

Code & Corpus at WMT 2026: Checkpoint-Pool Minimum Bayes Risk Decoding for Arabic–Hindi and Arabic–Bangla Translation

Raman Kumar Yadav
IIITDM Kurnool, India
rmnkumrydv@gmail.com

Alapan Kuila
IIITDM Kurnool, India
alapan.cse@iiitk.ac.in

Abstract

We describe the **Code & Corpus** submission to the WMT26 Low-Resource Arabic-Asian Language Translation shared task, covering bidirectional translation between Arabic and two Asian languages: Hindi (Sub-Tasks 1B and 2B) and Bangla (Sub-Tasks 1C and 2C). Our system fine-tunes NLLB-200 model on the official task dataset with a custom trainer that resolves a decoder-input conflict introduced by label smoothing in the M2M100 architecture. At inference time, rather than relying on a single checkpoint, we generate candidate translations from a pool of intermediate checkpoints and apply Minimum Bayes Risk (MBR) decoding with character-level automatic evaluation metric (chrF) as the utility function to select the final hypothesis for each source sentence. On the official leader-board, our primary system ranked 4th of 8 on Hindi→Arabic, 3rd of 6 on Bangla→Arabic, 3rd of 10 on Arabic→Hindi, and 2nd of 7 on Arabic→Bangla, evaluated using various evaluation metrics like BLEU, chrF2, TER, COMET, and mBERT.

1 Introduction

Arabic is spoken by hundreds of millions of people and plays a central role in education, commerce, and international relations, yet machine translation (MT) between Arabic and many Asian languages remains constrained by the scarcity of parallel corpora. The WMT26 Low-Resource Arabic-Asian Language Translation shared task (Laskar et al., 2026) addresses this gap by evaluating systems on bidirectional translation between Arabic and five Asian languages – English, Hindi, Bangla, Indonesian, and Urdu – under a constrained, official-data-only setting. We participate in:

- **Sub-Task 1B:** Hindi → Arabic **1C:** Bangla → Arabic
- **Sub-Task 2B:** Arabic → Hindi **2C:** Arabic → Bangla

These directions are challenging because Arabic (Semitic, right-to-left) is typologically distant from Hindi and Bangla (Indo-Aryan), and none of the three languages forms a high-resource pair with the others in existing corpora.

Prior low-resource shared tasks, notably the WMT Low-Resource Indic Language Translation series (Pal et al., 2023; Pakray et al., 2024), show that fine-tuning large pretrained multilingual models such as NLLB-200 (NLLB Team et al., 2022) or IndicTrans2 (Gala et al., 2023) beats training a Transformer (Vaswani et al., 2017) from scratch in constrained settings. However, most such submissions fine-tune a single checkpoint, select the one with the best development BLEU, and decode it with beam search – discarding information in neighbouring checkpoints and inheriting the length/frequency biases of mode-seeking beam search (Eikema and Aziz, 2020; Müller and Senrich, 2021). We address both issues jointly: rather than discarding intermediate checkpoints once early stopping picks a “best” model, we keep the checkpoint pool and apply MBR decoding (Kumar and Byrne, 2004) over it to pick the final translation.

We fine-tune a single instance of NLLB-200-distilled-600M¹ jointly on all four translation directions – Ar→Hi, Hi→Ar, Ar→Bn, Bn→Ar – by combining the Arabic–Hindi and Arabic–Bangla data into one unified four-way training corpus, rather than training one model per pair. We fix a label-smoothing/decoder-input conflict in the Hugging Face Seq2SeqTrainer with a custom `compute_loss` (Section 3.3), and at inference time ensemble the training checkpoints via MBR decoding (Section 3.4) rather than checkpoint averaging or single-best selection.

