

NERDS-NITS at WMT 2026: Pivot-Based and Direct Neural Machine Translation for Arabic–Asian Low-Resource Language Pairs

Misbahul Karim¹ Faruk Ahmed¹ Mishbahul Al Islam¹
Sahinur Rahman Laskar² Rabul Hussain Laskar¹

¹National Institute of Technology, Silchar, Assam, India

²UPES, Dehradun, India

{misbahul_ug_23, faruk_ug_23}@ei.nits.ac.in mishbahul_ug_24@ee.nits.ac.in
rhlaskar@ece.nits.ac.in sahinurlaskar.nits@gmail.com

Abstract

This paper presents NERDS-NITS’s submission to the WMT 2026 Low-Resource Arabic–Asian Language Translation Shared Task, which covers the following language pairs: Arabic↔English, Arabic↔Hindi, Arabic↔Bangla, Arabic↔Urdu, and Arabic↔Indonesian, under the primary task category. For the three Indic-language pairs, we propose a pivot-based framework that combines a fine-tuned Arabic↔English NLLB-200 model with pretrained IndicTrans2 checkpoints, motivated by the scarcity of direct Arabic–Indic parallel data. For Arabic↔Indonesian, where IndicTrans2 offers no coverage, we instead fine-tune NLLB-200-1.3B directly using QLoRA under memory-constrained conditions. Our Arabic↔English backbone, built on fine-tuned NLLB-200-600M model, forms the pivot leg for all Indic-language systems. Across all evaluated pairs, fine-tuning yields consistent improvements over zero-shot baselines, and the proposed pivot architecture further outperforms direct multilingual fine-tuning for Arabic–Indic pairs, with gains of up to 6.68 BLEU and 0.02 COMET. The fine-tuned model is made publicly available¹ for the community.

1 Introduction

Neural Machine Translation (NMT) has significantly improved with the development of large multilingual transformer models capable of translating hundreds of languages. Despite these advances, direct translation between Arabic and low-resource Asian languages such as Hindi, Bangla, Indonesian and Urdu remains challenging due to the limited availability of high-quality parallel datasets. Multilingual pretrained models such as NLLB-200 have

demonstrated strong cross-lingual transfer capabilities across a diverse set of languages, while IndicTrans2 has achieved competitive performance for English–Indic translation. Together, these models provide a robust foundation for developing translation systems for low-resource language pairs.

This paper explores translation between Arabic and several low-resource Asian languages, including Hindi, Bangla, Urdu, and Indonesian. For the Arabic–Hindi, Arabic–Bangla, and Arabic–Urdu directions, we employ pivot-based translation, combining a fine-tuned Arabic↔English NLLB model with pretrained IndicTrans2 models. For Arabic–Indonesian, we use the multilingual NLLB model for direct translation. Arabic–English translation is handled using the OPUS Helsinki-NLP model. The proposed systems are evaluated in the WMT’26 shared task.

2 Related Works

Recent multilingual machine translation heavily relies on large-scale pretraining. DeGenaro and Lupicki (2024) present a comparative study which shows that, for MT tasks with indigenous languages, transfer learning of NLLB-200 yields higher performance than a Mamba model trained from scratch. Mathewson (2026) explains that, within its word embeddings, NLLB-200 combines a geometrically separable conceptual core with a linguistic bias which allows high cross-lingual alignment. IndicTrans2 (Gala et al., 2023) handles regional parity for Indic languages, and the open-source OPUS-MT ecosystem (Tiedemann and Thottingal, 2020; Tiedemann et al., 2023) provides foundational baselines for diverse language pairs. To handle specific linguistic complexities, Kim et al. (2019) utilize pivot-language pretraining strategies to bridge language pairs lacking direct parallel data.

