

Scaling neural machine translation to 200 languages

<https://doi.org/10.1038/s41586-024-07335-x>


NLLB Team*

Received: 8 May 2023

Accepted: 19 March 2024

Published online: 5 June 2024

Open access

 Check for updates

The development of neural techniques has opened up new avenues for research in machine translation. Today, neural machine translation (NMT) systems can leverage highly multilingual capacities and even perform zero-shot translation, delivering promising results in terms of language coverage and quality. However, scaling quality NMT requires large volumes of parallel bilingual data, which are not equally available for the 7,000+ languages in the world¹. Focusing on improving the translation qualities of a relatively small group of high-resource languages comes at the expense of directing research attention to low-resource languages, exacerbating digital inequities in the long run. To break this pattern, here we introduce No Language Left Behind—a single massively multilingual model that leverages transfer learning across languages. We developed a conditional computational model based on the Sparsely Gated Mixture of Experts architecture^{2–7}, which we trained on data obtained with new mining techniques tailored for low-resource languages. Furthermore, we devised multiple architectural and training improvements to counteract overfitting while training on thousands of tasks. We evaluated the performance of our model over 40,000 translation directions using tools created specifically for this purpose—an automatic benchmark (FLORES-200), a human evaluation metric (XSTS) and a toxicity detector that covers every language in our model. Compared with the previous state-of-the-art models, our model achieves an average of 44% improvement in translation quality as measured by BLEU. By demonstrating how to scale NMT to 200 languages and making all contributions in this effort freely available for non-commercial use, our work lays important groundwork for the development of a universal translation system.

The recent advent of neural machine translation (NMT) has pushed translation technologies to new frontiers, but its benefits are unevenly distributed¹. The vast majority of improvements made have mainly benefited high-resource languages, leaving many low-resource languages behind. (For the purpose of our research, we define a high-resource language as a language for which we have at least 1 million sentences of aligned textual data (or bitext) with another language). This disparity could largely be attributed to a data gap: NMT models typically require large volumes of data to produce quality translations and, by definition, these volumes are not available for lower-resource languages. The No Language Left Behind (NLLB-200) project seeks to overcome this limitation by leveraging previously unknown approaches for building massively multilingual models with cross-lingual transfer abilities^{8,9}, thereby enabling related languages to learn from each other^{1,10,11}.

It has now been widely acknowledged that multilingual models have demonstrated promising performance improvement over bilingual models¹². However, the question remains whether massively multilingual models can enable the representation of hundreds of languages without compromising quality. Our results demonstrate that doubling the number of supported languages in machine translation and maintaining output quality are not mutually exclusive endeavours. Our final model—which includes 200 languages and three times as

many low-resource languages as high-resource ones—performs, as a mean, 44% better than the previous state-of-the-art systems. This paper presents some of the most important data-gathering, modelling and evaluation techniques used to achieve this goal.

First, compared with their high-resource counterparts, training data for low-resource languages are expensive and logistically challenging to procure^{13–15}. Publicly available digital resources are either limited in volume or difficult for automated systems to detect (particularly in large public web datasets such as CommonCrawl). Regardless of whether collecting a critical mass of human-translated seed data is necessary, sufficient data acquisition relies on large-scale data mining and monolingual data pipelines^{16–19}. The latter techniques are often affected by noise and biases, thereby making validating the quality of the datasets they generate tedious²⁰. In NLLB-200, we show that a distillation-based sentence encoding technique, LASER3 (ref. 21), facilitates the effective mining of parallel data for low-resource languages.

Second, on the modelling side, we use an assemblage of seed, mined, open-source and back-translated datasets to train multilingual conditional computational models (more specifically, Sparsely Gated Mixtures-of-Experts models^{2–7} that enable cross-lingual transfer between related languages without increasing interference between unrelated languages). We show how we can achieve state-of-the-art

*A list of authors and their affiliations appears at the end of the paper. ✉e-mail: costajussa@meta.com

performance with a more optimal trade-off between cross-lingual transfer and interference, and improve performance for low-resource languages.

Finally, for the purpose of quality evaluation, we created FLORES-200—a massive multilingual benchmark that enables the measurement of translation quality across any of the approximately 40,000 translation directions covered by the NLLB-200 models. Apart from automatic metrics, we also created Cross-lingual Semantic Text Similarity (XSTS) and Evaluation of Toxicity (ETOX). XSTS is a human evaluation protocol that provides consistency across languages; ETOX is a tool to detect added toxicity in translations using toxicity word lists.

Beyond creating these models, we also reflect on the potential societal impact of NLLB. To amplify the practical applicability of our work in service of low-resource-speaking communities, we provide all the benchmarks, data, code and models described in this effort as resources freely available for non-commercial use (<https://github.com/facebookresearch/fairseq/tree/nllb>) (see Data and Code availability statements for details).

Automatically creating translation training data

The current techniques used for training translation models are difficult to extend to low-resource settings, in which aligned bilingual textual data (or bitext data) are relatively scarce²². Many low-resource languages are supported only by small targeted bitext data consisting primarily of translations of the Christian Bible²³, which provide limited domain diversity.

To build a large-scale parallel training dataset that covers hundreds of languages, our approach centres around extending existing datasets by first collecting non-aligned monolingual data. Then, we used a semantic sentence similarity metric to guide a large-scale data mining effort aiming to identify sentences that have a high probability of being semantically equivalent in different languages¹⁸.

Language identification for monolingual data collection

Collecting monolingual data at scale requires a language identification (LID) system that accurately classifies textual resources for all NLLB-200 languages. Although LID could be seen as a solved problem in some domains²⁴, it remains an open challenge for web data^{25,26}. Specifically, issues coalesce around domain mismatch²⁶, similar language disambiguation²⁷ and successful massively multilingual scaling²⁸.

Devoted attention to advancing LID techniques led to a noticeable increase in both language coverage and accuracy over time. CLD3 (<https://github.com/google/cld3>) and fasttext²⁹ are two readily available models offering high detection performance for 107 and 187 languages, respectively. By using numerous public datasets, previous studies^{30,31} report even higher coverage—464 and 1,366 languages, respectively. Another study³² scales LID performance up to 1,629 languages using word lists and self-supervision to bootstrap training data found on the web. However, these approaches using found data suffer from domain imbalance. That is, because the available text domains vary by language, classifiers conflate different domains with different languages.

In our work, we curated FLORES-200 to use as a development set so that our LID system performance³³ is tuned over a uniform domain mix. Our approach combines a data-driven fasttext model trained on FLORES-200 with a small set of handwritten rules to address human feedback on classification errors. These rules are specifically mentioned in section 5.1.3 of ref. 34 and include linguistic filters to mitigate the learning of spurious correlations due to noisy training samples while modelling hundreds of languages.

We compare our LID model with three publicly available models: CLD3, LangId (<https://github.com/saffsd/langid.py>) and LangDetect (<https://pypi.org/project/langdetect/>). Table 1 reports the performance

Table 1 | Comparison of publicly available language identification models with various intersections of labels

	FLORES-200 ∩ CLD3 ∩ LangId ∩ LangDetect		FLORES-200 ∩ CLD3 ∩ LangId		FLORES-200 ∩ CLD3		
No. of supported languages	51 labels		78 labels		95 labels		
	F1	FPR	F1	FPR	F1	FPR	
LangDetect	55	97.3	0.0526	64.4	0.4503	53.1	0.4881
LangId	97	98.6	0.0200	92.0	0.0874	75.8	0.2196
CLD3	107	98.2	0.0225	97.7	0.0238	97.0	0.0283
Ours	218	99.4	0.0084	98.8	0.0133	98.5	0.0134

F1 is the micro-F1 score, and FPR is the micro-false-positive rate.

on three cascading sets of languages intersecting with NLLB-200: (1) 51 languages also supported by LangId, LangDetect and CLD3; (2) 78 languages also supported by LangId and CLD3; (3) 95 languages also supported by CLD3. We also report false-positive rates (FPR) to reflect the impact of false positives on unseen languages. Our results show that our model is equipped to handle all 200 languages found in FLORES-200 while achieving notably higher performance than LangId, LangDetect and CLD3. Furthermore, the gain in F1 score is accompanied by a notable improvement in FPR, suggesting a much stronger fit for extracting low-resource languages from web corpora³².

Mining for bitext

Previous work³⁵ notes that translation quality generally increases with the amount of high-quality training data, which is difficult to procure when working with low-resource languages. Existing parallel corpora for low-resource languages are often conveniently drawn from known multilingual collections, such as the Christian Bible or the publications of multinational organizations, which are limited in quantity and domain. To overcome this problem, we created training datasets through global bitext mining in publicly available web content (drawn from repositories such as CommonCrawl). The underlying idea of our bitext mining approach is first to learn a multilingual sentence embedding space and use a similarity measure in that space to decide whether two sentences are parallel. This comparison can be done for all possible pairs in two collections of monolingual texts.

As our mining approach requires a multilingual embedding space, there are several challenges when scaling this representation to all NLLB-200 languages. First, we had to ensure that all languages were well learnt and that we accounted for large imbalances in available training data. Second, training a massively multilingual sentence encoder from scratch each time a new set of languages is introduced is computationally expensive. Furthermore, the main drawback of this approach is that the learnt embedding spaces from each new model are not necessarily mutually compatible. This can make mining intractable as for each new encoder, the entirety of available monolingual data needs to be re-embedded (for example, for English alone, this means thousands of millions of sentences and considerable computational resources). We solved this problem using a teacher–student approach²¹ that extends the LASER embedding space³⁶ to all NLLB-200 languages. Languages are trained either as individual students or together with languages from the same family. The training of students follows the approach described in ref. 21.

