

North Small Translate: Advanced Cost-Effective Translation (Cohere CAT+)

Tom Kocmi*

Alexandre Bérard

Phil Blunsom

Samuel Cahyawijaya

Shaun Cassini

Nicholas Frosst

Ona de Gibert

Aidan Gomez

Nithya Govindarajan

Shun Kiyono

Olivia Lasche

Lawrence Rogers

Kelly Marchisio

Nikita Moghe

Yash More

Camila Moran-Hidalgo

Yiyang Nan

Michael Sachs

Trisha Starostina

Daan van Stigt

Spencer Rarrick

Sebastian Vincent

Ivan Zhang

Cohere

*kocmi@cohere.com

Abstract

We present *North Small Translate*, an open-weight, LLM-based machine translation (MT) model with instruction-following capabilities built on the same foundation as Cohere’s *Command A+*, a mixture-of-experts architecture with 25 billion active parameters out of 218 billion total parameters. *North Small Translate* is trained using difficulty sampling to obtain challenging documents and a five-step training protocol combining supervised fine-tuning, direct preference optimization, and online reinforcement learning. We prioritized throughput through a non-reasoning base model and supplemented it with optional agentic capabilities to unlock translation quality gains. *North Small Translate* is trained to perform MT-related tasks, including post-editing and quality estimation, as well as related tasks such as general instruction following. The model achieves top MT performance across 50 languages in the class of models under 1T parameters, with no need to run expensive reasoning at inference time.

1 Introduction

In recent years, machine translation (MT) has almost fully transitioned to large language models (LLMs), which treat cross-lingual generation as a complex instruction-following task. This shift lets models leverage broad context, translate robustly, and adapt to various requests.

We introduce *North Small Translate* (submitted to the WMT General MT shared task as *Cohere CAT+*), our best-performing MT system built upon the same foundation as Cohere’s *Command A+* (Cohere Team, 2026). It is a sparse mixture-of-experts

(MoE) transformer with 25 billion active parameters out of a total of 218 billion parameters.

North Small Translate is built for efficiency. While most LLMs rely on test-time reasoning to achieve state-of-the-art performance, long reasoning traces introduce prohibitive latency and computational overhead for production-level translation. Model throughput is a primary target for practical MT deployment, so we explicitly trained *North Small Translate* as a non-reasoning model. For latency-insensitive applications, our optional *Agentic Translation* framework bridges the potential quality gap caused by the absence of intermediate reasoning steps.

North Small Translate achieves top-tier translation quality through five distinct training steps: (i) Coarse SFT to establish general non-MT instruction-following capabilities and extend the base model’s language coverage; (ii) Fine-grained SFT using highly curated synthetic datasets to improve core MT quality and adjacent tasks like error detection and post-editing, followed by three steps focusing primarily on human-aligned translation quality: (iii) DPO; (iv) Online RL using an LLM-as-a-judge reward; and (v) a minimal final DPO with a reduced learning rate to resolve edge-case regressions introduced in RL.

We participate in the WMT 2026 General MT (Kocmi et al., 2026), Terminology (Charkiewicz et al., 2026), and Automated MT Evaluation shared tasks with our model, *Cohere CAT+*. Our submission uses no specialized search heuristics, inference tricks such as Minimum Bayes Risk decoding (Freitag et al., 2022), or Agentic Translation.¹ One

¹Disclosure of conflict: the main author is also an organizer

	Params total (active)	GEMBA WMT26	xCOMETxl WMT24++	Terminology	Long context	Structured translation
North Small Translate (Agentic)	218B (25B)	84.4	80.7	87.9	47.8	93.3
North Small Translate	218B (25B)	83.6	80.6	89.7	48.9	93.7
Mistral Large 3	675B (41B)	81.6	78.3	91.2	20.0	93.8
Qwen 3.5 397B	397B (17B)	81.6	78.5	87.0	19.5	59.4
DeepL NextGen		81.4	81.8	47.8	56.5	64.5
Gemma 4 31b	31B (31B)	79.5	79.2	92.1	19.4	95.1
Inkling Small	276B (12B)	78.5	78.7	91.1	21.5	91.3
Muse Glimmer 30B	30B (30B)	78.2	77.7	90.1	32.5	91.7
Command A+	218B (25B)	76.5	81.0	68.7	30.5	73.7
GLM 5.2	744B (40B)	76.5	78.8	78.2	28.6	92.1
Mistral Medium 3.5	128B (128B)	75.7	76.7	90.3	19.3	94.2
GPT-OSS 120B	117B (5B)	72.3	76.6	83.8	20.8	93.2
Command A Translate (2025)	111B (111B)	70.0	76.9	56.9	0.0	79.5
Google Translate		68.2	80.8	38.1	21.3	48.8
Nemotron 3 Ultra 550B	550B (55B)	67.1		84.8	19.7	
Qwen 3.6 27B	27B (27B)	61.9		89.2	28.9	

Table 1: Aggregated results of our model against other top systems with under 1T parameters.

domain for General MT contains videos as sources; we used *Cohere Transcribe*² to obtain textual transcripts for English videos, while relying on the officially provided transcripts for Chinese and Czech source data.

2 Training Details

2.1 Model Architecture

Our model is built on top of Command A+ (Cohere Team, 2026), a decoder-only sparse Mixture-of-Experts transformer (Vaswani et al., 2017) with 218B total parameters and 25B active.³

The list of pre-training languages is:

Arabic, Bengali, Czech, Danish, Dutch, English, Filipino, Finnish, French, German, Greek, Hebrew, Hindi, Indonesian, Italian, Japanese, Korean, Malay, Norwegian, Persian, Polish, Portuguese, Punjabi, Romanian, Russian, Simplified Chinese, Spanish, Swedish, Tamil, Telugu, Thai, Traditional Chinese, Turkish, Ukrainian, Urdu, Vietnamese.

The list of languages extended in post-training:

Albanian, Bulgarian, Catalan, Croatian, Estonian, Hungarian, Icelandic, Irish, Latvian, Lithuanian, Maltese, Serbian, Slovak, Slovenian.

2.2 Training Approach

We build on top of findings from our previous models and also our previous Command-A-Translate (Kocmi et al., 2025a), which showcased the im-

pact of the Direct Preference Optimization (DPO) (Rafailov et al., 2023) training step on MT quality.

We use five distinct training steps, each focused on a different aspect:

1. Initial coarse SFT: to cover general instruction-following capabilities and extend language coverage from the 35 pre-training languages to 50 languages. We fine-tune four different models and merge them (Cohere et al., 2025).
2. Fine-grained SFT: focusing on improving the MT quality using primarily synthetic datasets and also improving MT-related tasks such as error detection, terminology or post-editing.
3. DPO: primarily focused on improving quality.
4. Online RL: online RL with GSPO (Zheng et al., 2025) and LLM-as-a-Judge as the reward signal to polish performance and increase adequacy.
5. Light targeted DPO with a reduced learning rate: for fixing remaining problems, especially a few introduced by the RL step.

