr	27.9	50.0	77.65
ar-ru	6.5	22.7	55.33	en-nl	14.8	34.8	63.11
ar-zh	28.6	22.3	77.64	en-ru	13.3	29.0	61.43
de-ar	3.3	19.8	68.27	en-zh	36.1	27.7	80.31
de-fr	15.2	35.8	67.05	avg.	21.18	35.52	70.86
de-nl	8.2	26.0	53.35				
de-ru	4.4	17.9	54.79				
de-zh	12.0	9.9	65.20				
fr-ar	14.2	35.2	74.84		BLEU	chrF++	COMET
fr-de	8.9	28.4	53.81	ar-en	33.7	53.5	79.81
fr-nl	10.1	29.9	56.92	de-en	27.1	43.9	77.04
fr-ru	8.1	26.0	55.96	fr-en	29.6	51.0	79.60
fr-zh	30.2	25.6	79.43	nl-en	22.0	41.4	73.54
nl-ar	3.1	18.2	67.72	ru-en	25.1	43.9	74.05
nl-de	10.4	27.7	52.67	zh-en	32.6	54.3	79.75
nl-fr	16.9	37.3	68.46	avg.	28.35	48.00	77.30
nl-ru	4.8	17.8	54.71				
nl-zh	8.1	7.0	63.96				
ru-ar	11.9	31.5	72.45				
ru-de	6.1	23.7	52.74				
ru-fr	21.2	42.5	71.71				
ru-nl	6.8	22.6	53.91				
ru-zh	21.3	20.7	74.63				
zh-ar	13.1	34.1	74.92				
zh-de	4.1	22.3	52.13				
zh-fr	23.8	46.5	76.54				
zh-nl	4.8	19.9	54.26				
zh-ru	5.7	21.9	50.60				
avg.	11.79	26.36	63.22				
seen→seen	22.68	35.32	76.39	en→seen	26.20	37.77	78.22
seen→unseen	7.10	24.50	55.18	en→unseen	16.17	33.27	63.50
unseen→seen	12.56	24.74	68.83	seen→en	31.97	52.93	79.72
unseen→unseen	6.78	22.62	53.69	unseen→en	24.73	43.07	74.88

Table D.11: Detailed results of BLOOMZ-7b1 fine-tuned with MTInstruct for BLOOMZ+3 ar-de-fr-nl-ru-zh.

Zero-shot	BLEU	chrF++	COMET	Supervised	BLEU	chrF++	COMET
ar-de	6.7	24.2	57.45	en-ar	15.1	35.8	76.76
ar-fr	27.5	49.2	75.21	en-de	20.6	35.9	65.88
ar-nl	8.7	24.8	59.14	en-fr	27.5	49.4	77.46
ar-ru	6.7	21.6	55.04	en-nl	15.0	35.6	63.70
ar-zh	30.1	24.4	78.54	en-ru	13.5	29.5	61.62
de-ar	3.5	19.7	68.39	en-zh	36.3	27.7	80.52
de-fr	15.4	35.8	67.81	avg.	21.33	35.65	70.99
de-nl	9.6	27.3	53.74				
de-ru	4.7	17.9	54.23				
de-zh	12.0	9.9	65.40				
fr-ar	14.9	36.3	74.98		BLEU	chrF++	COMET
fr-de	9.2	28.3	52.96	ar-en	33.9	53.7	79.74
fr-nl	11.3	31.1	57.62	de-en	27.1	43.6	77.13
fr-ru	8.8	26.2	56.31	fr-en	29.7	51.0	80.03
fr-zh	31.1	26.9	79.93	nl-en	22.6	42.3	73.94
nl-ar	3.3	18.5	68.02	ru-en	25.8	44.5	74.07
nl-de	9.4	26.5	52.33	zh-en	32.5	54.5	80.01
nl-fr	17.2	37.3	68.38	avg.	28.60	48.27	77.49
nl-ru	4.4	17.1	53.63				
nl-zh	8.3	7.0	64.08				
ru-ar	12.4	32.1	72.40				
ru-de	5.7	22.9	51.90				
ru-fr	21.5	42.7	72.08				
ru-nl	6.3	22.1	53.32				
ru-zh	22.7	24.6	75.36				
zh-ar	13.9	35.4	75.68				
zh-de	3.6	21.3	51.32				
zh-fr	24.5	47.0	76.98				
zh-nl	4.9	20.3	54.30				
zh-ru	5.5	21.1	50.49				
avg.	12.13	26.65	63.23				
seen→seen	23.67	36.53	76.89	en→seen	26.30	37.63	78.25
seen→unseen	7.27	24.32	54.96	en→unseen	16.37	33.67	63.73
unseen→seen	12.92	25.29	69.10	seen→en	32.03	53.07	79.93
unseen→unseen	6.68	22.30	53.19	unseen→en	25.17	43.47	75.05

Table D.12: Detailed results of BLOOMZ-7b1 fine-tuned with MT+Align for BLOOMZ+3 ar-de-fr-nl-ru-zh.

Appendix E

Supplementary Materials of LayerNorm

E.1 Discussion about SVCCA score

In previous work [236, 108], the SVCCA score [168], a cosine similarity measure between the hidden states of neural models, was used to compare two ZST models. However, we demonstrate that this method is unsuitable for comparing different ZST systems through an experiment. We removed the final LayerNorm from the PreNorm encoder, denoting it as "PreNorm w/o Enc-Last." We then evaluated the BLEU scores of PreNorm, PostNorm, and "PreNorm w/o Enc-Last" on the OPUS dataset, as reported in Table E.1. We subsequently calculated the encoder layer-wise SVCCA score for each LayerNorm setting using the mean-pooled hidden states of each encoder layer. The average SVCCA score between all the "en-xx" and "xx-en" directions is reported in Figure E.1. When comparing Figure E.1 with Table E.1, we observe that PostNorm has a higher SVCCA score on top of the encoder (L6) than PreNorm, which suggests that the encoder of PostNorm is more language-agnostic and thus has a higher ZST BLEU score in Table E.1, aligning with the results found in Wu et al. [236] and Liu et al. [108]. However, "PreNorm w/o Enc-Last" shows an extremely high SVCCA score on top of the encoder, whereas its ZST BLEU performance is significantly lower than PostNorm by 6.3 BLEU points. This reveals the significant inconsistency between the SVCCA score

	Zero-shot	Supervised
PreNorm	9.8	33.8
PostNorm	17.5	33.8
PreNorm w/o Enc-Last	11.2	33.7

Table E.1: **BLEU scores of PreNorm, PostNorm, and "PreNorm w/o Enc-Last" on OPUS**. They are trained with the "S-ENC-T-DEC" tag, "Res.," and the random seed of 10. We report the mean of all the translation directions.

#	**LayerNorm-simple**	Language Tag	Res.	Zero-shot			Supervised		
				OPUS	IWSLT	Europarl	OPUS	IWSLT	Europarl
1	**PreNorm-simple**	S-ENC-T-DEC	w/	10.1 (+0.0)	5.9 (+1.0)	25.0 (+0.1)	33.9 (+0.2)	31.9 (+0.4)	34.4 (+0.1)
2	**PostNorm-simple**	S-ENC-T-DEC	w/	**15.8** (-1.0)	**11.5** (-0.9)	**28.7** (-0.5)	34.1 (+0.2)	32.1 (+0.6)	34.5 (+0.0)
3	**PreNorm-simple**	T-ENC	w/	13.7 (+0.4)	14.5 (+0.8)	29.4 (-0.1)	33.9 (+0.2)	31.9 (+0.3)	34.4 (+0.0)
4	**PostNorm-simple**	T-ENC	w/	**14.9** (+0.9)	**15.4** (-0.1)	**30.8** (+0.0)	34.0 (-0.1)	31.9 (+0.4)	34.6 (+0.1)
5	**PreNorm-simple**	S-ENC-T-DEC	w/o	15.4 (+1.1)	7.8 (-0.2)	19.4 (+2.7)	33.7 (+0.1)	31.3 (+0.4)	34.1 (-0.2)
6	**PostNorm-simple**	S-ENC-T-DEC	w/o	**16.4** (+0.4)	**16.0** (-1.4)	**29.2** (+0.2)	33.9 (+0.1)	31.3 (+0.6)	34.4 (+0.0)
7	**PreNorm-simple**	T-ENC	w/o	13.1 (-0.3)	16.8 (+0.6)	28.7 (-1.2)	33.7 (+0.2)	31.4 (+0.5)	34.3 (+0.0)
8	**PostNorm-simple**	T-ENC	w/o	**14.0** (+0.1)	**17.9** (+0.1)	**31.0** (+0.2)	33.7 (-0.2)	31.1 (+0.5)	34.4 (+0.0)

Table E.2: **BLEU scores of LayerNorm-simple**. We report the average score of three seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer. We mark better scores between PreNorm-simple and PostNorm-simple in **bold**. For each setting, significantly better or worse BLEU scores [88] compared with the results in Table 8.2 are marked in blue or red.

and the performance of ZST models. Therefore, it is crucial to carefully consider how to leverage SVCCA for ZST analysis in the future.

On the other hand, our proposed LLR score is consistent with the ZST BLEU score, as shown in Figure E.2. Specifically, we observe the lowest LLR score on top of the encoder of PostNorm for the source language and the highest LLR scores in all the decoder layers, which aligns with its best ZST performance among the three systems.

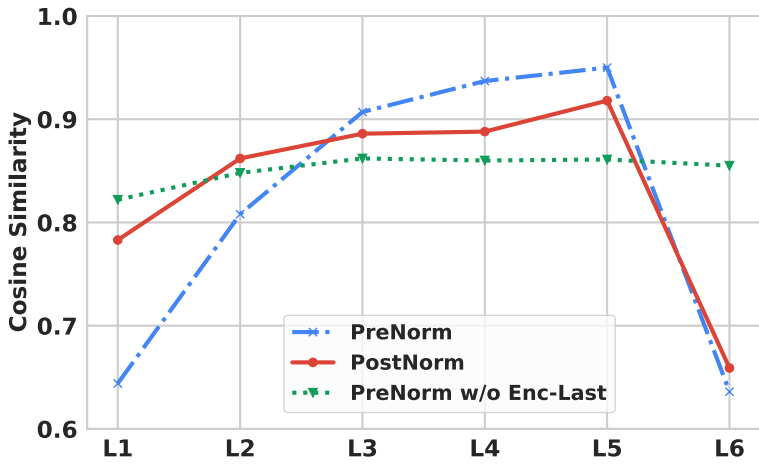

Figure E.1: **Encoder layer-wise SVCCA scores of PreNorm, PostNorm, and "PreNorm w/o Enc-Last" between "en-xx" and "xx-en" translation directions**. We report the mean of all the direction pairs.

E.2 Swap-PreNorm

Figure E.3 illustrates the implementation of Swap-PreNorm, which incorporates LayerNorm following the SA/FFN layers within the residual connection block. Compared with PostNorm, Swap-PreNorm alters the order of LayerNorm and residual connections. As depicted in the unraveled view of Swap-PreNorm in Figure E.3, it preserves the shallow sub-network characteristics of PreNorm, which is the main difference compared with PostNorm.

E.3 LayerNorm without Trainable Parameters

Xu et al. [240] demonstrated that the overfitting issue of PreNorm can be alleviated by removing the trainable parameters of LayerNorm (LayerNorm-simple). We apply this technique to our ZST experimental settings to investigate the overfitting state of PreNorm and PostNorm. PreNorm and PostNorm after applying this technique are denoted as PreNorm-simple and PostNorm-simple. As reported in Table E.2, the results indicate that PreNorm-simple and PostNorm-simple outperform their respective original versions in supervised directions, which aligns

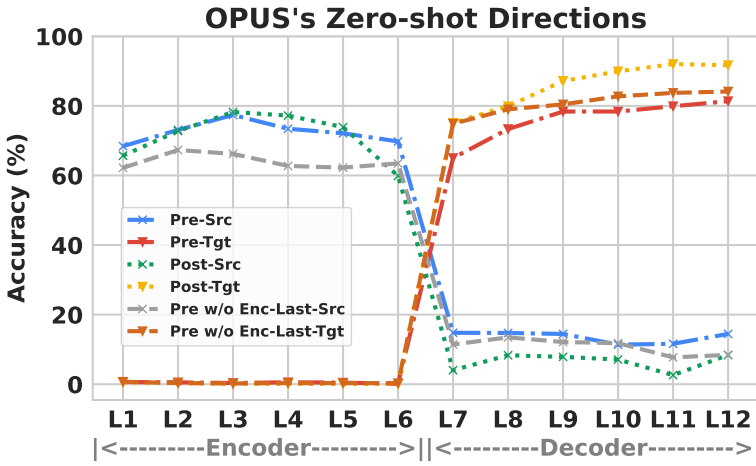

Figure E.2: **The LLR results of PreNorm, PostNorm, and "PreNorm w/o Enc-Last."** We report the mean of all the ZST directions. "-Src" and "-Tgt" indicate the LLR results for the source and target languages, respectively. "L1" to "L6" are 6 encoder layers and "L7" to "L12" are 6 decoder layers.

with the findings of Xu et al. [240]. Additionally, we observe comparable or better BLEU scores for PreNorm-simple than PreNorm (except for #7 on Europarl), indicating that the original PreNorm had low generalizability for ZST. For PostNorm-simple, we observe significant improvement only for #4 on OPUS, which suggests the superior generalizability of the original PostNorm for ZST. Despite this improvement, PreNorm-simple still underperforms PostNorm, highlighting the severe overfitting problem of the original PreNorm.

