LoResMT 2026

The Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026)

Proceedings of the Workshop

March 28, 2026

The LoResMT organizers gratefully acknowledge the support from the following organizations.

In cooperation with

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 979-8-89176-366-1

Preface

Based on the success of past low-resource machine translation (MT) workshops at AMTA 2018, MT Summit 2019, AACL-IJCNLP 2020, AMTA 2021, COLING 2022, EACL 2023, ACL 2024 and NAACL 2025, we introduce the LoResMT 2026 workshop at EACL 2026 (https://2026.eacl.org/). In the past few years, machine translation (MT) performance has improved significantly. With the development of new techniques such as multilingual translation and transfer learning, the use of MT is no longer a privilege for users of popular languages. However, the goal of expanding MT coverage to more diverse languages is hindered by the fact that MT methods require large amounts of data to train quality systems. This has made developing MT systems for low-resource languages challenging. Therefore, the need for developing comparable MT systems with relatively small datasets remains highly desirable.

Despite the advancements in MT technologies, creating an MT system for a new language or enhancing an existing one still requires a significant amount of effort to gather the necessary resources. The data-intensive nature of neural machine translation (NMT) approaches necessitates parallel and monolingual corpora in various domains, which are always in high demand. Developing MT systems also requires dependable evaluation benchmarks and test sets. Furthermore, MT systems rely on numerous natural language processing (NLP) tools to preprocess human-generated texts into the required input format and post-process MT output into the appropriate textual forms in the target language. These tools include word tokenizers/de-tokenizers, word segmenters, and morphological analyzers, among others. The quality of these tools significantly impacts the translation output, yet there is a limited discourse on their methods, their role in training different MT systems, and their support coverage in different languages.

LoResMT is a platform that aims to facilitate discussions among researchers who are working on machine translation (MT) systems and methods for low-resource, under-represented, ethnic, and endangered languages. The goal of the platform is to address the challenges associated with the development of MT systems for languages that have limited resources or are at risk of being lost.

This year, LoResMT received 57 papers covering many languages spoken worldwide. We have archived 15 scientific research papers from direct submission, 1 scientific research paper from ARR commitment and 9 system descriptions. Aside from the research and shared task papers, LoResMT also featured two invited talks. These talks allowed participants to hear from experts in the field of MT and learn about the latest developments and challenges in MT for low-resource languages.

The program committee members play a crucial role in ensuring the success of the peer-review workshop. They review the submissions and provide constructive feedback to help the authors refine their papers and ensure they meet the set standards. Without their dedication, expertise, and hard work, the workshop would not be possible. The authors who submitted their work to LoResMT are also an integral part of the workshop's success. Their research and contributions offer new insights into the field of machine translation for low-resource languages, and their participation enriches the discussions and fosters collaboration. We are sincerely grateful to both the program committee members and the authors for their invaluable contributions and for making LoResMT a success.

Atul, Chao, Kat, Nathaniel
(On behalf of the LoResMT chairs)

Organizing Committee

Workshop Chairs

Atul Kr. Ojha, Data Science Institute, Insight Research Ireland Centre for Data Analytics, University of Galway
Chao-hong Liu, Potamu Research Ltd
Ekaterina Vylomova, University of Melbourne, Australia
Flammie Pirinen, UiT Norgga árktalaš universitehta
Jonathan Washington, Swarthmore College
Nathaniel Oco, De La Salle University
Xiaobing Zhao, Minzu University of China

Program Committee

Abigail Walsh, ADAPT Centre, Dublin City University, Ireland
Aishwarya Jadhav, University of California, San Diego
Alberto Poncelas, Rakuten, Singapore
Ali Hatami, University of Galway
Alina Karakanta, Leiden University
Aswarth Abhilash Dara, Apple
Atul Kr. Ojha, University of Galway & Panlingua Language Processing LLP
Chao-hong Liu, Potamu Research Ltd
Constantine Lignos, Brandeis University, USA
Daan van Esch, Google
Ekaterina Vylomova, University of Melbourne, Australia
Flammie Pirinen, UiT Norgga árktalaš universitehta
Gaurav Negi, University of Galway
John Philip McCrae, University of Galway
Koel Dutta Chowdhury, Universität des Saarlandes
Manoj Yadav, Amazon
Mathias Müller, University of Zurich
Majid Latifi, University of York
Nathaniel Oco, De La Salle University
Pengwei Li, Meta
Rico Sennrich, University of Zurich
Sardana Ivanova, University of Helsinki
Sourabrata Mukherjee, Microsoft Research India
Surangika Ranathunga, Massey University
Valentin Malykh, International IT University
Yasmin Moslem, Trinity College Dublin

Keynote Talk: How (Not) to Find Errors in LLM Outputs

Ondřej Dušek
Institute of Formal and Applied linguistics, Charles University, Prague (Czech Republic)

Abstract: While LLMs have substantially improved the quality of generated texts, they still tend to make errors in their outputs, which can be subtle and harder to find than for older approaches. This needs to be reflected in the evaluation, where standard metrics or simple scores may not capture errors easily. While human evaluation should produce better results, we find a lot of inconsistency and underspecification in practice.