Table 2 shows the MT improvement from each training step. Each step yielded significant gains except RL, which improved some scores, especially adequacy, but degraded performance for a few low-resource languages because the LLM judge is weak in these languages.

of the WMT General MT shared task.

²<https://cohere.com/blog/transcribe>

³<https://huggingface.co/CohereLabs/command-a-plus-05-2026-bf16>

To fix the aforementioned regression, we introduced a final DPO step with a reduced learning rate and only 5 warm-up steps. For this step, we focused only on a few problematic languages and capabilities.

Step	MT Quality
Coarse SFT	71.3%
Fine-grained SFT	76.2%
DPO	80.1%
Online RL	80.0%
Light DPO	81.8%

Table 2: Training progression on the WMT25 evaluation with GEMBA-ESA averaged over 14 languages.

2.3 Data Preparation

While the main goal is to build a strong MT model, we additionally focused on building a strong instruction-following model that also excels in MT-related tasks, such as post-editing, terminology following, quality estimation, or structured translation.

We processed every translation pair with automated verifiers to remove noisy or malformed text. We enforced structural consistency by requiring source and target segments to have the same number of lines and paragraphs. Additionally, we applied language identification and filtered out instances of unwanted code-switching. Our filtering pipeline also removed segments containing broken Unicode characters and relied on various other internal classifiers to detect and discard anomalies.

2.3.1 Quality of Machine Translation

A primary challenge in curating training corpora for top-performing MT systems is the sparsity of genuinely difficult source material, as we highlighted in Kocmi et al. (2025a). After our initial coarse SFT stage, the model translates without major errors for more than 90% of our training documents. Consequently, randomly sampling source texts would yield a weak learning signal and fail to push the model’s quality boundaries.

We use difficulty sampling to focus our synthetic generation exclusively on problematic instances from which the model can learn. Building upon our previous work (Kocmi et al., 2025a), we extend difficulty sampling by using the initial base model with minimal post-training itself as the baseline model and keep only documents where the

model produces major errors, or its translation quality is lower than the human reference for parallel data. For each source document, we randomly select a high-resource target language and translate the document with an early-stage model. A judge then evaluates the translation for errors. For monolingual data, we classify a source segment as “easy” and remove it if its translation is largely error-free. For parallel data, we deem it “easy” if the translation outperforms the human reference. Filtering easily translated segments concentrates the corpus on difficult texts, directing downstream synthetic-data generation to documents where the model needs the most improvement.

To achieve top MT performance across all supported languages, we construct our training corpora using two distinct data synthesis and refinement strategies, which we call *Best-per-Language Forward Translation* and *Post-Edit Driven Preference Distillation*. Both approaches are designed to maximize the quality and diversity of the training signal without relying on human-annotated parallel data, which we use solely in the first stage of training.

Best-per-Language Forward Translation combines standard forward translation with Best-of-N selection. For each language, we select the top-performing language-specific expert to translate the source documents. To ensure broad multilingual coverage and improve cross-lingual transfer, we randomly assign the target language during generation. While the majority of the source texts are in English, a significant part consists of non-English sources, yielding a substantial volume of direct non-English to non-English translation pairs. A judge retains only candidates scoring above 70 GEMBA-ESA. Scoring details are in Section 2.4, as we use this approach throughout.

Our second strategy, *Post-Edit Driven Preference Distillation*, is the data preparation pipeline’s most critical component and the primary driver of DPO. It aims to generate near-on-policy data offline, creating effective preference pairs that target the model’s weaknesses. The orchestration pipeline operates as follows:

1. Generate a draft translation using the current candidate model.
2. A separate judge model identifies and annotates error spans in the generated draft. If no error is present, the sample is deemed “easy” and dropped from the training data.

3. A third model acts as an automated post-editor, taking the draft and fixing the errors.
4. We iterate steps 2 and 3 until no major error is discovered, or we pass 10 iterations.

This multi-model orchestration yields an aligned preference pair: the candidate model’s original, flawed output is the worse completion and the corrected version the preferred completion. *Because the flawed translation comes directly from the candidate model, the dataset closely approximates on-policy MT data. Offline DPO can thus function with the precision of online reinforcement learning by penalizing the model’s actual failure modes.* This orchestration approach was instrumental in fixing persistent, model-specific errors and driving the most significant quality improvements observed during our DPO stage.

2.3.2 Instruction Following

To give our MT model acceptable general instruction-following abilities and strengthen translation-related tasks and adherence to specific MT instructions, we included non-MT instruction data in the first SFT stage. We used some English-only single-turn datasets used to train Command A+ and multilingual datasets with general-purpose multilingual prompts. We removed reasoning traces from all reused datasets. We reused these data in the second SFT stage with significant downsampling.

2.3.3 Structured Translation

“Structured translation” is the process of translating natural language embedded in data formats such as JSON or HTML without breaking their underlying syntax. It is a critical feature especially for translating webpages, documents or training data.

While the general instruction-following datasets described above contain various formats, in structured translation, we focus solely on JSON and HTML formats. We prepare data from naturally occurring structures or by placing paragraphs or sentences from high-quality MT datasets into diverse structures. The former uses JSON from internal datasets and HTML from web datasets; the latter uses top segments from high-quality sentence corpora. An internal model translates the resulting sources into 35 languages, both field by field and as complete structures. Prompts controlled translation scope, from the entire structure to only a few JSON

keys. Lastly, the translations were filtered for structure retention and for translation quality. On top of SFT, we also created a DPO dataset targeting only specific structured errors that our fine-tuned models exhibit. These included over-copying or parsing issues, with the preferred completions remaining as pristine as possible.

2.3.4 Terminology Translation

Terminology translation is the task of producing a translation that adheres to a glossary of source-target term pairs supplied in the prompt. When a customer has an established vocabulary, such as product names, brand names or domain-specific jargon, it is critical that a translation solution render these consistently and have the ability to use translation memories. We focus on building a preference dataset for this capability.

For each document, an LLM identifies source named entities and their reference renderings, proposes an equally valid alternative target-language rendering for each, and rewrites the reference to use the alternatives while leaving other text unchanged. The rewrite is the preferred completion, the original reference is dispreferred, and the proposed term pairs are given to the model as the glossary. Because the completions differ only in term renderings, the preference signal isolates terminology adherence from general translation quality.

A model that blindly copies every glossary entry would score well on such data. To counter this, half the documents receive negative glossary entries: topically related source-language words and phrases that do *not* occur in the document. These are proposed by an LLM, filtered with a substring check against the source and both completions, and verified by a second LLM pass that also rejects inflected forms, compounds and transliterations. A document never receives more negative entries than genuine ones. The dataset contains preference pairs from English into ten languages: Arabic, German, Spanish, French, Hindi, Italian, Japanese, Korean, Portuguese and Russian. Following Section 2.3.8, each glossary uses a random format: a JSON object, a JSON list of pairs, or one term pair per line with a varying separator; it is attached to a random instruction. When negative entries are present, the instruction comes from a separate set warning that some entries may be irrelevant to the source.