Our approach enables us to focus on the specifics of each language while taking advantage of related languages, which is crucial for dealing with very low-resource languages. (A language is defined as very low-resource if it has fewer than 100,000 samples across all pairings with any other language in our dataset). Using this method, we generated more than 1,100 million new sentence pairs of training data for 148 languages. This additional training data, paired with back translation

Table 2 | Improvements from EOM and CL

	eng_Latn-xx				xx-eng_Latn				xx-yy	Average
	All	High	Low	Very low	All	High	Low	Very low	All	All
(1) Baseline MoE	44.8	54.3	41.4	39.0	56.2	64.0	53.4	52.5	41.9	47.6
(2) Baseline MoE+ CL	45.2	54.7	41.8	39.5	57.6	64.5	55.1	55.4	42.7	48.5
(2) Baseline MoE+CL+EOM	45.4	52.9	41.6	41.2	57.2	61.4	55.1	56.4	44.9	51.0

We report chrF++ scores on FLORES-200 dev set on different types of language pairs. For eng_Latn-xx and xx-eng_Latn, we included all 199 pairs. For xx-yy, we randomly chose 200 directions. We observe that combining EOM and CL is particularly helpful for low and very low-resource languages. A language is defined as a very low resource if it has fewer than 100,000 samples across all pairings with any other language in our dataset. The highest score in each column is shown in bold.

(a conventional technique for data augmentation in NMT; ref. 37), ushered notable improvements in translation quality—specifically, +12.5 chrF++ (ref. 38) for translating very low-resource languages into English. For more details, see Supplementary Information D.

Modelling

Even with marked data volume increases, the main challenge of low-resource translation is for training models to adequately represent 200 languages while adjusting to variable data capacity per language pair. Apart from techniques such as data augmentation (for example, with back translation) and self-supervision strategies on monolingual data, we used conditional computational models—more specifically, Sparsely Gated Mixture of Experts (henceforth MoE)—to minimize interference between unrelated language directions.

MoE transformer models differ from dense transformer models in that some of the feed-forward network layers are replaced with MoE layers in both the encoder and the decoder. An MoE layer consists of E experts (each is a feed-forward network) and a gating network to decide how to route input tokens to experts. The transformer encoder-decoder model, supplemented with MoE layers and their respective gating networks, learns to route input tokens to the corresponding top two experts by optimizing a linearly weighted combination of label-smoothed cross entropy³⁹ and an auxiliary load balancing loss⁶.

We find that vanilla MoE models with overall dropout are suboptimal for low-resource languages and significantly overfit on low-resource pairs. To remedy this issue, we designed Expert Output Masking (EOM), a regularization strategy specific to MoE architectures, and compared it with existing regularization strategies, such as Gating Dropout⁴⁰. We find that Gating Dropout performs better than vanilla MoE with overall dropout but is outperformed by EOM.

To further reduce overfitting on low-resource language pairs, we devised a curriculum learning that introduces language pairs in phases during model training. Pairs that empirically overfit within K updates are introduced with K updates before the end of training. This reduces overfitting while allowing pairs that benefit from additional training

to continue their learning. Table 2 shows that combining curriculum learning and EOM improves performance, especially on low and very low-resource language pairs (see section ‘Modelling’ for more details).

To understand how MoE models are helpful for multilingual machine translation, we visualize similarities of experts in the MoE layers using heat maps (Fig. 1a–d). These heat maps demonstrate that in late decoder layers (Fig. 1d), languages are being separated (that is, dispatched to different sets of experts). Moreover, we observe that languages within the same family are highly similar in their choice of experts (that is, the late decoder MoE layers are language-specific). This is particularly the case for the Arabic dialects (the six rows and columns in the top-left corner), languages in the Benue–Congo subgrouping, as well as languages in the Devanagari script. By contrast, the early decoder MoE layers (Fig. 1c) seem to be less language-specific. The late decoder MoE layers are particularly language-agnostic in how they route tokens as can be attested by the uniform heat map in Fig. 1b.

Combining data (see section ‘Automatically creating translation training data’) and modelling contributions, Table 3 shows that NLLB-200 outperforms the nearest state-of-the-art system by almost +7.3 spBLEU (ref. 41) on average, constituting a 44% improvement. We then compared NLLB-200 with a few other state-of-the-art models, such as Deepnet⁴² and M2M-100 (ref. 1), to report scores for 87 languages against FLORES-101. On this smaller subset, NLLB-200 again outperforms by +7.0 spBLEU on average. Overall, the results show that NLLB-200 improves on state-of-the-art systems by a notable margin despite supporting 200 languages, or twice as many languages (and more than 30,000 additional directions) compared with any previous work. We also show in additional experiments that NLLB-200 is a general-purpose NMT model, transferable to other domains by fine-tuning on small quantities of high-quality bitexts (see Supplementary Information E.3).

Evaluations

Among the many aspects of model performance that can be evaluated⁴³, this section emphasizes three aspects that have a marked impact on

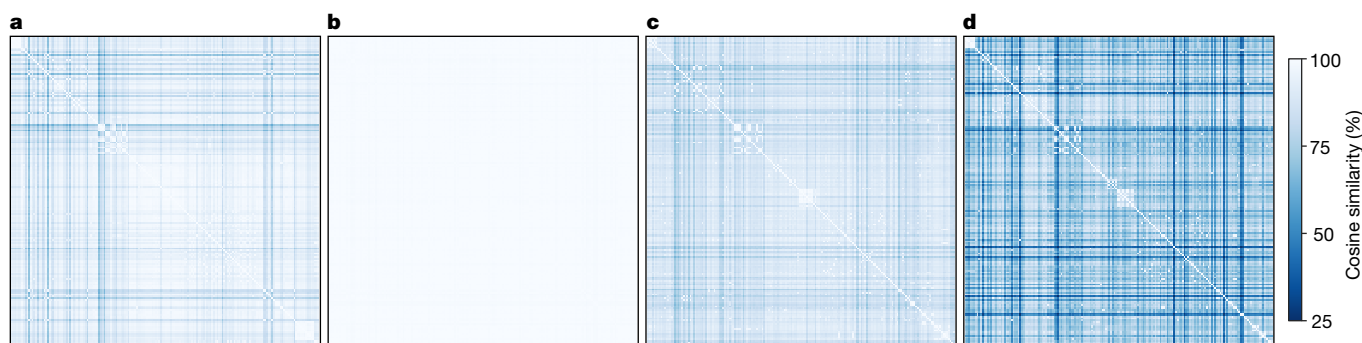


Fig. 1 | Cosine similarity scores between languages at different layers of the encoder-decoder architecture. a–d, The first (a) and last (b) encoder layers and then the first (c) and last (d) decoder layers. The similarity is measured with

respect to the gating decisions (expert choice) per language (source side in the encoder and target side in the decoder). Lighter colours represent higher experts similarity, hence, a language-agnostic processing.

Table 3 | Comparison of FLORES-101 devtest

	eng_Latn-xx	xx-eng_Latn	xx-yy	Average
87 languages				
M2M-100	-/-	-/-	-/-	13.6/-
Deepnet	-/-	-/-	-/-	18.6/-
NLLB-200	35.4 /52.1	42.4 /62.1	25.2 /43.2	25.5 /43.5
101 languages				
DeltaLM	26.6/-	33.2/-	16.4/-	16.7/-
NLLB-200	34.0 /50.6	41.2 /60.9	23.7 /41.4	24.0 /41.7

We evaluated using FLORES-101 for 10,000 directions. We report both spBLEU and chrF++ scores when available. Scores for DeltaLM are taken from the FLORES-101 leaderboard. M2M-100 and Deepnet averages only apply to 87 languages that overlap with FLORES-101. The performance of NLLB-200 was evaluated on this subset of languages. The highest score in each column and in each grouping of languages is shown in bold.

the overall quality assessment: benchmarks for automatic evaluation, human evaluation protocols and toxicity evaluation.

A benchmark for automatic evaluation using FLORES-200

The quality of NMT outputs is typically evaluated by automatic metrics such as BLEU⁴⁴ or spBLEU⁴¹. The computation of automatic quality scores using these metrics requires benchmark datasets that provide gold-standard human translations as references. In turn, the apples-to-apples evaluation of different approaches made possible by these benchmark datasets gives us a better understanding of what requires further research and development. For example, creating benchmark data sets at the Workshop on Machine Translation (WMT)⁴⁵ led to rapid progress in translation directions such as English to German and English to French.

For massively multilingual NMT, the largest benchmark dataset available was FLORES-101, which supports roughly half the number of languages in NLLB-200. The necessary expansion of FLORES-101 to FLORES-200 constitutes a further challenge in terms of quality assurance, in part because of differences in standardization practices and limited access to professional translators for all languages involved. To overcome this challenge, we adapted our workflow to pay particular attention to quality assurance mechanisms. The FLORES-200 workflow consists of four phases: (1) alignment; (2) translation, initial quality assurance and iteration(s); (3) final quality assurance; and (4) completion. A language FLORES-200 set is considered ready after passing a final human quality test with a 90 out of 100 quality score (that is, independent raters agreed with 90% of the FLORES-200 reference translations in that direction).

