The VolcTrans System for WMT22 Multilingual Machine Translation Task

Xian Qian[1], **Kai Hu**[1], **Jiaqiang Wang**[1], **Yifeng Liu**[2]
Xingyuan Pan[3], **Jun Cao**[1], **Mingxuan Wang**[1]
[1] ByteDance AI Lab, [2] Tsinghua University, [3] Wuhan University
{qian.xian, hukai.joseph,wangjiaqiang.sonian,
caojun.sh, wangmingxuan.89}@bytedance.com
liuyifen20@mails.tsinghua.edu.cn, panxingyuan209@gmail.com

Abstract

This report describes our VolcTrans system for the WMT22 shared task on large-scale multilingual machine translation. We participated in the unconstrained track which allows the use of external resources. Our system is a transformer-based multilingual model trained on data from multiple sources including the public training set from the data track, NLLB data provided by Meta AI, self-collected parallel corpora, and pseudo bitext from back-translation. A series of heuristic rules clean both bilingual and monolingual texts. On the official test set, our system achieves 17.3 BLEU, 21.9 spBLEU, and 41.9 chrF2++ on average over all language pairs. The average inference speed is 11.5 sentences per second using a single Nvidia Tesla V100 GPU. Our code and trained models are available at https://github.com/xian8/wmt22

1 Introduction

Multilingual Machine Translation attracts much attention in recent years due to its advantages in sharing cross-lingual knowledge for low-resource languages. It also dramatically reduces training and serving costs. Training a multilingual model is much faster and simpler than training many bilingual ones. Serving multiple low-traffic languages using one model could drastically improve GPU utilization.

The WMT22 shared task on large-scale multilingual machine translation includes 24 African languages (Adelani et al., 2022b). Inspired by previous research works, we train a deep transformer model to translate all languages since large models have been demonstrated effective for multilingual translation (Fan et al., 2021; Kong et al., 2021; Zhang et al., 2020). We participated in the unconstrained track that allows the use of external data. Besides the official dataset for the constrained track, and the NLLB corpus provided by MetaAI (NLLB Team et al., 2022), we also collect parallel and monolingual texts from public websites and sources. These raw data are cleaned by a series of commonly used heuristic rules, and a minimum description length (MDL) based approach to remove samples with repeat patterns. Monolingual texts are used for back translation. For some very low-resource languages such as Wolof, iterative back-translation is adopted for higher accuracy.

We compare different training strategies to balance efficiency and quality, such as streaming data shuffling, and dynamic vocabulary for new languages. Furthermore, we used the open-sourced LightSeq toolkit [1] to accelerate training and inference.

On the official test set, our system achieves 17.3 BLEU, 21.9 spBLEU, and 41.9 chrF2++ on average over all language pairs. Averaged inference speed is 11.5 sentences per second using a single Nvidia Tesla V100 GPU.

2 Data

2.1 Data Collection

Our training data are mainly from four sources: the official set for constrained track, NLLB data provided by Meta AI, self-collected corpora, and pseudo training set from back translation.

For each source, we collect both parallel sentence pairs and monolingual sentences. A parallel sentence pair is collected if one side is in African language and the other is in English or French. We did not collect African-African sentence pairs as we use English as the pivot language for African-to-African translation. Instead, they are added to the monolingual set. More specifically, we split every sentence pair into two sentences and add them to the monolingual set accordingly. For example, the source side of a fuv-fon sentence pair is added to the fuv set. This greatly enriches the monolingual dataset, especially for the very low-resource

[1] https://github.com/bytedance/lightseq

Proceedings of the Seventh Conference on Machine Translation (WMT), pages 1068 - 1075

languages.

We merge multiple corpora from the same source into one and use bloom filter [2](Bloom, 1970) for fast deduplication. To reduce false positive errors which over delete distinct samples, we set the error rate $1e-7$ and capacity of $4B$ samples which costs $100G$ host memory.

The official set includes the data from data track participants, OPUS collections, and the NLLB parallel corpora mined from Common Crawl (com) and other sources. All domains in OPUS collections are involved, such as Mozilla-I10n, which could introduce many noises such as programming languages, and needs extra rules to clean.

NLLB data provided by Meta AI has three subsets: primary bitext including a seed set that is carefully annotated for representative languages and a public bitext set downloaded from open sources and mined bitexts that are automatically discovered by LASER3 encoder in a global mining pipeline, back-translated data from a pretrained model. We add the first two subsets in our training set.