Adapting these massive architectures to spe-

¹<https://huggingface.co/collections/mishbahul11/arabic-asian-low-resource-translation-models-wmt-2026>

cialized, low-resource, or constrained environments requires strict parameter and data efficiency. Parameter-Efficient Fine-Tuning (PEFT) through LoRA (Hu et al., 2022) minimizes training overhead using low-rank matrices. This approach is further optimized by LoRA+ (Hayou et al., 2024), which balances adapter learning rates; MeTA-LoRA (Cheng et al., 2025), which isolates task-specific updates from shared meta-knowledge; and QLoRA (Dettmers et al., 2023), which quantizes and freezes the parameters of pretrained language models before loading adapters. For domain-specific applications, translation quality heavily degrades as text density and sentence complexity increase, as exposed in legal-domain evaluations by Damseh and Ghassemiazghandi (2026). However Alghamdi et al. (2024) demonstrate that injecting even minimal specialised in-domain data drastically mitigates these performance gaps during fine-tuning.

3 Dataset Description

The dataset provided by WMT’26 (Laskar et al. (2026)) consists of three splits, namely Train, Dev, and Devtest. The devtest is the public evaluation set, and an additional challenge test set was provided for inference, whose evaluation is private to the organizers. For each pair of languages, the same dataset is used for two pipelines, one for each direction of translation; the roles are matched by toggling between the source and target language. For Arabic–Indonesian, the train set contained 20,400 pairs, and the dev and devtest sets contained 1,000 and 500 pairs respectively. For Arabic-Hindi, Arabic-Bangla and Arabic-Urdu the trainset contained 20,493 pairs, and the dev and devtest sets contained 1,000 and 500 pairs whereas for the Arabic-English, the trainset contained 19,950 pairs, and the dev and devtest sets contained 1,050 and 500 pairs respectively.

3.1 Dataset Preprocessing

In order to filter out inconsistencies, we employed three criteria to clean up the data provided. Our pre-processing pipeline searched for empty samples, repeated samples and samples with extreme length mismatch. It counts the number of words in both the source and target sentences from a pair and checks if the ratio between them is less than the max ratio 3. After these checks, a total of 47 inconsistent samples were discarded from the

Arabic-Indonesian training set. The Fine Tuning of the Arabic-Indonesian models were then carried out with the remaining 20,353 samples.

4 System Description

We adopted two different techniques for the translation pipelines across the 5 language pairs. For Arabic-Indic language pairs (Hindi, Urdu, Bangla), we used a pivot-based approach as shown in Figure 1. This approach uses two different models in a single pipeline. The source language is first translated into English using one model, and then the translated English sentence is translated into the target language using another model. For the other two pairs (Arabic-English, Arabic-Indonesian), we used a direct approach which consisted of fine-tuning a single model.

4.1 For Pivot-Based Systems

4.1.1 Direct Fine-tuning: Arabic ↔ English

The Arabic ↔ English direction forms the backbone of our entire system. Owing to our approach of pivot-based translation for Arabic↔Bangla, Arabic↔Hindi, and Arabic↔Urdu, we first translate every non-English target language pair to English. A good Arabic↔English system directly improves the quality of every downstream pair. We therefore treated this direction as the primary target of our fine-tuning effort rather than a secondary component, and dedicated the majority of our experimentation to selecting the strongest possible Arabic↔English base model before building any pivot pipeline on top of it.

We adopted NLLB-200 (NLLB Team et al., 2022) as our starting point, given its explicit design for low-resource languages and its native support for both Arabic (arb_Arab) and English (eng_Latn) without requiring script conversion or transliteration. Two strategies were explored:

Parameter-efficient fine-tuning (LoRA). We first fine-tuned NLLB-200-1.3B using Low-Rank Adaptation (Hu et al., 2022), injecting rank-decomposition matrices into the query, key, value, and output projections, as well as the feed-forward layers of the Transformer, while keeping the pre-trained backbone frozen. This approach was motivated by the constraints of our single-GPU (T4, 16GB) setup.