As a result of this redesigned workflow, we produced a three-split (dev, devtest, test) data set of parallel human reference translations for all NLLB-200 languages meeting the 90% quality threshold in a maximum turnaround time of 287 days (119 days on average, 70 days minimum). (Note that to avoid leakage with our models, we filtered data from FLORES and other evaluation benchmarks used (such as WMT and IWSLT) from our training data. This was done by comparing

the hashes of training sentences against those of evaluation sentences, using the xxHash algorithm). Please refer to Supplementary Information C for more details on the evaluation process. Figure 2 shows the quality scores for all languages, some of which are labelled as examples.

Reliable human evaluation

State-of-the-art automatic metrics often fail to capture aspects of language that, while subtle, can have a notable bearing on translation quality. Human evaluations are, therefore, essential to ensuring meaningful quality assessments⁴⁶. That said, relying on them comes with two challenges: (1) any large-scale human evaluation of NMT quality, regardless of the number of translation directions involved, contends with potentially low inter-evaluator agreement (in the vicinity of 0.5 kappa); and (2) massively multilingual NMT introduces another complexity—that of quality evaluation consistency across language directions. We address these two issues by developing XSTS⁴⁷, a new scoring metric focused on meaning, and by using a protocol that allows for the calibration of scores across evaluators and language pairs.

XSTS is a human evaluation protocol inspired by STS⁴⁸, emphasizing meaning preservation over fluency. XSTS uses a five-point scale, in which 1 is the lowest score, and 3 represents the acceptability threshold. To ensure consistency not only across languages but also among different evaluators of any given language, we included the same subset of sentence pairs in the full set of sentence pairs given to each evaluator, making it possible to calibrate results.

We find that automated metrics such as spBLEU and chrF++ correlate reasonably well with calibrated human evaluations of translation quality, as shown in Fig. 3. Spearman’s *R* correlation coefficients between aggregated XSTS and spBLEU, chrF++ (corpus) and chrF++ (average sentence-level) are 0.710, 0.687 and 0.694, respectively. Other correlation coefficients (Kendall’s τ and Pearson’s *R*) have the same ordering. Corpus spBLEU provides the best nominal correlation, followed by average sentence-level chrF++.

We also find that calibrated human evaluation scores correlate more strongly with automated scores than uncalibrated human evaluation scores across all automated metrics and choices of correlation coefficient. In particular, uncalibrated human evaluation scores have a Spearman’s *R* correlation coefficient of 0.625, 0.607 and 0.611 for spBLEU, chrF++ (corpus) and chrF++ (average sentence-level), respectively.

Overall, a sample of 55 language directions were evaluated, including 8 into English, 27 out of English, and 20 other direct language directions. The overall mean of calibrated XSTS scores was 4.26, with 38/55 directions scoring over 4.0 (that is, high quality) and 52/56 directions scoring over 3.0.

We hypothesize that added toxicity may be because of the presence of toxicity in the training data and used our detectors to estimate, more specifically, unbalanced toxicity in the bitext data. We find that estimated levels of unbalanced toxicity vary from one corpus of bitext to the next and that unbalanced toxicity can be greatly attributed to misaligned bitext. In other words, training with this misaligned bitext could encourage mistranslations with added toxicity.

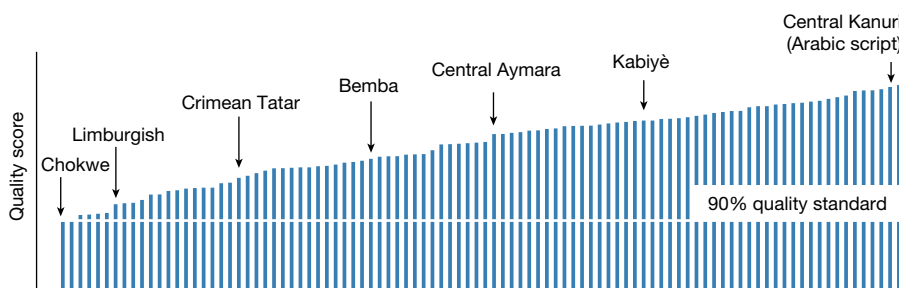


Fig. 2 | Quality of FLORES-200. Quality assurance scores for the languages in FLORES-200. The minimum acceptable standard is 90%.

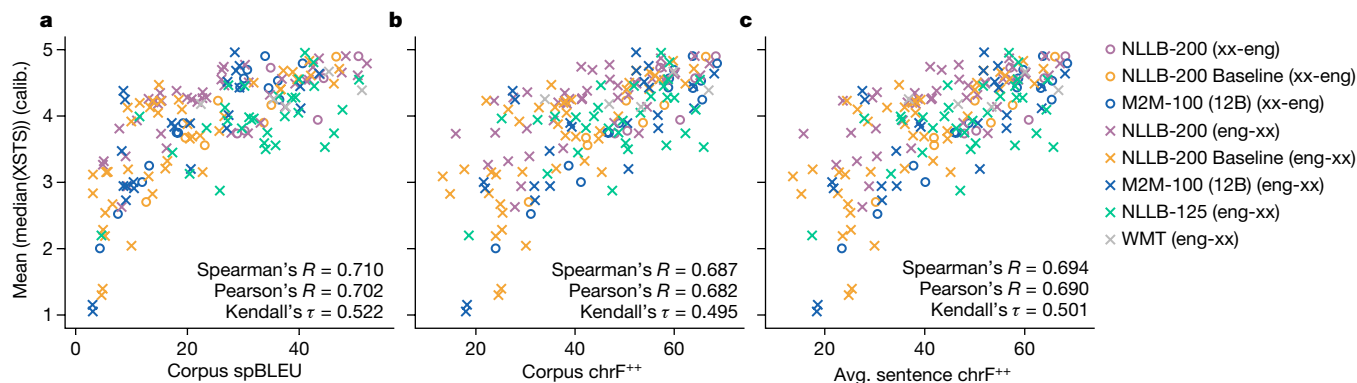


Fig. 3 | Correlations between aggregated human quality scores and automated metrics. **a**, The relationship between spBLEU and XSTS. **b**, The relationship between chrF++ and XSTS. **c**, The relationship between average sentence-level chrF++ and XSTS. All automated scores were computed only on

To mitigate this issue, we designed a bitext filtering procedure based on the detection of multiple instances of added toxicity (that is, cases in which one sentence in the bitext pair contains at least two more toxic items than the other sentence in the pair). (A previous detector quality analysis showed that a higher precision was reached in this situation). We added this toxicity filtering procedure as an option to the filtering process and experimented with or without it for comparison.

The experimental results on the FLORES-200 dev set for 10 translation directions (from and into English for Somali, Southern Sotho, Twi, Umbundu and Venetian) show that after filtering an average amount of around 30% parallel sentences, the translation quality (chrF++) improves by 5% and added toxicity (ETOX) reduces by the same amount. Therefore, the filtering pipeline that includes toxicity filtering not only reduces the number of toxic items in the translation output but also improves the overall translation performance.

Conclusion

In 2016, the United Nations declared internet access a basic human right. Although the intent of this declaration was to limit censorship and allow for information and ideas to flow without interference, much of the internet today remains inaccessible to many due to language barriers. Our effort was designed to contribute one solution to help alter this status quo.

For many low-resource language communities, NLLB-200 is one of the first models designed to support translation into or out of their languages. Although applications of these new translation capabilities could be found in several domains of everyday life, we believe their impact would be most significant in a domain such as education. In formal educational settings, for instance, students and educators belonging to low-resource language groups could, with the help of NLLB-200, tap into more books, research articles and archives than before. Within the realms of informal learning, low-resource language speakers could experience greater access to information from global news outlets and social media platforms, as well as online encyclopaedias such as Wikipedia. Access to machine translation motivates more low-resource language writers or content creators to share localized knowledge or various aspects of their culture. Giving individuals access to new translation tools could thus open up opportunities for bidirectional learning, thereby also challenging Western-centric modes of knowledge production and dissemination, ultimately aiding in revitalizing certain minority cultures and languages.

Since launching NLLB-200, we can already see the impact of the model across many directions. Four months after the launch of NLLB-200, Wikimedia reported that our model was the third most used machine translation engine used by Wikipedia editors (accounting

the sentences evaluated for a given model and translation direction (either the full FLORES-200 dataset or a subset). NLLB-200 refers to a 55B parameter MoE model, and NLLB-200 Baseline refers to a dense 3.3B parameter model.

for 3.8% of all published translations) (https://web.archive.org/web/20221107181300/https://nbviewer.org/github/wikimedia-research/machine-translation-service-analysis-2022/blob/main/mt_service_comparison_Sept2022_update.ipynb). Compared with other machine translation services and across all languages, articles translated with NLLB-200 has the lowest percentage of deletion (0.13%) and highest percentage of translation modification kept under 10%.

In many ways, the composition of the NLLB-200 effort speaks to the centrality of interdisciplinarity in shaping our vision. Machine translation and AI advancements lie at the intersection of technological, cultural and societal development, and thus require scholars with diverse training and standpoints to fully comprehend every angle^{49,50}. It is our hope that in future iterations, NLLB-200 continues to include scholars from fields underrepresented in the world of machine translation and AI, particularly those from humanities and social sciences backgrounds. More importantly, we hope that teams developing these initiatives would come from a wide range of race, gender and cultural identities, much like the communities whose lives we seek to improve.

