

Retrieval-Augmented Terminology Translation for English–Russian: A Multi-Domain Study

Maria Sukhareva
HSE University
Moscow, Russia
maryleyas@gmail.com

Ekaterina Enikeeva
HSE University, Yandex
Moscow, Russia
protoev@yandex.ru

Abstract

Neural machine translation attains high fluency on general-domain text but systematically fails on specialized terminology, where polysemous and low-frequency terms carry domain-specific meanings absent from general training data. We investigate two complementary interventions for English–Russian specialized translation across ten professional domains: (i) a four-stage retrieval-augmented pipeline of LLM-based term extraction, domain classification, lemma-based knowledge-base retrieval, and glossary-augmented translation; and (ii) a preference-based fine-tuning stage combining supervised fine-tuning (SFT) with Contrastive Preference Optimization (CPO) layered on top of retrieval. We release an open multi-domain English–Russian terminology knowledge base of 50,000 terms with 120,161 domain-tagged translations, a 200-item curated test set selected for terminology difficulty, and a 1,300-item Wikipedia-derived test set. Across four open base models, retrieval-augmented prompting alone delivers the bulk of the terminology gain and already exceeds the two commercial reference systems; the additional preference-tuning stage contributes only a small further gain at a measurable fluency cost. The pattern transfers to the WMT25 Terminology Translation Task, the most recent iteration of the shared task that included English–Russian as a language pair.

1 Introduction

Modern neural machine translation (NMT) is remarkably strong on general-domain text, but specialized text is where it breaks down: a drilling report’s *Christmas tree* becomes a holiday tree rather than a wellhead valve assembly, and a legal *consideration* reverts to its everyday sense. NMT and LLM architectures do condition on local context, but a strong frequency bias in the general training data often overrides that signal, driving the model toward the higher-frequency general sense because

specialized senses are under-represented in training data.

Retrieval-Augmented Generation (RAG) sidesteps retraining by querying an external knowledge base at inference time and inserting retrieved term translations into the prompt. The complementary route is fine-tuning the model itself, optionally with preference-based objectives such as the Contrastive Preference Optimization (CPO) stage of ALMA-R (Xu et al., 2024). Both directions have been studied separately; their combination – structured multi-domain glossary retrieval on top of CPO-style fine-tuning – has not been systematically evaluated for English–Russian. There is also a resource gap: openly available Russian termbases either spread coverage unevenly across domains or focus on a single specialization.

This paper asks two research questions.

RQ1: to what extent does domain-classified glossary retrieval improve EN–RU translation of specialized terminology across ten professional domains, compared to direct translation?

RQ2: does preference-based fine-tuning (SFT followed by CPO) complement retrieval?

Contributions. (1) An open multi-domain EN–RU terminology knowledge base of 50,000 terms with 120,161 domain-tagged translations across ten domains. (2) A 200-item test set whose items are specifically selected for terminology difficulty (polysemy, compositionality, rarity), plus a 1,300-item Wikipedia-derived test set for a larger distributional check. (3) A four-stage retrieval-augmented pipeline evaluated across four open-source base models and two commercial systems, with an SFT+CPO ablation on two of the open models. (4) External validation on the WMT25 Terminology Translation Task. All resources – code, knowledge base, both test sets, evaluation scripts, and verbatim prompts – are released at https://github.com/Maryleya/RAG_System_for_Specialized_Terms.

2 Related Work

MT of specialized text. Domain shift is a long-standing failure mode of NMT (Koehn and Knowles, 2017), and recent multi-domain studies confirm that terminology errors propagate to downstream quality (Oncevay et al., 2025). Domain-adaptation surveys (Chu and Wang, 2018; Saunders, 2022) address the domain as a whole rather than individual terms, which motivates term-level intervention.

Terminology-constrained translation. Constrained decoding (Hokamp and Liu, 2017; Hasler et al., 2018) enforces pre-specified lexical constraints but degrades fluency on morphologically rich targets. Training-time term injection (Dinu et al., 2019) avoids this at the cost of retraining. The WMT Terminology Shared Task (Semenov et al., 2025) has run in 2021, 2023, and 2025 and provides the closest evaluation precedent to the present work.

Retrieval-augmented MT. kNN-MT (Khandelwal et al., 2021) established inference-time retrieval for domain adaptation at the token level. RAG (Lewis et al., 2020) combines parametric and non-parametric knowledge; RAGtrans (Wang et al., 2024) applies this to MT with document-level retrieval. Dictionary-level prompting (Ghazvininejad et al., 2023; Moslem et al., 2023) injects bilingual entries directly into the LLM prompt, which we adopt. The closest prior system, DragFT (Zheng et al., 2024), combines dictionary prompting with SFT for EN–ZH but does not target Russian, omits a separate domain classifier, and does not extend to preference-based fine-tuning – the gap the present work addresses. WMT25 retrieval-augmented systems (Nieminen et al., 2025; Guttmann et al., 2025) assume the terminology dictionary is supplied as part of the task rather than retrieved end-to-end.

Fine-tuning for translation. Parameter-efficient adaptation of large translation models is now typically done with LoRA (Hu et al., 2022). On top of that, ALMA (Xu et al., 2023) and its preference-tuned successor ALMA-R (Xu et al., 2024) argue that CPO sidesteps the reward-margin collapse that plain DPO (Rafailov et al., 2023) runs into when the *chosen* side is a human reference translation.

Evaluation. chrF2++ (Popović, 2015), BLEU (Papineni et al., 2002), the COMET family (Rei et al., 2020, 2022), and MetricX (Juraska et al.,

2023) provide the general quality signal. The WMT Terminology Shared Task (Semenov et al., 2025) and its prior editions score term compliance with a lemmatized exact-match criterion: a constraint counts as satisfied only if some surface form sharing the reference lemma appears in the output, which penalises a system that conveys the correct meaning through a valid same-domain synonym the reference happens not to use. We build our own Terminology Success Rate (TSR) as a two-stage extension of that same idea: pymorphy3 (Korobov, 2015) lemma matching first – equivalent to the WMT-style criterion – and then, only for the items the first stage rejects, an LLM-as-judge pass that additionally credits morphological variants and legitimate same-domain synonyms, following Semenov and Bojar (2022) and Zheng et al. (2023). We add the second stage because our KB itself routinely lists several valid Russian variants per English term (§3.1), so a strict single-reference match would undercount correct translations by construction; the two stages are kept separable so that the first-stage-only number remains directly comparable to the WMT-style criterion. For external validation against WMT25 itself (§5.7) we accordingly report the shared task’s own official script rather than substituting our two-stage TSR, keeping that comparison on WMT’s own terms. Freitag et al. (2021) document the miscalibration of general-domain neural quality metrics on out-of-distribution translations – a finding sharpened by Zouhar et al. (2024), who show that fine-tuned MT metrics specifically degrade in unseen, specialized domains – a concern we return to in §5.5.