Building on previous works in machine translation, we examine annotating individual spans of texts for errors in order to get more detailed evaluation feedback. We explore span annotation through both human evaluation and LLM-as-judge evaluation. We provide a unified interface for both LLM and human authored error annotations, we examine different methods of obtaining LLM-annotated spans and introduce LLM ensembles for higher robustness. We directly compare LLMs and humans on the same task, finding that LLMs are able to reach high correlation with human assessments and, depending on the domain, can match trained human crowd workers in performance. However, we also report many caveats on both the human and LLM side, and we discuss potential further improvements of the evaluation setup.

Bio: Ondřej Dušek is an Assistant Professor at Charles University in Prague, working on natural language generation and human-computer dialogue. His research focuses on generative language models including large language models, mostly applied to the data-to-text and dialogue response generation tasks. He is specifically interested in evaluating the quality of generated content, especially its semantic accuracy.

After obtaining his PhD in Prague, Ondřej spent 2 years as a postdoc at Heriot-Watt University in Edinburgh. Back in Prague, he is currently the PI of an ERC Starting Grant which aims to produce fluent, accurate and explainable natural language generation systems.

Keynote Talk: TBD

TBD
TBD

Abstract:

Bio: TBD

Table of Contents

Program

Saturday, March 28, 2026

09:00 - 09:10 *Opening Remarks*

09:10 - 10:10 *Invited Talk 1: Ondřej Dušek (Institute of Formal and Applied Linguistics, Charles University, Prague (Czech Republic))*

10:10 - 10:30 *Session 1: Findings of Turkic Low Resource Machine Translation Challenge*

10:30 - 11:00 *Coffee/Tea Break*

11:00 - 12:30 *Session 2: Scientific Research Papers*

Are Small Language Models the Silver Bullet to Low-Resource Languages Machine Translation?
Yewei Song, Lujun LI, Cedric Lothritz, Saad Ezzini, Lama Sleem, Niccolo' Gentile, Radu State, Tegawendé F. Bissyandé and Jacques Klein

Tao–Filipino Neural Machine Translation: Strategies for Ultra–Low-Resource Settings
Adrian Denzel Macayan, Luis Andrew Sunga Madridijo, Ellexandrei Esponilla and Zachary Mitchell Francisco

Building and Evaluating a High Quality Parallel Corpus for English Urdu Low Resource Machine Translation
Munief Hassan Tahir, Hunain Azam, Sana Shams and Sarmad Hussain

Text Filter Based on Automatically Acquired Vocabularies for Multilingual Machine Translation
Kenji Imamura and Masao Utiyama

Improving Indigenous Language Machine Translation with Synthetic Data and Language-Specific Preprocessing
Aashish Dhawan, Christopher Driggers-Ellis, Christan Grant and Daisy Zhe Wang

Adapting Multilingual NMT to Language Isolates: The Role of Proxy Language Selection and Dialect Handling for Nivkh
Eleonora Izmailova, Alexey Sorokin and Pavel Grashchenkov

12:30 - 14:00 *Lunch*

14:00 - 15:00 *Invited Talk 2: TBD*

Saturday, March 28, 2026 (continued)

15:00 - 16:00 *Session 3: Poster Session*

Can LLMs Translate Italy's Language Varieties?
Edoardo Signoroni and Pavel Rychlý

Balancing Fluency and Adherence: Hybrid Fallback Term Injection in Low-Resource Terminology Translation
Kurt Abela, Marc Tanti and Claudia Borg

Assessing and Improving Punctuation Robustness in English-Marathi Machine Translation
Kaustubh Shivshankar Shejole, Sourabh Deoghare and Pushpak Bhattacharyya

Can Linguistically Related Languages Guide LLM Translation in Low-Resource Settings?
Aishwarya Ramasethu, Rohin Garg, Niyathi Allu, Harshwardhan Fartale and Dun Li Chan

Navigating Data Scarcity in Low-Resource English-Tatar Translation using LLM Fine-Tuning
Ahmed Khaled Khamis

No One-Size-Fits-All: Building Systems For Translation to Bashkir, Kazakh, Kyrgyz, Tatar and Chuvash Using Synthetic And Original Data
Dmitry Karpov