¹<https://huggingface.co/facebook/nllb-200-distilled-600M>

2 Related Work

Multilingual pretrained MT models. NLLB-200 (NLLB Team et al., 2022) is a single encoder-decoder Transformer supporting 200 languages, targeting low-resource coverage; its distilled 600M variant retains most of the full model’s quality at a fraction of the cost. IndicTrans2 (Gala et al., 2023) shows dedicated Indic-centric models can outperform general multilingual systems, though it excludes Arabic. The WMT Low-Resource Indic series (Pal et al., 2023; Pakray et al., 2024) is the closest precedent – fine-tuning a pretrained checkpoint on official low-resource data – but is English-centric; WMT26 (Laskar et al., 2026) extends this to Arabic-centric settings for the first time.

Checkpoint combination and MBR decoding.

Averaging the last k checkpoints stabilizes NMT training (Juncys-Dowmunt et al., 2016; Vaswani et al., 2017), but assumes mode-connectivity and still decodes one averaged model with beam search. We instead keep each checkpoint intact and combine *outputs* via MBR decoding: Eikema and Aziz (2020) argue the mode of an NMT model’s distribution – what beam search approximates – is often a poor translation, proposing selection by expected utility against sampled pseudo-references; Eikema and Aziz (2022) and Müller and Sennrich (2021) develop sampling-based approximations reducing length/copy bias; Freitag et al. (2022) show a strong utility metric narrows the gap to human judgment. We adapt this by drawing one hypothesis per checkpoint, rather than multiple samples from one model, as the MBR candidate pool.

3 Methodology

3.1 Data preprocessing

The official task data is released as line-aligned, language-specific plain-text files for each of the `train`, `dev`, and `devtest` splits, separately for the Ar–Hi and Ar–Bn pairs. We convert these into a unified JSONL format storing the source sentence, target sentence, source/target language codes, and a `lang_pair` tag. For each aligned pair we apply: (1) a **length filter** dropping sentences shorter than 5 or longer than 1000 characters per side; (2) a **length-ratio filter** dropping pairs whose character-length ratio exceeds 6.0; (3) a **script filter** requiring at least one Arabic-script codepoint (U+0600–U+06FF) on the Arabic side and a Devanagari codepoint (U+0900–U+097F) for Hindi or Bengali

codepoint (U+0980–U+09FF) for Bangla on the target side; and (4) **deduplication** of exact duplicate pairs.

Surviving pairs are expanded into *both* translation directions, doubling usable records per pair, and the Ar–Hi and Ar–Bn direction-expanded records are *concatenated* into a single four-way corpus (`train_4way.jsonl`, `dev_4way.jsonl`, `devtest_4way.jsonl`) covering Ar→Hi, Hi→Ar, Ar→Bn, Bn→Ar in one file. We fine-tune a single model on this combined corpus rather than training one model per language pair, so encoder and decoder parameters are shared across all four directions. The `devtest` split is held out entirely from training and model selection.

3.2 Base Model and Training Setup

We fine-tune `facebook/nllb-200-distilled-600M`, a 600M-parameter distilled variant of NLLB-200. Language identity is communicated via NLLB’s language-code tokens (Arabic: `arb_Arab`; Hindi: `hin_Deva`; Bangla: `ben_Beng`): the tokenizer’s `src_lang` is set per-example, and `forced_bos_token_id` is set to the target-language token at generation time so decoding is constrained to the correct script. Subword segmentation reuses NLLB-200’s pretrained SentencePiece model (Kudo and Richardson, 2018) unmodified, preserving the pretrained embedding table’s coverage of Arabic, Hindi, and Bangla.

Training uses the Hugging Face `Seq2SeqTrainer` with AdamW (Loshchilov and Hutter, 2019), dynamic padding (`DataCollatorForSeq2Seq`, padded to a multiple of 8 for FP16 efficiency), mixed-precision (FP16) computation, and gradient accumulation to reach an effective batch size of 32 (per-device batch 8×4 accumulation steps). Learning rate is 3×10^{-5} with 300 warmup steps, for up to 15 epochs, with label smoothing of 0.1. Evaluation runs every 200 steps on the development set; training stops early after 4 evaluations without SacreBLEU (Post, 2018) improvement, reloading the best-BLEU checkpoint at the end. Three rolling checkpoints are retained alongside `final_model`; inference uses beam size 5. The full hyperparameter list is given in Table 2 (Section 4).