2.3.5 Tone and Idiomatic Translation

Human evaluation in early stages revealed that our model struggled with adhering to an expected tone, formality levels and usage of idioms.

To improve the model’s adherence to formality instructions, we collected social-domain data in eight languages with explicit formality distinctions: Dutch, Italian, Polish, Greek, Japanese, Spanish, Korean, and Portuguese. Each sample was automatically assigned one of three labels using an LLM: *formal*, *informal*, or *ambivalent*. Both grammatical markers and stylistic cues were considered when determining the appropriate formality label.

We then back-translated each target-language sample into English using our internal MT model, producing supervised training triplets of a back-translated English source segment, a target-language segment with an authentic formality level, and an associated formality label. During training, each label was mapped to its corresponding natural-language instruction in the model prompt highlighting the expected formality level.

In order to tackle the idiom translation, we created multilingual synthetic datasets from existing English monolingual idiom datasets. The collected training set contained 50k idiomatic sentences paired with annotated idioms or literal paraphrases.

We translated literal paraphrases from English into 35 languages with our internal MT model. We then scored each translation using GEMBA-ESA (Kocmi and Federmann, 2023) with a strong LLM judge, retaining only examples scoring at least 70. For idioms without literal paraphrases, we used best-of-N: several models generated translations and the LLM judge selected the best. We otherwise followed the same process. Finally, we retained the top 10k candidates per language pair, yielding 512k sentences.

2.3.6 Error Detection and Quality Estimation

Error detection is the task of identifying erroneous spans in a given machine-translated text, while quality estimation assesses the quality of the translated text. Both are evaluated annually by the WMT Automatic Evaluation task (Lavie et al., 2025).

In our preliminary experiments, training on fully human labels alone was only marginally better than the baseline, lagging far behind state-of-the-art models, which may be caused by inconsistency in the human annotations. We therefore focus on

synthetic annotations of actual MT errors only, using a strong judge to assess error spans and scores.

Inspired by Treviso et al. (2024), we used an XML tag scheme to annotate the text. We designed a unified XML tag scheme that can represent both ESA-style tags (Kocmi et al., 2024b) encoding only severity (`<error severity="minor or major">`) and MQM-style tags (Lommel et al., 2014) that additionally carry an error category and subcategory (`<error severity="..." category="accuracy/mistranslation">`).

2.3.7 Post-editing

Post-editing is the process of improving an existing translation to capture inaccuracies in fluency and adequacy. Typically, one provides a tuple containing the source and raw translation, and expects an improved translation from an oracle. The oracle can be a human or an automated translation system or model. In our setup, we use another capable judge LLM to be an oracle.

WMT APE came up with the formulation for Automatic Post-editing, where people were given human post-edits of some unknown MT system and were allowed to train ML models that can perform similar fixes when trained on supervised datasets comprising the original translation, source text and the corrected output (Bojar et al., 2015; Chatterjee et al., 2018). Because human post-edits are scarce and hard to scale, researchers have relied on synthetic LLM edits (Bang et al., 2023; Ki and Carpuat, 2024; Singh et al., 2025). LLMs can also be fine-tuned for more faithful post-editing. We adopt this approach and consider inference-time post-editing useful for translation models.

We use post-editing both to refine training targets and teach our model to repair drafts at inference time. For post-training data, a base internal MT system produces a draft that a stronger LLM post-edits. We score the draft and revision with reference-free GEMBA-ESA (Kocmi and Federmann, 2023) and keep the pair only when the post-edit improves the draft, restricting supervision to major quality gaps. Each retained example has two instruction formats: (a) *Blind* post-editing, which provides the source and raw translation, and (b) *Span-assisted* post-editing, which marks erroneous spans in the original translation with XML severity tags (major or minor) and provides the annotated translation with the source (Lommel et al., 2014; Kocmi et al., 2024b).

Metric	Acc (ties)
GEMBA-ESA Fluency+Adequacy	51.0
GEMBA-ESA	50.6
xCOMETxl	44.9
spBLEU	43.7

Table 3: Metric meta-evaluation comparing LLM-as-a-judge (with GPT 5.4) versus traditional metrics on internal human-labeled translations.

2.3.8 Instruction Diversity

Instruction-tuned MT often relies on static prompts (e.g., "Translate this to [Language]"). Although functional, fixed phrasing causes overfitting. We therefore diversify prompt instructions during training to improve robustness.

We introduce variance across three distinct tiers. First, we seed instructions with curated, manually prepared paraphrases to capture human-written translation requests. Second, we automatically prepare thousands of paraphrased instruction variants for broad lexical and syntactic coverage. Finally, we translate each variant into all supported languages. When preparing the training dataset, we randomly select a prompt, alternating the instruction language between English, the source language, and the target language. Furthermore, we added various augmentations such as merging consecutive sentences into a single line, replacing newlines with random paragraph separators, enclosing the source and/or target text in XML tags or Markdown.

To strengthen instruction adherence, we included a small preference dataset in the final DPO step. It targets terminology, formality, custom formatting or style, and combinations of these constraints. We synthesize it by having an LLM add creative translation guidelines to existing prompts, generating translations with best-of-N, and using a judge focused on guideline adherence to select the best; the worst is the negative example.

2.4 Reliable Evaluation Metric

Reliable evaluation metrics were crucial to model development. We meta-evaluated metrics using internal human annotations for 24k translated segments across 10 languages and the segment-level pairwise accuracy method from Deutsch et al. (2023). As Table 3 shows, confirming Lavie et al. (2025), LLM-as-a-Judge methods significantly outperform traditional string-based metrics such as xComet and spBLEU. Consequently, we focused

on LLM-as-a-judge throughout our work. We also designed an updated GEMBA-ESA (Kocmi and Federmann, 2023) prompt that evaluates fluency and adequacy separately before averaging them. This yields a slightly higher correlation with human judgments while enabling independent diagnosis of translation accuracy and fluency and providing individual error annotations. The full prompt that we used throughout this work with various judges is in Section A.

2.5 Agentic Translation

To maximize inference throughput, we train a non-reasoning model, trading potential quality gains from intermediate reasoning for significantly faster generation. For quality-critical applications, we bridge this gap with *Agentic Translation*. It uses the same underlying model in three roles: a translation agent generates a draft, an evaluation agent identifies and annotates errors, and a post-editing agent corrects the draft using those error spans. By decomposing the translation process into iterative, targeted steps, the agentic approach recovers the performance benefits typically associated with reasoning models.

3 Evaluation and Results

We analyze the performance of our model over MT capabilities and related tasks. We compare it with our previous models, top-performing MT open-weight systems, and leading providers such as Google Translate and DeepL Pro (the full list is in Section 3.1). We evaluate all systems in an identical setup, in a clean zero-shot approach without any inference techniques.