Finally, we want to emphasize that overcoming the challenges that prevent the web from being accessible to speakers of all languages requires a multifaceted approach. At the technical level, NLLB-200 overcomes many data, modelling and evaluation challenges in NMT research, but it still has its limitations, some of which are documented in Supplementary Information G. As a single technological intervention, NLLB-200 is all but one piece of a massive puzzle; policy interventions aimed at more fundamental issues surrounding education, internet access and digital literacy are imperative to eradicate the structural problem of language disparities.

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at <https://doi.org/10.1038/s41586-024-07335-x>.

1. Fan, A. et al. Beyond English-centric multilingual machine translation. *J. Mach. Learn. Res.* **22**, 1–48 (2021).
2. Du, N. et al. GlM: efficient scaling of language models with mixture-of-experts. In *Proceedings of the 39th International Conference on Machine Learning* Vol. 162, 5547–5569 (PMLR, 2022).
3. Hwang, C. et al. Tutel: adaptive mixture-of-experts at scale. In *6th Conference on Machine Learning and Systems (MLSys, 2023)*.
4. Lepikhin, D. et al. GShard: scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations (ICLR, 2021)*.

5. Lewis, M., Bhosale, S., Dettmers, T., Goyal, N. & Zettlemoyer, L. BASE layers: simplifying training of large, sparse models. In *Proc. 38th International Conference on Machine Learning* Vol. 139, 6265–6274 (PMLR, 2021).
6. Shazeer, N. et al. Outrageously large neural networks: the sparsely-gated mixture-of-experts layer. In *Proc. 2017 International Conference on Learning Representations (ICLR)* 1–19 (ICLR, 2017).
7. Zoph, B. et al. ST-MoE: designing stable and transferable sparse expert models. Preprint at <https://arxiv.org/abs/2202.08906> (2022).
8. Zoph, B., Yuret, D., May, J. & Knight, K. Transfer learning for low-resource neural machine translation. In *Proc. 2016 Conference on Empirical Methods in Natural Language Processing* (eds Su, J. et al.) 1568–1575 (Association for Computational Linguistics, 2016).
9. Nguyen, T. Q. & Chiang, D. Transfer learning across low-resource, related languages for neural machine translation. In *Proc. Eighth International Joint Conference on Natural Language Processing* Vol. 2 (eds Kondrak, G. & Watanabe, T.) 296–301 (Asian Federation of Natural Language Processing, 2017).
10. Arivazhagan, N. et al. Massively multilingual neural machine translation in the wild: findings and challenges. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* Vol. 1, 3874–3884 (Association for Computational Linguistics, 2019).
11. Zhang, B., Williams, P., Titov, I. & Sennrich, R. Improving massively multilingual neural machine translation and zero-shot translation. In *Proc. 58th Annual Meeting of the Association for Computational Linguistics* (eds Jurafsky, D. et al.) 1628–1639 (ACL, 2020).
12. Tran, C. et al. Facebook AI's WMT21 news translation task submission. In *Proc. Sixth Conference on Machine Translation* (eds Barrault, L.) 205–215 (ACL, 2021); <https://aclanthology.org/2021.wmt-1.19>.
13. Orife, I. et al. Masakhane – machine translation for Africa. Preprint at <https://arxiv.org/abs/2003.11529> (2020).
14. Kuwanto, G. et al. Low-resource machine translation training curriculum fit for low-resource languages. Preprint at <https://arxiv.org/abs/2103.13272> (2021).
15. Nekoto, W. et al. Participatory research for low-resourced machine translation: a case study in African languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020* (eds Cohn, T. et al.) 2144–2160 (ACL, 2020).
16. Karakanta, A., Dehdari, J. & van Genabith, J. Neural machine translation for low-resource languages without parallel corpora. *Mach. Transl.* **32**, 167–189 (2018).
17. Bañón, M. et al. ParaCrawl: web-scale acquisition of parallel corpora. In *Proc. 58th Annual Meeting of the Association for Computational Linguistics* (eds Jurafsky, D. et al.) 4555–4567 (ACL, 2020).
18. Schwenk, H. et al. CCMatrix: mining billions of high-quality parallel sentences on the web. In *Proc. 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* Vol. 1 (eds Zong, C. et al.) 6490–6500 (ACL, 2021).
19. Ramesh, G. et al. *Samanantar*: the largest publicly available parallel corpora collection for 11 Indic languages. *Trans. Assoc. Comput. Linguist.* **10**, 145–162 (2022).
20. Kreutzer, J. et al. Quality at a glance: an audit of web-crawled multilingual datasets. *Trans. Assoc. Comput. Linguist.* **10**, 50–72 (2022).
21. Heffernan, K., Çelebi, O. & Schwenk, H. Bitext mining using distilled sentence representations for low-resource languages. Preprint at <https://arxiv.org/abs/2205.12654> (2022).
22. Gowda, T., Zhang, Z., Mattmann, C. & May, J. Many-to-English machine translation tools, data, and pretrained models. In *Proc. 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations* (eds Ji, H. et al.) 306–316 (ACL, 2021).
23. McCarthy, A. D. et al. The Johns Hopkins University Bible corpus: 1600+ tongues for typological exploration. In *Proc. 12th Language Resources and Evaluation Conference* (eds Calzolari, N. et al.) 2884–2892 (European Language Resources Association, 2020); <https://aclanthology.org/2020.lrec-1.352>.
24. McNamee, P. Language identification: a solved problem suitable for undergraduate instruction. *J. Comput. Sci. Coll.* **20**, 94–101 (2005).
25. Abadji, J., Suárez, P. J. O., Romary, L. & Sagot, B. Towards a cleaner document-oriented multilingual crawled corpus. Preprint at <https://arxiv.org/abs/2201.06642> (2022).
26. Widdows, D. & Brew, C. Language identification with a reciprocal rank classifier. Preprint at <https://arxiv.org/abs/2109.09862> (2021).
27. Goutte, C., Léger, S., Malmasi, S. & Zampieri, M. Discriminating similar languages: evaluations and explorations. Preprint at <http://arxiv.org/abs/1610.00031> (2016).
28. Jauhiainen, T., Lindén, K. & Jauhiainen, H. Evaluation of language identification methods using 285 languages. In *Proc. 21st Nordic Conference on Computational Linguistics* (eds Tiedemann, J. & Tahmasebi, N.) 183–191 (2017).
29. Grave, É., Bojanowski, P., Gupta, P., Joulin, A. & Mikolov, T. Learning word vectors for 157 languages. In *Proc. 11th International Conference on Language Resources and Evaluation (LREC 2018)* (eds Calzolari, N. et al.) (ELRA, 2018).
30. Dunn, J. Mapping languages: the corpus of global language use. *Lang. Resour. Eval.* **54**, 999–1018 (2020).
31. Brown, R. D. Non-linear mapping for improved identification of 1300+ languages. In *Proc. 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (eds Moschitti, A. et al.) 627–632 (ACL, 2014).
32. Caswell, I., Breiner, T., van Esch, D. & Bapna, A. Language ID in the wild: unexpected challenges on the path to a thousand-language web text corpus. In *Proc. 28th International Conference on Computational Linguistics* (eds Scott, D. et al.) 6588–6608 (International Committee on Computational Linguistics, 2020); <https://aclanthology.org/2020.coling-main.579>.
33. Joulin, A., Grave, É., Bojanowski, P. & Mikolov, T. Bag of tricks for efficient text classification. In *Proc. 15th Conference of the European Chapter of the Association for Computational Linguistics* Vol. 2 (eds Lapata, M. et al.) 427–431 (ACL, 2017).
34. NLLB Team et al. No language left behind: scaling human-centered machine translation. Preprint at <https://arxiv.org/abs/2207.04672> (2022).
35. Koehn, P. & Knowles, R. Six challenges for neural machine translation. In *Proc. First Workshop on Neural Machine Translation* (eds Luong, T. et al.) 28–39 (ACL, 2017).
36. Artetxe, M. & Schwenk, H. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Trans. Assoc. Comput. Linguist.* **7**, 597–610 (2019).
37. Sennrich, R., Haddow, B. & Birch, A. Improving neural machine translation models with monolingual data. In *Proc. 54th Annual Meeting of the Association for Computational Linguistics (ACL)* Vol. 1 (eds Erk, K. & Smith, N. A.) 86–96 (ACL, 2016).
38. Popović, M. chrF++: words helping character n-grams. In *Proc. Second Conference on Machine Translation* Vol. 2 (eds Bojar, O. et al.) 612–618 (ACL, 2017).
39. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2818–2826 (IEEE, 2016).
40. Liu, R., Kim, Y. J., Muzio, A., Mozafari, B. & Awadalla, H. H. Gating dropout: communication-efficient regularization for sparsely activated transformers. In *Proceedings of the 39th International Conference on Machine Learning* (PMLR, 2022).
41. Goyal, N. et al. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Trans. Assoc. Comput. Linguist.* **10**, 522–538 (2022).
42. Wang, H. et al. DeepNet: scaling transformers to 1,000 layers. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* <https://doi.org/10.1109/TPAMI.2024.3386927> (IEEE, 2024).
43. Freitag, M. et al. Results of the WMT21 metrics shared task: evaluating metrics with expert-based human evaluations on TED and news domain. In *Proc. Sixth Conference on Machine Translation* (eds Barrault, L. et al.) 733–774 (ACL, 2021); <https://aclanthology.org/2021.wmt-1.73>.
44. Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proc. 40th annual meeting of the Association for Computational Linguistics* (eds Isabelle, P. et al.) 311–318 (ACL, 2002).
45. Akhbardeh, F. et al. Findings of the 2021 conference on machine translation (WMT21). In *Proc. Sixth Conference on Machine Translation* (eds Barrault, L. et al.) 1–88 (ACL, 2021); <https://aclanthology.org/2021.wmt-1.1>.
46. Kocmi, T. et al. To ship or not to ship: an extensive evaluation of automatic metrics for machine translation. In *Proc. Sixth Conference on Machine Translation* (eds Barrault, L. et al.) 478–494 (ACL, 2021).
47. Licht, D. et al. Consistent human evaluation of machine translation across language pairs. In *Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas* Vol. 1, 309–321 (Association for Machine Translation in the Americas, 2022).
48. Agirre, E. et al. SemEval-2012 task 6: a pilot on semantic textual similarity. In *Proc. *SEM 2012: The First Joint Conference on Lexical and Computational Semantics* Vols 1–2 (eds Voglre, E. et al.) 385–393 (ACL, 2012).
49. Kusters, R. et al. Interdisciplinary research in artificial intelligence: Challenges and opportunities. *Front. Big Data* **3**, 577974 (2020).
50. Wang, S., Cooper, N., Eby, M. & Jo, E. S. From human-centered to social-centered artificial intelligence: assessing ChatGPT's impact through disruptive events. Preprint at <https://arxiv.org/abs/2306.00227> (2023).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© Meta 2024