3.3 Fixing a Label-Smoothing / Decoder-Input Conflict

When `label_smoothing_factor` is non-zero, Hugging Face’s default `Trainer.compute_loss` pops labels out of the input batch *before* calling the model, so the loss can be computed externally from raw logits. However, `M2M100ForConditionalGeneration` (the architecture underlying NLLB-200) only auto-constructs `decoder_input_ids` from labels when labels is present in the call. With labels removed, both `decoder_input_ids` and `decoder_inputs_embeds` remain `None`, and `M2M100Decoder` raises an error.

We resolve this with a custom `NLLBSeq2SeqTrainer` that overrides `compute_loss`: before the parent pops labels, we right-shift the labels ourselves (replacing the `-100` padding sentinel with `pad_token_id`) via `shift_tokens_right`, and inject the result as `decoder_input_ids`. A lightweight `SafeCollator` wrapper additionally strips any stray `decoder_input_ids/decoder_inputs_embeds` keys before and after collation as a defensive safety net.

3.4 Checkpoint-Pool Minimum Bayes Risk Decoding

Rather than decoding only the single checkpoint with the best development BLEU, we retain the pool of checkpoints saved during training (`checkpoint-5000`, `checkpoint-10000`, `checkpoint-12500`, and `final_model`) and use all four at inference time. For each source sentence x , we generate one beam-search hypothesis (beam size 5) from each checkpoint $c \in C$, giving a hypothesis set $H(x) = \{h_c = \text{decode}(x \mid \theta_c) : c \in C\}$. Because different checkpoints correspond to different points along training, they tend to make different, only weakly correlated errors, giving $H(x)$ more useful diversity than k -best beam-search outputs from a single checkpoint.

Following the Monte Carlo MBR formulation of Eikema and Aziz (2020, 2022), we select the hypothesis in $H(x)$ maximizing expected utility when every other hypothesis is treated as a pseudo-reference:

$$\hat{y} = \arg \max_{h \in H(x)} \frac{1}{|H(x)|} \sum_{h' \in H(x)} u(h, h'), \quad (1)$$

where $u(\cdot, \cdot)$ is chrF (Popović, 2015), which is fast to compute at inference scale and correlates reasonably well with human judgment for morphologically rich languages such as Hindi and Bangla. The outer loop runs once per sentence; candidate generation is $O(K)$ and pairwise utility scoring is $O(K^2)$ ($K=4$ checkpoints), so cost per sentence is $O(K \cdot b + K^2)$, dominated by the K beam-search decode calls.

Compared to *single-best-checkpoint* decoding, this recovers value from checkpoints early stopping would otherwise discard. Compared to *parameter averaging* (Junczys-Dowmunt et al., 2016; Vaswani et al., 2017), it makes no assumption that checkpoints are linearly mode-connected – only that they produce plausible, moderately diverse translations. The main limitation is cost: generating from $|C|$ checkpoints and scoring $|H(x)|^2$ utility pairs per sentence is $|C|$ times more expensive than single-checkpoint beam search, practical here only because $|C|$ is small and chrF is cheap.

4 Experimental Setup

4.1 Dataset Statistics and Hyperparameters

Table 1 reports the size of each split, broken down by language pair, as filtered into the unified four-way corpus (Section 3.1). Raw (pre-filter) pair counts are not reported, since they are only recoverable from the original per-pair .txt release and were not available at the time of writing. Table 2 lists the training hyperparameters (Section 3.2), taken directly from our training script (`train_hi_ar.py`).

Split	Kept pairs	Records
Train (Ar–Hi)	20,100	40,200
Dev (Ar–Hi)	1,000	2,000
Devtest (Ar–Hi)	500	1,000
Train (Ar–Bn)	20,500	41,000
Dev (Ar–Bn)	1,000	2,000
Devtest (Ar–Bn)	500	1,000

Table 1: Dataset statistics after filtering. “Records” counts both directions (2× kept pairs).