3 Resources

3.1 Terminology Knowledge Base

The KB pulls entries from five sources, each covering a different gap in EN–RU terminological coverage. **Multitran**¹ is a large, crowd-sourced online dictionary that functions as the de facto standard reference tool for Russian professional translators; entries are contributed by volunteer translators, each tagged by its contributor with a subject field. It supplies the bulk of our material, with its editor-assigned domain tags mapped down to our ten canonical labels. We add three further sources to extend coverage beyond Multitran: **Wikipedia interlanguage titles**, i.e. pairs of article titles linked across language editions of Wikipedia, which yield

¹<https://www.multitran.com>

a term pair whenever an encyclopedia entry exists in both English and Russian; **Wikidata label pairs**, the English and Russian labels attached to the same item in Wikidata (Wikipedia’s structured companion knowledge base of entities); and **EngCom**², an open, community-maintained English–Russian dictionary of computing terminology, used only to extend the IT domain. Finally, **Wiktionary**³, the collaboratively edited multilingual dictionary, contributes general-vocabulary pairs; since Wiktionary entries carry no subject-field tag of their own, they inherit the domain tag of whichever KB entry they align with. Polysemy is preserved at the entry level rather than collapsed: *draft* keeps both *осадка* for shipbuilding and *черновик* for general use, and *haircut* keeps the financial discount reading separately from the cosmetic one. The final KB has 50,000 unique English terms and 120,161 domain-tagged Russian translations (2.46 per term on average) across IT, aviation, medicine, oil and gas, agriculture, construction, shipbuilding, finance, energy, and law.

3.2 Test Sets

Two test sets serve complementary roles. The **curated set** has 200 items, 20 per domain. It was assembled semi-automatically from three specialized reference sources – **UNTERM**⁴, the United Nations’ own multilingual terminology database of terms used in UN documentation; **Linguee**⁵, a bilingual search engine that pairs a conventional dictionary with real translated sentence pairs mined from the web; and **Gufo.me**⁶, an online aggregator of Russian-language dictionaries and encyclopedias spanning many subject fields – topped up by hand through **ProZ.com**⁷, the largest online community and job marketplace for professional translators, whose crowd-sourced KudoZ glossary lets translators propose and vote on translations for difficult terms. Every candidate was then checked manually item by item and kept only if it presented at least one of three concrete difficulties – polysemy, compositionality, or rarity – so that a strong general-purpose system is expected to get it wrong even though the intended translation is known. The **wiki set** is much larger (1,300 items), drawn from

Wikipedia-aligned parallel data without any difficulty filter, and gives a broader distributional view. Because it lacks aligned reference sentences, it is scored with reference-free metrics and TSR only.

3.3 Fine-Tuning Pool

The fine-tuning pool holds 10,000 (source, chosen, rejected) triplets mined from WikiMatrix (Schwenk et al., 2021) and ParaCrawl (Bañón et al., 2020). Selection is lemma-aware: sentences are kept when they match a KB term pair. About 86% of the pool is an “other”-domain portion mixed in on purpose so the model does not drift away from general-domain text. Rejected candidates come from the very model being fine-tuned, run zero-shot without the glossary, which points the preference signal at that model’s own failure modes rather than at some generic baseline. A three-stage filter runs before mining is complete: it drops proper nouns and short generic terms, transliteration cognates such as *router* – *роутер*, and any term Google Translate already handles correctly. Google Translate is used here purely as a difficulty proxy – an external, off-pipeline system that flags examples on which a strong general-purpose engine already succeeds, so the surviving pool concentrates on terminology cases where a fine-tuning gain is actually possible; the fine-tuned models themselves are never compared to or trained against Google Translate. Any sentence that overlaps with the curated or wiki test set – by source sentence or by target term – is excluded from the mining pool throughout, so that no fine-tuning example leaks into evaluation.

4 Methodology

4.1 Pipeline

The pipeline turns a single English source sentence into a Russian translation in four stages (Figure 1), each consuming the previous stage’s output and producing one new artifact.

Stage 1: Term Extraction. A few-shot prompt to Qwen3-4B-Instruct-2507 (Appendix B) reads the source sentence and returns a newline-separated list of the noun phrases it judges to be specialized terminology, with no other output. We compared this LLM extractor against three lighter-weight alternatives – a supervised BERT-NER model trained on ACTER (Rigouts Terryn et al., 2020), and the unsupervised keyword extractors KeyBERT and YAKE – and adopted the LLM extractor for its recall–precision trade-off and, more importantly,

²<https://github.com/Etersoft/engcom>

³<https://www.wiktionary.org>

⁴<https://unterm.un.org/unterm2/>

⁵<https://www.linguee.com>

⁶<https://gufo.me>

⁷<https://www.proz.com>

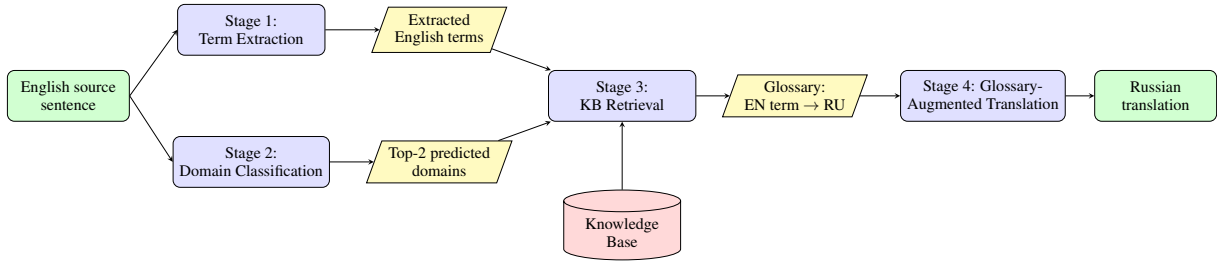


Figure 1: Overview of the four-stage retrieval-augmented translation pipeline. Blue rounded boxes are processing stages; yellow parallelograms are intermediate data artifacts passed between stages; the red cylinder is the external terminology knowledge base queried at Stage 3; green boxes are the source input and target output.

because it needs no domain-specific retraining to generalize across our ten target domains (Table 1). Its average of 4.04 extracted terms per sentence is in the same range as the unsupervised baselines (5.00 / 4.91), so some of what it extracts will not be genuine specialized terminology; this is tolerable because Stage 3 only ever inserts a glossary entry for a string that is actually present in the KB, so a spurious extraction simply fails to retrieve anything rather than propagating as a translation error.

Stage 2: Domain Classification. In parallel with extraction, the same source sentence is classified into one of the ten KB domains by a second Qwen3-4B-Instruct-2507 prompt (Appendix C). The classifier is primed with 30 synthetic (sentence, domain) examples – three per domain, themselves LLM-generated from a short keyword description of that domain (Appendix C.1) – so bootstrapping the classifier needs no manual per-domain annotation. Rather than commit to one label, the classifier outputs a `Primary` and an `Alternative` domain, and both are passed forward to filter Stage 3 retrieval; we found this top-2 handoff meaningfully more robust than using the top-1 label alone (Table 2), since some sentences are genuinely ambiguous between adjacent domains (e.g. energy vs. oil and gas). We chose this LLM classifier over an embedding-centroid alternative – paraphrase-multilingual-mpnet-base-v2 sentence embeddings, with class centroids computed from the same 30 synthetic examples – because it gave consistently higher top-1/top-2 accuracy.