Full fine-tuning. In parallel, we fine-tuned the smaller NLLB-200-600M distilled checkpoint with full-parameter updates, updating all model weights

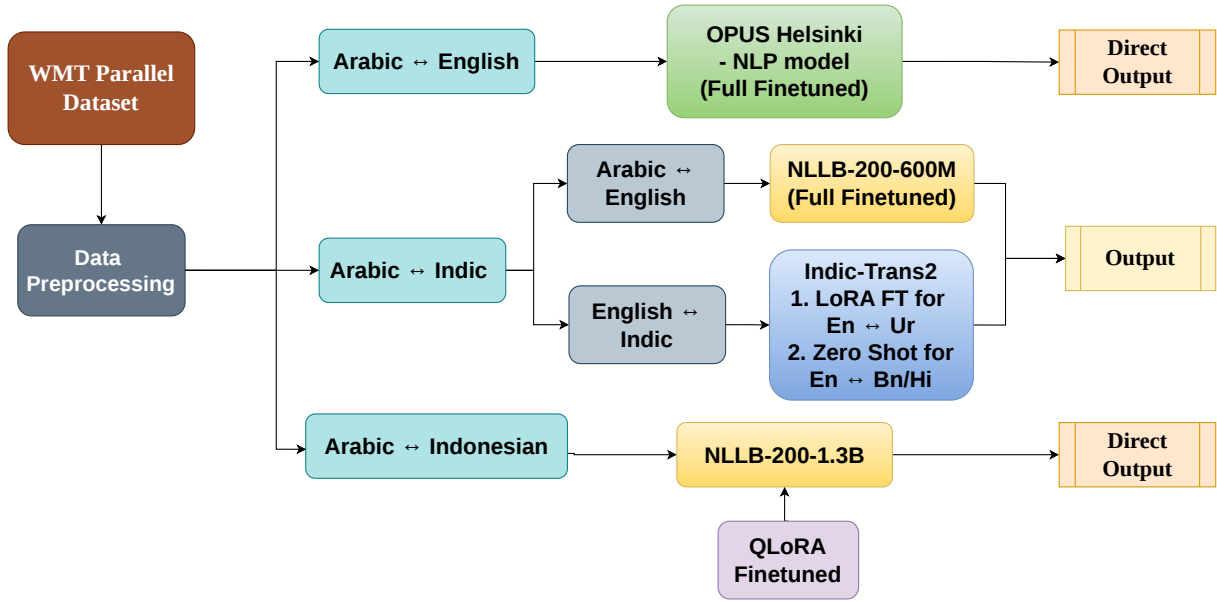


Figure 1: Overview of the pivot-based and direct translation pipelines used in our system.

rather than a low-rank subset.

It was observed that the fully fine-tuned 600M model outperformed the 1.3B parameter model across BLEU, chrF, and COMET on our held-out development set, in both translation directions. Hence, we selected the fully fine-tuned NLLB-200-600M model as our Arabic↔English backbone for the entire pivot system.

4.1.2 The En-X Pivot

Direct multilingual translation between Arabic and low-resource Indic languages is constrained by the scarcity of high-quality Arabic–Indic parallel corpora. Although the NLLB-200 family supports Arabic↔Indic translation, we observed that, due to data sparsity, its zero-shot and fully fine-tuned performance for the 600M model remained limited. As strong dedicated resources exist for both Arabic↔English and English↔Indic translation independently, exploiting a pivot architecture yielded consistent improvement.

For the English↔Indic leg of the pivot, we use IndicTrans2 (Gala et al., 2023), specifically the distilled 200M-parameter variants: ai4bharat/indictrans2-en-indic-dist-200M for English→Indic directions and ai4bharat/indictrans2-indic-en-dist-200M for Indic→English directions. Since it was trained on large-scale Indic data, it natively supports Urdu (urd_Arab), Bangla (ben_Beng), and Hindi (hin_Deva). The English↔Hindi and English↔Bangla legs of our pivot pipelines used

the pretrained IndicTrans2-200M checkpoints without any fine-tuning, due to compute constraints, whereas we were able to fine-tune the English↔Urdu leg of our pipeline.