Hyperparameter	Value
Base model	NLLB-200-distilled-600M
Max src/tgt length	256 tokens
Batch size (per device)	8
Gradient accumulation	4 steps
Effective batch size	32
Learning rate	3×10^{-5}
Warmup steps	300
Max epochs	15
Label smoothing	0.1
Early stopping patience	4 evaluations
Eval / save frequency	every 200 steps
Checkpoints retained	3 (rolling)
Precision	FP16
Beam size (inference)	5

Table 2: Training hyperparameters, as configured in `train_hi_ar.py`.

4.2 Baseline Systems

We did not train a from-scratch baseline; the most directly comparable baselines are the other constrained *primary*-track submissions to the same WMT26 sub-tasks, evaluated by the organizers on the same challenge test set with the same five metrics (Section 5.1). We found no peer-reviewed system reporting MT quality specifically for Arabic $\leftrightarrow$ Hindi or Arabic $\leftrightarrow$ Bangla, consistent with the task’s low-resource framing and the fact that WMT26 (co-located with EMNLP 2026) has not yet taken place. The closest external reference points are NLLB-200’s aggregate non-English-centric FLORES quality (25.5 spBLEU; NLLB Team et al., 2022) and the English-centric WMT Low-Resource Indic series (Pal et al., 2023; Pakray et al., 2024), neither pair-specific. Given the absence of a pair-specific external baseline, the same-task leaderboard comparison in Table 4 is our primary basis for judging competitiveness.

4.3 Evaluation Metrics and Inference

Systems are evaluated with five automatic metrics: BLEU (Post, 2018) (via SacreBLEU), chrF2 (Popović, 2015), TER (Snover et al., 2006) (lower is better), COMET (Rei et al., 2020), and mBERT-based similarity (Devlin et al., 2019); all except TER are higher-is-better. We additionally monitor development BLEU/chrF for early stopping. At inference, the final model runs on each challenge test set in batches of 16 sentences with the source-language tag set and `forced_bos_token_id` set to the target-language tag; we generate one hypothesis per pooled checkpoint (Section 3.4) with beam size 5, then apply chrF-based MBR selection to produce the single output submitted per sentence.

5 Results and Error Analysis

5.1 Official Results and Standing

Table 3 reports the official primary-track evaluation of our submitted system on each sub-task, as released by the WMT26 organizers, computed on the same held-out challenge test set under the constrained (official-data-only) condition. Table 4 situates these numbers without reproducing the full leaderboard, reporting our rank, field size, and BLEU gap to the top-ranked system.

Task	Rank	Field	Δ top (BLEU)
1B	4	8	5.08
1C	3	6	4.14
2B	3	10	3.84
2C	2	7	0.51

Table 4: Leaderboard standing of our primary submission.

Across all four sub-tasks, our system places in the upper half of the field, evidence that joint four-way training with checkpoint-pool MBR decoding is a competitive, low-effort strategy given only official data and a single 600M-parameter base model. Performance is direction-dependent: BLEU/chrF2 are higher translating *out of* Arabic (2B, 2C) than *into* Arabic (1B, 1C), and the gap to the top system is smallest on Arabic $\rightarrow$ Bangla ($\Delta = 0.51$ BLEU) and largest on Hindi $\rightarrow$ Arabic ($\Delta = 5.08$ BLEU) – consistent with Arabic’s templatic morphology making *generation* into Arabic harder than into the more regular Devanagari/Bengali decoding space NLLB-200 was pretrained on.

5.2 Error Analysis: Arabic $\rightarrow$ Hindi

Table 5 shows four representative devtest examples for Arabic $\rightarrow$ Hindi (single-checkpoint output from `final_model`), spanning four error categories. English glosses are given under every line for readability.