Stage 3: KB Retrieval. The extracted terms and the predicted top-2 domains are jointly used to query the KB (§3.1): for each extracted term, its lemma is looked up, and only the Russian translations tagged with either predicted domain are kept, discarding translations tagged for other domains even when the lemma matches. This restricts a

polysemous entry like *draft* to only its shipbuilding sense when Stage 2 predicts shipbuilding, rather than surfacing all 2.46 average translations regardless of context. We chose this lemma-based lookup over two dense-retrieval alternatives we also tried – embedding the single-word KB entries and matching by cosine similarity, and embedding whole sentences to match against single-word entries – because both degraded on exactly the polysemy cases the pipeline is meant to fix: a single embedding for a polysemous word collapses toward its dominant sense, and sentence-to-word matching compares the wrong granularity. The output is a compact glossary: English term → one or more domain-tagged Russian translations.

Stage 4: Glossary-Augmented Translation.

The Stage 3 glossary is inserted into a translation prompt (Appendix D) that instructs the model to strongly prefer a glossary translation whenever the corresponding English term appears in the source, to pick among several domain-tagged variants by context when more than one survives filtering, and to inflect the chosen Russian lemma for the correct case, number, and gender rather than pasting it in unchanged. Terms absent from the glossary – because Stage 1 missed them or the KB has no entry – are left entirely to the model’s own judgment. This is a soft preference rather than a hard decoding constraint, a deliberate choice: it lets the model fall back to a fluent translation when the glossary is empty, wrong, or grammatically awkward in context, at the cost of not guaranteeing that a supplied glossary term actually surfaces in the output – which is exactly what TSR measures downstream (§3.1, §5.5).

4.2 Pipeline Variants

The four-stage pipeline above is referred as `rag_clf` below. We compare it against six other

configurations, each isolating one design choice by changing exactly one stage relative to `rag_clf`.

baseline. No glossary at all: the source sentence is translated with a plain zero-shot translation prompt (Appendix D), so that the reported baseline is a standard ordinary-MT number, comparable to how the four base models are used off the shelf. This means the baseline and the `rag_*` variants differ both in retrieval content and in prompt template, so some fraction of the observed `rag_clf` lift may be attributable to the glossary prompt itself rather than to retrieved content. A prompt-only control (the glossary prompt run with an empty dictionary) would separate the two contributions and we flag its absence in Limitations.

rag_auto. Stages 1, 3, and 4 run as usual, but Stage 2 is skipped entirely: Stage 3 retrieves *every* domain-tagged translation for each extracted term regardless of predicted domain, so the model sees every sense of a polysemous term at once rather than a domain-filtered subset. Comparing `rag_auto` to `rag_clf` isolates the contribution of domain classification specifically, separately from the contribution of retrieval itself.

rag_clf. The full four-stage pipeline described above.

rag_clf_sft and rag_clf_sft_cpo. Identical to `rag_clf` at Stages 1–3; only the Stage 4 translation model changes, to the SFT checkpoint and the SFT+CPO checkpoint respectively (§4.3). Fine-tuning is applied only to Hy-MT and Qwen, so these two variants exist only for those two base models.

term_substitution. A non-retrieval-augmented alternative that never shows the model a glossary: Stage 1’s extracted terms are instead substituted, in Russian, directly into the *source* sentence before translation (a code-switched source, cf. Dinu et al., 2019), and the model is prompted only to keep the inserted Russian words as-is while translating the rest of the sentence (Appendix D). This trades away Stage 4’s soft preference for a much stronger guarantee that the term survives into the output, at a fluency cost we quantify in §5.5.

rag_oracle. Identical to `rag_clf` except that Stage 2’s prediction is replaced by the ground-truth domain label. This is not a deployable configuration – it requires a human to specify the domain

– but it serves two purposes: as an upper bound that isolates classifier error from every other source of pipeline error (§5.5), and as a description of a practical translator-in-the-loop workflow, where a human specifies the domain once per document and the pipeline automates term extraction and retrieval underneath it.

4.3 Base Models and Fine-Tuning

The open side of the comparison uses four models: Llama-3.2-3B-Instruct, Qwen3-4B-Instruct-2507, TranslateGemma-4B, and Hy-MT2-7B. Google Translate and Yandex Translate serve as commercial reference points. Fine-tuning is applied only to two of the four open models – Hy-MT (the strongest baseline) and Qwen (a general-purpose control) – via LoRA at rank 64, $\alpha = 128$. The training recipe is 2 epochs of SFT at 10^{-4} on the chosen translations, with the glossary already sitting in the training prompt exactly as at inference; then 1 epoch of CPO at 5×10^{-6} , $\beta = 0.1$. An early DPO attempt collapsed with the same reward-margin pattern reported for ALMA-R (Xu et al., 2024); switching to CPO, whose NLL anchor keeps the chosen side pinned, stabilised it.

4.4 Evaluation

Surface quality is measured with sacrebleu (Post, 2018) chrF2++ and BLEU. On the neural side we report COMET (Unbabel/wmt22-comet-da) and COMET-Kiwi (Unbabel/wmt22-cometkiwi-da) together with MetricX-23 XXL and its reference-free variant MetricX-23-QE XXL (google/metricx-23-v2p0 and google/metricx-23-qe-v2p0). Terminology accuracy uses our two-stage TSR: a lemma-based first pass – equivalent to the exact-match criterion used by the WMT Terminology Shared Task – then a RuQwen-2.5-32B judge that decides whether the remaining misses are actually valid same-domain synonyms. We report the combined two-stage number throughout the main results because it better reflects whether the intended meaning was conveyed; the first stage alone recovers the WMT-style score whenever a strictly comparable number is needed. For the WMT25 numbers (§5.7) we run the shared task’s official script rather than our own pipeline, so that comparison stays on WMT’s own terms.

Extractor	Recall	Terms/sent.
LLM (Qwen3-4B-Instruct-2507)	95.2%	4.04
KeyBERT	84.1%	5.00
YAKE	74.5%	4.91
BERT-NER (ACTER)	37.5%	2.26

Table 1: Term extraction results on the curated test set.

Classifier	Top-1	Top-2
Embedding centroid	74%	86%
LLM (Qwen3-4B-Instruct-2507)	80%	90%

Table 2: Domain classification results on the curated test set. Embedding centroid was made using paraphrase-multilingual-mpnet-base-v2.

5 Results

5.1 Component Ablations

The LLM-based term extractor reaches 95.2% recall at 4.04 terms per sentence, clearly ahead of KeyBERT (84.1%) and YAKE (74.5%). A supervised BERT-NER model trained on ACTER (Rigouts Terryn et al., 2020) does not transfer across our ten domains and finishes at 37.5% recall. For domain classification the same pattern holds: the LLM classifier reaches 80%/90% top-1/top-2 accuracy, while an embedding-centroid baseline plateaus at 74%/86%.