4.2 For Direct Systems

4.2.1 For Arabic ↔ Indonesian System

For the Arabic–Indonesian pair, the pivot approach was not adopted, since IndicTrans2 does not include Indonesian. Instead, we selected and fine-tuned NLLB-200-1.3B, a larger variant of NLLB-200-600M, for its direct coverage of the Arabic–Indonesian pair and its strong balance of speed and performance. However, the substantial memory footprint of NLLB-200-1.3B makes full fine-tuning computationally infeasible on our setup. To address this, we adopted QLoRA (Quantized Low-Rank Adaptation) (Dettmers et al., 2023) into our pipeline. Taking the high morphological complexity of Arabic into account, we chose a LoRA rank of $r = 64$, ensuring high model capacity; roughly 4.2% of the model parameters were thereby selected and retained for training, with the remainder frozen. Upon evaluation, notable improvements were observed for the fine-tuned model over the zero-shot baseline. For inference, we used Beam Search with beam width 5.

4.2.2 For Arabic ↔ English System

Our baseline system relies on the publicly available Helsinki-NLP/opus-mt-tc-big-ar-en and Helsinki-NLP/opus-mt-tc-big-en-ar models,

handling Arabic→English and English→Arabic translation respectively. Both are built on the Marian transformer-big architecture and implemented through the Hugging Face transformers library, with a 6-layer encoder and 6-layer decoder, a hidden size of 1024, a feed-forward dimension of 4096, and 16 attention heads per layer, totaling around 240 million parameters, with subword tokenization handled by SentencePiece trained on the OPUS parallel corpus. Rather than using these pretrained checkpoints as-is, we treat them as a starting point: since they were originally trained on a broad, general-domain corpus, we fine-tune each one further on our own task-specific parallel data to better adapt it to Arabic↔English translation for our target domain. This fine-tuning is done using the Hugging Face Seq2SeqTrainer framework, which optimizes the models by minimizing the sequence-to-sequence cross-entropy loss between the generated output and the reference translations.

5 Experimental Setup

All models were trained on a single NVIDIA T4 (16GB) using fp16/bf16 mixed precision, gradient checkpointing, and cosine learning rate scheduling, with early stopping on validation loss (patience = 2). Table 1 summarizes the training configuration for each of our fine-tuned models.

For Arabic↔English specifically, we trained two independent directional checkpoints, one for Arabic→English and one for English→Arabic. Both directional checkpoints were fully fine-tuned from the same NLLB-200-600M initialization, using identical hyperparameters (Table 1) so that any downstream performance difference between directions reflects genuine asymmetries in the translation task rather than differences in training configuration. Throughout, we follow $\alpha = 2r$ across different LoRA ranks, so that the effective update magnitude does not change unpredictably between configurations.

For Arabic–Indonesian, a common training configuration was used for both directions of translation. The baseline NLLB-200-1.3B was loaded onto an NVIDIA T4 GPU with 16GB of usable VRAM. The model was compressed by quantizing its weights using NF4, mitigating quantization error. The data was tokenized using the AutoTokenizer class, which automatically loaded the correct tokenizer architecture and vocabulary

configuration for the original model. During training, the sequence length was limited to 192 tokens to minimize truncation issues and fit the pipeline into our limited VRAM without out-of-memory errors. We used an effective batch size of 32 (batch size 4, gradient accumulation 8). A learning rate of 2×10^{-4} with a warmup ratio of 0.05 was chosen, as LoRA fine-tuning tolerates larger steps than full fine-tuning. We used Adafactor as the optimizer; since LoRA converges faster than full fine-tuning, 6 epochs were sufficient.

6 Results and Analysis

The organizers of the WMT 2026 Low-Resource Arabic–Asian Language Translation Shared Task released the official evaluation results using BLEU, chrF2, TER, COMET, and mBERT metrics. Our team, NERDS-NITS, participated in both primary tracks, and the submitted systems demonstrated competitive performance across all evaluated language pairs. We secured 1st position in two of the subtasks: Arabic → Indonesian and Arabic → Bangla; 2nd position in four subtasks: Arabic → Urdu, Arabic → Hindi, Bangla → Arabic and Indonesian → Arabic; 3rd position in one of the subtasks: Hindi → Arabic; and 4th position in three subtasks: Arabic → English, English → Arabic and Urdu → Arabic.