3.1 Benchmark Models

Next to our new model, we evaluate Cohere’s two previous models: Command A Translate (Kocmi et al., 2025a) and general-purpose Command A+ (Cohere Team, 2026). We compare them with a broad selection of top-performing open-weight models below 1T total parameters: Google’s Gemma 4 31B (Gemma Team, 2026); Mistral Large 3 (Mistral AI Team, 2026) and Mistral Medium 3.5 (Mistral AI, 2026); Alibaba’s Qwen3.5 397B (Qwen Team, 2026a) and Qwen3.6 27B (Qwen Team, 2026b); Z.ai’s GLM-5.2 (GLM-5-Team et al., 2026); OpenAI’s GPT-OSS 120B (OpenAI et al., 2025); NVIDIA’s Nemotron 3 Ultra (NVIDIA Research, 2026); Thinking Machines Lab’s Inkling-Small (Thinking

Machines Lab, 2026); and Meta’s Muse Glimmer 30B (Cuenca et al., 2026). In addition, we compare against two leading commercial translation services: Google Translate and DeepL’s NextGen.

We keep the reasoning on for every model that supports reasoning capabilities: specifically, Command A+, Gemma 4 31B, Qwen3.5-397B, Qwen3.6-27B, GLM-5.2, GPT-OSS 120B, and Muse Glimmer 30B.

3.2 Machine Translation Capabilities

Our core focus is MT quality across all supported languages. To honestly evaluate capabilities, we evaluate them on a test set released after we finished building the model: the WMT 2026 General MT test set (Kocmi et al., 2026). This dataset contains four distinct domains and translation modes.

Lavie et al. (2025) showed that traditional metrics such as xComet (Guerreiro et al., 2024) and MetricX (Juraska et al., 2024) no longer correlate well with human evaluation, and instead, strong frontier models excel at evaluation. Our internal evaluation in Section 2.4 agrees. Thus, we use GPT-5.6 Sol as a reference-free judge, asking it to score adequacy and fluency separately and averaging the scores.

WMT26 is a document-level test set, where the number of documents differs for each domain. Building on the recommendation of the original test set, we weight each score by the number of segments in the document. This way, each domain is roughly equally represented with about 180 segments.

The speech domain tests multimodal capabilities, but all reported models are evaluated text to text. We use the provided automatic transcript as source text and give the human-cleaned transcript to the LLM judge to reduce ASR noise.

Table 4 shows that North Small Translate ranks first overall, particularly in European and Latin-script languages. It ranks second for Middle Eastern and Asian languages. Per-language results are in Table 6.

We supplement results with a well-known evaluation in the MT community: WMT24++ (Deutsch et al., 2025). WMT24++ contains English to 55 human-translated languages and dialects. The original source text is from Kocmi et al. (2024a) and covers multiple domains. Each language pair contains 171 documents split into 998 mostly paragraph-level segments. We use the prompt instruction from Deutsch et al. (2025) and we evaluate with

xComet-XL (Guerreiro et al., 2024), a metric widely used for system rankings. The metric is a 3.5B-parameter model fine-tuned on human judgment data.

Table 1 shows that North Small Translate reaches near-top performance. That said, caution is needed; as discussed in Section 3.6, WMT24++ is significantly easier than WMT26 and mostly “solved” with top-performing systems.

3.3 Long-Context Translation

We evaluate long-context translation using a subset of the WMT25 General MT test set (Kocmi et al., 2025b). The English long-context source comprises two documents with approximately 9,900 words divided into two sections. We concatenate them into a single source document, yielding one long-form evaluation instance. As part of the evaluation, we translate it into Egyptian Arabic, Czech, Estonian, Icelandic, Japanese, Korean, Russian, Simplified Chinese, and Ukrainian. The complete document is provided to the model in a single request.

Translations are evaluated against the WMT25 references using the reference-based xCOMET-XL metric (Guerreiro et al., 2024). Since the metric is not able to evaluate a full document, we divide the translations at segment boundaries. xCOMET-XL independently scores each aligned source–hypothesis–reference triple and returns the arithmetic mean over the 85 segments.

Table 1 shows that our model reaches second-best MT quality, outperformed only by DeepL.⁴

3.4 Terminology Translation

We evaluate glossary-conditioned translation using the WMT25 Terminology Translation Task (Semenov et al., 2025). We refer readers to that work for the dataset construction, domains, and official shared-task protocol. Our evaluation covers five translation directions: English to German, Spanish, Russian, and Traditional Chinese, and Traditional Chinese to English.

For the scoring, we use only the proper-terminology condition setup. Each model receives the source text together with a source–target glossary and is instructed to apply the supplied terminology while returning only the translation.

⁴DeepL is a closed-source system accessed by API, so we cannot confirm implementation: for instance, whether it might pre-parse the source text.

	Params total (active)	All languages	Europe	ME + Asia	Latin script	Other scripts
North Small Translate (Agentic)	218B (25B)	84.4	82.5	87.2	83.0	86.5
North Small Translate	218B (25B)	83.6	82.0	86.1	82.4	85.4
Mistral Large 3	675B (41B)	81.6	80.3	83.5	80.3	83.5
Qwen 3.5 397B	397B (17B)	81.6	81.2	82.0	81.0	82.4
DeepL NextGen		81.4	82.3	80.0	82.4	79.9
Gemma 4 31b	31B (31B)	79.5	73.5	88.4	74.9	86.3
Inkling Small	276B (12B)	78.5	77.8	79.6	78.2	79.0
Muse Glimmer 30B	30B (30B)	78.2	76.8	80.3	77.4	79.5
Command A+	218B (25B)	76.5	76.0	77.4	76.0	77.3
GLM 5.2	744B (40B)	76.5	77.5	75.0	77.8	74.6
Mistral Medium 3.5	128B (128B)	75.7	73.6	78.7	73.6	78.7
GPT-OSS 120B	117B (5B)	72.3	70.3	75.2	71.4	73.5
Command A Translate (2025)	111B (111B)	70.0	68.8	71.7	69.6	70.5
Google Translate		68.2	68.0	68.4	68.9	67.1
Nemotron 3 Ultra 550B	550B (55B)	67.1	62.9	73.3	64.6	70.8
Qwen 3.6 27B	27B (27B)	61.9	59.9	64.9	59.9	65.0

Table 4: Machine translation capabilities aggregated over different groups of languages over the WMT26 test set.

We measure the corpus-level terminology success rate: a dictionary pair is successful when its source term occurs in the input and its prescribed target form occurs in the translation. As a final score in Table 1, we micro-average language-pair scores from both WMT25 Terminology Tracks.