5.2 Main Pipeline Comparison

On the curated test set, `rag_clf` alone lifts TSR by at least +0.15 absolute over baseline on every open model, with the largest absolute gain (+0.295) on the weakest baseline (Llama), consistent with retrieval helping most where parametric terminology knowledge is weakest (Zheng et al., 2024). `rag_clf_sft_cpo` reaches 0.845 on Hy-MT (+0.17, 25% relative) and 0.84 on Qwen (+0.26, 45% relative), as shown in Table 3. The pattern replicates on the 1,300-item wiki set (Hy-MT: 0.755→0.86; Qwen: 0.607→0.813), and the classifier-filtered variant alone already exceeds both commercial systems on TSR (Table 4: 0.852 vs. Yandex 0.764 / Google 0.769 on wiki).

5.3 Per-Domain Analysis

Table 6 breaks down Hy-MT TSR by domain (both test sets combined). The baseline varies substantially, from 0.53 (shipbuilding) to 0.91 (IT). The `rag_clf` stage yields its largest gains on shipbuilding (+0.21) and aviation (+0.23), followed by oil and gas (+0.12), while providing little or no

benefit on domains where the baseline is already strong (IT decreases slightly; law gains only +0.03). `rag_clf_sft_cpo` adds a further small gain on some domains but reverses the `rag_clf` gain on shipbuilding, energy, and finance, indicating that fine-tuning does not benefit all domains equally.

5.4 Qualitative Analysis

Individual items illustrate the four main patterns behind the aggregate numbers.

Polysemy resolved by domain filtering. In sentence (1) the aviation term *canopy* refers to the transparent cockpit cover:

(1) When he looked up through the *canopy*, Taylor could barely make out the distant black specks on the horizon.

All four models at baseline render it as *крона деревьев* (tree canopy), whereas the glossary-augmented pipeline substitutes the aviation-correct *фонарь (кабины)*. The same pattern holds for *bulkhead* in a shipbuilding context, rendered as generic *перегородки* at baseline but correctly as *переборки* under `rag_clf`.

Classifier noise. When the classifier misroutes a sentence, the glossary steers translation toward the wrong sense. In (2) the medical *vehicle* means *растворитель* (solvent):

(2) Water is often used as a *vehicle* in pharmaceuticals.

Hy-MT baseline is correct, but under `rag_clf` the misclassified domain injects a pharmacy-related gloss that shifts the output away from the solvent reading. This is the mechanism behind the classifier-noise fluency cost quantified in §5.5.

Fine-tuning: targeted gains and false substitutions. Adding SFT and CPO on top of `rag_clf` sometimes recovers terms that retrieval alone misses: aviation *wet leasing* is rendered as *аренда без экипажа* only after fine-tuning. But the CPO preference signal also amplifies unrelated glossary entries. On oil and gas *break-out torque* (*развинчивание*), `rag_clf` produces the acceptable *отвертывание*, whereas the fine-tuned variant produces *борьба с появлением прыщей на лице*, a completely different English sense of *breakout* that CPO has over-weighted. On construction *truss* (*ферма*), `rag_clf` correctly produces *ферм* but the fine-tuned variant replaces it with *кустей*.

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	0.580	0.200	0.650	0.675	0.745	0.760
rag_auto	0.755	0.385	0.780	0.825	—	—
rag_clf	0.830	0.495	0.800	0.830	—	—
rag_clf_sft	0.830	—	—	0.825	—	—
rag_clf_sft_cpo	0.840	—	—	0.845	—	—
term_substitute	0.905	0.800	0.845	0.920	—	—
rag_oracle	0.950	0.745	0.895	0.975	—	—

Table 3: Terminology success rate on the curated test set (200 items). Column headers denote base model versions: Qwen = Qwen3-4B-Instruct-2507, Llama = Llama-3.2-3B-Instruct, Gemma = TranslateGemma-4B, Hy-MT = Hy-MT2-7B (§4.3); Google / Yandex = Google Translate / Yandex Translate via their public APIs (§A). Fine-tuning (SFT/CPO) is applied only to the two strongest open models (Qwen and Hy-MT). Commercial systems are evaluated at their default settings; `rag_*` variants are not applicable to them.

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	0.607	0.280	0.674	0.755	0.769	0.764
rag_auto	0.809	0.470	0.809	0.847	—	—
rag_clf	0.810	0.476	0.812	0.852	—	—
rag_clf_sft	0.805	—	—	0.856	—	—
rag_clf_sft_cpo	0.813	—	—	0.860	—	—
term_substitute	0.901	0.841	0.913	0.919	—	—
rag_oracle	0.910	0.757	0.885	0.955	—	—

Table 4: Model versions as in Table 3. Terminology success rate on the 1,300-item wiki test set. The classifier-filtered retrieval variant alone (`rag_clf`) already exceeds both commercial reference systems on Hy-MT, indicating that the terminology gain is not specific to a difficulty-filtered distribution.

Cross-model validation and a possible size effect.

The weakest model (Llama-3.2-3B) exhibits a qualitatively different failure mode: it frequently fails to translate specialized terms at all and instead transliterates or copies the English word. Shipbuilding *bilge* (лъяло) is rendered as *билг*; *pig* (pipeline scraper, *скребок*) becomes *интеллектуальный свинья* (“intelligent pig”); and *bollard* (кнехт) is copied as *BOLLARD*. This failure mode is absent from the three stronger models, which is consistent with a lower capacity bound below which a model cannot integrate glossary terms into fluent Russian output even when the correct term is explicitly provided in the prompt. We stop short of calling this a confirmed size effect, however: Llama-3.2-3B is the only sub-4B model in our comparison, and it also differs from the other three base models in pretraining data and instruction-tuning recipe, none of which we control for. Isolating parameter count from these family-specific differences would require an ablation across multiple sizes within a single model family, which we did not run (see the Limitations section).

5.5 Fluency–Terminology Trade-off

Surface metrics (chrF2++, BLEU) move in the same direction as TSR, but reference-free QE

metrics move the opposite way. On Hy-MT curated, COMET-Kiwi drops from 0.800 (baseline) to 0.729 (`rag_clf_sft_cpo`) and MetricX-QE rises from 1.285 to 2.091; the same effect appears on wiki (Table 7: 0.839 → 0.762 and 0.914 → 1.809). Both QE metric families and both surface- and reference-free evaluation agree on the sign of the effect, ruling out a single-metric artifact.