Table 3 reports devtest performance, while Table 2 presents official challenge test set results as evaluated by the WMT 2026 organizers. Across both evaluation sets, Helsinki-NLP achieves the strongest Arabic↔English performance, while the pivot-based NLLB + IndicTrans2 framework yields the best results for all three Arabic–Indic pairs (Hindi, Bengali, Urdu), and fine-tuned NLLB-1.3B remains the top performer for Arabic↔Indonesian, consistent with the direct fine-tuning approach adopted for that pair.

Comparing the two evaluation sets, most systems show a modest performance drop on the challenge test set relative to devtest, most notably for Arabic→Hindi, where COMET falls from 0.7186 to 70.0510² and English→Arabic BLEU drops from 24.25 to 19.7713. This gap likely reflects the challenge test set’s broader domain coverage relative to the devtest split, and is consistent with the pivot-based systems’ reliance on two independently trained components, where compounding

²Note: challenge test set COMET values are reported on a 0–100 scale by the organizers, while our devtest COMET values use the standard 0–1 scale.

Model	Pair	Method	LR	Epochs	Batch (eff.)	LoRA (r, α)	Train size
OPUS Helsinki	AR $\leftrightarrow$ EN	Full FT	2e-5	3	16	—	19.95K
NLLB-600M	AR $\leftrightarrow$ EN	Full FT	2e-5	3	64	—	21K
IndicTrans2-200M	EN $\leftrightarrow$ UR	LoRA	3e-5	3	64	32, 64	21K
NLLB-1.3B	AR $\leftrightarrow$ ID	QLoRA	2e-4	6	32	64, 128	20.35K

Table 1: Training configurations across all fine-tuned models. AR = Arabic, EN = English, UR = Urdu, ID = Indonesian.

Direction	Best System	BLEU	chrF2	TER	COMET	mBERT
EN $\rightarrow$ AR	Helsinki-NLP (fine-tuned)	19.7713	53.4896	71.8221	83.4870	84.3868
AR $\rightarrow$ EN	Helsinki-NLP (fine-tuned)	29.3462	55.8251	59.8500	80.8750	81.9188
HI $\rightarrow$ AR	NLLB + IndicTrans (pivot)	15.5918	49.0983	75.1866	81.9127	83.4993
AR $\rightarrow$ HI	NLLB + IndicTrans (pivot)	27.7756	52.9529	63.7534	70.0510	85.0161
BN $\rightarrow$ AR	NLLB + IndicTrans (pivot)	15.0870	48.5534	75.7664	81.6119	83.3574
AR $\rightarrow$ BN	NLLB + IndicTrans (pivot)	17.1027	51.8287	74.1051	78.6902	83.2766
UR $\rightarrow$ AR	NLLB + IndicTrans (pivot)	14.1150	45.8411	76.7996	80.5890	82.8330
AR $\rightarrow$ UR	NLLB + IndicTrans (pivot)	25.3846	50.9115	65.1215	75.1896	83.5103
ID $\rightarrow$ AR	NLLB-1.3B (fine-tuned)	17.9214	50.9884	72.7991	82.9557	84.0537
AR $\rightarrow$ ID	NLLB-1.3B (fine-tuned)	30.5145	59.8177	59.1446	84.1057	86.0613

Table 2: Official challenge test set results per language pair and direction, as evaluated by the WMT 2026 organizers and published on the shared task website. AR = Arabic, EN = English, HI = Hindi, BN = Bangla, UR = Urdu, ID = Indonesian.

errors are more exposed under distribution shift. Overall, these results confirm that both fine-tuning and the proposed pivot-based framework provide consistent and meaningful improvements over baseline systems across all evaluated language pairs.

7 Conclusion

We presented the NERDS-NITS submission to the WMT 2026 Low-Resource Arabic–Asian Language Translation Shared Task, addressing Arabic $\leftrightarrow$ English, Arabic $\leftrightarrow$ Hindi, Arabic $\leftrightarrow$ Bangla, Arabic $\leftrightarrow$ Urdu, and Arabic $\leftrightarrow$ Indonesian. We demonstrated that a pivot-based architecture, combining a fine-tuned Arabic $\leftrightarrow$ English backbone with pretrained IndicTrans2 checkpoints, consistently outperforms direct multilingual fine-tuning for Arabic–Indic pairs where direct parallel data is scarce, while direct fine-tuning with QLoRA remains the stronger choice for Arabic–Indonesian, where no suitable pivot model exists. Future work includes exploring synthetic pivot generation to further reduce error compounding in the two-hop pipeline, and extending fine-tuning to the English–Hindi and English–Bangla legs, which were left as zero-shot in this submission due to compute constraints.