Some public bitext data that are no longer available or require authorization such as JW300 (Agić and Vulić, 2019), Lorelei[3] and Chichewa News [4] are not included. We noticed that the NLLB team released another version of mined data recently in hugging-face [5], which is different from the version on the WMT22 website. We merge the new version into the old one and remove duplicates.

We collected additional bitexts in two ways: large-scale mining from general web pages, and manually crawling from specific websites and sources.

Large-scale mining focused on two scenarios, parallel sentences appearing on a single web page such as dictionary web pages that use multiple bilingual sentences to exemplify the usage of a word, and parallel web pages that describe the same content but are written in different languages. We extract these pages from the Common Crawl corpus. Then we utilized Vecalign (Thompson and Koehn, 2019), an accurate and efficient sentence alignment algorithm to mine parallel bilingual sentences. We use LASER (Schwenk and Douze, 2017) encoders released by WMT to obtain multilingual sentence embeddings and facilitate the alignment work. We collected about 3 million sentence pairs namely LAVA corpus and submitted them to the data track. And another $150M$ pairs for the unconstrained track.

Specific websites and sources have fewer but higher-quality sentence pairs. For example, the bible website[6] labels the order of sentences across languages so we can align them easily without sentence segmentation. Since JW300 is not publicly available, we crawled pages from Jehovah's Witnesses[7] to recover the dataset.

Monolingual texts have richer sources such as VOA news in Amharic [8] and OSCAR (Abadji et al., 2022), which improve English/French → African translation using back-translation. Monolingual texts from parallel data are also collected as described above. For African → English/French translation, we clean Wikipedia pages in English/French to get monolingual texts. For languages that gain significantly from back-translation such as Wolof, we run another round of back-translation to generate high-quality pseudo data.

2.2 Data Cleaning

We used the following rules to clean parallel datasets, except the NLLB mined bitext.

- Filter out parentheses and texts in between if the numbers of parentheses in two sentences are different.
- Filter out sentence pairs if numbers mismatch or one sentence ends with punctuation *: ! ?* ... and the other mismatches.
- Filter out sentences shorter than 30 characters, sentences having URLs or emails, or words longer than 100 characters.
- De-duplication: remove sentence pairs sharing the same source or target but having different translations.
- Sentences having programming languages are removed. We manually create a set of keywords to detect programming languages, such as *if (* , == and *.getAttribute* .
- Language identification using the NLLB language identification model trained by fastText (Joulin et al., 2017)

[2]https://pypi.org/project/bloom-filter
[3]https://catalog.ldc.upenn.edu/LDC2021T02
[4]https://zenodo.org/record/4315018#.YypJWezML0p
[5]https://huggingface.co/datasets/allenai/nllb
[6]https://www.bible.com/languages
[7]https://www.jw.org
[8]https://amharic.voanews.com/

One type of noisy text could survive the rules above, which has repeat patterns and commonly exists in many datasets. Here are some examples,

> *Download Bongeziwe Mabandla mini esadibana ngayo (#001) Mp3 Bongeziwe Mabandla - mini esadibana ngayo (#001).*
>
> *Coaster Gift,Paper-Cut Coaster Zodiac,Red Coaster Cute,Paper-Cut Zodiac Coaster*
>
> *mm mm mm MPEE(um) MPEP(um) mm mm mm mm mm mm kg kg*

A natural choice to detect these repeating patterns is the minimum description length (MDL) which finds the optimal compression by encoding frequent substrings with shorter codes.

Specifically, given a sentence $\mathbf{s}$, our MDL objective minimizes the length of the codebook plus the bits to encode the sentence:

$$\text{MDL}(\mathbf{s}) = \min_{\mathbf{s}=w_1 w_2 \ldots w_n} \left(C \sum_{\text{distinct } w} |w| - \sum_i \log\left(p(w_i|w_{i-1})\right) \right)$$

where $w_1, w_2, \ldots w_n$ is the word (coding entry) sequence, C is a positive constant, which balances the contribution of the codebook and length of the encoded sequence. $|w|$ is the length of word w. In our experiments, we set $C = 2$. $p(w_i|w_{i-1}) = \frac{\#w_{i-1}w_i}{\#w_{i-1}}$ is the conditional probability of word bigrams in the sequence.