Decomposing the cost via `rag_oracle` isolates classifier noise from glossary content, with different metrics showing different amounts of recovery. The cleanest decomposition is Hy-MT MetricX on the curated set: under gold domain classification MetricX is 2.57, essentially the baseline value of 2.60, so the entire `rag_clf` MetricX cost of +0.58 is attributable to classifier noise rather than glossary content, and the subsequent SFT+CPO stages add a further +0.31 MetricX on top – making classifier noise the larger of the two additive components. On the reference-free QE side, the same oracle diagnostic recovers only part of the drop: on Hy-MT wiki COMET-Kiwi baseline is 0.839 and `rag_oracle` sits at 0.768, closer to `rag_clf` (0.780) than to baseline, so on this metric a substantial residual fluency cost survives perfect domain classification. We take the two decompositions together as evidence that

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	0.594 / 0.603	0.240 / 0.269	0.662 / 0.671	0.715 / 0.745	0.757 / 0.766	0.762 / 0.763
rag_auto	0.782 / 0.763	0.428 / 0.459	0.795 / 0.805	0.836 / 0.844	—	—
rag_clf	0.820 / 0.813	0.486 / 0.479	0.806 / 0.811	0.841 / 0.849	—	—
rag_clf_sft	0.818 / 0.809	—	—	0.841 / 0.852	—	—
rag_clf_sft_cpo	0.827 / 0.817	—	—	0.853 / 0.858	—	—
term_substitute	0.903 / 0.901	0.820 / 0.835	0.879 / 0.904	0.920 / 0.919	—	—
rag_oracle	0.930 / 0.915	0.751 / 0.755	0.890 / 0.887	0.965 / 0.958	—	—

Table 5: Model versions as in Table 3. Overall TSR averaged across the curated (200 items) and wiki (1,300 items) test sets, reported as macro / micro. Macro treats the two sets equally; micro weights by item count.

Domain	base	rag_clf	+sft_cpo
IT	0.91	0.85	0.85
Aviation	0.63	0.86	0.96
Medicine	0.85	0.91	0.91
Oil and gas	0.63	0.75	0.76
Agriculture	0.73	0.85	0.86
Construction	0.71	0.83	0.83
Shipbuilding	0.53	0.74	0.72
Finance	0.86	0.92	0.89
Energy	0.81	0.89	0.86
Law	0.82	0.85	0.84

Table 6: Per-domain TSR on Hy-MT, both test sets combined.

classifier noise is the dominant but not the only contributor to the QE regression.

We consider two non-exclusive explanations for the residual gap that oracle cannot close. First, calibration: COMET-Kiwi and MetricX-QE are trained primarily on general-domain human judgments where naturalness is weighted more heavily than terminology fidelity, so they may under-reward the more constrained but terminology-correct output that any glossary-augmented pipeline produces – consistent with Zouhar et al. (2024)’s finding that fine-tuned MT metrics degrade sharply outside their training domain. Second, a genuine fluency cost: inserting a specialized Russian lemma from the glossary can produce awkward collocations even when it is the domain-correct translation, so at least some of the residual may reflect a real drop in naturalness rather than a metric artifact. Distinguishing the two cleanly would require a human evaluation we did not run (Limitations), and we do not claim to isolate them here. The counter-example that reinforces both readings is `term_substitute`, which attains the second-highest TSR on Hy-MT curated (0.92) but scores worst of all variants on both QE metrics – source-side substitution produces terminology-correct but noticeably less fluent Russian, and both explanations predict this direction.

5.6 Fine-Tuning Contribution (RQ2)

Table 5 decomposes the fine-tuning delta over `rag_clf` into its SFT and CPO components. SFT alone is essentially $\Delta = 0$ on both fine-tuned models and both test sets ($-0.005 / +0.005$ on Hy-MT curated / wiki; $0.0 / -0.005$ on Qwen). CPO adds $+0.02 / +0.004$ on Hy-MT and $+0.01 / +0.008$ on Qwen. Neither stage restores the QE-metric fluency cost introduced by `rag_clf`; MetricX on Hy-MT curated is unchanged from `rag_clf` 3.18 to `rag_clf_sft` 3.20 and rises further to 3.49 under `rag_clf_sft_cpo`. SFT therefore contributes neither the expected fluency recovery nor a terminology gain.

We attribute the muted CPO effect (relative to the gains reported for ALMA-R on general translation) to a bounded-headroom mechanism. With the gold-domain glossary already in the prompt under `rag_clf`, the residual headroom for fine-tuning is capped by cases where the base model fails to use a glossary entry already present in its context. The `rag_oracle` diagnostic sets this gap at $+0.13$ (curated) / $+0.10$ (wiki) above `rag_clf`, of which CPO recovers roughly one-sixth. SFT is retained not for its evaluation contribution but as a training-stability prerequisite: preliminary runs in which CPO was applied directly on top of the base model reproduced the reward-margin collapse pattern reported by Xu et al. (2024), and only the SFT warm-up on the chosen translations stabilised the CPO loss.

5.7 External Validation on WMT25

We validate on WMT25 because it is the most recent iteration of the WMT Terminology Translation Task that includes English–Russian; the current WMT26 edition covers Spanish–Basque, English–Polish, and Traditional Chinese–English, so no contemporaneous en–ru terminology benchmark exists. Using the official WMT25 dictionaries directly (bypassing our own extraction/classification/retrieval

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	0.814 / 1.394	0.685 / 3.109	0.825 / 1.119	0.839 / 0.914	0.836 / 1.130	0.840 / 1.019
rag_auto	0.795 / 1.668	0.646 / 4.457	0.809 / 1.396	0.766 / 1.822	—	—
rag_clf	0.796 / 1.625	0.636 / 4.592	0.808 / 1.362	0.780 / 1.553	—	—
rag_clf_sft	0.789 / 1.721	—	—	0.807 / 1.447	—	—
rag_clf_sft_cpo	0.781 / 1.842	—	—	0.762 / 1.809	—	—
term_substitute	0.780 / 1.691	0.663 / 3.411	0.793 / 1.395	0.806 / 1.142	—	—
rag_oracle	0.792 / 1.605	0.606 / 5.995	0.811 / 1.333	0.768 / 1.610	—	—

Table 7: Model versions as in Table 3. Reference-free QE metrics on the wiki test set: COMET-Kiwi ($\uparrow$) / MetricX-QE ($\downarrow$). `rag_oracle` recovers only part of the COMET-Kiwi drop (Hy-MT baseline 0.839 vs. `rag_oracle` 0.768, closer to `rag_clf` 0.780 than to baseline); the cleaner oracle-vs-baseline recovery holds for MetricX on the curated set (§5.5). We read the two decompositions together as classifier noise being the dominant but not the only contributor to the QE regression.

Condition	baseline	sft_cpo
noterm (chrF2++)	53.11	52.85
proper (chrF2++ / TSR)	59.32 / 0.918	60.53 / 0.931
random (chrF2++ / TSR)	56.47 / 0.911	54.44 / 0.917

Table 8: WMT25 Terminology Translation Task, Hy-MT results (Hy-MT = Hy-MT2-7B, §4.3; `sft_cpo` is the released `maryleya/hymt2-7b-sft-cpo` checkpoint, §A).

stages, since gold terminology is supplied), the SFT+CPO Hy-MT checkpoint improves +1.21 chrF2++ and +0.014 TSR over baseline under the *proper* terminology condition (Table 8), confirming that the gains transfer to an external benchmark. Under *noterm*, the two configurations are within noise of each other, as expected, since there is no glossary signal for fine-tuning to express itself through. Under *random* terminology, TSR is essentially unchanged (0.911 vs. 0.917), indicating the model does not blindly substitute unrelated glossary entries, though chrF2++ drops by 2.03 – a mild distractor sensitivity consistent with known RAG failure modes (Barnett et al., 2024).