The named-entity row is a genuine positive finding rather than a model error: the *official reference* mistransliterates *Al-Sawaha* as “*Sarfar*,” while our hypothesis produces the closer “*Al-Suaha*” – automatic chrF/BLEU here penalizes a translation for being *more* correct than a noisy reference. The second row shows the Arabic adjective مدرّوس (“measured”) transliterated as if it were a proper noun. The third shows a source-attribution phrase (“*according to Anadolu Agency*”) dropped entirely. The fourth shows a case where our hypothesis is a

Task	BLEU	chrF2	TER↓	COMET	mBERT
1B (Hi→Ar)	14.77	47.59	76.50	81.16	82.99
1C (Bn→Ar)	13.79	46.32	77.74	80.39	82.69
2B (Ar→Hi)	25.44	50.40	66.08	69.83	84.44
2C (Ar→Bn)	16.59	49.80	74.28	77.97	83.01

Table 3: Official primary-submission results across all four sub-tasks.

faithful, literal translation, but the human reference adds an unsourced closing sentence not present in the Arabic – a stylistic elaboration that lowers chrF despite no error in the hypothesis. Together, these point to named-entity/loanword handling, source-attribution preservation, and reference quality itself as dominant sources of apparent error, more than genuine word-order or fluency failures.

6 Contribution and Future Work

We presented the Code & Corpus submission to WMT26 Low-Resource Arabic-Asian Translation, fine-tuning a single NLLB-200-distilled-600M jointly on all four Arabic–Hindi/Bangla directions via a unified four-way corpus. Our two contributions are: (1) a custom trainer fixing a decoder-input conflict between label smoothing and M2M100 (Section 3.3), and (2) checkpoint-pool MBR decoding (Section 3.4), reusing intermediate checkpoints as a hypothesis pool for chrF-based MBR selection instead of discarding them after early stopping. Our primary systems placed 2nd–4th across all four sub-tasks.

Replacing chrF with a learned metric such as COMET as the MBR utility function (Freitag et al., 2022) may better align selection with the task’s judging metric. Sampling-based MBR (multiple stochastic samples per checkpoint) could enlarge the hypothesis pool without extra training runs. We also plan to explore Hindi transliteration as an auxiliary signal for the Arabic↔Hindi direction to help bridge script differences on rare tokens.

Limitations

Our evaluation relies solely on automatic metrics; we did not conduct human evaluation. The MBR checkpoint pool comes from a single training run rather than multiple seeds, so its diversity reflects training-trajectory diversity rather than independent-model diversity. Constrained-track results are not directly comparable to systems using additional data.

References

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Bryan Eikema and Wilker Aziz. 2020. [Is MAP decoding all you need? the inadequacy of the mode in neural machine translation](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4506–4520, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Bryan Eikema and Wilker Aziz. 2022. [Sampling-based approximations to minimum Bayes risk decoding for neural machine translation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10978–10993, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Markus Freitag, Yaser Al-Onaizan, Katsuhito Sudoh, Yuki Kishimoto, Patrick Fernandes, Marjan Ghazvininejad, and Miguel Domingo. 2022. High quality rather than high model probability: Minimum Bayes risk decoding with neural metrics. *Transactions of the Association for Computational Linguistics*, 10:811–825.
- Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Atul Nawale, Anupama Sujatha, Ratish Pudupully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *arXiv preprint arXiv:2305.16307*.
- Marcin Junczys-Dowmunt, Tomasz Dwojak, and Rico Sennrich. 2016. [The AMU-UEDIN submission to the WMT16 news translation task: Attention-based NMT models as feature functions in phrase-based SMT](#). In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 319–325, Berlin, Germany. Association for Computational Linguistics.
- Taku Kudo and John Richardson. 2018. [SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). In