A sentence is noisy if the ratio of MDL over sentence length is less than a predefined threshold:

$$\mathbf{s} \text{ is noisy if } \frac{\text{MDL}(\mathbf{s})}{\text{len}(\mathbf{s})} < T$$

If a sentence has no repeat patterns at all, then the length of the codebook should be $C\text{len}(\mathbf{s})$, and $\text{MDL}(\mathbf{s}) \geq C\text{len}(\mathbf{s})$. Thus we choose $T = C$.

For the NLLB mined corpus, we remove pairs with laser score < 1.06 or language score < 0.95 provided by LASER. Monolingual texts are cleaned using language scores only.

Table 1 and Figure 1 summarize the size of our training data after data cleaning and deduplication.

2.3 Preprocessing and Post Processing

There are thousands of languages in the world, thus statically training a tokenizer on a predefined list of languages is not flexible for new languages. There are several studies on dynamic vocabulary for new language adaption, the general principle is to maximize the overlap with the old vocabulary. (Lakew et al., 2018, 2019)

Source	Sentence Pairs
Constrained Track	$50.5M$
NLLB	$29.1M$
Self Collected	$151.6M$
Back Translation	$1.41B$
Total	$1.64B$

Table 1: Number of sentence pairs from different sources after data cleaning.

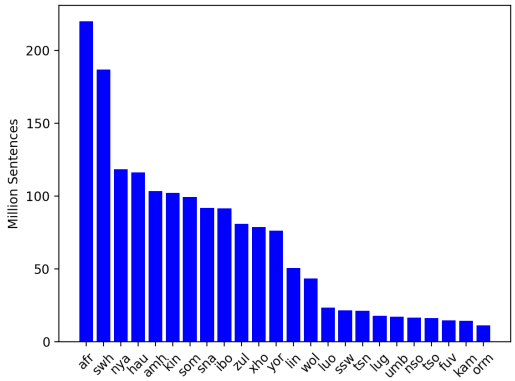

Figure 1: Number of sentences (in millions) in different African languages after data cleaning.

We reuse the mRASP2 tokenizer, a unigram model trained on 150 languages using SentencePiece (Pan et al., 2021). To support new African languages, we train another tokenizer for new languages and merge it to the mRASP2 tokenizer. To ensure that the merged tokenizer produces the same segmentation for old languages, new words that can be made by joining two or more old words are removed and the rest new words' probabilities are scaled down.

We notice that the Yoruba text in FLORES200 has more accented characters than other corpora. According to NLLB team's report, the way FLORES200 marks the tone of vowels is similar to MAFAND dataset (Adelani et al., 2022a). Thus, we use the MAFAND data to train an accent model to post-process the translated sentences for $X \rightarrow$ Yoruba translation. It takes the Yoruba character sequence with accents removed as the input and outputs the accented characters. The structure of the model is a two-layer bidirectional LSTM having 50 hidden units in each layer. Correspondingly, we train another accent model using non-MAFAND datasets to preprocess source text for Yoruba$\rightarrow X$ translation.

3 Model

3.1 Model Architecture

Existing research works demonstrate that small models suffer from the underfitting problem for multilingual machine translation. On the other hand, training and serving large models are expensive. Sometimes model parallelism or pipeline parallelism is necessary if it is impossible to run training on a single GPU due to memory constraints. And quantization is required to reduce the latency of inference. Our compromised model is a pre-layer norm transformer with $2.1B$ parameters which can be trained using $A100$ GPUs with $80G$ memory without parallelism. Details of the model are described in Table 2

Parameter	Value
Encoder Layer	64
Decoder Layer	64
Hidden Size	1024
FFN dimension	4096
Max Length	512
Shared Embedding	Decoder input output
Positional Embedding	Learned

Table 2: Architecture of our transformer model

3.2 Language Tag

There are two popular language tag strategies for multilingual MT: S-ENC-T-DEC which adds source language token to encoder input and target language token to decoder input (Fan et al., 2021; Liu et al., 2020; Wu et al., 2021), and T-ENC which adds target language token to encoder input (Yang et al., 2021; Wu et al., 2021). Our system uses T-ENC-T-DEC which adds the target language token to both encoder and decoder inputs. We did not use source language information for two reasons. First, most translation engines detect input languages automatically, which may introduce incorrect source language tokens. Second, a source sentence may be written in mixed languages.

4 Training and Optimization

4.1 Platform

Our models are trained on 6 machines each equipped with 8 Nvidia $A100$ $80G$ GPUs. We use our internal version of ParaGen [9] (Feng et al., 2022) , a self-developed text generation framework, to train the model. For back-translation, monolingual data are split and translated in parallel using 50 Nvidia Tesla $V100$ GPUs.