E.4 Details of the LLR Results

We show the LLR results of #3 - #8 (Table 8.2) for ZST and supervised directions in Figure E.4.

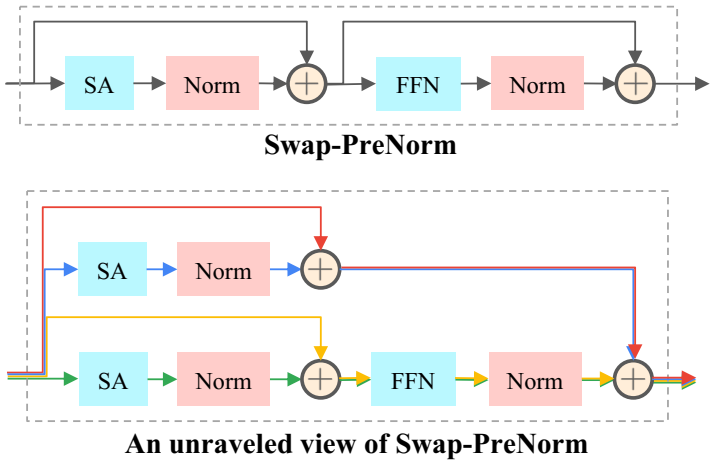

Figure E.3: **Swap-PreNorm, and an unraveled view of Swap-PreNorm in a Transformer encoder layer**. "Norm," "SA," and "FFN" denote LayerNorm, self-attention, and feed-forward network. $\oplus$ is residual connection. Paths with different colors in the unraveled view of PreNorm indicate respective sub-networks.

E.5 Details of the Main Results

We report the specific BLEU score for each translation direction and each random seed in Tables E.3, E.4, E.5, E.6, E.7, and E.8.[1] In addition to BLEU scores, we present model-based evaluation results obtained using BLEURT [185][2] in Table E.9. The results trend is consistent with those obtained from BLEU scores.

[1] Refer to details of setting random seeds in PyTorch at `https://pytorch.org/docs/stable/notes/randomness.html`.

[2] `https://github.com/google-research/bleurt`

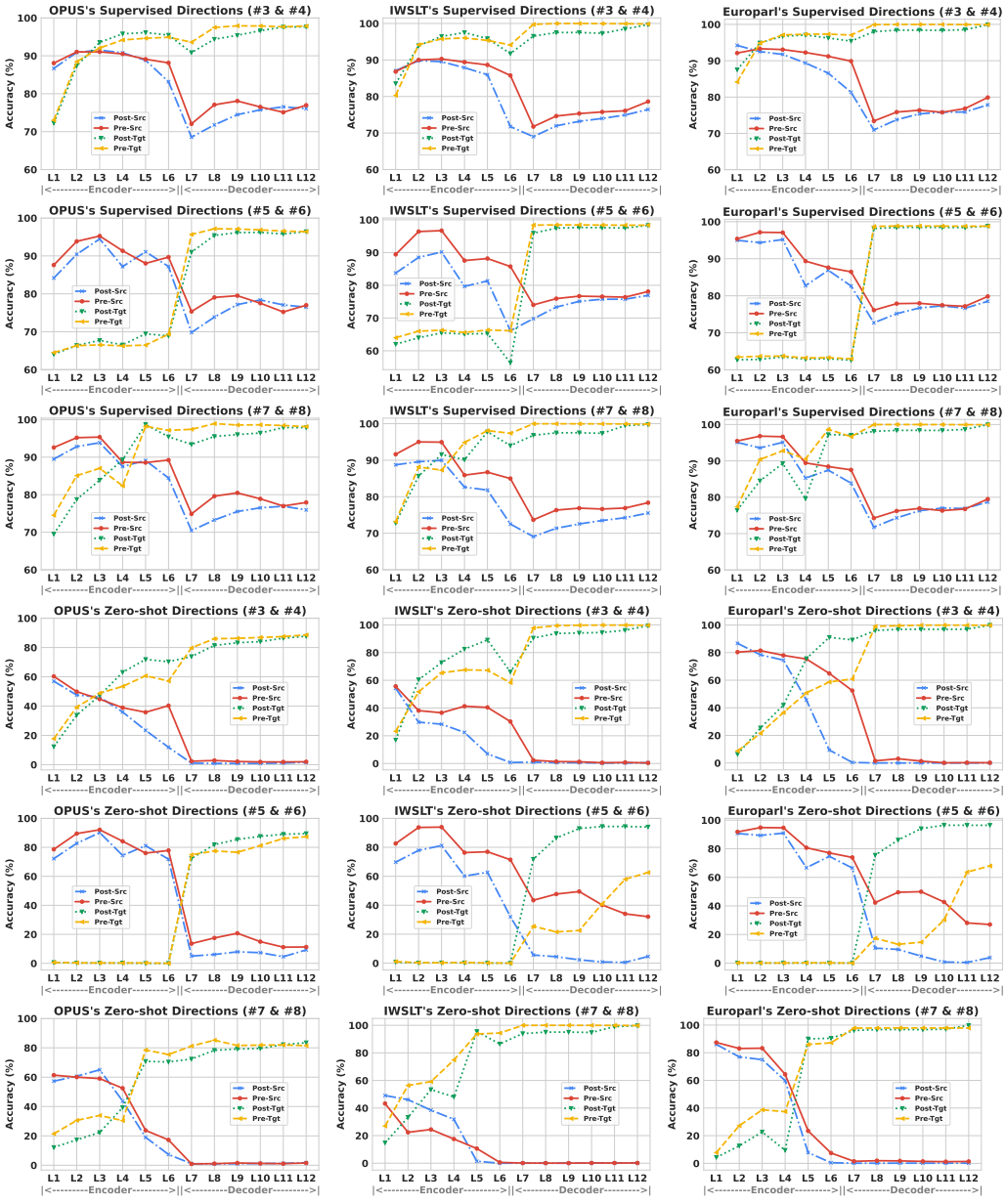

Figure E.4: **The LLR results of #3 - #8 (Table 8.2) for both ZST and supervised directions for each dataset**. "Pre-Src" and "Pre-Tgt" indicate the layer-wise source and target language recognition for a PreNorm system (#3, #5, or #7), while "Post-Src" and "Post-Tgt" denote similary for a PostNorm system (#4, #6, or #8).

Layer Norm	Direction	S-ENC-T-DEC w/ Res.				T-ENC w/ Res.				S-ENC-T-DEC w/o Res.				T-ENC w/o Res.			
		1	10	20	avg.	1	10	20	avg.	1	10	20	avg.	1	10	20	avg.
	ar-de	5.3	5.9	5.2	5.5	10.0	11.0	10.0	10.3	9.8	8.4	9.5	9.2	11.4	8.3	10.8	10.2
	ar-fr	17.5	16.1	17.2	16.9	16.3	19.9	18.3	18.2	19.9	20.3	20.8	20.3	20.9	18.6	20.4	20.0
	ar-nl	8.6	6.3	7.9	7.6	13.2	13.1	12.6	13.0	13.3	14.0	12.2	13.2	13.5	12.8	13.6	13.3
	ar-ru	8.1	9.1	9.5	8.9	14.8	16.2	15.9	15.6	13.0	10.9	13.0	12.3	19.6	17.8	19.6	19.0
	ar-zh	12.7	13.4	13.8	13.3	28.1	28.1	27.3	27.8	25.1	19.8	24.4	23.1	31.2	31.0	31.3	31.2
	de-ar	3.6	3.3	2.5	3.1	5.6	5.0	3.9	4.8	6.9	6.4	5.6	6.3	6.4	5.1	3.3	4.9
	de-fr	15.5	16.0	16.2	15.9	5.1	5.7	3.8	4.9	18.8	17.6	18.9	18.4	5.9	4.7	5.2	5.3
	de-nl	19.4	15.9	18.8	18.0	12.4	8.9	8.6	10.0	21.4	20.4	20.7	20.8	9.1	7.1	7.7	8.0
	de-ru	6.0	6.1	5.8	6.0	5.0	5.6	3.7	4.8	9.4	9.2	9.0	9.2	6.4	4.5	3.8	4.9
	de-zh	7.6	9.5	8.8	8.6	15.6	12.4	11.9	13.3	14.4	12.8	13.2	13.5	16.4	4.1	6.0	8.8
	fr-ar	9.5	7.5	8.5	8.5	15.5	16.2	13.2	15.0	15.4	13.1	14.4	14.3	18.5	16.5	15.8	16.9
	fr-de	10.4	10.6	11.6	10.9	6.3	7.2	4.9	6.1	14.1	11.0	15.2	13.4	4.5	4.6	4.0	4.4
	fr-nl	17.5	13.7	18.0	16.4	16.0	12.5	13.2	13.9	20.5	19.9	20.2	20.2	11.1	9.1	8.6	9.6
	fr-ru	8.8	8.4	9.3	8.8	12.1	12.8	10.9	11.9	13.3	9.2	12.0	11.5	16.5	7.4	8.4	10.8
	fr-zh	14.3	13.1	15.3	14.2	31.2	30.0	28.0	29.7	27.9	21.0	25.8	24.9	34.1	16.0	27.9	26.0
Pre.	nl-ar	2.6	2.0	1.7	2.1	5.2	5.6	5.3	5.4	4.3	5.8	4.2	4.8	5.5	5.0	5.0	5.2
	nl-de	14.3	14.4	13.9	14.2	12.8	13.9	11.3	12.7	16.9	14.8	18.3	16.7	13.8	6.9	10.9	10.5
	nl-fr	18.3	17.4	18.5	18.1	13.1	16.1	12.4	13.9	21.5	19.9	22.3	21.2	15.0	7.1	13.8	12.0
	nl-ru	4.2	4.4	3.4	4.0	9.5	9.8	8.6	9.3	7.2	6.5	7.3	7.0	10.3	6.6	7.3	8.1
	nl-zh	2.2	3.2	3.0	2.8	10.8	10.0	10.4	10.4	7.0	8.0	6.3	7.1	11.1	7.5	10.0	9.5
	ru-ar	9.7	7.6	7.6	8.3	15.6	16.1	14.6	15.4	15.9	13.3	14.0	14.4	18.6	19.1	18.0	18.6
	ru-de	7.7	9.1	7.2	8.0	8.5	10.0	6.0	8.2	10.5	10.0	10.9	10.5	8.4	5.6	6.8	6.9
	ru-fr	18.1	17.5	17.4	17.7	18.1	20.5	17.6	18.7	19.9	19.5	20.7	20.0	22.4	17.4	21.1	20.3
	ru-nl	10.2	8.6	9.9	9.6	11.5	11.7	9.5	10.9	13.0	13.1	12.4	12.8	12.7	8.2	10.1	10.3
	ru-zh	11.3	11.6	12.5	11.8	28.4	28.3	27.6	28.1	25.3	17.7	21.6	21.5	31.9	20.0	30.7	27.5
	zh-ar	9.1	7.6	7.2	8.0	15.2	16.6	14.5	15.4	15.6	12.7	15.1	14.5	18.4	18.8	18.7	18.6
	zh-fr	16.7	15.6	16.4	16.2	20.1	21.4	18.4	20.0	20.9	19.3	20.6	20.3	23.5	23.3	23.7	23.5
	zh-de	4.7	5.8	5.4	5.3	7.8	8.1	7.0	7.6	7.5	6.9	7.1	7.2	8.6	8.6	8.8	8.7
	zh-nl	6.9	5.4	6.0	6.1	8.6	8.6	8.2	8.5	8.5	8.0	8.0	8.2	9.1	9.2	8.8	9.0
	zh-ru	6.9	8.2	7.8	7.6	13.7	15.7	12.9	14.1	12.8	10.0	11.8	11.5	18.7	19.8	19.7	19.4
	avg.	10.3	9.8	10.2	**10.1**	13.5	13.9	12.4	**13.3**	15.0	13.3	14.5	**14.3**	15.1	11.7	13.3	**13.4**
	ar-de	11.4	11.0	10.3	10.9	10.1	10.4	9.9	10.1	10.1	11.9	9.9	10.6	11.0	11.0	10.0	10.7
	ar-fr	20.7	23.2	20.3	21.4	16.2	18.7	19.3	18.1	20.7	24.0	19.2	21.3	20.4	21.8	15.9	19.4
	ar-nl	13.3	13.7	12.5	13.2	12.8	13.5	13.3	13.2	13.4	14.4	12.5	13.4	13.2	13.9	13.0	13.4
	ar-ru	16.9	18.7	16.1	17.2	17.4	17.2	18.6	17.7	13.5	19.1	14.7	15.8	20.4	20.7	18.7	19.9
	ar-zh	28.6	29.4	29.2	29.1	29.2	30.4	30.3	30.0	26.1	30.7	27.4	28.1	32.9	32.9	31.9	32.6
	de-ar	7.2	7.2	6.6	7.0	5.7	5.6	5.8	5.7	6.9	7.6	7.6	7.4	4.4	4.1	3.1	3.9
	de-fr	17.6	19.3	18.2	18.4	5.1	6.6	5.8	5.8	17.3	20.3	17.3	18.3	5.4	7.9	4.1	5.8
	de-nl	21.4	21.8	20.4	21.2	9.1	9.5	7.9	8.8	20.0	22.3	20.5	20.9	9.7	11.9	7.1	9.6
	de-ru	12.3	13.8	12.8	13.0	6.0	6.3	7.2	6.5	10.1	13.3	10.5	11.3	5.2	4.0	3.7	4.3
	de-zh	16.1	16.9	16.5	16.5	8.9	15.3	15.0	13.1	11.2	16.9	13.5	13.9	14.1	11.1	3.1	9.4
	fr-ar	17.9	17.8	18.9	18.2	16.4	17.1	16.4	16.6	14.6	19.5	16.3	16.8	16.4	16.6	14.8	15.9
	fr-de	15.0	17.3	17.0	16.4	5.4	6.7	6.5	6.2	13.1	17.0	13.5	14.5	4.9	7.0	4.8	5.6
	fr-nl	21.4	21.8	20.3	21.2	11.3	13.3	11.6	12.1	20.6	22.7	20.5	21.3	11.6	14.1	10.1	11.9
	fr-ru	17.7	19.5	15.9	17.7	16.7	13.3	18.5	16.2	12.9	20.7	13.3	15.6	10.9	15.5	13.3	13.2
	fr-zh	30.5	32.0	31.8	31.4	29.8	32.0	31.4	31.1	25.9	32.5	28.4	28.9	31.7	32.0	30.3	31.3
Post.	nl-ar	5.3	5.9	5.6	5.6	6.0	5.3	5.8	5.7	5.2	6.1	6.4	5.9	5.0	5.2	4.5	4.9
	nl-de	17.9	19.7	19.1	18.9	10.9	12.8	10.5	11.4	16.5	19.8	17.1	17.8	9.0	10.4	10.4	9.9
	nl-fr	21.1	22.5	21.2	21.6	13.8	13.4	13.0	13.4	21.2	22.9	19.6	21.2	10.1	12.6	9.5	10.7
	nl-ru	10.0	11.2	10.2	10.5	9.7	9.1	8.8	9.2	8.4	10.9	8.6	9.3	8.6	7.6	8.2	8.1
	nl-zh	9.6	11.1	9.6	10.1	10.2	10.4	10.0	10.2	5.4	11.1	7.3	7.9	9.9	9.9	7.5	9.1
	ru-ar	18.7	18.7	18.2	18.5	16.9	17.9	17.5	17.4	14.8	19.7	16.2	16.9	17.9	18.9	17.0	17.9
	ru-de	12.9	12.9	12.9	12.9	8.7	8.1	9.0	8.6	10.8	13.3	10.5	11.5	8.6	9.2	7.9	8.6
	ru-fr	21.5	24.0	21.2	22.2	19.4	17.9	19.0	18.8	20.1	24.8	19.0	21.3	16.8	22.0	13.8	17.5
	ru-nl	13.0	13.6	12.7	13.1	10.9	11.8	12.4	11.7	13.3	14.2	13.0	13.5	11.0	12.0	9.7	10.9
	ru-zh	27.6	29.8	28.6	28.7	30.1	30.4	30.6	30.4	23.6	30.2	24.6	26.1	32.5	32.2	29.0	31.2
	zh-ar	18.0	17.4	17.3	17.6	16.9	17.5	17.1	17.2	16.3	19.3	17.0	17.5	19.1	19.8	19.4	19.4
	zh-fr	20.2	21.3	20.2	20.6	21.4	22.3	21.5	21.7	20.5	24.1	18.3	21.0	23.1	24.4	24.5	24.0
	zh-de	8.6	9.1	8.8	8.8	7.3	7.4	7.1	7.3	8.3	9.9	7.5	8.6	8.7	8.5	8.0	8.4
	zh-nl	8.7	8.5	8.1	8.4	8.9	8.7	8.4	8.7	8.9	9.0	8.1	8.7	8.9	9.3	9.0	9.1
	zh-ru	15.3	15.8	14.1	15.1	16.7	17.3	17.6	17.2	13.3	17.8	12.8	14.6	20.2	20.5	20.2	20.3
	avg.	16.5	17.5	16.5	**16.8**	13.6	14.2	14.2	**14.0**	14.8	18.2	15.0	**16.0**	14.1	14.9	12.8	**13.9**