Category	Source (Arabic)	Reference (Hindi)	Hypothesis (Hindi)
Named entity / reference noise	من يتطلب الأمر هذا أن السواحة ويرى العمل تعزيزي العشرين مجموعة دول الإجماع. ("Al-Sawaha believes this requires G20 countries to strengthen joint action.")	सरफर का मानना है कि इस तरह के मुद्दे के लिए G20 देशों के बीच सहयोग बढ़ाने की आवश्यकता है। ("Sarfar believes...")	अल-सुआहा ने इस बात पर जोर दिया कि इसके लिए जी-20 देशों को सामूहिक प्रयासों को बढ़ावा देने की आवश्यकता है। ("Al-Suaha emphasized...")
Lexical (OOV)	مدرس "ارتفاع عن الوزارة وأعلنت 2026م. الاعام بنه اية الدين لمحفظة" ("The ministry announced a 'measured' rise in the debt portfolio by end-2026.")	मंत्रालय ने... ऋण पोर्टफोलियो में नियोजित वृद्धि का खुलासा किया। ("...a planned rise...")	मंत्रालय ने... ऋण पोर्टफोलियो में "मदरूस" की वृद्धि की घोषणा की है। ("...a 'Madroos' rise" – word left untranslated, treated as a name)
Omission (dropped)	عائلاتهم احتجاجات... وتزوج لولالة وفدا ضده، المستمرة... ("...and, according to Anadolu Agency, fuels their families' ongoing protests against him.")	...और अनादोलु एजेंसी के अनुसार, ...व्यापक वरिष्ठ को बढ़ावा दे रहा है। ("...and, according to Anadolu Agency, is fueling protests...")	...और उनके परिवारों द्वारा उनके खिलाफ चल रहे वरिष्ठ प्रदर्शन को तेज करता है। ("...intensifies ongoing protests..." – attribution dropped)
Reference elaboration	الهندسية السلع صادرات 4.2 إلى 12% بنسبة والإلكترونية دول... مليار ("Engineering and electronics exports rose 12% to \$4.2bn...")	...जो पछिल्ले वर्षों की तुलना में पर्याप्त वृद्धि दर्शाता है... ये आंकड़े अंतरराष्ट्रीय व्यापार के संदर्भ में... (reference adds an editorial closing sentence absent from the source)	इंजीनियरिंग और इलेक्ट्रॉनिक वस्तुओं का निर्यात 12% बढ़कर \$4.2 बिलियन हो गया... (faithful, concise translation; penalized by chrF for not matching the reference's added elaboration)

Table 5: Arabic→Hindi devtest error examples across four categories, with English glosses; underlining marks where the hypothesis diverges from the reference, or where the reference itself is the noisy/expanded element.

- Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Shankar Kumar and Bill Byrne. 2004. Minimum bayes-risk decoding for statistical machine translation. In *Proceedings of the human language technology conference of the North American chapter of the association for computational linguistics: HLT-NAACL 2004*, pages 169–176.
- Sahinur Rahman Laskar, Firoj Alam, Bishwaraj Paul, Irfan Ahmad, Maya Silvi Lydia, and Pankaj Dadure. 2026. WMT26 low-resource Arabic-Asian language translation shared task. <https://www2.statmt.org/wmt26/arabic-asian-mt-task.html>. Accessed 2026-07-27.
- Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.
- Mathias Müller and Rico Sennrich. 2021. [Understanding the properties of minimum Bayes risk decoding in neural machine translation](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 259–272, Online. Association for Computational Linguistics.
- 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. *arXiv preprint arXiv:2207.04672*.
- Partha Pakray, Santanu Pal, Advaita Vetagiri, Reddi Krishna, Arnab Kumar Maji, Sandeep Dash, Lenin Laitonjam, Lyngdoh Sarah, and Riyanka Manna. 2024. Findings of WMT 2024 shared task on low-resource Indic languages translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 654–668.
- Santanu Pal, Partha Pakray, Sahinur Rahman Laskar, Lenin Laitonjam, Vanlalmuansangi Khenglawt, Sunita Warjri, Pankaj Kundan Dadure, and Sandeep Kumar Dash. 2023. Findings of the WMT 2023 shared task on low-resource Indic language translation. In *Proceedings of the Eighth Conference on Machine Translation (WMT)*, pages 682–694.
- Maja Popović. 2015. [chrF: character n-gram F-score for automatic MT evaluation](#). In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. [COMET: A neural framework for MT evaluation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.