3.5 Evaluation of Structured Translation

Because this task is the first of its kind, we built dedicated test suites and metrics. Our test suites were developed using internal datasets from the same distribution as the training data and also naturally occurring sentences that were combined to create diverse JSON structures. They test instruction following through two tasks: translating all fields and translating only user-specified fields. We dropped a Markdown suite because our base models already saturated it.

For structured translation, we tracked translation quality and, primarily, structural consistency, which we report here. The metrics target errors observed during model development. Table 1 reports their average.

- Overall structure: *Parsing accuracy* checks whether the delimiters permit structured-output extraction. *Decoding accuracy* checks whether the extracted output decodes into a valid object (e.g., JSON or YAML). *Schema accuracy* checks whether the decoded output matches the source schema (e.g., the same

JSON keys). Failures cascade to subsequent structured metrics.

- Untranslation issues: The intermediate models exhibited different levels of copying in their outputs, from leaving intermediate fields untranslated to retaining entire translations in the source language. In addition, the instruction-following test suite involved only translating a limited number of fields, requiring the remaining fields to be retained in their source language. Thus, the evaluation measured (i) the percentage of samples where all non-translatable content was copied correctly and (ii) the average per-sample percentage of translatable fields that were incorrectly copied. Lastly, a separate metric evaluated the proportion of examples that were left completely untranslated.
- Line count consistency: Per field, this checks whether the generated and source translations have the same line count.

3.6 Quantifying Test Set Saturation

To measure headroom on popular WMT test sets, we ran a human evaluation of the same system with the ESA protocol (Kocmi et al., 2024b). Table 5 shows that for high-resource languages, WMT24 is much easier than WMT26—and nearly solved. This is especially problematic with xComet: as actual errors grow rare, the metric’s word preferences

Language	WMT24	WMT26
Arabic (MSA)	19.7%	34.9%
German	22.9%	37.1%
Spanish (Mexico)	14.2%	34.9%
French	7.6%	13.9%
Italian	11.9%	11.5%
Japanese	14.6%	28.9%
Korean	29.9%	19.3%
Portuguese (Brazil)	6.6%	17.5%
Chinese (Simplified)	14.4%	18.1%
Overall difficulty	15.8%	24.0%

Table 5: Percentage of segments containing at least one major error, highlighting the saturation of the test sets.

affect score differences more than the errors. We also found that in WMT26, the software domain is easiest, while other domains remain considerably harder.

These results require some caution. Although both evaluations used the Pearmut framework (Zouhar and Kocmi, 2026), some details differ, most notably those affecting the slider anchors. We argue that these changes minimally affected the major-error annotations, which remained comparable. The test sets were also evaluated in different months, alongside different systems, and by slightly different annotator pools. While the pools could not be strictly controlled, we selected one system that was identical across both evaluations. The table shows this model’s error percentages in both setups. This strong system outperforms Command A+ but underperforms North Small Translate.

4 Conclusion

We introduce *North Small Translate*, Cohere’s state-of-the-art machine translation system. A sparse mixture-of-experts non-reasoning transformer-based LLM with 25 billion active and 218 billion total parameters, it achieves top-tier performance across 50 languages.

We build *North Small Translate* via a five-step training protocol spanning SFT, DPO, and online RL. By using difficulty sampling and an orchestrated preference distillation pipeline, we focused our synthetic data generation and training signal strictly on the most challenging documents. Consequently, the model excels in standard machine translation as well as complex related tasks, including structured translation, terminology adherence, and error detection.

For latency-insensitive applications requiring maximum quality, we incorporated an Agentic

Translation framework, enabling the model to iteratively evaluate and post-edit its own drafts. We submitted our model to the WMT 2026 General MT shared task as *Cohere CAT+*, demonstrating the model’s strong out-of-the-box capabilities.

Limitations

MT evaluation is limited by noise and the limited discriminative power of automated benchmarks and even human evaluators. Translation quality can be subjective, and high quality in one domain does not guarantee it in another, even for the same language. Preferred system recommendations can thus vary by use case.

References

- Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. [A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity](#). In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718, Nusa Dua, Bali. Association for Computational Linguistics.
- Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. [Findings of the 2015 workshop on statistical machine translation](#). In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisbon, Portugal. Association for Computational Linguistics.
- Adrian Charkiewicz, Pinzhen Chen, Thierry Etchegoyhen, Harritxu Gete Ugarte, Kamil Guttman, Xu Huang, David Ponce, Artur Nowakowski, Frédéric Oudermatt, Arturo Oncevay, Kirill Semenov, Dawei Zhu, and Vilém Zouhar. 2026. Findings of the WMT26 terminology translation task. In *To appear in Proceedings of the Eleventh Conference on Machine Translation*. Association for Computational Linguistics.
- Rajen Chatterjee, Matteo Negri, Raphael Rubino, and Marco Turchi. 2018. [Findings of the WMT 2018 shared task on automatic post-editing](#). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 710–725, Belgium, Brussels. Association for Computational Linguistics.
- Team Cohere, Arash Ahmadian, Marwan Ahmed, Jay Alammam, Milad Alizadeh, Yazeed Alnumay, Sophia Althammer, Arkady Arkhangorodsky, Viraat