Table E.3: **BLEU scores of OPUS in ZST directions**. Scores in **bold** are the results reported in Table 8.2. "1," "10," and "20" indicates three random seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer.

Layer Norm	Direction	S-ENC-T-DEC w/ Res.				T-ENC w/ Res.				S-ENC-T-DEC w/o Res.				T-ENC w/o Res.			
		1	10	20	avg.	1	10	20	avg.	1	10	20	avg.	1	10	20	avg.
Pre.	en-ar	23.6	24.1	23.2	23.6	23.7	23.9	24.1	23.9	24.0	23.2	23.1	23.4	22.8	23.8	23.8	23.5
	ar-en	37.6	37.1	37.3	37.3	37.5	37.1	37.5	37.4	37.4	37.2	36.9	37.2	36.4	36.7	37.0	36.7
	en-de	29.7	30.1	30.4	30.1	30.4	29.6	30.4	30.1	30.1	30.1	30.1	30.1	30.3	30.5	30.7	30.5
	de-en	34.3	34.5	34.2	34.3	34.5	34.1	34.3	34.3	35.0	34.7	34.3	34.7	33.8	34.1	34.4	34.1
	en-fr	33.5	33.7	33.6	33.6	33.4	33.8	33.6	33.6	33.7	33.1	33.8	33.5	33.0	33.6	33.1	33.2
	fr-en	35.6	35.4	35.3	35.4	35.0	35.0	35.5	35.2	35.6	35.2	35.1	35.3	34.4	35.2	35.0	34.9
	en-nl	27.7	28.4	28.2	28.1	28.4	27.9	28.3	28.2	27.6	28.0	27.9	27.8	28.1	28.1	28.0	28.1
	nl-en	31.3	30.8	31.2	31.1	30.9	30.7	30.8	30.8	31.0	30.8	31.0	30.9	30.4	30.9	30.5	30.6
	en-ru	29.2	29.7	29.6	29.5	29.4	29.8	29.8	29.7	29.5	29.1	29.6	29.4	29.4	29.9	29.2	29.5
	ru-en	35.2	34.6	35.0	34.9	34.7	34.6	35.0	34.8	35.2	34.8	35.1	35.0	34.3	34.8	34.7	34.6
	en-zh	40.7	40.8	40.9	40.8	40.6	40.3	40.7	40.5	40.7	40.4	40.6	40.6	39.6	40.7	40.6	40.3
	zh-en	46.2	46.1	45.9	46.1	46.1	46.1	46.2	46.1	46.2	45.9	45.8	46.0	45.6	46.4	46.3	46.1
	avg.	33.7	33.8	33.7	**33.7**	33.7	33.6	33.9	**33.7**	33.8	33.5	33.6	**33.7**	33.2	33.7	33.6	**33.5**
Post.	en-ar	23.9	23.4	23.7	23.7	24.6	24.4	24.3	24.4	23.7	23.8	23.8	23.8	24.0	23.8	24.0	23.9
	ar-en	37.8	37.3	37.5	37.5	37.8	37.5	37.2	37.5	37.7	37.2	37.6	37.5	37.8	37.3	37.7	37.6
	en-de	30.8	31.0	29.3	30.4	31.2	29.9	31.2	30.8	31.1	30.5	31.2	30.9	31.1	30.5	31.5	31.0
	de-en	34.6	34.6	34.8	34.7	34.9	34.6	34.7	34.7	34.8	34.6	34.7	34.7	34.4	34.6	34.4	34.5
	en-fr	33.9	33.4	34.1	33.8	34.1	33.8	33.9	33.9	33.5	33.5	33.2	33.4	33.7	33.8	33.6	33.7
	fr-en	35.5	35.6	35.4	35.5	35.6	35.7	35.4	35.6	35.0	35.5	35.2	35.2	35.3	35.3	35.5	35.4
	en-nl	27.8	28.4	28.2	28.1	27.9	28.8	28.3	28.3	28.0	27.9	28.3	28.1	27.7	27.9	28.4	28.0
	nl-en	31.5	30.9	31.2	31.2	31.3	30.9	31.4	31.2	30.8	30.8	30.7	30.8	31.1	31.1	30.9	31.0
	en-ru	29.4	29.6	29.9	29.6	30.1	29.8	30.0	30.0	29.9	30.0	29.2	29.7	30.0	29.5	29.5	29.7
	ru-en	35.1	34.6	35.1	34.9	34.9	34.9	35.2	35.0	34.8	34.9	35.2	35.0	34.8	34.8	35.0	34.9
	en-zh	41.2	40.9	40.9	41.0	41.2	40.9	40.8	41.0	40.8	40.5	40.7	40.7	40.7	40.7	41.0	40.8
	zh-en	46.4	46.0	46.1	46.2	46.7	46.3	46.2	46.4	46.1	46.3	46.1	46.2	46.7	46.6	46.0	46.4
	avg.	34.0	33.8	33.9	**33.9**	34.2	34.0	34.1	**34.1**	33.9	33.8	33.8	**33.8**	33.9	33.8	34.0	**33.9**

Table E.4: **BLEU scores of OPUS in supervised directions**. Scores in **bold** are the results reported in Table 8.2. "1," "10," and "20" indicates three random seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer.

Layer Norm	Direction	S-ENC-T-DEC w/ Res.				T-ENC w/ Res.				S-ENC-T-DEC w/o Res.				T-ENC w/o Res.			
		1	10	20	avg.	1	10	20	avg.	1	10	20	avg.	1	10	20	avg.
Pre.	it-nl	5.2	3.7	4.3	4.4	13.4	14.4	14.0	13.9	6.4	3.6	13.8	7.9	16.3	17.7	17.2	17.1
	nl-it	5.5	4.3	4.3	4.7	13.9	14.7	14.4	14.3	6.1	4.6	10.8	7.2	15.5	17.0	17.1	16.5
	it-ro	5.5	5.7	5.1	5.4	13.4	13.5	14.4	13.8	7.8	7.4	14.2	9.8	16.0	16.6	16.9	16.5
	ro-it	7.2	5.5	5.3	6.0	14.9	15.1	15.4	15.1	7.1	4.3	11.4	7.6	17.8	18.1	18.4	18.1
	nl-ro	4.5	4.9	4.2	4.5	12.1	12.5	12.4	12.3	6.1	7.1	11.8	8.3	12.8	14.1	14.1	13.7
	ro-nl	4.4	4.3	3.9	4.2	12.1	13.4	12.5	12.7	5.6	3.1	12.4	7.0	15.1	16.1	15.6	15.6
	avg.	5.4	4.7	4.5	**4.9**	13.3	13.9	13.9	**13.7**	6.5	5.0	12.4	**8.0**	15.6	16.6	16.6	**16.2**
Post.	it-nl	13.7	11.8	13.1	12.9	15.9	16.3	17.0	16.4	17.7	18.3	17.4	17.8	18.4	18.0	18.6	18.3
	nl-it	14.5	12.8	12.2	13.2	15.7	17.0	16.1	16.3	18.0	18.5	18.4	18.3	17.9	18.3	18.3	18.2
	it-ro	12.3	11.2	12.4	12.0	14.8	14.3	15.8	15.0	17.0	17.3	17.0	17.1	17.9	17.8	18.2	18.0
	ro-it	14.6	13.7	13.0	13.8	17.2	16.8	17.5	17.2	19.5	20.0	20.0	19.8	19.2	19.8	20.8	19.9
	nl-ro	11.1	10.4	10.2	10.6	13.5	13.4	13.6	13.5	14.9	14.9	14.7	14.8	15.4	15.2	15.5	15.4
	ro-nl	12.3	10.9	12.2	11.8	14.5	15.0	15.2	14.9	16.5	16.6	16.0	16.4	16.9	16.2	17.1	16.7
	avg.	13.1	11.8	12.2	**12.4**	15.3	15.5	15.9	**15.5**	17.3	17.6	17.3	**17.4**	17.6	17.6	18.1	**17.8**

Table E.5: **BLEU scores of IWSLT in ZST directions**. Scores in **bold** are the results reported in Table 8.2. "1," "10," and "20" indicates three random seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer.