NLLB Team

Marta R. Costa-jussà^{1,2}, James Cross², Onur Çelebi¹, Maha Elbayad³, Kenneth Heafield⁴, Kevin Heffernan⁴, Elaha Kalbassi³, Janice Lam³, Daniel Licht³, Jean Maillard³, Anna Sun³, Skyler Wang^{3,5}, Guillaume Wenzek¹, Al Youngblood³, Bapi Akula³, Loic Barrault¹, Gabriel Mejia Gonzalez³, Prangthip Hansanti³, John Hoffman³, Smerley Jarrett³, Kaushik Ram Sadagopan³, Dirk Rowe³, Shannon Spruit¹, Chau Tran³, Pierre Andrews¹, Necip Fazil Ayan³, Shruti Bhosale³, Sergey Edunov³, Angela Fan³, Cynthia Gao³, Vedanuj Goswami³, Francisco Guzmán³, Philipp Koehn^{2,6}, Alexandre Mourachko¹, Christophe Ropers³, Safiyyah Saleem², Holger Schwenk¹ & Jeff Wang³

¹Foundational AI Research (FAIR), Meta, Paris, France. ²Foundational AI Research (FAIR), Meta, New York, NY, USA. ³Foundational AI Research (FAIR), Meta, Menlo Park, CA, USA. ⁴Foundational AI Research (FAIR), Meta, London, UK. ⁵University of California, Berkeley, CA, USA. ⁶Johns Hopkins University, Baltimore, MD, USA.

Methods

Data

This section describes the steps taken to design our language identification system and bitext mining protocol.

Language identification. To train language identification models, we used fasttext^{33,51}, which has been widely used for text classification tasks because of its simplicity and speed. We embedded character-level n -grams from the input text and leveraged a multiclass linear classifier on top. The lightweight nature of fasttext enables our LID models to handle web-scale data. Furthermore, a linear model has the benefit of being easily explainable, allowing us to trace any classification error back to its root cause. This is instrumental in addressing common pitfalls that arise when detecting language on web corpora³².

Classifier design. We experimented with two different designs. First, we used a combination of multiple binary classifiers in which the final decision was obtained by selecting the language with the highest score after applying a threshold. We applied threshold optimization so that when the confidence of a classifier is low, the corresponding language is not considered for the final decision. A sentence was filtered out if none of the classifiers surpassed its threshold. Second, we built a multiclass classifier using softmax over all possible languages. In this case, the threshold optimization is done after the softmax.

Our results directed us to focus on the second approach, which offers several advantages. First, changing the threshold for one language did not affect the performance of the other (which is not true in the first setting). Second, this approach generalizes better to out-of-domain data, which is our primary use case (Wikipedia \rightarrow web data). Finally, a single classifier has the added benefit of being computationally simpler, thus streamlining the language identification process.

Training data and handling massive class imbalance. We used publicly available datasets to train our LID system, partially covering our languages of interest. The public datasets deployed were mostly built from web pages such as CommonCrawl. We then supplemented these with NLLB-Seed data (Supplementary Information B) for any missing languages. However, this supplementation is insufficient in ensuring balance in the raw training data^{32,30}. For example, English alone represents 10.1% of our training data, whereas Minangkabau (Latin script) represents only 0.06%. Following ref. 10, we experimented with multiple settings of temperature upsampling for underrepresented languages, in which sentences from a language l representing p_l per cent of the data set are sampled proportionally to $p_l^{1/T}$. Optimal performance was obtained at $1/T = 0.3$ (for more details, see section 5.1 of ref. 34).

Training parameters. Our best-performing model was trained with softmax loss over two epochs with a learning rate of 0.8 and embeddings with 256 dimensions. We discarded words with less than a thousand occurrences after upsampling and selecting a minimum and maximum character n -gram length of two and five, respectively (which were assigned a slot in buckets of size 1,000,000). (In fasttext, we refer to ‘word’ when it is separated by spaces. When it is a non-segmenting language, there is only one ‘word’ for the whole sentence (and we take character n -grams)). All hyperparameters were tuned on FLORES-200 dev (see section 5.1.2 of ref. 34).

Improving LID with linguistic analysis. Language identification is a challenging task in which numerous failure modes exist, often exacerbated by the gaps between the clean data on which LID models are trained and noisy data on which LID models are applied. In other words, LID models trained in a supervised manner on fluently written sentences may have difficulty identifying grammatically incorrect and incomplete strings extracted from the web. Furthermore, models can easily learn spurious correlations that are not meaningful for the task itself. Given these challenges, we collaborated closely with a team of linguists throughout different stages of LID development to identify

proper focus areas, mitigate issues and explore solutions (see section 5.1.3 of ref. 34).

Bitext mining. The overall approach for bitext mining focused on starting with a massively multilingual sentence encoder teacher model and adapting it to several different low-resource student models. This approach enabled us to add low-resource languages without competing with high-resource languages for capacity. Doing so circumvents the need to retrain the entire model from scratch while maintaining compatibility with the multilingual embedding spaces for subsequent mining. Extended data Fig. 1 summarizes the overall architecture of the teacher–student approach. The teacher, LASER2, is an improved version of the open-source LASER encoder (<https://github.com/facebookresearch/LASER>). The original training procedure³⁶ was adapted to include SentencePiece tokenization (including a vocabulary of 7,000 tokens) and the upsampling of low-resource languages.

The architecture of the five-layer BiLSTM encoder and the max pooling method to obtain sentence embeddings were left unchanged. The training was then performed on the same 93 languages with public resources obtained from OPUS⁵². See ref. 36 for details on the original LASER training procedure. Training of the students followed the approach described in greater detail in ref. 21, summarized below:

- students specialized in one language or several similar languages;
- students were randomly initialized because we wanted to handle low-resource language for which we did not have a pre-trained language model;
- students may have a dedicated SentencePiece vocabulary different from the teacher to better accommodate scripts and tokens in the student languages;
- as we used cosine distance for bitext mining (Fig. 1), students learnt to minimize the cosine loss with the teacher;
- students can have an MLM loss to leverage student language monolingual data (Fig. 1).

Training parameters. Our student encoders used a 12-layer transformer with a hidden size of 1,024, four attention heads, and around 250 million parameters. All students were trained with available bitexts in their respective language, complemented by 2 million sentences of English/English and English/Spanish. The motivation behind this approach is to anchor the students to the English embedding space, increasing robustness by including English/Spanish bitexts from CCMatrix and allowing for the joint learning of new languages. This technique is particularly useful when only limited amounts of bitexts are available to train the students. Teacher–student training was performed on 16 GPUs, the ADAM optimizer, a learning rate of 0.0005 and a batch size of 10,000. We trained student encoders for 148 languages and named these models LASER3.

Proxy metric for new encoders. Mined bitexts were subsequently used to improve translation quality for the languages of NLLB-200. However, mining and NMT training are computationally expensive, and it is intractable to perform this evaluation systematically for many different sentence encoder variants. As an evaluation proxy, we used a mining-based multilingual similarity search error rate, referred to here as xsim. In contrast to cosine accuracy, which aligns embeddings based on the highest cosine score, xsim aligns source and target embeddings based on the highest margin score, which has been shown to be beneficial in mining⁵³. The margin-based score is defined as

$$\text{score}(x, y) = \text{margin} \left(\cos(x, y), \sum_{z \in NN_k(x)} \frac{\cos(x, z)}{2k} + \sum_{v \in NN_k(y)} \frac{\cos(y, v)}{2k} \right) \quad (1)$$

Article

where x and y are the source and target sentences, and $NN_k(x)$ denotes the k nearest neighbours of x in the other language. We set k to 4. All xsim results are calculated on FLORES-200 devtest, using the ratio margin, where $\text{margin}(a, b) = a/b$. Moreover, all scores are calculated for translations into English (that is, $\text{xxx} \rightarrow \text{eng}$). English is encoded by the teacher, and the other language is encoded by the LASER3 student. To facilitate further research using xsim, we also provide this evaluation method as an open-source resource (<https://github.com/facebookresearch/LASER/>).