To accelerate training, LightSeq is integrated. Unlike approaches that proposed alternative model structures to trade quality for speed, LightSeq used a series of GPU optimization techniques tailored to the specific computation flow and memory access patterns of transformer models. It has been demonstrated 50% to 250% faster than Apex [10] on machine translation tasks. (Wang et al., 2021, 2022) Its inference speed is about 11.5 sentences per second using a single Nvidia Tesla $V100$ GPU, which allows us to translate all monolingual texts within a month.

As the training set's size exceeds the local disk's capacity, it is stored on a remote Hadoop file system.

4.2 Hyper-parameter Tuning

We tune the hyperparameters using a hill climbing approach where each iteration searches along one direction with a different value in the hyperparameter space while keeping the others constant in order to converge to the locally optimal solution on the validation set. To search efficiently, we fix a small batch size and tune other parameters, then increase the batch size after the other parameters have been tuned.

The final configuration is listed in Table 3.

Hyperparameter	Value
Initial Learning Rate	0.001
Warmup Steps	1000
Learning Rate Scheduler	Inverse Square Root
Dropout Rate	0.1
Sampling Temperature	5
Label Smoothing	0.1
Optimizer	AdamW$(0.9, 0.98)$
Activation Function	ReLU
Batch Size	$21M$ tokens

Table 3: Hyperparameters for training.

4.3 Streaming Data Shuffling

Data Shuffling reduces the variance of mini-batches and lowers the risk of local optimum. However, it is challenging to shuffle a Terabyte-scale dataset

[9] https://github.com/bytedance/ParaGen

[10] https://github.com/NVIDIA/apex

dynamically. Our system uses multi-source streaming data based shuffling, which maintains a small in-memory buffer and a set of file pointers that point to random offsets of the training set. Each time a file pointer is selected randomly and loads the next sample to the buffer. A batch of samples is drawn from the buffer randomly once the buffer is full. This approach takes the advantage of data prefetching for sequential access in the Hadoop file system. The randomness of the sampling is controlled by the number of file pointers and the size of the buffer. In our experiments, we use about $5k$ file pointers and $300G$ host memory for the buffer.

To compare with global dynamic shuffling, we run a simulation experiment. We train a model until convergence, then shuffle the full dataset statically, and continue training on the shuffled data. Repeat shuffling until no significant change in loss or performance. For clarity, the original model is named as M_0, and the model trained with $i-th$ round of shuffled data is named as M_i.

Table 4 shows the averaged per token loss of the last 100 training steps and averaged BLEU of M_i on English $\leftrightarrow$ African language translations. We observed a slight improvement in the first round, but no significant change in the second round. This experiment suggests that our shuffling method combined with a limited number of static shuffling is a good approximation of global dynamic shuffling.

	M_0	M_1	M_2
Averaged Loss	1.95	1.91	1.91
Averaged BLEU	21.39	21.48	21.49

Table 4: Simulation Experiment of global dynamic data shuffling: M_0 is the model trained on original training data. M_i is the model trained on the $i-th$ round of statically shuffled data using M_{i-1} as the initial point. The averaged training loss over the last 100 steps and averaged BLEU of English $\leftrightarrow$ African translations are reported.

4.4 Small Dynamic Vocabulary vs Large Static Vocabulary

Existing studies on vocabulary size do not reach a consensus. Large vocabularies often outperform small ones (Gowda and May, 2020), but not always (Liao et al., 2021)

Our vocabulary has $100k$ words, smaller than most of the other systems. Another difference is that our vocabulary is incrementally built for more than 150 languages, it may miss important words in new languages.

To understand the impact of vocabulary, we train another large unigram model with $200K$ words on the 26 languages in this shared task. Table 5 shows the performance with different vocabularies. It is obvious that the $100K$ vocabulary outperforms the $200K$ vocabulary, about 0.3 improvement in BLEU on average.

Vocabulary Size	Languages	BLEU
$100k$ words	173	21.97
$200k$ words	26	21.64

Table 5: Average BLEU of English $\leftrightarrow$ African translations on the FLORES200 devtest set for the models with different vocabularies.