Layer Norm	Direction	S-ENC-T-DEC w/ Res.				T-ENC w/ Res.				S-ENC-T-DEC w/o Res.				T-ENC w/o Res.			
		1	10	20	avg.	1	10	20	avg.	1	10	20	avg.	1	10	20	avg.
Pre.	en-it	33.9	33.8	33.6	33.8	33.7	33.4	33.7	33.6	33.6	32.9	33.3	33.3	32.4	33.3	33.4	33.0
	it-en	37.5	37.1	37.1	37.2	37.4	37.2	37.0	37.2	35.8	36.3	36.5	36.2	35.8	36.7	36.5	36.3
	en-nl	29.6	29.5	29.4	29.5	29.6	29.5	29.6	29.6	29.2	29.7	29.5	29.5	29.0	29.2	29.2	29.1
	nl-en	31.9	32.4	32.0	32.1	32.0	32.1	31.9	32.0	30.9	31.3	31.7	31.3	31.2	31.5	31.5	31.4
	en-ro	24.4	25.1	25.1	24.9	25.2	25.1	25.4	25.2	24.4	24.6	24.4	24.5	24.6	24.7	24.6	24.6
	ro-en	31.3	31.6	31.3	31.4	32.1	31.6	31.4	31.7	30.3	30.7	30.9	30.6	30.3	31.2	31.2	30.9
	avg.	31.4	31.6	31.4	**31.5**	31.7	31.5	31.5	**31.6**	30.7	30.9	31.1	**30.9**	30.6	31.1	31.1	**30.9**
Post.	en-it	33.9	33.3	33.5	33.6	33.8	34.0	33.5	33.8	33.1	33.2	32.6	33.0	32.4	32.6	33.4	32.8
	it-en	37.1	36.9	37.0	37.0	37.1	37.1	36.9	37.0	35.7	35.4	36.1	35.7	36.4	35.7	35.8	36.0
	en-nl	29.6	30.1	30.1	29.9	30.4	30.4	30.0	30.3	29.2	29.0	29.0	29.1	29.2	29.0	29.5	29.2
	nl-en	31.9	32.0	31.6	31.8	31.3	31.9	31.8	31.7	31.0	31.1	31.7	31.3	30.9	30.7	31.3	31.0
	en-ro	25.4	25.2	24.6	25.1	25.3	25.2	25.5	25.3	24.7	25.0	24.6	24.8	24.4	24.4	25.0	24.6
	ro-en	31.5	31.6	31.6	31.6	30.8	31.4	31.1	31.1	30.4	29.6	30.8	30.3	30.4	30.1	30.4	30.3
	avg.	31.6	31.5	31.4	**31.5**	31.5	31.7	31.5	**31.5**	30.7	30.6	30.8	**30.7**	30.6	30.4	30.9	**30.6**

Table E.6: **BLEU scores of IWSLT in supervised directions**. Scores in **bold** are the results reported in Table 8.2. "1," "10," and "20" indicates three random seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer.

Layer Norm	Direction	S-ENC-T-DEC w/ Res.				T-ENC w/ Res.				S-ENC-T-DEC w/o Res.				T-ENC w/o Res.			
		1	10	20	avg.	1	10	20	avg.	1	10	20	avg.	1	10	20	avg.
Pre.	es-de	23.2	22.0	16.1	20.4	26.7	26.9	27.3	27.0	6.2	14.1	11.2	10.5	24.9	28.5	28.3	27.2
	de-es	30.3	30.0	27.6	29.3	32.4	32.0	32.3	32.2	15.5	25.7	18.7	20.0	32.9	33.1	33.4	33.1
	es-fr	35.0	35.6	34.0	34.9	38.8	38.8	39.3	39.0	27.8	29.8	28.2	28.6	39.9	39.8	39.9	39.9
	fr-es	36.0	35.5	32.8	34.8	38.6	38.7	38.7	38.7	18.7	30.7	22.3	23.9	39.7	39.7	40.0	39.8
	es-nl	22.7	23.0	14.2	20.0	26.4	26.3	26.3	26.3	7.0	12.8	15.0	11.6	23.2	27.7	27.5	26.1
	nl-es	27.2	27.1	24.9	26.4	29.1	29.1	29.1	29.1	13.9	23.0	16.9	17.9	29.6	29.7	29.8	29.7
	de-fr	28.6	28.1	26.9	27.9	31.4	31.3	31.7	31.5	21.9	23.0	22.5	22.5	31.9	32.3	32.2	32.1
	fr-de	23.5	22.0	15.9	20.5	26.3	26.5	26.8	26.5	6.3	14.3	11.5	10.7	25.0	28.1	28.2	27.1
	de-nl	23.2	23.4	15.0	20.5	26.3	26.2	26.0	26.2	7.0	12.8	16.2	12.0	22.5	27.5	27.2	25.7
	nl-de	21.4	20.3	14.3	18.7	23.2	23.8	23.5	23.5	6.4	13.3	11.9	10.5	21.6	24.6	24.6	23.6
	fr-nl	22.9	23.3	14.1	20.1	26.0	25.9	25.8	25.9	6.8	12.2	15.3	11.4	21.6	27.4	27.1	25.4
	nl-fr	26.0	25.9	25.0	25.6	28.1	28.3	28.2	28.2	19.9	20.9	19.9	20.2	28.9	28.8	28.7	28.8
	avg.	26.7	26.4	21.7	**24.9**	29.4	29.5	29.6	**29.5**	13.1	19.4	17.5	**16.7**	28.5	30.6	30.6	**29.9**
Post.	es-de	26.0	26.9	26.8	26.6	28.2	28.4	28.7	28.4	26.1	26.3	26.1	26.2	28.7	28.7	28.7	28.7
	de-es	32.3	32.6	32.1	32.3	33.2	33.7	33.5	33.5	32.7	31.9	32.1	32.2	33.5	33.3	33.5	33.4
	es-fr	37.7	38.8	37.5	38.0	40.2	40.0	40.1	40.1	37.9	37.8	37.7	37.8	40.1	39.9	40.5	40.2
	fr-es	37.8	38.5	38.2	38.2	40.0	39.9	40.1	40.0	38.4	37.7	38.0	38.0	39.7	39.7	40.1	39.8
	es-nl	25.6	26.0	26.2	25.9	27.9	27.7	27.8	27.8	26.0	25.7	25.5	25.7	27.8	28.0	27.9	27.9
	nl-es	29.3	29.3	29.1	29.2	29.8	30.0	29.6	29.8	29.4	29.0	29.2	29.2	29.7	29.8	29.8	29.8
	de-fr	30.6	31.7	30.8	31.0	32.8	32.8	33.1	32.9	31.0	30.7	30.8	30.8	32.9	32.4	33.3	32.9
	fr-de	25.9	26.4	26.6	26.3	27.8	28.6	28.8	28.4	26.3	26.0	25.1	25.8	28.2	28.5	28.3	28.3
	de-nl	25.8	26.0	25.9	25.9	27.5	27.7	27.5	27.6	25.7	25.6	25.5	25.6	27.8	27.6	27.5	27.6
	nl-de	23.5	23.4	23.9	23.6	24.2	24.6	24.4	24.4	23.6	23.5	23.2	23.4	24.4	24.5	24.5	24.5
	fr-nl	25.3	25.8	25.6	25.6	27.4	27.4	27.3	27.4	25.5	25.5	25.3	25.4	27.8	27.6	27.5	27.6
	nl-fr	28.1	28.4	27.9	28.1	29.3	29.0	29.3	29.2	28.3	28.0	27.9	28.1	29.2	29.1	29.3	29.2
	avg.	29.0	29.5	29.2	**29.2**	30.7	30.8	30.9	**30.8**	29.2	29.0	28.9	**29.0**	30.8	30.8	30.9	**30.8**

Table E.7: **BLEU scores of Europarl in ZST directions**. Scores in **bold** are the results reported in Table 8.2. "1," "10," and "20" indicates three random seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer.

Layer Norm	Direction	S-ENC-T-DEC w/ Res.				T-ENC w/ Res.				S-ENC-T-DEC w/o Res.				T-ENC w/o Res.			
		1	10	20	avg.	1	10	20	avg.	1	10	20	avg.	1	10	20	avg.
Pre.	en-de	28.0	28.0	28.3	28.1	28.2	28.2	28.4	28.3	28.0	28.1	28.4	28.2	28.5	28.5	28.3	28.4
	de-en	35.2	35.1	35.3	35.2	35.1	35.0	35.1	35.1	34.9	35.0	35.0	35.0	34.8	35.1	35.0	35.0
	en-es	37.6	37.4	37.4	37.5	37.5	37.4	37.7	37.5	37.5	37.5	37.4	37.5	37.5	37.5	37.3	37.4
	es-en	39.3	38.9	39.0	39.1	39.0	39.0	38.9	39.0	38.8	39.0	39.1	39.0	38.6	39.0	38.9	38.8
	en-fr	36.2	36.6	36.5	36.4	36.5	36.4	36.8	36.6	36.3	36.4	36.5	36.4	36.7	36.7	36.2	36.5
	fr-en	38.2	38.2	38.0	38.1	38.0	38.2	38.0	38.1	38.0	37.9	38.2	38.0	37.8	38.2	38.0	38.0
	en-nl	28.5	28.8	28.7	28.7	28.8	28.7	28.6	28.7	28.5	28.6	28.6	28.6	28.3	28.6	28.3	28.4
	nl-en	31.7	31.6	31.5	31.6	31.5	31.7	31.9	31.7	31.6	31.3	31.6	31.5	31.3	31.7	31.6	31.5
	avg.	34.3	34.3	34.3	**34.3**	34.3	34.3	34.4	**34.4**	34.2	34.2	34.4	**34.3**	34.2	34.4	34.2	**34.3**
Post.	en-de	28.4	28.4	28.7	28.5	28.6	28.7	29.0	28.8	28.5	28.2	28.4	28.4	28.7	28.5	28.3	28.5
	de-en	35.2	35.0	35.5	35.2	34.8	35.1	34.9	34.9	35.2	35.2	35.0	35.1	35.1	35.1	34.7	35.0
	en-es	37.6	37.8	37.5	37.6	37.6	37.7	37.6	37.6	37.6	37.5	37.6	37.6	37.3	37.4	37.5	37.4
	es-en	39.4	39.0	39.0	39.1	39.0	39.3	38.8	39.0	39.2	38.9	39.1	39.1	39.0	39.1	39.1	39.1
	en-fr	36.8	36.8	36.4	36.7	36.8	36.7	37.0	36.8	36.6	36.5	37.1	36.7	36.9	36.8	36.7	36.8
	fr-en	38.3	38.2	38.4	38.3	38.2	38.2	38.4	38.3	38.2	38.1	38.2	38.2	38.1	38.3	37.9	38.1
	en-nl	28.8	28.8	28.6	28.7	28.7	28.7	28.9	28.8	28.6	28.6	28.9	28.7	28.7	28.7	28.5	28.6
	nl-en	31.5	31.6	31.7	31.6	32.1	31.7	31.7	31.8	31.7	31.9	31.5	31.7	31.7	31.4	31.4	31.5
	avg.	34.5	34.5	34.5	**34.5**	34.5	34.5	34.5	**34.5**	34.5	34.4	34.5	**34.4**	34.4	34.4	34.3	**34.4**

Table E.8: **BLEU scores of Europarl supervised directions**. Scores in **bold** are the results reported in Table 8.2. "1," "10," and "20" indicates three random seeds. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer.

#	**Layer Norm**	Language Tag	Res.	Zero-shot			Supervised		
				OPUS	IWSLT	Europarl	OPUS	IWSLT	Europarl
0	*Pivot*			55.8	64.6	73.8	-	-	-
1	**PreNorm**	S-ENC-T-DEC	w/	35.9	34.6	66.5	63.8	70.6	74.9
2	**PostNorm**	S-ENC-T-DEC	w/	**49.1**	**51.2**	**73.0**	64.1	70.6	75.0
3	**PreNorm**	T-ENC	w/	42.5	53.0	73.0	63.7	70.6	74.9
4	**PostNorm**	T-ENC	w/	**43.8**	**56.0**	**73.8**	64.0	70.7	75.0
5	**PreNorm**	S-ENC-T-DEC	w/o	44.5	41.7	50.3	63.7	70.0	74.8
6	**PostNorm**	S-ENC-T-DEC	w/o	**47.6**	**60.8**	**72.9**	64.0	69.7	74.9
7	**PreNorm**	T-ENC	w/o	42.5	57.1	72.5	63.6	69.9	74.8
8	**PostNorm**	T-ENC	w/o	**43.1**	**60.2**	**73.8**	64.0	69.7	74.9

Table E.9: **BLEURT scores**. We report the mean of three seeds and all the translation directions. "Res." indicates the residual connection of self-attention in the 4^{th} encoder layer. We mark better scores between PreNorm and PostNorm in **bold** for ZST.

Bibliography

[1] Z. Agic and I. Vulic. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 3204–3210, 2019.

[2] S. Agrawal, C. Zhou, M. Lewis, L. Zettlemoyer, and M. Ghazvininejad. In-context examples selection for machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8857–8873, Toronto, Canada, July 2023. Association for Computational Linguistics.