End-to-end encoder evaluation. Once we had identified the best sentence encoder for each language using the xsim scores, we performed mining, added the mined data to the existing bitexts and trained a bilingual NMT system. Initial experiments indicated that a threshold on the margin of 1.06 seems to be the best compromise between precision and recall for most languages. For these NMT baselines, we do not apply extra filtering on the bitexts and leave this to the training procedure of our massively multilingual NMT system.

We did not attempt to optimize the architecture and parameters of the bilingual NMT systems to the characteristics of each language pair but used the same architecture for all. Therefore, the reported results should not be interpreted as the best possible ones given the available resources—they are mainly provided to validate the mined bitexts. We used a 12-layer encoder and decoder and trained for 100 epochs. Moreover, we looked for the best performance on the FLORES-200 development set and report detokenized BLEU on the FLORES-200 devtest.

Modelling

In this section, we first describe the multilingual machine translation task setup, which includes tokenization and base model architecture. Then, we outline how we leveraged conditional computation for massively multilingual machine translation with EOM regulation and our Curriculum Learning (CL) strategy for low-resource languages.

Task setup. We modelled multilingual NMT as a sequence-to-sequence task, in which we conditioned on an input sequence in the source language with an encoder and generated the output sequence in the expected target language with a decoder⁵⁴. With the source sentence S , source language ℓ_s , and target language ℓ_t in hand, we trained to maximize the probability of the translation in the target language T —that is, $P(T|S, \ell_s, \ell_t)$. Below, we discuss details of the (1) tokenization of the text sequences in the source and target languages; and (2) model architecture with the input and output designed specifically for multilingual machine translation. For further details on the task setup, such as the amount of training data per language pair, please refer to Supplementary Information F or section 8 of ref. 34.

Segmentation with SentencePiece. To tokenize our text sequences, we trained a single SentencePiece model (SPM)⁵⁵ for all languages. We sampled a total of 100 million sentences from primary bitext data. To ensure low-resource languages are well-represented in the vocabulary, we downsampled high-resource and upsampled low-resource languages with a sampling temperature of five (ref. 10). Notably, vocabulary size is an important hyperparameter in multilingual translation models involving low-resource languages^{56–58}. The vocabulary size of our trained SPM model is 256,000. Such a large vocabulary ensures adequate representation across the wide spectrum of languages we support.

Model architecture. Our sequence-to-sequence multilingual machine translation model is based on the transformer encoder–decoder architecture⁵⁹. The encoder transforms the source token sequence into a sequence of token embeddings. Then, the decoder attends to the encoder output and autoregressively generates the target sentence token by token. More precisely, the encoder takes the sequence of tokens $W = (w_1, \dots, w_s)$ and the source language ℓ_s , and produces a

sequence of embeddings $H = (h_1, \dots, h_s)$, which are then provided to the decoder with the target language ℓ_t to produce the target tokens $V = (v_1, \dots, v_T)$ sequentially. In sum,

$$H = \text{encoder}(W, \ell_s), \quad (2)$$

$$\forall i \in [1, \dots, T], v_{i+1} = \text{decoder}(H, \ell_t, v_1, \dots, v_i). \quad (3)$$

Note that we prefixed the source sequence with the source language, as opposed to the target language, as done in previous work^{10,60}. We did so because we prioritized optimizing the zero-shot performance of our model on any pair of 200 languages at a minor cost to supervised performance. Empirically, we find zero-shot performance to be negatively affected when conditioning the encoder on the target language. When the source is conditioned on only the source language, the encoder generalizes better to pairs of source and target languages not encountered during training⁷.

Conditional computation for multilingual machine translation.

A massively multilingual translation (MMT) model uses the same shared model capacity to train on several translation directions simultaneously. While doing so can lead to beneficial cross-lingual transfer between related languages, it can also add to the risk of interference between unrelated languages^{1,61}. MoE models are a type of conditional computational models^{62,63} that activate a subset of model parameters per input, as opposed to dense models that activate all model parameters per input. MoE models unlock marked representational capacity while maintaining the same inference and training efficiencies in terms of FLOPs compared with the core dense architecture.

However, as we increase the model capacity and the computational cost per update, the propensity for low or very low-resource languages to overfit increases, thus causing performance to deteriorate. In this section, we examine how we can use Sparsely Gated Mixture of Experts models^{2–7} to achieve a more optimal trade-off between cross-lingual transfer and interference and improve performance for low-resource languages.

Sparsely gated mixture of experts. To build our MoE models, we substitute a quarter of the encoder and decoder feed-forward network layers with MoE layers, each with E distinct experts. We followed the Top- k -Gating algorithm in ref. 4 and dispatched each token to at most $k = 2$ experts. For more details on the training of MoE models, see Supplementary Information E.

Expert output masking. In this proposed regularization strategy, we masked the expert output for a random fraction (p_{com}) of the input tokens. For input tokens with dropped expert outputs, the first and/or second expert is effectively skipped. As shown in the second panel of Extended data Fig. 2, we masked both experts for the first token (x_1 in red), chose not to mask any of the expert outputs for the second token (x_2 in blue) and in the final scenario, masked only one expert for the last token (x_3 in green).

Curriculum learning for MMT. Orthogonal to model-side regularization methods such as dropout, we explored regularizing MMT models by means of CL. We proposed starting training with high-resource pairs first, then introducing low-resource pairs—prone to overfitting—in later phases. To derive the phases of the curriculum, we first trained a vanilla MoE model (without CL), followed by partitioning the translation directions into n bins $\{b_1, \dots, b_n\}$. If T is the total number of training updates, we introduced each bin b_i after $T - k_i$ updates. We based when $(k_i)_i$ and what $(b_i)_i$ directions to add at every phase of the step when we observed a language pair starting to overfit. Review the step-based CL algorithm in ref. 64 for more on how the directions are partitioned. See Supplementary Information E.2 for the list of directions added at each stage.

Evaluations

Automatic evaluation. Many automatic translation quality assessment metrics exist, including model-based ones such as COMET⁶⁵ and BLEURT⁶⁶. Although model-based metrics have shown better correlation with human judgement in recent metrics shared tasks of the WMT⁴³, they require training and are not easily extendable to a large set of low-resource languages. In this work, we rely on BLEU (and a variant of it) and chrF++. Both measures draw on the idea that translation quality can be quantified based on how similar a machine translation output is compared with that produced by a human translator.

BLEU and spBLEU. The BLEU score⁴⁴ has been the standard metric for machine translation evaluation since its inception two decades ago. It measures the overlap between machine and human translations by combining the precision of 1-grams to 4-grams with a brevity penalty. The main disadvantage of BLEU is that it is tokenization-dependent. Efforts such as sacrebleu⁶⁷ have taken strides towards standardization, supporting the use of community-standard tokenizers under the hood. However, these tokenizers do not extend to many languages. Reference 41 proposes spBLEU, a BLEU metric based on a standardized SentencePiece model (SPM) covering 101 languages, released alongside FLORES-101. In this work, we provide SPM-200 along with FLORES-200 to enable the measurement of spBLEU. (Our analyses demonstrate that there are minor differences between SPM-200 from FLORES-200 and SPM-100 from FLORES-101 when measuring on the FLORES-101 languages. The major advantage of SPM-200 is that it covers 200 languages. More details on SPM-200 are reported in section 8.1.1 of ref. 34).

chrF++. The chrF++ score³⁸ overcomes the limitation of the BLEU score, which requires that a sentence can be broken up into word tokens. However, some languages, such as Chinese or Thai, do not use spaces to separate words, and word segmentation tools may not be readily available. There is also a concern about highly agglutinative languages in which BLEU fails to assign any credit to morphological variants. chrF++ overcomes these weaknesses by basing the overlap calculation on character-level n -grams F -score (n ranging from 1 to 6) and complementing with word unigrams and bi-grams. In this work, we primarily evaluated using chrF++ using the settings from sacrebleu. However, when comparing with other published work, we used BLEU and spBLEU where appropriate.

Human evaluation methodology. When building machine translation systems for thousands of different language pairs, a core question is which pairs reach certain levels of quality. Therefore, we needed meaningful scores that are comparable across language pairs.

XSTS evaluation protocol. We adapted the recently proposed XSTS methodology⁴⁸. In short, XSTS is a human evaluation protocol focusing on meaning preservation above fluency. See details on this protocol in Supplementary Information F. For low-resource languages, translations are usually of poorer quality, and so we focused more on usable (that is, meaning-preserving) translations, even if they are not fully fluent. Compared with Direct Assessment⁶⁸ with a 5-point scale (the original direct assessment uses a 100-point scale), it is found that XSTS yields higher inter-annotator agreement⁴⁷. XSTS rates each source sentence and its machine translation on a 5-point scale, in which 1 is the lowest and 5 is the highest.