4.5 Pivot vs Direct

As reported in Microsoft's work, pivot-based translation is more robust, especially for directions between low-resource languages since corpora of $X \leftrightarrow Y$ are commonly sparser than $X \leftrightarrow$ English. (Yang et al., 2021) Therefore we use English as the pivot language for African-African translation. For French-African translation, the size of $X \leftrightarrow$ French data is comparable to $X \leftrightarrow$ English. Thus, we train a model for both English and French and choose the better one during inference time.

4.6 Model Averaging

As suggested by other works, model averaging is a simple trick that could significantly improve the performance without changing the model structure or slowing the inference speed. The only cost is the external disk spaces to save intermediate checkpoints, which is trivial compared with GPU and memory costs.

We save the checkpoints every 100 updates of gradients and average the last K checkpoints. By enumerating K from 1 to 20, we find that $K = 10$ is large enough to capture most of the gains.

5 Results

5.1 System Tuning

We tune our model on the FLORES200 devtest dataset, starting with a base model trained on the official data for the constrained track. Then we add more datasets and apply the optimization described above to boost performance. Table 6 reports the averaged BLEU over 56 directions includ-

ing 24 African languages from and to English and 4 African languages from and to French.

Model Description	BLEU
Base model	16.92
+ NLLB and self-collected data	18.89
+ Data cleaning	19.64
+ Back-translation data	22.85
+ $X \rightarrow$ English $\rightarrow$ French	22.95
+ French $\rightarrow$ English $\rightarrow X$ †	22.90
+ Yoruba Accent for $X \rightarrow$ Yoruba	23.20
+ Yoruba Accent for Yoruba $\rightarrow X$ †	23.17
+ Model Averaging	23.35

Table 6: System tuning on FLORES200 devtest set, averaged BLEU over 56 directions is reported. Superscript † means the modification is not included in the final submission.

We can see that the amount of training data is proportional to the performance of the model, especially when back-translation data is added. For some very low resource languages such as Wolof, back-translation improves Wolof $\rightarrow$ English from 11.1 to 19.3, and English $\rightarrow$ Wolof from 4.17 to 7.07.

Another observation is that pivot translation outperforms direct translation for $X \rightarrow$ French directions, but underperforms for French $\rightarrow X$, which indicates that the final step in pivot translation dominates the overall performance.

The impact of Yoruba accent models also shows mixed results. There is a significant improvement for $X \rightarrow$ Yoruba translation, but a little damage to Yoruba$\rightarrow$ X translation. One possible reason is that the non-MAFAND dataset has multiple sources with different accent annotation standards, making the accent model confused. Therefore we only apply post-processing for $X \rightarrow$ Yoruba translations.

5.2 Final Result

Official evaluation metrics include BLEU, sentence-piece BLEU (spBLEU) score, and chrF++. Table 7 shows the results of our primary submission on FLORES200 dev, FLORES200 devtest set, and hidden test sets respectively. The sentence-piece model for calculating spBLEU is SPM-200 provided by Meta AI [11]

[11] https://github.com/facebookresearch/fairseq/tree/nllb

Dataset	BLEU	spBLEU	chrF++
FLORES dev	17.41	21.70	42.01
FLORES devtest	17.43	21.71	41.99
Official test	17.30	21.90	41.87

Table 7: Results of our primary submission on FLORES200 dev, FLORES200 devtest and official test datasets respectively. Metrics are averaged over 100 language pairs.

6 Conclusion

This paper presents our system for the WMT22 shared task on Multilingual Machine Translation for African Languages. We focus on data collection, augmentation, and cleaning. Due to the limited time, we did not try modeling tricks such as reranking and ensemble. Our finding is that the amount of data is crucial for translation quality, especially monolingual data in low-resource languages.

Acknowledgements

We thank Ying Xiong and Yang Wei for building the LightSeq package for this submission. We also thank the GMU-eval team for their effort to make our system work on the evaluation platform.

References

Common crawl. https://commoncrawl.org/. Accessed: 2022-07-18.

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a Cleaner Document-Oriented Multilingual Crawled Corpus. *arXiv e-prints*, page arXiv:2201.06642.