[3] R. Aharoni, M. Johnson, and O. 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.

[4] M. Al-Shedivat and A. Parikh. Consistency by agreement in zero-shot 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 1184–1197, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[5] R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen, E. Chu, J. H. Clark, L. E. Shafey, Y. Huang, K. Meier-Hellstern, G. Mishra, E. Moreira, M. Omernick, K. Robinson,

S. Ruder, Y. Tay, K. Xiao, Y. Xu, Y. Zhang, G. H. Ábrego, J. Ahn, J. Austin, P. Barham, J. A. Botha, J. Bradbury, S. Brahma, K. Brooks, M. Catasta, Y. Cheng, C. Cherry, C. A. Choquette-Choo, A. Chowdhery, C. Crepy, S. Dave, M. Dehghani, S. Dev, J. Devlin, M. Díaz, N. Du, E. Dyer, V. Feinberg, F. Feng, V. Fienber, M. Freitag, X. Garcia, S. Gehrmann, L. Gonzalez, and et al. Palm 2 technical report. *CoRR*, abs/2305.10403, 2023.

[6] A. Araabi and C. Monz. Optimizing transformer for low-resource neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3429–3435, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Linguistics.

[7] N. Arivazhagan, A. Bapna, O. Firat, R. Aharoni, M. Johnson, and W. Macherey. The missing ingredient in zero-shot neural machine translation. *CoRR*, abs/1903.07091, 2019.

[8] N. Arivazhagan, A. Bapna, O. Firat, D. Lepikhin, M. Johnson, M. Krikun, M. X. Chen, Y. Cao, G. F. Foster, C. Cherry, W. Macherey, Z. Chen, and Y. Wu. Massively multilingual neural machine translation in the wild: Findings and challenges. *CoRR*, abs/1907.05019, 2019.

[9] S. Arora, Y. Liang, and T. Ma. A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations*, 2017.

[10] M. Artetxe, S. Ruder, and D. 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.

[11] M. Artetxe and H. 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.

[12] M. Artetxe and H. Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610, Mar. 2019.

[13] M. Aulamo, U. Sulubacak, S. Virpioja, and J. Tiedemann. Opustools and parallel corpus diagnostics. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3782–3789, 2020.

[14] L. J. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

[15] A. Baevski and M. Auli. Adaptive input representations for neural language modeling. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[16] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[17] M. Bañón, P. Chen, B. Haddow, K. Heafield, H. Hoang, M. Esplà-Gomis, M. L. Forcada, A. Kamran, F. Kirefu, P. Koehn, S. Ortiz-Rojas, L. P. Sempere, G. Ramírez-Sánchez, E. Sarrías, M. Strelec, B. Thompson, W. Waites, D. Wiggins, and J. Zaragoza. Paracrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, 2020.

[18] A. Bapna and O. Firat. Simple, scalable adaptation for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1538–1548, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[19] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In A. P. Danyluk, L. Bottou, and M. L. Littman, editors, *Proceedings of the 26th Annual International Conference on Machine Learning, ICML*

2009, Montreal, Quebec, Canada, June 14-18, 2009, volume 382 of *ACM International Conference Proceeding Series*, pages 41–48. ACM, 2009.

[20] G. Blackwood, M. Ballesteros, and T. Ward. Multilingual neural machine translation with task-specific attention. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3112–3122, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics.

[21] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, 2015.

[22] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[23] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[24] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[25] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, B. Strope, and R. Kurzweil. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 169–174, 2018.

[26] M. Cettolo, M. Federico, L. Bentivogli, J. Niehues, S. Stüker, K. Sudoh, K. Yoshino, and C. Federmann. Overview of the IWSLT 2017 evaluation campaign. In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 2–14, Tokyo, Japan, Dec. 14-15 2017. International Workshop on Spoken Language Translation.

[27] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020.

[28] Q. Cheng, X. Yang, T. Sun, L. Li, and X. Qiu. Improving contrastive learning of sentence embeddings from AI feedback. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11122–11138, 2023.

[29] M. Chidambaram, Y. Yang, D. Cer, S. Yuan, Y. Sung, B. Strope, and R. Kurzweil. Learning cross-lingual sentence representations via a multi-task dual-encoder model. In *Proceedings of the 4th Workshop on Representation Learning for NLP*, pages 250–259, 2019.

[30] Y. J. Choe, K. Park, and D. Kim. word2word: A collection of bilingual lexicons for 3,564 language pairs. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3036–3045, Marseille, France, May 2020. European Language Resources Association.

[31] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang,

B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24:240:1–240:113, 2023.

[32] C. Christodoulopoulos and M. Steedman. A massively parallel corpus: the bible in 100 languages. *Lang. Resour. Evaluation*, 49(2):375–395, 2015.

[33] C. Chu, R. Dabre, and S. 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.

[34] C. Chu, T. Nakazawa, and S. Kurohashi. Constructing a Chinese—Japanese parallel corpus from Wikipedia. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 642–647, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA).

[35] C. Chu, T. Nakazawa, and S. Kurohashi. Integrated parallel sentence and fragment extraction from comparable corpora: A case study on chinese-japanese wikipedia. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 15(2):10:1–10:22, 2016.

[36] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, S. Narang, G. Mishra, A. Yu, V. Y. Zhao, Y. Huang, A. M. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416, 2022.

[37] K. Clark, M. Luong, Q. V. Le, and C. D. Manning. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[38] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. 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.

[39] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, 2017.

[40] A. Conneau and G. Lample. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7057–7067, 2019.

[41] A. Conneau, R. Rinott, G. Lample, A. Williams, S. R. Bowman, H. Schwenk, and V. Stoyanov. XNLI: evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, 2018.

[42] M. R. Costa-jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard, A. Sun, S. Wang, G. Wenzek, A. Youngblood, B. Akula, L. Barrault, G. M. Gonzalez, P. Hansanti, J. Hoffman, S. Jarrett, K. R. Sadagopan, D. Rowe, S. Spruit, C. Tran, P. Andrews, N. F. Ayan, S. Bhosale, S. Edunov, A. Fan, C. Gao, V. Goswami, F. Guzmán, P. Koehn, A. Mourachko, C. Ropers, S. Saleem, H. Schwenk, and J. Wang. No language left behind: Scaling human-centered machine translation. *CoRR*, abs/2207.04672, 2022.

[43] R. Dabre, C. Chu, and A. Kunchukuttan. A survey of multilingual neural machine translation. *ACM Comput. Surv.*, 53(5):99:1–99:38, 2021.

[44] R. Dabre, A. Fujita, and C. Chu. Exploiting multilingualism through multistage fine-tuning for low-resource neural machine translation. In *Proceed-*

ings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1410–1416, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[45] J. de la Rosa and A. Fernández. Zero-shot reading comprehension and reasoning for spanish with BERTIN GPT-J-6B. In M. Montes-y-Gómez, J. Gonzalo, F. Rangel, M. Casavantes, M. Á. Á. Carmona, G. Bel-Enguix, H. J. Escalante, L. A. de Freitas, A. Miranda-Escalada, F. J. Rodríguez-Sanchez, A. Rosá, M. A. S. Cabezudo, M. Taulé, and R. Valencia-García, editors, *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2022) co-located with the Conference of the Spanish Society for Natural Language Processing (SEPLN 2022), A Coruña, Spain, September 20, 2022*, volume 3202 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.

[46] J. Devlin, M.-W. Chang, K. Lee, and K. 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.

[47] C. Ding, H. T. Z. Aye, W. P. Pa, K. T. Nwet, K. M. Soe, M. Utiyama, and E. Sumita. Towards burmese (myanmar) morphological analysis: Syllable-based tokenization and part-of-speech tagging. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 19(1):5:1–5:34, 2020.

[48] D. Dong, H. Wu, W. He, D. Yu, and H. 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.

[49] C. Dyer, V. Chahuneau, and N. A. Smith. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia, June 2013. Association for Computational Linguistics.

[50] A. Ebrahimi and K. Kann. How to adapt your pretrained multilingual model to 1600 languages. 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 4555–4567, Online, Aug. 2021. Association for Computational Linguistics.

[51] S. Edunov, M. Ott, M. Auli, and D. Grangier. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.

[52] C. España-Bonet, Á. C. Varga, A. Barrón-Cedeño, and J. van Genabith. An empirical analysis of nmt-derived interlingual embeddings and their use in parallel sentence identification. *IEEE J. Sel. Top. Signal Process.*, 11(8):1340–1350, 2017.

[53] A. Fan, S. Bhosale, H. Schwenk, Z. Ma, A. El-Kishky, S. Goyal, M. Baines, O. Celebi, G. Wenzek, V. Chaudhary, N. Goyal, T. Birch, V. Liptchinsky, S. Edunov, M. Auli, and A. Joulin. Beyond english-centric multilingual machine translation. *J. Mach. Learn. Res.*, 22:107:1–107:48, 2021.

[54] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[55] O. Firat, K. Cho, and Y. 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.

[56] O. Firat, K. Cho, B. Sankaran, F. T. Yarman-Vural, and Y. Bengio. Multi-way, multilingual neural machine translation. *Comput. Speech Lang.*, 45:236–252, 2017.

[57] T. Gao, X. Yao, and D. Chen. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, 2021.

[58] S. Garg, S. Peitz, U. Nallasamy, and M. Paulik. Jointly learning to align and translate with transformer models. 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 4453–4462, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[59] M. Ghazvininejad, H. Gonen, and L. Zettlemoyer. Dictionary-based phrase-level prompting of large language models for machine translation. *CoRR*, abs/2302.07856, 2023.

[60] J. M. Giorgi, O. Nitski, B. Wang, and G. D. Bader. Declutr: Deep contrastive learning for unsupervised textual representations. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *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, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 879–895. Association for Computational Linguistics, 2021.

[61] K. Goswami, S. Dutta, H. Assem, T. Fransen, and J. P. McCrae. Cross-lingual sentence embedding using multi-task learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9099–9113, 2021.

[62] N. Goyal, C. Gao, V. Chaudhary, P. Chen, G. Wenzek, D. Ju, S. Krishnan, M. Ranzato, F. Guzmán, and A. Fan. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Trans. Assoc. Comput. Linguistics*, 10:522–538, 2022.

[63] F. Grégoire and P. Langlais. A deep neural network approach to parallel sentence extraction. *CoRR*, abs/1709.09783, 2017.

[64] J. Gu, Y. Wang, K. Cho, and V. O. Li. Improved zero-shot neural machine translation via ignoring spurious correlations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1258–1268, Florence, Italy, July 2019. Association for Computational Linguistics.

[65] B. Gunel, J. Du, A. Conneau, and V. Stoyanov. Supervised contrastive learning for pre-trained language model fine-tuning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[66] M. Guo, Q. Shen, Y. Yang, H. Ge, D. Cer, G. Hernandez Abrego, K. Stevens, N. Constant, Y.-H. Sung, B. Strope, and R. Kurzweil. Effective parallel corpus mining using bilingual sentence embeddings. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 165–176, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics.

[67] T.-L. Ha, J. Niehues, and A. Waibel. Toward multilingual neural machine translation with universal encoder and decoder. In *Proceedings of the 13th International Conference on Spoken Language Translation*, Seattle, Washington D.C, Dec. 8-9 2016. International Workshop on Spoken Language Translation.

[68] D. Han, K. Sudoh, X. Wu, K. Duh, H. Tsukada, and M. Nagata. Head finalization reordering for Chinese-to-Japanese machine translation. In *Proceedings of the Sixth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 57–66, Jeju, Republic of Korea, July 2012. Association for Computational Linguistics.

[69] Z. He, T. Liang, W. Jiao, Z. Zhang, Y. Yang, R. Wang, Z. Tu, S. Shi, and X. Wang. Exploring human-like translation strategy with large language models. *CoRR*, abs/2305.04118, 2023.

[70] K. Heffernan, O. Çelebi, and H. Schwenk. Bitext mining using distilled sentence representations for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2101–2112, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[71] G. E. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[72] V. C. D. Hoang, P. Koehn, G. Haffari, and T. Cohn. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[73] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997.

[74] S. Hoshino, Y. Miyao, K. Sudoh, and M. Nagata. Two-stage pre-ordering for Japanese-to-English statistical machine translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1062–1066, Nagoya, Japan, Oct. 2013. Asian Federation of Natural Language Processing.

[75] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[76] J. Hu, S. Ruder, A. Siddhant, G. Neubig, O. Firat, and M. Johnson. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*,

volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR, 2020.

[77] H. Isozaki, K. Sudoh, H. Tsukada, and K. Duh. Head finalization: A simple reordering rule for SOV languages. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 244–251, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[78] T. Jiang, S. Huang, Z. Luan, D. Wang, and F. Zhuang. Scaling sentence embeddings with large language models. *CoRR*, abs/2307.16645, 2023.

[79] W. Jiao, W. Wang, J. Huang, X. Wang, and Z. Tu. Is chatgpt A good translator? A preliminary study. *CoRR*, abs/2301.08745, 2023.