We reuse WMT25’s released dictionaries and official scoring script to evaluate our own system offline, rather than through WMT’s centralized, ranked evaluation – so we lack a like-for-like basis for placing our absolute chrF2++/TSR values on the official leaderboard. One qualitative comparison is worth flagging though: WMT25’s own headline finding is that proper terminology helps more for already-strong participating systems (Se-menov et al., 2025), our within-model results show that the gain is largest where the underlying model or domain is weakest (§5.4). The two findings operate on different axes (between-system comparison across a leaderboard vs. within-model comparison

across domains) and are not strictly contradictory, but we do not claim a consistency we cannot verify.

6 Discussion

6.1 RQ1: Retrieval Improves Terminology

Domain-classified glossary retrieval substantially improves EN–RU terminology accuracy on all four open base models, with the largest gains where the base model’s parametric terminology knowledge is weakest (Zheng et al., 2024) – both across models (largest gain on the weakest baseline, Llama) and, at a finer grain, across domains within a single model (largest gains on Hy-MT’s weakest domains; no benefit on its strongest, IT). The pattern transfers to the WMT25 external benchmark under the *proper* glossary condition (§5.7).

6.2 RQ2: Preference Fine-Tuning is Bounded by the Prompt

SFT and CPO deliver far less than the ALMA-R precedent (Xu et al., 2024) would suggest, which we attribute to a bounded-headroom mechanism rather than a design flaw: ALMA-R was developed for a setting with no inference-time glossary access, so the entire burden of producing the correct term falls on the fine-tuned weights. Once `rag_clf` already supplies the correct glossary entry in the prompt, the only headroom left for fine-tuning is cases where the base model fails to apply an entry already in front of it – a gap the `rag_oracle` diagnostic quantifies, of which CPO recovers roughly one-sixth (§5.5). This also explains why the CPO fluency cost partly compounds rather than adds to the classifier-noise cost: CPO amplifies whatever the classifier routed to, correct or not.

6.3 Practical Implications

The fluency–terminology trade-off maps onto a simple configuration rule: default to `rag_clf` (optionally with SFT+CPO) on domains where the baseline is weak and terminology fidelity is high-stakes (legal, medical, oil and gas, shipbuilding, aviation), and default to the plain baseline where it is already strong (IT, law), since the modular design supports switching per domain without a separate model. Where a translator-in-the-loop workflow is feasible, `rag_oracle` is the stronger option: because the dominant fluency cost comes from classifier noise rather than glossary content (§5.5), supplying the domain manually recovers most of the lost fluency while keeping the terminology gain. Either configuration already outperforms both commercial systems on TSR while running on a single consumer GPU (§A), making the pipeline a practical option where data confidentiality or API cost rules out commercial cloud MT.

6.4 Position Relative to Related Work

The cross-model TSR improvement under `rag_clf` confirms the general principle established by DiPMT (Ghazvininejad et al., 2023), Moslem et al. (2023), and DragFT (Zheng et al., 2024) that prompt-level dictionary injection improves specialized translation, extending it to EN–RU across ten domains. Unlike WMT25 retrieval-augmented systems that assume the dictionary is supplied (Guttmann et al., 2025; Nieminen et al., 2025), our pipeline closes the loop with its own extraction and classification stages, at the cost of the classifier-noise component in §5.5; and unlike constrained decoding (Hokamp and Liu, 2017; Hasler et al., 2018), our prompt-level conditioning leaves the model free to fall back to a fluent translation where the glossary is absent or wrong.

The systematic divergence between surface metrics (which agree with TSR) and reference-free QE metrics (which penalise the same translations) admits two non-exclusive readings, as detailed in §5.5: a calibration mismatch on specialized-domain outputs, and a genuine but small fluency cost from inserting glossary lemmas into contexts that were not necessarily selected for them. We do not claim to isolate the two without a human evaluation (Limitations), only that the QE regression is bounded above by what the oracle diagnostic and the surface metrics jointly imply.

Limitations

Training pool size and source. The fine-tuning pool contains approximately 10,000 items after the filtering cascade, small relative to the tens to hundreds of thousands of pairs used by ALMA and ALMA-R (Xu et al., 2023, 2024). At this scale SFT may have too few examples to reshape the output distribution of a 7B model and the CPO preference signal may be similarly undersupplied. A substantial share of the pool is mined from WikiMatrix (Schwenk et al., 2021), whose Russian sides can themselves be translations from English and may not carry fully natural Russian fluency – a plausible contributor to the QE-metric fluency regression.

Curated test set size. With 200 items, the curated set is small. TSR gains are large enough to stand well above evaluation noise, but the neural-metric deltas are close to it and should be read together with the 1,300-item wiki set and the WMT25 numbers, both of which show the same trade-off at a smaller magnitude.

Knowledge-base quality and provenance. Multitran, the main source behind the KB, has no editorial moderation – entries are added by volunteers. Individual glosses can be noisy or dated, and the classifier-filtering step will not catch these on domains where Multitran is the only supplier of candidates. The same volunteer sourcing shapes what is covered and which variants get treated as canonical; we did not audit these biases directly.

Single language pair. Our experiments cover only English–Russian. Nothing in the pipeline design assumes Russian specifically, but we have not tested it on other morphologically rich targets, on lower-resource languages, or on pairs that lack a Multitran-scale open termbase.

Model-size confound. The size-floor observation in §5.4 rests on a single small model, Llama-3.2-3B, compared against three larger models from three different model families. We cannot separate the effect of parameter count from other differences between Llama and the other base models – pre-training corpus, instruction-tuning recipe, tokenizer. A controlled ablation varying model size within a single model family would be needed to confirm whether the observed failure mode is a genuine capacity floor rather than a Llama-specific artifact; we leave this to future work.

Ethics Statement

All source materials for the knowledge base and both test sets come from openly licensed or publicly accessible dictionaries and parallel corpora. Nothing in the data is personally identifiable or otherwise sensitive. Because Multitran is user-contributed, domain coverage and the choice of preferred variants inevitably reflect the editorial habits of its volunteer contributors, and some of that bias carries over into what we release. We publish everything to make the results reproducible and easier to build on.