Acknowledgments

We thank the WMT 2026 shared task organizers for providing the datasets and organizing this shared task.

References

- Emad A. Alghamdi, Jezia Zakraoui, and Fares A. Abanmy. 2024. [Domain adaptation for arabic machine translation: Financial texts as a case study](#). *Applied Sciences*, 14(16).
- Bo Cheng, Xiaoyang Wang, Jian Liu, Yong Chang, and Yun Wu. 2025. [MeTA-LoRA: Data-efficient multi-task fine-tuning for large language models](#). *Preprint*, arXiv:2510.11598.
- Karam Damseh and Mozghan Ghassemiazghandi. 2026. A quantitative evaluation of neural machine translation systems for english-arabic customs legal texts. *AWEJ for Translation & Literary Studies*, 10(2).
- Dan DeGenaro and Tom Lupicki. 2024. Experiments in mamba sequence modeling and nllb-200 fine-tuning for low resource multilingual machine translation. In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 188–194.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. [QLoRA: Efficient finetuning of quantized LLMs](#). *Preprint*, arXiv:2305.14314.

Direction	Best System	BLEU	Chrf	COMET
EN→AR	Helsinki-NLP (fine-tuned)	24.25	49.35	87.77
AR→EN	Helsinki-NLP (fine-tuned)	31.16	54.95	83.75
HI→AR	NLLB + IndicTrans (pivot)	16.56	46.3	85.04
AR→HI	NLLB + IndicTrans (pivot)	28.64	51.8	71.86
BN→AR	NLLB + IndicTrans (pivot)	15.73	45.86	84.98
AR→BN	NLLB + IndicTrans (pivot)	17.2	48.1	80.75
UR→AR	NLLB + IndicTrans (pivot)	14.91	43.27	85.09
AR→UR	NLLB + IndicTrans (pivot)	26.73	50.32	79.49
ID→AR	NLLB-1.3B (fine-tuned)	18.67	52.85	85.57
AR→ID	NLLB-1.3B (fine-tuned)	31.97	61.25	85.82

Table 3: Evaluation results on the devtest set for the best performing model across all language pairs. AR = Arabic, EN = English, HI = Hindi, BN = Bangla, UR = Urdu, ID = Indonesian.

- Jay Gala, Pranjal A. Chitale, Raghavan A K, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. [Indic-trans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages](#).
- Soufiane Hayou, Nikhil Ghosh, and Bin Yu. 2024. [LoRA+: Efficient low rank adaptation of large models](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 17783–17806. PMLR.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019. Pivot-based transfer learning for neural machine translation between non-English languages. 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 866–876.
- Sahinur Rahman Laskar, Firoj Alam, Bishwaraj Paul, Irfan Ahmad, Maya Silvi Lydia, and Pankaj Kundan Dadure. 2026. Findings of the WMT 2026 shared task on low-resource arabic-asian language translation. In *Proceedings of the Eleventh Conference on Machine Translation*, Budapest, Hungary. Association for Computational Linguistics.
- Kyle Elliott Mathewson. 2026. Universal conceptual structure in neural translation: Probing nllb-200’s multilingual geometry. *arXiv preprint arXiv:2603.02258*.
- NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Hefernan, 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, and 20 others. 2022. [No language left behind: Scaling human-centered machine translation](#). *Preprint*, arXiv:2207.04672.
- Jörg Tiedemann, Mikko Aulamo, Daria Bakshandaeva, Michele Boggia, Stig-Arne Grönroos, Tommi Nieminen, Alessandro Raganato, Yves Scherrer, Raul Vazquez, and Sami Virpioja. 2023. [Democratizing neural machine translation with OPUS-MT](#). *Preprint*, arXiv:2212.01936.
- Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, Lisboa, Portugal. European Association for Machine Translation.