[80] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu. Tinybert: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, volume EMNLP 2020, pages 4163–4174, 2020.

[81] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, M. Hughes, and J. Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

[82] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020.

[83] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov. Fasttext.zip: Compressing text classification models. *CoRR*, abs/1612.03651, 2016.

[84] P. Keung, Y. Lu, G. Szarvas, and N. A. Smith. The multilingual amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 4563–4568, 2020.

[85] T. Kim, K. M. Yoo, and S. Lee. Self-guided contrastive learning for BERT sentence representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 2528–2540, 2021.

[86] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[87] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. Rush. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[88] P. Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[89] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers, MTSummit 2005, Phuket, Thailand, September 13-15, 2005*, pages 79–86, 2005.

[90] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[91] M. Komachi, Y. Matsumoto, and M. Nagata. Phrase reordering for statistical machine translation based on predicate-argument structure. In *2006 International Workshop on Spoken Language Translation, IWSLT 2006, Kei-*

hanna Science City, Kyoto, Japan, November 27-28, 2006, pages 77–82, 2006.

[92] X. Kong, A. Renduchintala, J. Cross, Y. Tang, J. Gu, and X. Li. Multilingual neural machine translation with deep encoder and multiple shallow decoders. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1613–1624, Online, Apr. 2021. Association for Computational Linguistics.

[93] T. Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 66–75, 2018.

[94] T. Kudo and J. Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics.

[95] S. Kurohashi, T. Nakamura, Y. Matsumoto, and M. Nagao. Improvements of Japanese morphological analyzer JUMAN. In *Proceedings of the International Workshop on Sharable Natural Language Resources*, pages 22–28, 1994.

[96] P. Lambert, S. Petitrenaud, Y. Ma, and A. Way. What types of word alignment improve statistical machine translation? *Mach. Transl.*, 26(4):289–323, 2012.

[97] G. Lample, M. Ott, A. Conneau, L. Denoyer, and M. 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, Oct.-Nov. 2018. Association for Computational Linguistics.

[98] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations*, 2020.

[99] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. RCV1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004.

[100] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. 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.

[101] J. Li, H. Zhou, S. Huang, S. Chen, and J. Chen. Eliciting the translation ability of large language models via multilingual finetuning with translation instructions. *CoRR*, abs/2305.15083, 2023.

[102] Z. Li, S. Huang, Z. Zhang, Z. Deng, Q. Lou, H. Huang, J. Jiao, F. Wei, W. Deng, and Q. Zhang. Dual-alignment pre-training for cross-lingual sentence embedding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 3466–3478, 2023.

[103] X. Liang, N. Zhang, S. Cheng, Z. Zhang, C. Tan, and H. Chen. Contrastive demonstration tuning for pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 799–811, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[104] X. V. Lin, T. Mihaylov, M. Artetxe, T. Wang, S. Chen, D. Simig, M. Ott, N. Goyal, S. Bhosale, J. Du, R. Pasunuru, S. Shleifer, P. S. Koura, V. Chaudhary, B. O'Horo, J. Wang, L. Zettlemoyer, Z. Kozareva, M. Diab, V. Stoyanov, and X. Li. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in*

Natural Language Processing, pages 9019–9052, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[105] Z. Lin, X. Pan, M. Wang, X. Qiu, J. Feng, H. Zhou, and L. 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, Nov. 2020. Association for Computational Linguistics.

[106] Z. Lin, L. Wu, M. Wang, and L. Li. Learning language specific sub-network for multilingual machine translation. 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 293–305, Online, Aug. 2021. Association for Computational Linguistics.

[107] P. Lison and J. Tiedemann. Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, 2016.

[108] D. Liu, J. Niehues, J. Cross, F. Guzmán, and X. Li. Improving zero-shot translation by disentangling positional information. 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 1259–1273, Online, Aug. 2021. Association for Computational Linguistics.

[109] H. Liu, D. Tam, M. Muqeeth, J. Mohta, T. Huang, M. Bansal, and C. Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *NeurIPS*, 2022.

[110] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742, 2020.

[111] L. Logeswaran and H. Lee. An efficient framework for learning sentence representations. In *6th International Conference on Learning Representations*, 2018.

[112] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[113] Y. Lu, P. Keung, F. Ladhak, V. Bhardwaj, S. Zhang, and J. Sun. A neural interlingua for multilingual machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 84–92, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics.

[114] Z. Mao, C. Chu, R. Dabre, H. Song, Z. Wan, and S. Kurohashi. When do contrastive word alignments improve many-to-many neural machine translation? In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1766–1775, Seattle, United States, July 2022. Association for Computational Linguistics.

[115] Z. Mao, C. Chu, and S. Kurohashi. EMS: efficient and effective massively multilingual sentence representation learning. *CoRR*, abs/2205.15744, 2022.

[116] Z. Mao, C. Chu, and S. Kurohashi. Linguistically driven multi-task pre-training for low-resource neural machine translation. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 21(4):68:1–68:29, 2022.

[117] Z. Mao, C. Chu, and S. Kurohashi. Efficiently learning multilingual sentence representation for cross-lingual sentence classification. In **言語処理学会 第*29*回年次大会**, pages 2830–2835, 2023.

[118] Z. Mao, F. Cromieres, R. Dabre, H. Song, and S. Kurohashi. JASS: Japanese-specific sequence to sequence pre-training for neural machine translation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3683–3691, Marseille, France, May 2020. European Language Resources Association.

[119] Z. Mao, R. Dabre, F. Cromieres, H. Song, R. Nakao, and S. Kurohashi. ニューラル機械翻訳のための言語知識に基づくマルチタスク事前学習. In **言語処理学会 第 *26* 回年次大会**, pages 1061–1064, 2020.

[120] Z. Mao, R. Dabre, Q. Liu, H. Song, C. Chu, and S. Kurohashi. Exploring the impact of layer normalization for zero-shot neural machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1300–1316, Toronto, Canada, July 2023. Association for Computational Linguistics.

[121] Z. Mao, P. Gupta, C. Chu, M. Jaggi, and S. Kurohashi. Learning cross-lingual sentence representations for multilingual document classification with token-level reconstruction. In **言語処理学会 第 *27* 回年次大会**, pages 1049–1053, 2021.

[122] Z. Mao, P. Gupta, C. Chu, M. Jaggi, and S. Kurohashi. Lightweight cross-lingual sentence representation learning. 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 2902–2913, Online, Aug. 2021. Association for Computational Linguistics.

[123] Z. Mao and T. Nakagawa. LEALLA: learning lightweight language-agnostic sentence embeddings with knowledge distillation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1878–1886, 2023.

[124] Z. Mao, Y. Shen, C. Chu, S. Kurohashi, and C. Jin. Meta ensemble for Japanese-Chinese neural machine translation: Kyoto-U+ECNU participation to WAT 2020. In *Proceedings of the 7th Workshop on Asian Translation*, pages 64–71, Suzhou, China, Dec. 2020. Association for Computational Linguistics.

[125] Z. Mao, H. Song, R. Dabre, C. Chu, and S. Kurohashi. Variable-length neural interlingua representations for zero-shot neural machine translation. *CoRR*, abs/2305.10190, 2023.

[126] Z. Mao and Y. Yu. Tuning llms with contrastive alignment instructions for machine translation in unseen, low-resource languages. *CoRR*, abs/2401.05811, 2024.

[127] P. Micikevicius, S. Narang, J. Alben, G. F. Diamos, E. Elsen, D. García, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu. Mixed precision training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[128] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, Workshop Track Proceedings*, 2013.

[129] S. Mishra, D. Khashabi, C. Baral, and H. Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[130] H. Morita, D. Kawahara, and S. Kurohashi. Morphological analysis for unsegmented languages using recurrent neural network language model. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2292–2297, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics.

[131] Y. Moslem, R. Haque, J. D. Kelleher, and A. Way. Adaptive machine translation with large language models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 227–237, Tampere, Finland, June 2023. European Association for Machine Translation.

[132] N. Muennighoff, T. Wang, L. Sutawika, A. Roberts, S. Biderman, T. Le Scao, M. S. Bari, S. Shen, Z. X. Yong, H. Schoelkopf, X. Tang, D. Radev, A. F. Aji, K. Almubarak, S. Albanie, Z. Alyafeai, A. Webson,

E. Raff, and C. Raffel. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada, July 2023. Association for Computational Linguistics.

[133] B. Muller, A. Anastasopoulos, B. Sagot, and D. Seddah. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online, June 2021. Association for Computational Linguistics.

[134] M. Müller and F. Laurent. Cedille: A large autoregressive french language model. *CoRR*, abs/2202.03371, 2022.

[135] R. Murthy, A. Kunchukuttan, and P. Bhattacharyya. Addressing word-order divergence in multilingual neural machine translation for extremely low resource languages. 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 3868–3873, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[136] M. Nagao. A framework of a mechanical translation between japanese and english by analogy principle. In *Proc. of the International NATO Symposium on Artificial and Human Intelligence*, page 173–180, USA, 1984. Elsevier North-Holland, Inc.

[137] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[138] T. Nakazawa, H. Mino, I. Goto, G. Neubig, S. Kurohashi, and E. Sumita. Overview of the 2nd workshop on Asian translation. In *Proceedings of the 2nd Workshop on Asian Translation (WAT2015)*, pages 1–28, Kyoto, Japan, Oct. 2015. Workshop on Asian Translation.

[139] T. Nakazawa, K. Sudoh, S. Higashiyama, C. Ding, R. Dabre, H. Mino, I. Goto, W. P. Pa, A. Kunchukuttan, and S. Kurohashi. Overview of the 5th workshop on Asian translation. In *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation: 5th Workshop on Asian Translation: 5th Workshop on Asian Translation*, Hong Kong, 1–3 Dec. 2018. Association for Computational Linguistics.

[140] T. Nakazawa, M. Yaguchi, K. Uchimoto, M. Utiyama, E. Sumita, S. Kurohashi, and H. Isahara. ASPEC: Asian scientific paper excerpt corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2204–2208, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).

[141] G. Neubig and J. 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, Oct.-Nov. 2018. Association for Computational Linguistics.

[142] T. Q. Nguyen and D. Chiang. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan, Nov. 2017. Asian Federation of Natural Language Processing.

[143] T. Q. Nguyen and J. Salazar. Transformers without tears: Improving the normalization of self-attention. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong, Nov. 2-3 2019. Association for Computational Linguistics.

[144] J. Ni, G. H. Ábrego, N. Constant, J. Ma, K. B. Hall, D. Cer, and Y. Yang. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874, 2022.

[145] OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023.

[146] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[147] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

[148] L. Pan, C.-W. Hang, H. Qi, A. Shah, S. Potdar, and M. Yu. Multilingual BERT post-pretraining alignment. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 210–219, Online, June 2021. Association for Computational Linguistics.

[149] X. Pan, M. Wang, L. Wu, and L. Li. Contrastive learning for many-to-many multilingual neural machine translation. 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 244–258, Online, Aug. 2021. Association for Computational Linguistics.

[150] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[151] J. Park, J.-P. Hong, and J.-W. Cha. Korean language resources for everyone. In *Proceedings of the 30th Pacific Asia Conference on Language, Informa-*

tion and Computation: Oral Papers, pages 49–58, Seoul, South Korea, Oct. 2016.

[152] K. Peng, L. Ding, Q. Zhong, L. Shen, X. Liu, M. Zhang, Y. Ouyang, and D. Tao. Towards making the most of ChatGPT for machine translation. In H. Bouamor, J. Pino, and K. Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5622–5633, Singapore, Dec. 2023. Association for Computational Linguistics.

[153] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. 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.

[154] J. Pfeiffer, I. Vulić, I. Gurevych, and S. Ruder. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online, Nov. 2020. Association for Computational Linguistics.

[155] N.-Q. Pham, J. Niehues, T.-L. Ha, and A. Waibel. Improving zero-shot translation with language-independent constraints. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 13–23, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

[156] J. Philip, A. Berard, M. Gallé, and L. Besacier. Monolingual adapters for zero-shot neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4465–4470, Online, Nov. 2020. Association for Computational Linguistics.

[157] S. S. Pierre Zweigenbaum and R. Rapp. Overview of the third bucc shared task: Spotting parallel sentences in comparable corpora. In R. Rapp,

P. Zweigenbaum, and S. Sharoff, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France, may 2018. European Language Resources Association (ELRA).

[158] C. Pollard and I. A. Sag. *Information-Based Syntax and Semantics: Vol. 1: Fundamentals*. Center for the Study of Language and Information, USA, 1988.

[159] C. Pollard and I. A. Sag. *Head-Driven Phrase Structure Grammar*. The University of Chicago Press, Chicago, 1994.

[160] M. 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, Sept. 2015. Association for Computational Linguistics.

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

[162] P. Prettenhofer and B. Stein. Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1118–1127, 2010.

[163] I. Provilkov, D. Emelianenko, and E. Voita. Bpe-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, 2020.