References

- Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrias, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza. 2020. ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Scott Barnett, Stefanus Kurniawan, Srikanth Thudumu, Zach Brannelly, and Mohamed Abdelrazek. 2024. Seven failure points when engineering a retrieval augmented generation system. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering – Software Engineering for AI (CAIN)*.
- Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*.
- Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. Training neural machine translation to apply terminology constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.
- Marjan Ghazvininejad, Hila Gonen, and Luke Zettlemoyer. 2023. Dictionary-based phrase-level prompting of large language models for machine translation. *arXiv preprint arXiv:2302.07856*.
- Kamil Guttman, Zofia Rostek, Adrian Charkiewicz, Aleksander Solarski, Mikolaj Pokrywka, and Artur Nowakowski. 2025. Laniqo at WMT25 general translation task: Self-improved and retrieval-augmented translation. In *Proceedings of the Tenth Conference on Machine Translation (WMT)*.
- Eva Hasler, Adrià de Gispert, Gonzalo Iglesias, and Bill Byrne. 2018. Neural machine translation decoding with terminology constraints. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Juraj Juraska, Mara Finkelstein, Daniel Deutsch, Aditya Siddhant, Mehdi Mirzazadeh, and Markus Freitag. 2023. MetricX-23: The Google submission to the WMT 2023 metrics shared task. In *Proceedings of the Eighth Conference on Machine Translation (WMT)*.
- Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2021. Nearest neighbor machine translation. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*.
- Mikhail Korobov. 2015. Morphological analyzer and generator for Russian and Ukrainian languages. In *Analysis of Images, Social Networks and Texts (AIST)*, volume 542 of *Communications in Computer and Information Science*.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Yasmin Moslem, Rejwanul Haque, and Andy Way. 2023. Adaptive machine translation with large language models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation (EAMT)*.
- Tommi Nieminen, Jörg Tiedemann, and Sami Virpioja. 2025. Implementing and evaluating multi-source retrieval-augmented translation. In *Proceedings of the Tenth Conference on Machine Translation (WMT)*.

- Arturo Oncevay, Charese Smiley, and Xiaomo Liu. 2025. The impact of domain-specific terminology on machine translation for finance in European languages. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*.
- Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers (WMT)*.
- 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 *Advances in Neural Information Processing Systems (NeurIPS)*.
- 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)*.
- Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C. Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte M. Alves, Alon Lavie, et al. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*.
- Ayla Rigouts Terryn, Véronique Hoste, and Els Lefever. 2020. In no uncertain terms: A dataset for monolingual and multilingual automatic term extraction from comparable corpora. *Language Resources and Evaluation*, 54(2):385–418.
- Danielle Saunders. 2022. Domain adaptation and multi-domain adaptation for neural machine translation: A survey. In *Journal of Artificial Intelligence Research*, volume 75.
- Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.
- Kirill Semenov and Ondřej Bojar. 2022. Automated evaluation metric for terminology consistency in MT. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*.
- 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 (WMT)*, pages 554–576.
- Jiaan Wang, Fandong Meng, Yingxue Zhang, and Jie Zhou. 2024. Retrieval-augmented machine translation with unstructured knowledge. *arXiv preprint arXiv:2412.04342*.
- Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*.
- Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.
- Jiawei Zheng, Hui Hong, Feng Liu, Xiaoli Wang, Jiahui Su, Yu Liang, and Shuting Wu. 2024. Fine-tuning large language models for domain-specific machine translation. *arXiv preprint arXiv:2402.15061*.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Vilém Zouhar, Shuoyang Ding, Anna Currey, Tatyana Badeka, Jinyuan Wang, and Brian Thompson. 2024. [Fine-tuned machine translation metrics struggle in unseen domains](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 488–500, Bangkok, Thailand. Association for Computational Linguistics.

A Reproducibility

Compute. All SFT and CPO fine-tuning fits on a single A100 (40 GB). At inference, each of the four open models, along with the RuQwen-2.5-32B judge, runs on one 24 GB consumer GPU.

Fine-tuning setup. We use LoRA with $r=64$ and $\alpha=128$. SFT runs for 2 epochs, learning rate 10^{-4} , on the chosen translations, with the same glossary block in the training prompt that is used at inference. CPO then runs for 1 epoch at 5×10^{-6} with $\beta=0.1$. Decoding is greedy throughout.

Commercial systems. Outputs from Google Translate and Yandex Translate were pulled through their public APIs. The exact timestamps and API versions are logged in the repository.

Randomness and reproducibility. Open-model inference is deterministic (greedy decoding), so it repeats exactly. Fine-tuning uses a fixed seed. Everything needed to rerun the experiments – seed, full hyperparameters, KB, both test sets, evaluation scripts, verbatim prompts – is at https://github.com/Maryleya/RAG_System_for_Specialized_Terms.

Model checkpoints. The four fine-tuned LoRA adapters are released on the Hugging Face Hub as `maryleya/hymt2-7b-sft`, `maryleya/hymt2-7b-sft-cpo`, `maryleya/qwen3-4b-sft`, and `maryleya/qwen3-4b-sft-cpo`. Each adapter loads on top of its declared base model (`tencent/Hy-MT2-7B` or `Qwen/Qwen3-4B-Instruct-2507`) via the `peft` library.

B Term Extractor Prompt

The term extractor is a few-shot prompt to Qwen3-4B-Instruct-2507, structured as a system instruction, three in-context examples from heterogeneous domains, and the source sentence.

```
System: You are a domain
terminology specialist.
Extract specialized
technical/professional terms
from the given English sentence.
A specialized term is a noun
or noun phrase from a specific
field that has a domain-specific
meaning, not a common word.
Output terms one per line.
Output ONLY the terms - no
explanations, no numbering.
```

```
Example 1:
Input: The patient was
diagnosed with chronic
obstructive pulmonary disease
and prescribed long-acting
bronchodilators.
Output:
chronic obstructive pulmonary
disease
long-acting bronchodilators
```

```
Example 2:
Input: The contract included
a force majeure clause and was
governed by the law of England
and Wales.
Output:
```

```
force majeure clause
law of England and Wales
```

```
Example 3:
Input: The wellhead Christmas
tree is rated for 10,000
psi working pressure during
production testing.
Output:
wellhead Christmas tree
working pressure
production testing

Input: {SOURCE_SENTENCE}
Output:
```

C Domain Classifier Prompt

The classifier is a two-step synthetic few-shot prompt. Before the first classification pass, Qwen3-4B-Instruct-2507 generates three example English sentences per domain from a keyword description (Appendix C.1). The resulting 30 (sentence, domain) pairs are reused as in-context labelled examples in the classifier prompt for every test sentence (Appendix C.2). Domain names are kept in their mixed English–Russian form to match the KB labels. Output is parsed as a fixed `Primary/Alternative` pair; an alias map and a difflib fuzzy match recover misspellings, transliterations, and Latin–Cyrillic confusables at parse time.

C.1 Synthetic Example Generation

```
System: You are an expert
technical writer. Given
a domain and its keywords,
generate authentic example
English sentences that would
appear in real specialized texts.
Output EXACTLY 3 sentences as a
numbered list.

User:
Domain: {DOMAIN}
Keywords/topics:
{DOMAIN_DESCRIPTION}

Generate 3 distinct English
sentences.
```

The ten keyword descriptions are: *IT* (software, programming, computers, networks, code); *авиация* (aircraft, aviation, flight, runway, fuselage); *медицина* (medical, anatomy, diseases, symptoms, treatment, surgery); *нефтегаз* (oil and gas extraction, drilling, wells, reservoirs, pipelines); *сельское хозяйство* (agriculture, crops, livestock, soil, fertilizer, irrigation); *строительство* (construction, buildings, concrete, walls, beams); *судостроение* (shipbuilding, naval, hulls, navigation, ports); *финансы* (fi-

nance, banking, investments, securities, accounting); *энергетика* (energy, electricity, grids, turbines, transformers); *юриспруденция* (law, contracts, courts, regulations, liability).