- Aryabumi, Dennis Aumiller, et al. 2025. Command a: An enterprise-ready large language model. *arXiv preprint arXiv:2504.00698*.
- Cohere Team. 2026. [Introducing command a+: Making sovereign agentic capabilities available to all](#).
- Pedro Cuenca, Ben Burtenshaw, and Aritra Roy Gosthipaty. 2026. [Meta is back with muse glimmer: Local, agentic, multimodal, and open source](#).
- Daniel Deutsch, Eleftheria Briakou, Isaac Rayburn Caswell, Mara Finkelstein, Rebecca Galor, Juraj Juraska, Geza Kovacs, Alison Lui, Ricardo Rei, Jason Riesa, Shruti Rijhwani, Parker Riley, Elizabeth Salesky, Firas Trabelsi, Stephanie Winkler, Biao Zhang, and Markus Freitag. 2025. [WMT24++: Expanding the language coverage of WMT24 to 55 languages & dialects](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12257–12284, Vienna, Austria. Association for Computational Linguistics.
- Daniel Deutsch, George Foster, and Markus Freitag. 2023. [Ties matter: Meta-evaluating modern metrics with pairwise accuracy and tie calibration](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12914–12929, Singapore. Association for Computational Linguistics.
- Markus Freitag, David Grangier, Qijun Tan, and Bowen Liang. 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.
- Gemma Team. 2026. [Gemma 4 technical report](#). Preprint, arXiv:2607.02770.
- GLM-5-Team, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, and 167 others. 2026. [Glm-5: from vibe coding to agentic engineering](#). Preprint, arXiv:2602.15763.
- Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. [xCOMET: Transparent machine translation evaluation through fine-grained error detection](#). *Transactions of the Association for Computational Linguistics*, 12:979–995.
- Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. 2024. [MetricX-24: The Google submission to the WMT 2024 metrics shared task](#). In *Proceedings of the Ninth Conference on Machine Translation*, pages 492–504, Miami, Florida, USA. Association for Computational Linguistics.
- Dayeon Ki and Marine Carpuat. 2024. [Guiding large language models to post-edit machine translation with error annotations](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4253–4273, Mexico City, Mexico. Association for Computational Linguistics.
- Tom Kocmi, Arkady Arkhangorodsky, Alexandre Berard, Phil Blunsom, Samuel Cahyawijaya, Théo Dehaze, Marzieh Fadaee, Nicholas Frosst, Matthias Galle, Aidan Gomez, Nithya Govindarajan, Wei-Yin Ko, Julia Kreutzer, Kelly Marchisio, Ahmet Üstün, Sebastian Vincent, and Ivan Zhang. 2025a. [Command-a-translate: Raising the bar of machine translation with difficulty filtering](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 789–799, Suzhou, China. Association for Computational Linguistics.
- Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakounga, Jessica Lundin, Christof Monz, Kenton Murray, and 10 others. 2025b. [Findings of the WMT25 general machine translation shared task: Time to stop evaluating on easy test sets](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 355–413, Suzhou, China. Association for Computational Linguistics.
- Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Sergey Dukanov, Anton Dvorkovich, Mark Fishel, Markus Freitag, Samuel Frontull, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Stan Kharevich, Philipp Koehn, Zheng Li, Jean Maillard, Christof Monz, Alexander Murauski, and 12 others. 2026. [Findings of the WMT26 general machine translation shared task: Contrastive dynamic human evaluation at scale](#). In *Proceedings of the Eleventh Conference on Machine Translation*. Association for Computational Linguistics.
- Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popović, and 3 others. 2024a. [Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet](#). In *Proceedings of the Ninth Conference on Machine Translation*, pages 1–46, Miami, Florida, USA. Association for Computational Linguistics.
- Tom Kocmi and Christian Federmann. 2023. [Large language models are state-of-the-art evaluators of translation quality](#). In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203, Tampere, Finland. European Association for Machine Translation.
- Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja

- Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024b. [Error span annotation: A balanced approach for human evaluation of machine translation](#). In *Proceedings of the Ninth Conference on Machine Translation*, pages 1440–1453, Miami, Florida, USA. Association for Computational Linguistics.
- Alon Lavie, Greg Hanneman, Sweta Agrawal, Diptesh Kanojia, Chi-Kiu Lo, Vilém Zouhar, Frederic Blain, Chrysoula Zerva, Eleftherios Avramidis, Sourabh Deoghare, Archchana Sindhujan, Jiayi Wang, David Ifeoluwa Adelani, Brian Thompson, Tom Kocmi, Markus Freitag, and Daniel Deutsch. 2025. [Findings of the WMT25 shared task on automated translation evaluation systems: Linguistic diversity is challenging and references still help](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 436–483, Suzhou, China. Association for Computational Linguistics.
- Arle Lommel, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional quality metrics (mqm): A framework for declaring and describing translation quality metrics. *Tradumàtica*, pages 0455–463.
- Mistral AI. 2026. [Mistral medium 3.5](#).
- Mistral AI Team. 2026. [Ministral 3](#). *Preprint*, arXiv:2601.08584.
- NVIDIA Research. 2026. [Nemotron 3 ultra: Open, efficient mixture-of-experts hybrid mamba](#). *Preprint*, arXiv:2606.15007.
- OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, and 107 others. 2025. [gpt-oss-120b and gpt-oss-20b model card](#). *Preprint*, arXiv:2508.10925.
- Qwen Team. 2026a. [Qwen3.5-omni technical report](#). *Preprint*, arXiv:2604.15804.
- Qwen Team. 2026b. [Qwen3.6-27B: Flagship-level coding in a 27b dense model](#).
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS ’23*, Red Hook, NY, USA. Curran Associates Inc.
- Kirill Semenov, Xu Huang, Vilém Zouhar, Nathaniel Berger, Dawei Zhu, Arturo Oncevay, and Pinzhen Chen. 2025. [Findings of the WMT25 terminology translation task: Terminology is useful especially for good MTs](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 554–576, Suzhou, China. Association for Computational Linguistics.
- Anushka Singh, Aarya Pakhale, Mitesh M. Khapra, and Raj Dabre. 2025. [Quality estimation and post-editing using LLMs for Indic languages: How good is it?](#) In *Proceedings of Machine Translation Summit XX: Volume 1*, pages 388–398, Geneva, Switzerland. European Association for Machine Translation.
- Thinking Machines Lab. 2026. [Introducing inkling-small](#).
- Marcos V Treviso, Nuno M Guerreiro, Sweta Agrawal, Ricardo Rei, José Pombal, Tania Vaz, Helena Wu, Beatriz Silva, Daan van Stigt, and Andre Martins. 2024. [xTower: A multilingual LLM for explaining and correcting translation errors](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15222–15239, Miami, Florida, USA. Association for Computational Linguistics.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. [Group sequence policy optimization](#). *Preprint*, arXiv:2507.18071.
- Vilém Zouhar and Tom Kocmi. 2026. [Pearmut: Human evaluation of translation made trivial](#). *Preprint*, arXiv:2601.02933.