[164] W. Qi, Y. Yan, Y. Gong, D. Liu, N. Duan, J. Chen, R. Zhang, and M. Zhou. ProphetNet: Predicting future n-gram for sequence-to-SequencePre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2401–2410, Online, Nov. 2020. Association for Computational Linguistics.

[165] Y. Qi, D. Sachan, M. Felix, S. Padmanabhan, and G. Neubig. When and why are pre-trained word embeddings useful for neural machine translation?

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 529–535, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[166] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.

[167] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020.

[168] M. Raghu, J. Gilmer, J. Yosinski, and J. Sohl-Dickstein. SVCCA: singular vector canonical correlation analysis for deep learning dynamics and interpretability. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6076–6085, 2017.

[169] P. Ramachandran, B. Zoph, and Q. V. Le. Searching for activation functions. In *6th International Conference on Learning Representations, Workshop Track Proceedings*, 2018.

[170] J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In R. Gupta, Y. Liu, J. Tang, and B. A. Prakash, editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3505–3506. ACM, 2020.

[171] R. Rei, C. Stewart, A. C. Farinha, and A. Lavie. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online, Nov. 2020. Association for Computational Linguistics.

[172] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3980–3990, 2019.

[173] N. Reimers and I. Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online, Nov. 2020. Association for Computational Linguistics.

[174] S. Ren, Y. Wu, S. Liu, M. Zhou, and S. Ma. Explicit cross-lingual pre-training for unsupervised machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 770–779, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[175] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[176] D. E. Rumelhart and J. L. McClelland. *Learning Internal Representations by Error Propagation*, pages 318–362. 1987.

[177] A. Sabet, P. Gupta, J. Cordonnier, R. West, and M. Jaggi. Robust cross-lingual embeddings from parallel sentences. *CoRR*, abs/1912.12481, 2019.

[178] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[179] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, A. Raja, M. Dey, M. S. Bari, C. Xu, U. Thakker, S. S. Sharma, E. Szczechla, T. Kim, G. Chhablani, N. V. Nayak, D. Datta, J. Chang, M. T. Jiang, H. Wang, M. Manica, S. Shen, Z. X. Yong, H. Pandey,

R. Bawden, T. Wang, T. Neeraj, J. Rozen, A. Sharma, A. Santilli, T. Févry, J. A. Fries, R. Teehan, T. L. Scao, S. Biderman, L. Gao, T. Wolf, and A. M. Rush. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[180] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilic, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, J. Tow, A. M. Rush, S. Biderman, A. Webson, P. S. Ammanamanchi, T. Wang, B. Sagot, N. Muennighoff, A. V. del Moral, O. Ruwase, R. Bawden, S. Bekman, A. McMillan-Major, I. Beltagy, H. Nguyen, L. Saulnier, S. Tan, P. O. Suarez, V. Sanh, H. Laurençon, Y. Jernite, J. Launay, M. Mitchell, C. Raffel, A. Gokaslan, A. Simhi, A. Soroa, A. F. Aji, A. Alfassy, A. Rogers, A. K. Nitzav, C. Xu, C. Mou, C. Emezue, C. Klamm, C. Leong, D. van Strien, D. I. Adelani, and et al. BLOOM: A 176b-parameter open-access multilingual language model. *CoRR*, abs/2211.05100, 2022.

[181] H. Schwenk, V. Chaudhary, S. Sun, H. Gong, and F. Guzmán. WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online, Apr. 2021. Association for Computational Linguistics.

[182] H. Schwenk and M. Douze. Learning joint multilingual sentence representations with neural machine translation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 157–167, 2017.

[183] H. Schwenk and X. Li. A corpus for multilingual document classification in eight languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, 2018.

[184] H. Schwenk, G. Wenzek, S. Edunov, E. Grave, A. Joulin, and A. Fan. CCMatrix: Mining billions of high-quality parallel sentences on the web. 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 6490–6500, Online, Aug. 2021. Association for Computational Linguistics.

[185] T. Sellam, D. Das, and A. Parikh. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online, July 2020. Association for Computational Linguistics.

[186] S. Sen, K. K. Gupta, A. Ekbal, and P. Bhattacharyya. Multilingual unsupervised NMT using shared encoder and language-specific decoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3083–3089, Florence, Italy, July 2019. Association for Computational Linguistics.

[187] R. Sennrich and B. Haddow. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 83–91, Berlin, Germany, Aug. 2016. Association for Computational Linguistics.

[188] R. Sennrich, B. Haddow, and A. Birch. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 371–376, Berlin, Germany, Aug. 2016. Association for Computational Linguistics.

[189] R. Sennrich, B. Haddow, and A. 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, Aug. 2016. Association for Computational Linguistics.

[190] R. Sennrich, B. Haddow, and A. 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, Aug. 2016. Association for Computational Linguistics.

[191] R. Sennrich and B. Zhang. Revisiting low-resource neural machine translation: A case study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221, Florence, Italy, July 2019. Association for Computational Linguistics.

[192] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. V. Le, G. E. Hinton, and J. Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[193] Y. Shen, Q. Liu, Z. Mao, F. Cheng, and S. Kurohashi. Textual enhanced contrastive learning for solving math word problems. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4297–4307, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[194] A. Siddhant, A. Bapna, Y. Cao, O. Firat, M. Chen, S. Kudugunta, N. Arivazhagan, and Y. Wu. Leveraging monolingual data with self-supervision for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2827–2835, Online, July 2020. Association for Computational Linguistics.

[195] H. Song, R. Dabre, C. Chu, A. Fujita, and S. Kurohashi. Bilingual corpus mining and multistage fine-tuning for improving machine translation of lecture transcripts. *CoRR*, abs/2311.03696, 2023.

[196] H. Song, R. Dabre, C. Chu, and S. Kurohashi. Large pre-trained language models with multilingual prompt for japanese natural language tasks. In **言語処理学会 第 *29* 回年次大会**, pages 810–814, 2023.

[197] H. Song, R. Dabre, C. Chu, S. Kurohashi, and E. Sumita. Self-supervised dynamic programming encoding for neural machine translation. In **言語処理学会 第 *27* 回年次大会**, pages 632–636, 2021.

[198] H. Song, R. Dabre, C. Chu, S. Kurohashi, and E. Sumita. Selfseg: A self-supervised sub-word segmentation method for neural machine translation. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 22(8):215:1–215:24, 2023.

[199] H. Song, R. Dabre, A. Fujita, and S. Kurohashi. Coursera corpus mining and multistage fine-tuning for improving lectures translation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3640–3649, Marseille, France, May 2020. European Language Resources Association.

[200] H. Song, R. Dabre, A. Fujita, and S. Kurohashi. Domain adaptation of neural machine translation through multistage fine-tuning. In **言語処理学会第 *26* 回年次大会**, pages 461–464, 2020.

[201] H. Song, R. Dabre, Z. Mao, F. Cheng, S. Kurohashi, and E. Sumita. Pre-training via leveraging assisting languages for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 279–285, Online, July 2020. Association for Computational Linguistics.

[202] H. Song, R. Dabre, Z. Mao, C. Chu, and S. Kurohashi. BERTSeg: BERT based unsupervised subword segmentation for neural machine translation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 85–94, Online only, Nov. 2022. Association for Computational Linguistics.

[203] H. Song, R. Dabre, Z. Mao, C. Chu, and S. Kurohashi. Representative data selection for sequence-to-sequence pre-training. In **言語処理学会 第 *28* 回年次大会**, pages 1–5, 2022.

[204] K. Song, X. Tan, T. Qin, J. Lu, and T. Liu. MASS: masked sequence to sequence pre-training for language generation. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference*

on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936. PMLR, 2019.

[205] H. Su, W. Shi, J. Kasai, Y. Wang, Y. Hu, M. Ostendorf, W. Yih, N. A. Smith, L. Zettlemoyer, and T. Yu. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, 2023.

[206] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang. ERNIE 2.0: A continual pre-training framework for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8968–8975, 2020.

[207] Z. Sun, H. Yu, X. Song, R. Liu, Y. Yang, and D. Zhou. MobileBERT: a compact task-agnostic BERT for resource-limited devices. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2158–2170, Online, July 2020. Association for Computational Linguistics.

[208] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112, 2014.

[209] W. Tan, K. Heffernan, H. Schwenk, and P. Koehn. Multilingual representation distillation with contrastive learning. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1469–1482, 2023.

[210] X. Tan, J. Chen, D. He, Y. Xia, T. Qin, and T.-Y. Liu. Multilingual neural machine translation with language clustering. 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 963–973, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[211] R. Tang, Y. Lu, L. Liu, L. Mou, O. Vechtomova, and J. Lin. Distilling task-specific knowledge from BERT into simple neural networks. *CoRR*, abs/1903.12136, 2019.

[212] Y. Tang, C. Tran, X. Li, P. Chen, N. Goyal, V. Chaudhary, J. Gu, and A. Fan. Multilingual translation with extensible multilingual pretraining and finetuning. *CoRR*, abs/2008.00401, 2020.

[213] Y. Tang, C. Tran, X. Li, P.-J. Chen, N. Goyal, V. Chaudhary, J. Gu, and A. Fan. Multilingual translation from denoising pre-training. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3450–3466, Online, Aug. 2021. Association for Computational Linguistics.

[214] J. Tiedemann. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 2214–2218, 2012.

[215] A. Tolmachev, D. Kawahara, and S. Kurohashi. Juman++: A morphological analysis toolkit for scriptio continua. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 54–59, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics.

[216] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971, 2023.

[217] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton-Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288, 2023.

[218] A. Vaswani, S. Bengio, E. Brevdo, F. Chollet, A. N. Gomez, S. Gouws, L. Jones, L. Kaiser, N. Kalchbrenner, N. Parmar, R. Sepassi, N. Shazeer, and J. Uszkoreit. Tensor2tensor for neural machine translation. In C. Cherry and G. Neubig, editors, *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas, AMTA 2018, Boston, MA, USA, March 17-21, 2018 - Volume 1: Research Papers*, pages 193–199. Association for Machine Translation in the Americas, 2018.

[219] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[220] R. Vázquez, A. Raganato, J. Tiedemann, and M. Creutz. Multilingual NMT with a language-independent attention bridge. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 33–39, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

[221] A. Veit, M. J. Wilber, and S. J. Belongie. Residual networks behave like ensembles of relatively shallow networks. In D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 550–558, 2016.

[222] D. Vilar, M. Freitag, C. Cherry, J. Luo, V. Ratnakar, and G. Foster. Prompting PaLM for translation: Assessing strategies and performance. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15406–15427, Toronto, Canada, July 2023. Association for Computational Linguistics.

[223] Z. Wan, F. Cheng, Q. Liu, Z. Mao, H. Song, and S. Kurohashi. Relation extraction with weighted contrastive pre-training on distant supervision. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2580–2585, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[224] Z. Wan, F. Cheng, Z. Mao, Q. Liu, H. Song, and S. Kurohashi. Improving medical relation extraction with distantly supervised pre-training. In **言語処理学会 第** *28* **回年次大会**, pages 610–614, 2022.

[225] Z. Wan, F. Cheng, Z. Mao, Q. Liu, H. Song, J. Li, and S. Kurohashi. GPT-RE: In-context learning for relation extraction using large language models. In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, Singapore, Dec. 2023. Association for Computational Linguistics.

[226] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, and F. Wei. Text embeddings by weakly-supervised contrastive pre-training. *CoRR*, abs/2212.03533, 2022.

[227] L. Wang, W. Zhao, R. Jia, S. Li, and J. Liu. Denoising based sequence-to-sequence pre-training for text generation. 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 4003–4015, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[228] W. Wang, Z. Zhang, Y. Du, B. Chen, J. Xie, and W. Luo. Rethinking zero-shot neural machine translation: From a perspective of latent variables. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4321–4327, Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.

[229] X. Wang, H. Pham, P. Arthur, and G. Neubig. Multilingual neural machine translation with soft decoupled encoding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[230] X. Wang, Y. Tsvetkov, and G. Neubig. Balancing training for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8526–8537, Online, July 2020. Association for Computational Linguistics.

[231] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Naik, A. Ashok, A. S. Dhanasekaran, A. Arunkumar, D. Stap, E. Pathak, G. Karamanolakis, H. Lai, I. Purohit, I. Mondal, J. Anderson, K. Kuznia, K. Doshi, K. K. Pal, M. Patel, M. Moradshahi, M. Parmar, M. Purohit, N. Varshney, P. R. Kaza, P. Verma, R. S. Puri, R. Karia, S. Doshi, S. K. Sampat, S. Mishra, S. Reddy A, S. Patro, T. Dixit, and X. Shen. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[232] Y. Wang, A. Wu, and G. Neubig. English contrastive learning can learn universal cross-lingual sentence embeddings. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9122–9133, 2022.

[233] Y. Wang, C. Zhai, and H. Hassan. Multi-task learning for multilingual neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1022–1034, Online, Nov. 2020. Association for Computational Linguistics.