David Adelani, Jesujoba Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, Ernie Chang, Tajuddeen Gwadabe, Freshia Sackey, Bonaventure F. P. Dossou, Chris Emezue, Colin Leong, Michael Beukman, Shamsuddeen Muhammad, Guyo Jarso, Oreen Yousuf, Andre Niyongabo Rubungo, Gilles Hacheme, Eric Peter Wairagala, Muhammad Umair Nasir, Benjamin Ajibade, Tunde Ajayi, Yvonne Gitau, Jade Abbott, Mohamed Ahmed, Millicent Ochieng, Anuoluwapo Aremu, Perez Ogayo, Jonathan Mukiibi, Fatoumata Ouoba Kabore, Godson Kalipe, Derguene Mbaye, Allahsera Auguste Tapo, Victoire Memdjokam Koagne, Edwin Munkoh-Buabeng, Valencia Wagner, Idris Abdulmumin, Ayodele Awokoya, Happy Buzaaba, Blessing Sibanda, Andiswa Bukula, and Sam Manthalu. 2022a. A few thousand translations go a long way! leveraging pre-trained models for African news translation. In *Proceedings of*

the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3053–3070, Seattle, United States. Association for Computational Linguistics.

David Ifeoluwa Adelani, Md Mahfuz Ibn Alam, Antonios Anastasopoulos, Akshita Bhagia, Marta Costa-Jussá, Jesse Dodge, Fahim Faisal, Christian Federmann, Natalia Fedorova, Francisco Guzmán, Sergey Koshelev, Jean Maillard, Vukosi Marivate, Jonathan Mbuya, Safiyyah Saleem, and Holger Schwenk. 2022b. Findings of the WMT 2022 shared task on large-scale machine translation evaluation for african languages. In *Proceedings of the Seventh Conference on Machine Translation*, Online. Association for Computational Linguistics.

Željko Agić and Ivan Vulić. 2019. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy. Association for Computational Linguistics.

Burton H. Bloom. 1970. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Michael Auli, and Armand Joulin. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Jiangtao Feng, Yi Zhou, Jun Zhang, Xian Qian, Liwei Wu, Zhexi Zhang, Yanming Liu, Mingxuan Wang, Lei Li, and Hao Zhou. 2022. Paragen : A parallel generation toolkit.

Thamme Gowda and Jonathan May. 2020. Finding the optimal vocabulary size for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3955–3964, Online. Association for Computational Linguistics.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Xiang Kong, Adithya Renduchintala, James Cross, Yuqing Tang, Jiatao Gu, and Xian Li. 2021. 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. Association for Computational Linguistics.

Surafel M. Lakew, Aliia Erofeeva, Matteo Negri, Marcello Federico, and Marco Turchi. 2018. Transfer learning in multilingual neural machine translation with dynamic vocabulary. In *Proceedings of the 15th International Conference on Spoken Language Translation*, pages 54–61, Brussels. International Conference on Spoken Language Translation.

Surafel M. Lakew, Alina Karakanta, Marcello Federico, Matteo Negri, and Marco Turchi. 2019. Adapting multilingual neural machine translation to unseen languages. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics.

Baohao Liao, Shahram Khadivi, and Sanjika Hewavitharana. 2021. Back-translation for large-scale multilingual machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 418–424, Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8(0):726–742.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Xiao Pan, Mingxuan Wang, Liwei Wu, and Lei Li. 2021. 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. Association for Computational Linguistics.

Holger Schwenk and Matthijs Douze. 2017. Learning joint multilingual sentence representations with neural machine translation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017*, pages 157–167. Association for Computational Linguistics.

Brian Thompson and Philipp Koehn. 2019. Vecalign: Improved sentence alignment in linear time and space. 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 1342–1348, Hong Kong, China. Association for Computational Linguistics.

Xiaohui Wang, Yang Wei, Ying Xiong, Guyue Huang, Xian Qian, Yufei Ding, Mingxuan Wang, and Lei Li. 2022. Lightseq2: Accelerated training for transformer-based models on gpus. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '22, Dallas, Texas. Association for Computing Machinery.

Xiaohui Wang, Ying Xiong, Yang Wei, Mingxuan Wang, and Lei Li. 2021. LightSeq: A high performance inference library for transformers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pages 113–120, Online. Association for Computational Linguistics.

Liwei Wu, Shanbo Cheng, Mingxuan Wang, and Lei Li. 2021. Language tags matter for zero-shot neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3001–3007, Online. Association for Computational Linguistics.

Jian Yang, Shuming Ma, Haoyang Huang, Dongdong Zhang, Li Dong, Shaohan Huang, Alexandre Muzio, Saksham Singhal, Hany Hassan, Xia Song, and Furu Wei. 2021. Multilingual machine translation systems from Microsoft for WMT21 shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 446–455, Online. Association for Computational Linguistics.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. 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. Association for Computational Linguistics.