	Params total (active)	All languages	Albanian	Bengali	Bulgarian	Catalan	Croatian	Czech	Danish	Dutch	Egyptian Arabic	Estonian	European Portuguese
North Small Translate (Agentic)	218B (25B)	84.4	78.7	87.5	80.1	84.0	79.0	88.8	89.1	89.5	84.2	73.5	90.4
North Small Translate	218B (25B)	83.6	77.9	82.8	79.5	83.9	78.3	89.2	85.3	89.5	84.0	72.6	89.6
Mistral Large 3	675B (41B)	81.6	76.2	84.9	83.2	85.3	81.4	83.9	86.0	87.9	78.7	72.0	86.4
Qwen 3.5 397B	397B (17B)	81.6	81.2	83.8	83.3	82.0	83.4	81.5	85.5	87.1	75.4	75.6	80.7
DeepL NextGen		81.4	76.5	75.8	84.5	78.7	76.4	85.3	88.6	89.5	57.6	81.0	87.2
Gemma 4 31b	31B (31B)	79.5	76.2	88.8	78.5	78.3	79.4	75.5	88.5	88.9	87.5	53.1	87.0
Inkling Small	276B (12B)	78.5	73.8	80.2	78.4	82.4	76.7	78.8	82.5	83.9	80.1	73.1	81.1
Muse Glimmer 30B	30B (30B)	78.2	76.0	82.4	76.9	80.5	77.0	79.1	79.6	82.0	81.6	73.8	83.1
Command A+	218B (25B)	76.5	73.4	79.3	75.4	78.8	76.0	83.3	85.0	82.9	71.8	65.9	79.5
GLM 5.2	744B (40B)	76.5	79.0	72.6	78.2	84.1	86.2	86.3	85.0	80.9	73.8	77.3	69.5
Mistral Medium 3.5	128B (128B)	75.7	60.4	80.9	78.4	83.6	74.1	80.1	83.0	85.0	67.7	61.1	84.3
GPT-OSS 120B	117B (5B)	72.3	66.4	73.5	70.8	73.4	71.8	71.6	77.4	78.0	75.0	62.5	78.8
Command A Translate (2025)	111B (111B)	70.0	55.7	63.7	59.7	77.2	66.3	81.1	75.3	82.9	81.7	51.1	83.5
Google Translate		68.2	67.7	72.9	68.0	69.8	69.2	65.4	70.8	76.8	53.0	65.6	51.6
Nemotron 3 Ultra 550B	550B (55B)	67.1	52.5	65.8	61.3	74.6	60.8	65.1	70.5	73.6	79.5	49.2	85.4
Qwen 3.6 27B	27B (27B)	61.9	30.1	68.4	64.7	52.2	77.7	70.0	79.0	79.9	63.6	52.2	83.2
	Filipino	Finnish	French	German	Greek	Hebrew	Hindi	Hungarian	Icelandic	Indonesian	Irish	Italian	Japanese
North Small Translate (Agentic)	82.8	86.1	90.2	89.7	89.4	87.1	88.8	74.8	67.2	90.1	69.6	90.3	87.9
North Small Translate	82.7	86.6	90.2	89.7	89.0	83.3	88.3	73.8	67.9	85.8	68.4	90.2	87.4
Mistral Large 3	77.5	80.0	88.7	87.7	81.1	80.6	84.6	81.4	69.1	86.3	62.6	88.0	86.7
Qwen 3.5 397B	67.9	83.4	81.9	87.0	79.7	82.2	79.6	82.7	71.4	86.1	67.1	86.5	81.2
DeepL NextGen	75.3	85.0	86.6	86.4	82.7	85.7	77.8	82.7	67.2	87.7	70.6	88.4	84.6
Gemma 4 31b	86.1	82.6	90.4	89.6	74.1	80.8	89.5	55.4	41.3	90.0	18.9	89.0	90.9
Inkling Small	74.1	78.5	82.2	82.0	79.1	78.9	79.0	77.5	69.2	81.2	67.9	84.4	80.4
Muse Glimmer 30B	76.3	76.2	83.8	84.1	77.8	77.2	79.9	78.5	65.7	82.9	59.1	82.8	83.7
Command A+	73.6	78.2	85.5	84.5	81.9	81.6	79.4	70.3	53.7	80.1	54.2	84.8	79.4
GLM 5.2	60.7	72.4	89.1	85.9	58.5	72.0	76.1	73.8	65.1	78.1	51.7	84.1	90.0
Mistral Medium 3.5	70.8	72.8	88.1	86.8	76.1	75.8	81.3	75.1	48.8	83.9	44.9	86.3	84.9
GPT-OSS 120B	75.0	69.5	82.5	79.1	67.9	67.8	76.7	70.2	55.7	80.0	50.7	79.4	79.9
Command A Translate (2025)	64.7	62.1	84.2	83.4	80.2	79.6	77.2	68.9	42.1	82.0	38.4	84.3	79.2
Google Translate	66.9	66.4	78.5	75.1	64.9	64.9	71.7	64.4	64.2	77.5	65.0	76.7	71.8
Nemotron 3 Ultra 550B	70.0	57.8	86.9	83.9	58.4	84.4	84.9	59.0	45.2	76.8	43.4	83.9	89.9
Qwen 3.6 27B	50.1	58.8	78.3	75.8	59.4	19.4	81.1	69.3	25.4	83.2	16.8	84.4	75.6
	Korean	Latvian	Lithuanian	Malay	Maltese	Norwegian Bokmal	Panjabi	Persian	Polish	Romanian	Russian	Serbian	Simplified Chinese
North Small Translate (Agentic)	88.9	71.6	69.6	87.3	65.0	90.2	87.8	87.4	87.2	88.1	88.8	74.9	89.1
North Small Translate	84.4	71.2	68.5	87.0	65.6	90.0	87.6	87.0	87.2	87.6	88.1	73.9	85.4
Mistral Large 3	86.7	71.2	77.8	83.1	46.4	84.5	83.5	83.5	85.3	85.6	85.9	80.3	87.6
Qwen 3.5 397B	83.2	79.2	78.3	79.7	70.9	81.2	84.7	83.9	83.1	84.8	86.5	75.9	86.2
DeepL NextGen	87.0	80.1	80.8	77.7	69.4	87.8	77.8	74.3	82.1	85.8	84.3	73.5	85.5
Gemma 4 31b	92.1	48.5	56.8	89.2	30.6	88.1	88.1	89.4	88.9	83.3	89.9	69.8	90.4
Inkling Small	84.1	69.6	73.9	79.3	68.2	81.7	78.5	81.6	80.7	80.1	80.1	63.0	86.1
Muse Glimmer 30B	82.7	68.4	74.4	80.7	58.0	80.8	80.5	80.4	78.6	81.1	80.9	68.7	85.4
Command A+	81.1	65.7	65.7	78.2	52.6	85.2	78.7	79.5	80.0	82.9	79.5	74.6	81.3
GLM 5.2	86.0	73.8	78.0	79.3	58.6	83.9	37.0	85.7	73.4	83.3	88.4	75.2	92.1
Mistral Medium 3.5	81.5	55.0	59.3	77.0	34.7	82.2	77.0	81.0	81.0	80.5	82.8	74.9	86.8
GPT-OSS 120B	78.8	59.4	62.9	76.7	48.0	77.9	74.2	71.3	74.9	76.6	74.6	61.2	81.5
Command A Translate (2025)	79.5	49.9	59.7	71.9	36.3	74.3	48.7	79.8	80.0	82.7	78.4	53.0	80.9
Google Translate	72.6	65.0	62.7	67.1	68.3	71.1	65.2	66.0	68.4	67.3	69.6	66.6	75.4
Nemotron 3 Ultra 550B	87.4	46.7	49.6	69.6	31.6	61.3	69.9	66.4	67.9	54.9	77.5	43.5	87.4
Qwen 3.6 27B	63.9	36.7	8.7	79.8	10.6	75.0	26.2	82.1	77.8	48.1	85.3	59.6	91.0
	Slovak	Slovenian	Spanish	Standard Arabic	Swedish	Tamil	Telugu	Thai	Traditional Chinese	Turkish	Ukrainian	Urdu	Vietnamese
North Small Translate (Agentic)	84.6	76.1	90.9	86.6	89.2	85.7	85.9	86.7	88.8	85.3	86.8	87.1	89.7
North Small Translate	81.2	77.2	90.5	86.7	88.9	85.7	85.7	86.7	88.5	85.7	86.9	86.9	89.4
Mistral Large 3	79.3	74.5	88.4	80.2	82.7	82.2	82.5	84.2	87.8	82.9	84.7	80.8	86.6
Qwen 3.5 397B	79.1	81.0	86.7	85.4	84.6	83.9	83.8	82.0	82.8	84.0	85.9	78.2	86.5
DeepL NextGen	84.0	81.9	88.4	82.0	88.4	77.9	79.8	81.9	84.6	85.1	84.1	75.7	86.7
Gemma 4 31b	71.2	63.1	90.6	88.1	88.8	85.8	87.5	90.5	87.4	88.1	87.7	88.5	90.4
Inkling Small	77.8	76.6	85.2	78.9	84.0	75.6	77.7	78.1	81.6	77.2	82.0	76.3	82.8
Muse Glimmer 30B	77.5	74.0	84.7	74.3	80.9	79.7	78.7	76.1	82.5	78.2	80.9	79.3	82.8
Command A+	77.2	73.9	87.2	82.0	83.0	76.5	75.9	78.3	53.6	77.8	78.4	77.4	81.9
GLM 5.2	83.6	73.8	81.7	86.7	81.3	46.7	31.8	84.6	88.8	85.3	83.8	84.1	87.7
Mistral Medium 3.5	71.3	65.6	87.9	77.8	82.9	73.9	73.9	79.0	84.2	77.8	81.8	75.3	83.8
GPT-OSS 120B	69.0	66.6	82.3	72.9	77.7	70.1	72.7	72.1	81.9	73.5	72.5	75.3	74.6
Command A Translate (2025)	73.3	59.4	85.6	82.0	74.7	66.4	44.2	61.2	79.1	76.3	79.3	56.6	79.9
Google Translate	65.6	65.4	76.5	73.0	69.4	71.4	78.3	76.5	41.3	70.4	65.5	54.0	78.8
Nemotron 3 Ultra 550B	60.7	56.3	88.1	68.5	71.9	54.1	52.1	65.0	83.8	67.6	65.7	70.6	73.3
Qwen 3.6 27B	70.9	42.5	81.6	61.9	74.5	34.7	73.2	53.7	89.9	39.7	69.7	75.9	84.6