[234] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[235] X. Wei, R. Weng, Y. Hu, L. Xing, H. Yu, and W. Luo. On learning universal representations across languages. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[236] L. Wu, S. Cheng, M. Wang, and L. Li. Language tags matter for zero-shot neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3001–3007, Online, Aug. 2021. Association for Computational Linguistics.

[237] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T. Liu. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 10524–10533. PMLR, 2020.

[238] C. Xu, T. Ge, C. Li, and F. Wei. Unihanlm: Coarse-to-fine chinese-japanese language model pretraining with the unihan database. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 201–211, 2020.

[239] H. Xu, Y. J. Kim, A. Sharaf, and H. H. Awadalla. A paradigm shift in machine translation: Boosting translation performance of large language models. *CoRR*, abs/2309.11674, 2023.

[240] J. Xu, X. Sun, Z. Zhang, G. Zhao, and J. Lin. Understanding and improving layer normalization. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 4383–4393, 2019.

[241] Y. Yan, R. Li, S. Wang, F. Zhang, W. Wu, and W. Xu. Consert: A contrastive framework for self-supervised sentence representation transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 5065–5075, 2021.

[242] J. Yang, Y. Yin, S. Ma, H. Huang, D. Zhang, Z. Li, and F. Wei. Multilingual agreement for multilingual neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 233–239, Online, Aug. 2021. Association for Computational Linguistics.

[243] W. Yang, C. Li, J. Zhang, and C. Zong. Bigtrans: Augmenting large language models with multilingual translation capability over 100 languages. *CoRR*, abs/2305.18098, 2023.

[244] Y. Yang, G. H. Ábrego, S. Yuan, M. Guo, Q. Shen, D. Cer, Y. Sung, B. Strope, and R. Kurzweil. Improving multilingual sentence embedding using bi-directional dual encoder with additive margin softmax. In S. Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5370–5378. ijcai.org, 2019.

[245] Y. Yang, D. Cer, A. Ahmad, M. Guo, J. Law, N. Constant, G. Hernandez Abrego, S. Yuan, C. Tar, Y.-h. Sung, B. Strope, and R. Kurzweil. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings*

of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 87–94, Online, July 2020. Association for Computational Linguistics.

[246] Y. Yang, A. Eriguchi, A. Muzio, P. Tadepalli, S. Lee, and H. Hassan. Improving multilingual translation by representation and gradient regularization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7266–7279, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.

[247] Y. Yang, S. Yuan, D. Cer, S. Kong, N. Constant, P. Pilar, H. Ge, Y. Sung, B. Strope, and R. Kurzweil. Learning semantic textual similarity from conversations. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 164–174, 2018.

[248] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764, 2019.

[249] Z. Yang, B. Hu, A. Han, S. Huang, and Q. Ju. CSP:code-switching pre-training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2624–2636, Online, Nov. 2020. Association for Computational Linguistics.

[250] Z. Yang, Y. Yang, D. Cer, J. Law, and E. Darve. Universal sentence representation learning with conditional masked language model. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6216–6228, 2021.

[251] E. Yankovskaya, A. Tättar, and M. Fishel. Quality estimation and translation metrics via pre-trained word and sentence embeddings. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task*

Papers, Day 2), pages 101–105, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

[252] J. Yim, D. Joo, J. Bae, and J. Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 7130–7138. IEEE Computer Society, 2017.

[253] W. Yin, H. Schütze, B. Xiang, and B. Zhou. ABCNN: attention-based convolutional neural network for modeling sentence pairs. *Trans. Assoc. Comput. Linguistics*, 4:259–272, 2016.

[254] Z. X. Yong, H. Schoelkopf, N. Muennighoff, A. F. Aji, D. I. Adelani, K. Almubarak, M. S. Bari, L. Sutawika, J. Kasai, A. Baruwa, G. Winata, S. Biderman, E. Raff, D. Radev, and V. Nikoulina. BLOOM+1: Adding language support to BLOOM for zero-shot prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11682–11703, Toronto, Canada, July 2023. Association for Computational Linguistics.

[255] K. Yu, H. Li, and B. Oguz. Multilingual seq2seq training with similarity loss for cross-lingual document classification. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 175–179, 2018.

[256] Z. Z. Yu, L. J. Jaw, W. Q. Jiang, and Z. Hui. Fine-tuning language models with generative adversarial feedback. *CoRR*, abs/2305.06176, 2023.

[257] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia, W. L. Tam, Z. Ma, Y. Xue, J. Zhai, W. Chen, Z. Liu, P. Zhang, Y. Dong, and J. Tang. GLM-130B: an open bilingual pre-trained model. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[258] B. Zhang, A. Bapna, R. Sennrich, and O. Firat. Share or not? learning to schedule language-specific capacity for multilingual translation. In *9th*

International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021.

[259] B. Zhang, B. Haddow, and A. Birch. Prompting large language model for machine translation: A case study. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 41092–41110. PMLR, 2023.

[260] B. Zhang and R. Sennrich. Root mean square layer normalization. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 12360–12371, 2019.

[261] B. Zhang, P. Williams, I. Titov, and R. Sennrich. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online, July 2020. Association for Computational Linguistics.

[262] J. Zhang and C. 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, Nov. 2016. Association for Computational Linguistics.

[263] S. Zhang, Q. Fang, Z. Zhang, Z. Ma, Y. Zhou, L. Huang, M. Bu, S. Gui, Y. Chen, X. Chen, and Y. Feng. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models. *CoRR*, abs/2306.10968, 2023.

[264] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. T. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster,

D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer. OPT: open pre-trained transformer language models. *CoRR*, abs/2205.01068, 2022.

[265] Y. Zhang, R. He, Z. Liu, L. Bing, and H. Li. Bootstrapped unsupervised sentence representation learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 5168–5180, 2021.

[266] Y. Zhang, R. He, Z. Liu, K. H. Lim, and L. Bing. An unsupervised sentence embedding method by mutual information maximization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 1601–1610, 2020.

[267] Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, and X. Zhou. Semantics-aware BERT for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9628–9635, 2020.

[268] C. Zhou, X. Ma, J. Hu, and G. Neubig. Handling syntactic divergence in low-resource machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1388–1394, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[269] J. Zhou, Z. Zhang, H. Zhao, and S. Zhang. LIMIT-BERT : Linguistics informed multi-task BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4450–4461, Online, Nov. 2020. Association for Computational Linguistics.

[270] C. Zhu, H. Yu, S. Cheng, and W. Luo. Language-aware interlingua for multilingual neural machine translation. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics, pages 1650–1655, Online, July 2020. Association for Computational Linguistics.

[271] M. Ziemski, M. Junczys-Dowmunt, and B. Pouliquen. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).

[272] B. Zoph and K. Knight. Multi-source neural translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 30–34, San Diego, California, June 2016. Association for Computational Linguistics.

[273] B. Zoph, D. Yuret, J. May, and K. Knight. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[274] P. Zweigenbaum, S. Sharoff, and R. 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.

[275] P. Zweigenbaum, S. Sharoff, and R. 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.

List of Major Publications

[1] Zhuoyuan Mao, Fabien Cromieres, Raj Dabre, Haiyue Song and Sadao Kurohashi. JASS: Japanese-specific Sequence to Sequence Pre-training for Neural Machine Translation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3683–3691, 2020.

[2] Zhuoyuan Mao, Prakhar Gupta, Chenhui Chu, Martin Jaggi and Sadao Kurohashi. Lightweight Cross-Lingual Sentence Representation Learning. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 2902–2913, 2021.

[3] Zhuoyuan Mao, Chenhui Chu and Sadao Kurohashi. EMS: efficient and effective massively multilingual sentence representation learning. Under review at *IEEE ACM Trans. Audio Speech Lang. Process.*, 2022.

[4] Zhuoyuan Mao, Chenhui Chu, Raj Dabre, Haiyue Song, Zhen Wan and Sadao Kurohashi. When do Contrastive Word Alignments Improve Many-to-many Neural Machine Translation? In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1766–1775, 2022.

[5] Zhuoyuan Mao, Chenhui Chu and Sadao Kurohashi. Linguistically-driven Multi-task Pre-training for Low-resource Neural Machine Translation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 21, 4, Article 68 (Jul. 2022), 29 pages. https://doi.org/10.1145/3491065

[6] Zhuoyuan Mao and Tetsuji Nakagawa. LEALLA: Learning Lightweight Language-agnostic Sentence Embedding with Knowledge Distillation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1886–1894, 2023.

[7] Zhuoyuan Mao, Haiyue Song, Raj Dabre, Chenhui Chu and Sadao Kurohashi. Variable-length Neural Interlingua Representations for Zero-shot Neural Machine Translation. In *the 1st Workshop on Multilingual, Multimodal and Multitask Language Generation*, pages 16–25, 2023.

[8] Zhuoyuan Mao, Raj Dabre, Qianying Liu, Haiyue Song, Chenhui Chu and Sadao Kurohashi. Exploring the Impact of Layer Normalization for Zero-shot Neural Machine Translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 1300–1316, 2023.

[9] Zhuoyuan Mao and Yen Yu. Tuning LLMs with Contrastive Alignment Instructions for Machine Translation in Unseen, Low-resource Languages. Under review at *ACL Rolling Review*, 2023.

List of Other Publications

[1] Zhuoyuan Mao, Raj Dabre, Fabien Cromieres, Haiyue Song, Ryota Nakao and Sadao Kurohashi. ニューラル機械翻訳のための言語知識に基づくマルチタスク事前学習. In *Proceedings of the 26th Annual Meeting of the Association for Natural Language Processing*, pages 1061–1064, 2020. (in Japanese).

[2] Zhuoyuan Mao, Yibin Shen, Chenhui Chu, Sadao Kurohashi and Cheqing Jin. Meta Ensemble for Japanese-Chinese Neural Machine Translation: Kyoto-U+ECNU Participation to WAT 2020. In *Proceedings of the 7th Workshop on Asian Translation*, pages 64–71, 2020.

[3] Haiyue Song, Raj Dabre, Zhuoyuan Mao, Fei Cheng, Sadao Kurohashi and Eiichiro Sumita. Pre-training via Leveraging Assisting Languages and Data Selection for Neural Machine Translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 279–285, 2020.

[4] Zhuoyuan Mao, Prakhar Gupta, Chenhui Chu, Martin Jaggi and Sadao Kurohashi. Learning Cross-lingual Sentence Representations for Multilingual Document Classification with Token-level Reconstruction. In *Proceedings of the 27th Annual Meeting of the Association for Natural Language Processing*, pages 1049-1053, 2021.

[5] Haiyue Song, Raj Dabre, Zhuoyuan Mao, Chenhui Chu and Sadao Kurohashi. Representative Data Selection for Sequence-to-Sequence Pre-training.

In *Proceedings of the 28th Annual Meeting of the Association for Natural Language Processing*, pages 1–5, 2022.

[6] Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song and Sadao Kurohashi. Improving Medical Relation Extraction with Distantly Supervised Pre-training. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 610–614, 2022.

[7] Chenhui Chu, Zhuoyuan Mao, Toshiaki Nakazawa, Daisuke Kawahara and Sadao Kurohashi. SCTB-V2: the 2nd Version of the Chinese Treebank in the Scientific Domain. *Language Resources and Evaluation*, pages 1–15, Oct. 2022.

[8] Yibin Shen, Qianying Liu, Zhuoyuan Mao, Zhen Wan, Fei Cheng and Sadao Kurohashi. Seeking Diverse Reasoning Logic: Controlled Equation Expression Generation for Solving Math Word Problems. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 254–260, 2022.

[9] Haiyue Song, Raj Dabre, Zhuoyuan Mao, Chenhui Chu and Sadao Kurohashi. BERTSeg: BERT Based Unsupervised Subword Segmentation for Neural Machine Translation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 85–94, 2022.

[10] Zhen Wan, Qianying Liu, Zhuoyuan Mao, Fei Cheng, Sadao Kurohashi and Jiwei Li. Rescue Implicit and Long-tail Cases: Nearest Neighbor Relation Extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1731–1738, 2022.

[11] Yibin Shen, Qianying Liu, Zhuoyuan Mao, Fei Cheng and Sadao Kurohashi. Textual Enhanced Contrastive Learning for Solving Math Word Problems. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4297–4307, 2022.

[12] Zhuoyuan Mao, Chenhui Chu and Sadao Kurohashi. Efficiently Learning Multilingual Sentence Representation for Cross-lingual Sentence Classification. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 2830–2835, 2023.

[13] Zhen Wan, Fei Cheng, Qianying Liu, Zhuoyuan Mao, Haiyue Song and Sadao Kurohashi. Relation Extraction with Weighted Contrastive Pre-training on Distant Supervision. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2580–2585, 2023.

[14] Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li and Sadao Kurohashi. GPT-RE: In-context Learning for Relation Extraction using Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, 2023.

[15] Haiyue Song, Zhuoyuan Mao, Raj Dabre, Chenhui Chu and Sadao Kurohashi. DiverSeg: Leveraging Diverse Segmentations with Cross-granularity Alignment for Neural Machine Translation. *Journal of Natural Language Processing*, to appear in 31(1), 2024.