Calibration set. To enable meaningful scores comparable across language pairs, we asked each evaluator to provide assessments using the XSTS scale on precisely the same set of sentence pairs. This aims to identify annotators who have a systematic tendency to be more harsh or generous in their scoring and correct for this effect. The calibration set consists of the machine translation output paired with the reference translation only in English. Based on how evaluators used the XSTS scale on this calibration set, we adjusted their raw scores on the actual evaluation task to ensure consistency across evaluators. Although this monolingual calibration task does not precisely mimic the bilingual

XSTS task, it is a reasonable first approximation and has been shown to increase the correlation between human and automatic metrics primarily by reducing one source of ‘noise’ in the human evaluations—the lack of score calibration between annotators.

Obtaining aggregated human quality metrics from multiple studies. To obtain an aggregate human quality metric for each language direction in an evaluation study, we take the majority XSTS score (that is, mean–median score) for each sentence and average these majority scores over all evaluated sentences. In a given study, the aggregate human evaluation score for any translation direction $l_s \rightarrow l_t$ is

$$H_{l_s \rightarrow l_t} = \frac{1}{|T_{l_s \rightarrow l_t}|} \sum_{(S, T) \in T_{l_s \rightarrow l_t}} \text{median}\{X_{l_s \rightarrow l_t, i}(S, T) | 1 \leq i \leq M_{l_s \rightarrow l_t}\}, \quad (4)$$

where l_s and l_t denote the source language and the target language, respectively; $X_{l_s \rightarrow l_t, i}(S, T)$ denotes the XSTS score of the i th evaluator who evaluates sentences in a given translation direction $l_s \rightarrow l_t$ for a source sentence S and a target sentence T ; $M_{l_s \rightarrow l_t}$ denotes the total number of evaluators who evaluate the (source, translation) sentence pair (S, T) for translation direction $l_s \rightarrow l_t$; $T_{l_s \rightarrow l_t} = \{(S_{l_s \rightarrow l_t, k}, T_{l_s \rightarrow l_t, k}) | 1 \leq k \leq N_{l_s \rightarrow l_t}\}$ is the set of $N_{l_s \rightarrow l_t}$ (source, translation) sentence pairs being evaluated for translation direction $l_s \rightarrow l_t$.

Every evaluator in a given study s is also asked to provide ratings for all or parts of a calibration set $C_s = \{(S_{s, k}, T_{s, k}) | 1 \leq k \leq K_s\}$. $S_{s, k}$ denotes the k th source sentence in the calibration set for an evaluation study; $T_{s, k}$ denotes the translated sentence corresponding to $S_{s, k}$; and $K_s = |C_s|$ is the number of sentence pairs in the calibration set for an evaluation study.

For each language direction evaluated in a study, we obtained the majority score on the calibration set as follows:

$$C_{l_s \rightarrow l_t}^{(s)} = \frac{1}{|C_s|} \sum_{(S, T) \in C_s} \text{median}\{X_{l_s \rightarrow l_t, i}^{(s)}(S, T) | 1 \leq i \leq M_{l_s \rightarrow l_t}^{(s)}\}, \quad (5)$$

where $X_{l_s \rightarrow l_t, i}^{(s)}(S, T)$ denotes the XSTS score provided by the i th evaluator, for the language direction $l_s \rightarrow l_t$, in study s , for a given source sentence S and a translated sentence T , in the calibration set C_s of the study.

To obtain aggregated calibrated XSTS scores on the language direction level, we explored several different calibration methodologies. None of the calibration methods we investigated showed a marked difference in correlation with automated scores, and all calibration methodologies we explored provided superior correlation compared with uncalibrated XSTS scores. For more details on these calibration methodologies, see section 7.2 of ref. 34.

Added toxicity detection for 200 languages. To enable toxicity detection at scale, we used a detector based on word lists. In this section, we provide more details about our toxicity definition and describe the detector (ETOX) and associated word lists.

Toxic content. Owing to the subjective nature of toxicity, definitions of toxic language can vary. We included items that are commonly referred to as vulgar or profane language. (Note that vulgar or profane language is not always necessarily toxic. Some common slang, for instance, may be considered vulgar but is not necessarily toxic). Moreover, we also included items associated with depictions of pornographic content or sexual acts, some frequently used hate speech expressions and some expressions tied to bullying. We also included items, vulgar or not, referring to body parts that are commonly associated with sexual practices.

The ETOX detector. We started with the assumption that general-purpose machine translation systems should remain faithful to the source content and not add any toxic elements during the translation process. We define toxic elements as word tokens or short phrases present in our lists. ETOX identifies added toxicity using the following two

Article

criteria: number of toxic items and matched or non-matched toxicity. A toxic item is considered detected if it is present in a line and surrounded by spaces or the start or end of a line. ETOX tracks the number of unique toxic items found in a line but does not count a phrase again if it has multiple occurrences. Matched toxicity indicates that the number of toxic items is the same in both the source and the translated content (that is, no added toxicity). Added toxicity is an instance of non-matched toxicity in which more toxic items are found in the translation output than in the source. For non-segmenting languages or some languages that use complex diacritics, space tokenization is insufficient to distinguish words from one another. In those cases, we used SentencePiece tokenization of both the sentence and toxicity word list.

Toxicity-200 lists. Lists are based on professional translations from English, which were then heuristically adapted by linguists to better serve the target language. As toxicity is culturally sensitive, attempting to find equivalents in a largely multilingual setting constitutes a challenge when starting from one source language. To address this issue, translators were allowed to forgo translating some of the source items and add more culturally relevant items.

In the initial release of the Toxicity-200 lists, the average number of items in a toxicity detection list was 271 entries, whereas the median number of entries was 143. The latter may be a better measure of central tendency than the mean average, given that languages with a rich inflectional morphology constitute extreme outliers (for example, the Czech list had 2,534 entries and the Polish list 2,004). The shortest list had 36 entries, and the longest 6,078.

Data availability

All data generated and described in the Article and its Supplementary Information are available at GitHub (<https://github.com/facebookresearch/fairseq/tree/nllb>)⁶⁹ as follows. The FLORES-200 dataset contains human-translated evaluation data in 204 languages. The NLLB-Seed database contains human-translation seed training data in 39 languages (Supplementary Information I). The NLLB-MD database contains human-translated seed data in different domains in six languages to assess generalization (Supplementary Information J). The Toxicity-200 database contains wordlists to detect toxicity in 200 languages. Mined bitext database contains publicly available web data for 148 English-centric and 1,465 non-English-centric language pairs. Publicly available data used to train NLLB models with references to download the data are listed in Supplementary Table 2.

Code availability

To make our work available to the community, we provide the following models and supporting code as resources freely available for non-commercial use, available at GitHub (<https://github.com/facebookresearch/fairseq/tree/nllb>)⁶⁹ as follows. The translation models cover 200 languages; the NLLB models come in multiple sizes (54.5B MoE, 3.3B and 1.3B Dense, and 1.3B and 600M distilled). The language identification models contain more than 200 languages. LASER3 comprises sentence encoders for identifying aligned bitext for 148 languages. Stopes consists of a data-mining library that can be used to process and clean monolingual data, followed by the creation of aligned bitext. Scripts to recreate our training data and training and generation scripts to reproduce our models are also included.

51. Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T. Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguist.* **5**, 135–146 (2017).
52. Tiedemann, J. Parallel data, tools and interfaces in OPUS. In *Proc. Eighth International Conference on Language Resources and Evaluation* (eds Calzolari, N. et al.) 2214–2218 (ACL, 2012).
53. Artetxe, M. & Schwenk, H. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proc. 57th Annual Meeting of the Association for Computational Linguistics* (eds Korhonen, A.) 3197–3203 (ACL, 2019).