Table 6: Machine translation quality of every supported language of the WMT26 test set.

A GEMBA-ESA Fluency+Adequacy Template

A.1 System prompt

You are an impartial judge whose task is to analyze machine
↪ translation outputs and evaluate their quality.

Based on the provided source text, target language,
↪ translation guidelines, and translation output, identify
↪ translation errors and assess their severity. Then
↪ evaluate the quality of the translation according to the
↪ provided rubric criteria. For each criterion, provide a
↪ brief rationale in English and a score within the allowed
↪ range.

Errors have "quote", "severity", and "category" attributes.
↪ The "quote" value should quote the part of the
↪ translation that contains the error. Use minor severity
↪ for imperfections or stylistic issues that do not impact
↪ the core message (e.g., awkward phrasing). Use major
↪ severity for errors that obscure the meaning,
↪ misrepresent the source, or change the message (e.g.,
↪ incorrect information or confusing wording). The category
↪ must be one of: "accuracy", "fluency", "terminology",
↪ "style", "locale", "design", "mistranslation",
↪ "omission", "addition", "untranslated", "grammar",
↪ "spelling", "punctuation", "register", "unidiomatic",
↪ "inconsistency", or "other". If the error span is longer
↪ than a few words, use an ellipsis (...) to shorten it. If
↪ the error concerns the entire translation, such as a
↪ translation in the wrong language, set "quote" to
↪ "FULL_TRANSLATION". For omissions (missing source
↪ content), set "quote" to the adjacent translation words
↪ with "OMISSION" between them. If there are no errors, set
↪ "errors" to an empty array.

Return your error analysis and quality evaluation as a
↪ single, strictly valid JSON object matching the provided
↪ output format.

A.2 User prompt

Evaluation rubric

- Fluency score [key: fluency]: Measures how natural and
↪ grammatically correct the translation reads in the target
↪ language. (score: 0 to 100)
- Adequacy score [key: adequacy]: Measures how completely the
↪ meaning of the source is conveyed by the translation.
↪ (score: 0 to 100)

Additional judging rules

Important rules

- Hallucinations are major adequacy errors and should be
↪ penalized severely.
- Outputs in the wrong language, dialect, or script are major
↪ fluency errors and should be penalized severely. Borrowed
↪ words are allowed if valid in the target language.
- Check translation consistency (e.g., technical terms)
↪ across the entire text.
- Over-translation and extraneous text (e.g., introductory
↪ statements like "Here is your translation" or unprompted
↪ explanations) are major adequacy errors.
- Count guideline adherence issues toward the adequacy score;
↪ for these errors, set "quote" to the corresponding
↪ guideline.

Adequacy scale

- 81-100 (Very Good): Complete meaning transfer; no
↪ information is lost, added, or distorted.
- 61-80 (Good): Near-complete transfer; minor inaccuracies or
↪ omissions that do not affect the core message.
- 41-60 (Acceptable): Main ideas conveyed, but noticeable
↪ inaccuracies, omissions, or additions.
- 21-40 (Borderline): Partial transfer; frequent
↪ misinterpretation or omission confusing the core message.
- 0-20 (Not acceptable): Complete violation of meaning; large
↪ portions mistranslated, hallucinated, or missing.

Fluency scale

- 81-100 (Very Good): Perfectly natural and grammatical;
↪ reads like a native text; requires no or minimal
↪ proofreading.
- 61-80 (Good): Mostly natural; minor awkwardness or slight
↪ grammatical imperfections; needs light proofreading.

- 41-60 (Acceptable): Uneven naturalness; noticeable awkward
↪ phrasing or structural issues; usable only after
↪ substantial revision.
- 21-40 (Borderline): Often unnatural; frequent grammatical
↪ errors that impede reading flow; requires major rewrite.
- 0-20 (Not acceptable): Incoherent, structurally broken,
↪ incomprehensible, or in the wrong language, dialect, or
↪ script; unusable without complete retranslation.

Source text
...
{{SOURCE_TEXT}}
...

Translation language or dialect
{{TARGET_LANGUAGE_OR_DIALECT}}

Translation guidelines
{{TRANSLATION_GUIDELINES}}

Translation output
...
{{TRANSLATION_OUTPUT}}
...

Expected output format
Return a single, strictly valid JSON object with the
↪ following schema:
```json  
{  
 "errors": [  
 {  
 "quote": "<string>",  
 "severity": "minor|major",  
 "category": "<string>"  
 },  
 ],  
 "fluency": {  
 "rationale": "<string>",  
 "score": "<number>"  
 },  
 "adequacy": {  
 "rationale": "<string>",  
 "score": "<number>"  
 }  
}  
```

Every string value must be valid JSON: escape each backslash
↪ as \\ and each double quote as \". Do not emit raw
↪ control characters or trailing commas.