C.2 Classifier Prompt

System: You are a domain classifier. You will be given a list of allowed domain names, descriptions, a few labeled example sentences per domain, and a target English sentence. Output EXACTLY two lines:
Primary: <domain>
Alternative: <domain>
where both <domain> values are copied verbatim from the allowed list (including Cyrillic). Primary is your best guess; Alternative is the second-best guess (must differ from Primary). Do NOT translate, transliterate, abbreviate, or rephrase domain names. No explanations.

User:
Allowed domain names:
{TEN_DOMAIN_NAMES}
Descriptions:
{TEN_KEYWORD_DESCRIPTIONS}
Labeled examples:
{30_SYNTHETIC_PAIRS}
Target sentence:
{SOURCE_SENTENCE}
Output:

D Translation Prompts

Three prompt variants are used across all open-source translation models; only the glossary contents differ across models and conditions.

D.1 Baseline

Translate the following text to Russian. Provide only the translation, without any explanations or comments.

{SOURCE_SENTENCE}

D.2 Term Substitution

System: You are a professional English-to-Russian translator. The source sentence may already contain some Russian words - keep them as-is and adapt their grammatical form (case, number, gender) to fit the Russian sentence naturally. Translate the rest of the sentence to fluent Russian. Output ONLY the Russian translation.

User: {SOURCE_WITH_
RU_TERM_SUBSTITUTED}

D.3 Glossary-Augmented Translation

System: You are a professional English-to-Russian translator specializing in technical texts. You will be given a source English sentence and a dictionary of specialized terms with their Russian translations.

RULES:

1. Strongly prefer the Russian translation from the dictionary when an English term appears in the source.
2. When several variants under different domains - choose the one whose domain best fits the context.
3. Adapt the translation to the correct grammatical case, number, gender - keep the lemma.
4. For words NOT in the dictionary, translate naturally.
5. Output ONLY the Russian translation - no explanations.

User:

Dictionary:
{ENGLISH_TERM_1}:
- {RU_TRANSLATION_1A}
[{DOMAINS}]
- {RU_TRANSLATION_1B}
[{DOMAINS}]
...
Source sentence:
{SOURCE_SENTENCE}
Translation:

D.4 LLM-as-Judge Prompt for TSR

The second-stage verifier (RuQwen-2.5-32B, temperature 0) is invoked only on terms that the first-stage lemma matcher classified as not-matched, to recover valid synonyms and paraphrases. The prompt is in Russian and is structured around an explicit FAIL-only criterion list with seven worked PASS/FAIL examples across IT, medicine, ship-building, and aviation.

System: You are evaluating whether the translator correctly conveyed the meaning of a specialized term.
Given: domain, English sentence, gold Russian term, candidate translation.
Rule: PASS if the candidate correctly expresses the term from the source - this may be the gold term (in any grammatical form) OR ANY valid Russian synonym with the same meaning in this domain. Be LENIENT toward synonyms.
FAIL only if:
- a word from a DIFFERENT domain is used ("partition" for the aviation term *bulkhead*);

- transliteration without translation ("pig" instead of "scraper");
- a generic word that loses the domain-specific meaning ("empty" for the medical term *void*);
- the term is missing or rendered with the opposite meaning.

Seven worked PASS/FAIL examples covering IT, medicine, shipbuilding, aviation, and oil & gas omitted for brevity; released in full at the repository URL above.

```

Answer with STRICTLY one word:
PASS or FAIL.
User:
Domain: {DOMAIN}; source
(EN): {SOURCE_EN}; specialized
term: {TERM_EN} → gold
Russian: {TERM_RU}; candidate:
{CAND_RU}.
Answer:

```

E Strict Lemma-Match TSR

Tables 9 and 10 report the first-stage (pymorphy3 lemma-match) TSR alone, without the LLM-as-judge second stage. This is exactly the criterion used by the WMT Terminology Shared Task and is reproducible without any LLM call. The direction of every claim in the main text – retrieval improves over baseline, oracle sets the upper bound, fine-tuning adds a small residual, commercial systems trail retrieval-augmented open models – holds under this stricter criterion as well; the absolute numbers are correspondingly lower because same-domain synonyms that the KB and the judge both accept do not count here.

F Reference-Based Surface Metrics on the Curated Set

Table 11 reports chrF2++ and BLEU on the 200-item curated set, aligned to the pipeline variants used elsewhere. These are the reference-based counterparts to the reference-free QE metrics in Table 7. On Hy-MT the strongest retrieval-augmented variant `rag_clf_sft_cpo` lifts chrF2++ from 45.69 (baseline) to 50.52 and BLEU from 19.81 to 22.75, moving in the same direction as TSR.

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	0.190	0.070	0.245	0.240	0.410	0.385
rag_auto	0.555	0.280	0.540	0.650	—	—
rag_clf	0.655	0.395	0.565	0.665	—	—
rag_clf_sft	0.645	—	—	0.600	—	—
rag_clf_sft_cpo	0.680	—	—	0.705	—	—
term_substitute	0.790	0.750	0.650	0.755	—	—
rag_oracle	0.825	0.630	0.705	0.865	—	—

Table 9: First-stage strict lemma-match TSR on the curated test set (200 items). Complements Table 3: the two-stage combined number there additionally credits same-domain synonyms accepted by RuQwen-2.5-32B.

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	0.067	0.055	0.075	0.086	0.031	0.092
rag_auto	0.489	0.277	0.415	0.568	—	—
rag_clf	0.482	0.288	0.433	0.549	—	—
rag_clf_sft	0.489	—	—	0.498	—	—
rag_clf_sft_cpo	0.569	—	—	0.625	—	—
term_substitute	0.714	0.762	0.688	0.651	—	—
rag_oracle	0.668	0.601	0.497	0.760	—	—

Table 10: First-stage strict lemma-match TSR on the wiki test set (1,300 items). Complements Table 4.

Variant	Qwen	Llama	Gemma	Hy-MT	Google	Yandex
baseline	42.87 / 18.38	34.96 / 13.56	45.39 / 19.87	45.69 / 19.81	50.37 / 25.58	49.63 / 23.88
rag_auto	46.27 / 20.25	36.66 / 14.78	47.49 / 20.86	45.51 / 17.35	—	—
rag_clf	46.07 / 20.11	35.90 / 13.80	47.45 / 20.79	45.92 / 17.21	—	—
rag_clf_sft	46.90 / 20.86	—	—	48.00 / 22.79	—	—
rag_clf_sft_cpo	46.10 / 20.02	—	—	50.52 / 22.75	—	—
term_substitute	46.05 / 19.71	37.99 / 14.58	46.59 / 20.07	47.73 / 19.90	—	—
rag_oracle	46.43 / 20.48	34.24 / 12.37	47.89 / 21.71	47.61 / 19.43	—	—

Table 11: Reference-based surface metrics on the curated test set: chrF2++ (↑) / BLEU (↑). Model versions as in Table 3.