54. Bahdanau, D., Cho, K. H. & Bengio, Y. Neural machine translation by jointly learning to align and translate. In *Proc. of the 3rd International Conference on Learning Representations* (ICLR, 2015).
55. Kudo, T. & Richardson, J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proc. 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018* (eds Blanco, E. & Lu, W.) 66–71 (ACL, 2018); <https://doi.org/10.18653/v1/d18-2012>.
56. Gu, J., Hassan, H., Devlin, J. & Li, V. O. Universal Neural Machine Translation for Extremely Low Resource Languages. In *Proc. 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies Vol. 1* (eds Walker, M. et al.) 344–354 (ACL, 2018); <https://aclanthology.org/N18-1032>.
57. Wang, X., Pham, H., Arthur, P. & Neubig, G. Multilingual neural machine translation with soft decoupled encoding. Preprint at <https://arxiv.org/abs/1902.03499> (2019).
58. Rajab, J. Effect of tokenisation strategies for low-resourced Southern African languages. In *3rd Workshop on African Natural Language Processing* (ICLR, 2022).
59. Vaswani, A. et al. Attention is all you need. In *Proc. 31st Conference on Neural Information Processing Systems* 5998–6008 (NIPS, 2017).
60. Johnson, M. et al. Google's multilingual neural machine translation system: enabling zero-shot translation. *Trans. Assoc. Comput. Linguist.* **5**, 339–351 (2017).
61. Conneau, A. et al. Unsupervised cross-lingual representation learning at scale. In *Proc. 58th Annual Meeting of the Association for Computational Linguistics* (eds Jurafsky, D. et al.) 8440–8451 (ACL, 2020).
62. Bengio, Y., Léonard, N. & Courville, A. C. Estimating or propagating gradients through stochastic neurons for conditional computation. Preprint at <http://arxiv.org/abs/1308.3432> (2013).
63. Almahairi, A. et al. Dynamic capacity networks. In *Proc. 33rd International Conference on International Conference on Machine Learning* Vol. 48, 2091–2100 (PMLR, 2016).
64. Elbayad, M., Sun, A. & Bhosale, S. Fixing MoE over-fitting on low-resource languages in multilingual machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023* (eds Rogers, A. et al.) 14237–14253 (ACL, 2023); <https://aclanthology.org/2023.findings-acl.897>.
65. Rei, R., Stewart, C., Farinha, A. C. & Lavie, A. COMET: a neural framework for MT evaluation. In *Proc. 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (eds Webber, B. et al.) 2685–2702 (ACL, 2020).
66. Sellam, T., Das, D. & Parikh, A. BLEURT: learning robust metrics for text generation. In *Proc. 58th Annual Meeting of the Association for Computational Linguistics* (eds Jurafsky, D. et al.) 7881–7892 (ACL, 2020).
67. Post, M. A Call for Clarity in Reporting BLEU Scores. In *Proc. Third Conference on Machine Translation: Research Papers* (eds Bojar, O. et al.) 186–191 (ACL, 2018); <https://aclanthology.org/W18-6319>.
68. Graham, Y., Baldwin, T., Moffat, A. & Zobel, J. Continuous measurement scales in human evaluation of machine translation. In *Proc. 7th Linguistic Annotation Workshop and Interoperability with Discourse* 33–41 (eds Graham, Y. et al.) (ACL, 2013).
69. NLLB Team et al. No Language Left Behind: scaling human-centered machine translation. *GitHub* <https://github.com/facebookresearch/fairseq/tree/nllb> (2022).

Acknowledgements We thank the following interns for their contributions to the project: C. Baziotis, D. Dua, A. Guo, O. Ignat, A. Kamran, T. Mohiuddin, A. N. Rubungo, S. Sun, S. Tan, H. Xu, S. Wu and Y. Zhang. We are grateful to all the Wikimedia Foundation staff and volunteers who worked with us and provided helpful feedback on our project. We thank V. Chaudhary for help with the data pipeline; E. Grave for his help in scaling fasttext to all FLORES-200 languages; M. Diab for her work on XSTS; L. Specia for her feedback on toxicity and XSTS; J. Ferrando and C. Escolano for their help in using the ALTI+ method; G. Chang, C.-J. Wu and R. Raghavendra for helping us to compute the CO₂ cost of training our models; A. Sridhar for helping with FSDP; S. Jeschonek, G. Anantharaman, D. Sarina, J. Colombo, S. Krishnan, D. Kannappan, K. Saladi, V. Pai, A. Yajurvedi and S. Sengupta for their assistance with training infrastructure; K. Johnson for his help with UXR studies and model evaluation; B. O'Horo and J. Kao for their generative insights and guidance; P. Fung, N. Usunier, S. Riedel, S. Sengupta and E. Dinan for their helpful feedback on the paper. We would also like to thank A. Bordes, M. Zannoli and C. Moghbel for their overall support of this project. Finally, we are indebted to the translators, reviewers, human evaluators, linguists, as well as the translation and quality assurance agencies we partnered with, for helping to create FLORES-200, NLLB-Seed, NLLB-MD and Toxicity-200; performing human evaluations; and teaching us about their native languages.

Author contributions B.A., P.A., O.Ç., K. Heafield, K. Heffernan, S.J., H.S. and G.W. contributed to the data workstream of the project, which includes developing tools to facilitate data mining, cleaning and consolidation. L.B., S.B., J.C., M.E., V.G., J.M., K.R.S., A.S. and C.T. conducted research and experiments that gave rise to the models in this work. M.R.C., C.G., J.H., E.K., P.K., D.L., D.R., S. Spruit, S.W. and A.Y. implemented automatic and human evaluations of NLLB, including but not limited to quality, bias and toxicity. G.M.G., P.H., J.L. and C.R. performed all linguistics work in this project. N.F.A., S.E., A.F., F.G., A.M., S.S. and J.W. provided crucial technical and organizational leadership to help materialize this overall project. M.R.C., C.R., M.E. and S.W. prepared the paper for publication.

Competing interests The authors declare no competing interests.

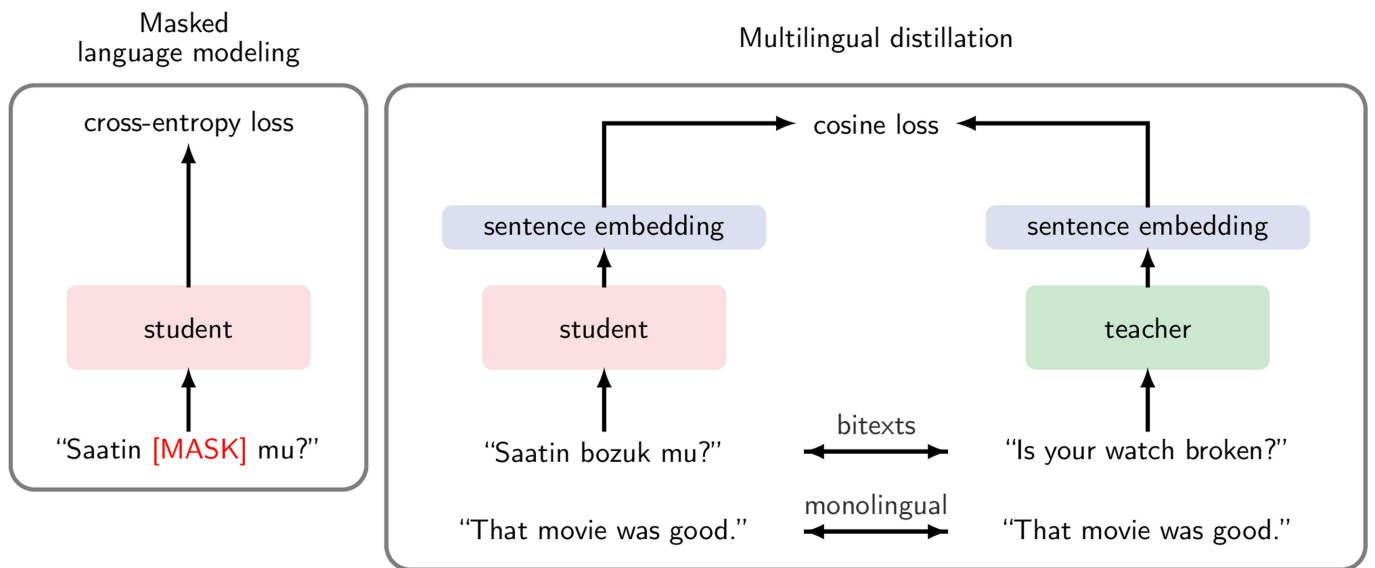
Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41586-024-07335-x>.

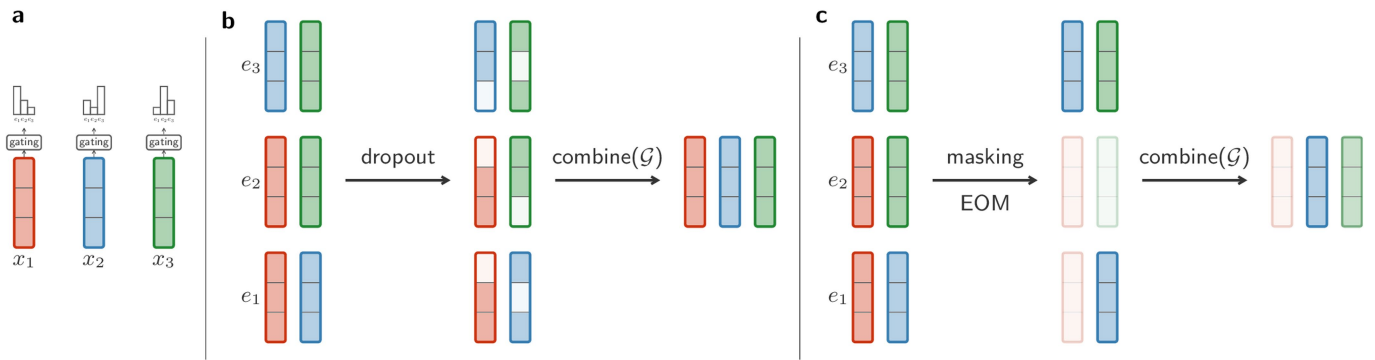
Correspondence and requests for materials should be addressed to Marta R. Costa-jussà.

Peer review information Nature thanks David Adelani, Sunipa Dev and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Peer reviewer reports are available.

Reprints and permissions information is available at <http://www.nature.com/reprints>.



Extended Data Fig. 1 | Architecture of the LASER3 teacher-student approach. See²¹ for more details.



Extended Data Fig. 2 | Illustration of EOM (panel c) in contrast to overall dropout (panel b) for MoE layers. A color represents a token, and each token is dispatched to two experts (Top-2-Gating) depending on the gating decision (panel a). Faded colors correspond to dropped units or masked outputs.