DevLake at LoResMT 2026: The Impact of Pre-training and Model Scale on Russian-Bashkir Low-Resource Translation
Vyacheslav Tyurin

A Comparative Evaluation of Open-Source Models for Russian-Kazakh Translation
Gleb Shanshin

Script Correction and Synthetic Pivoting: Adapting Tencent HY-MT for Low-Resource Turkic Translation
Bolgov Maxim

Machine Translation for Low Resource Turkic Languages: English-Tatar
Alexander Dikov

Saturday, March 28, 2026 (continued)

Data-Centric Approach at the LoResMT 2026 Turkic Translation Challenge: Russian-Kyrgyz
Dmitry Novokshanov

LoResMT 2026 Shared Task System Description
Vladimir Panov

Ensemble Methods for Low-Resource Russian-Kyrgyz Machine Translation: When Diverse Models Beat Better Models
Adilet Metinov

15:30 - 16:00 *Coffee/Tea Break*

16:00 - 17:15 *Session 4: Scientific Research Papers*

Comparing LLM-Based Translation Approaches for Extremely Low-Resource Languages
Jared Coleman, Ruben Rosales, Kira Toal, Diego Cuadros, Nicholas Leeds, Bhaskar Krishnamachari and Khalil Iskarous

Context Volume Drives Performance: Tackling Domain Shift in Extremely Low-Resource Translation via RAG
David Samuel Setiawan, Raphael Merx and Jey Han Lau

Semi-Automatic construction of a Quechua-Spanish dictionary
Maximiliano Duran and Max Silberztein

A Fine-Grained Linguistic Evaluation of Low-Resource Luxembourgish–English MT
Nils Rehlinger

CTC Regularization for Low-Resource Speech-to-Text Translation
Zachary William Hopton and Rico Sennrich

17:15 - 17:25 *Closing remarks*

Are Small Language Models the Silver Bullet to Low-Resource Languages Machine Translation?

Yewei Song[♠1], **Lujun Li**[♠1], **Cedric Lothritz**[2],
Saad Ezzini[3], **Lama Sleem**[1], **Niccolo' Gentile**[4],
Radu State[1], **Tegawendé F. Bissyandé**[1], **Jacques Klein**[1],
[1]University of Luxembourg, [2]Luxembourg Institute of Science and Technology,
[3]King Fahd University of Petroleum and Minerals, [4]Foyer S.A.,
Correspondence: yewei.song@uni.lu

Abstract

Small language models (SLMs) offer computationally efficient alternatives to large language models, yet their translation quality for low-resource languages (LRLs) remains severely limited. This work presents the first large-scale evaluation of SLMs across 200 languages, revealing systematic underperformance in LRLs and identifying key sources of linguistic disparity. We show that knowledge distillation from strong teacher models using predominantly monolingual LRL data substantially boosts SLM translation quality—often enabling 2B–3B models to match or surpass systems up to 70B parameters. Our study highlights three core findings: (1) a comprehensive benchmark exposing the limitations of SLMs on 200 languages; (2) evidence that LRL-focused distillation improves translation without inducing catastrophic forgetting, with full-parameter fine-tuning and decoder-only teachers outperforming LoRA and encoder–decoder approaches; and (3) consistent cross-lingual gains demonstrating the scalability and robustness of the method. These results establish an effective, low-cost pathway for improving LRL translation and provide practical guidance for deploying SLMs in truly low-resource settings. [1].

1 Introduction

Persistent LRL underperformance Low-resource languages (LRLs) continue to face substantial challenges due to the scarcity of linguistic resources, rooted in socioeconomic, geographical, and political constraints, which limits their representation in both academic and industrial contexts (Nigatu et al., 2024); despite advances in multilingual transfer learning and pretraining approaches (Conneau et al., 2020; Artetxe and Schwenk, 2019), exemplified by No Language Left Behind (NLLB; (Costa-jussa et al., 2022)), translation quality for LRLs still lags behind that of high-resource languages (HRLs), particularly in sensitive domains such as finance and government, where privacy and offline deployment are crucial (Zhong et al., 2024). Transformer-based models (Zhao et al., 2023), whether encoder-decoder with attention (Bahdanau et al., 2015; Vaswani et al., 2017; Naveed et al., 2024) or decoder-only frameworks like GPT (Gao et al., 2022; Hendy et al., 2023), have driven progress through techniques such as back-translation (Sennrich et al., 2016), unsupervised training (Lample et al., 2018), and multilingual initiatives like OPUSMT (Tiedemann and Thottingal, 2020), yet decoder-only models often underperform for LRLs due to English-centric data distributions (Brown et al., 2020; Hasan et al., 2024), leading to inaccuracies and hallucinations (Benkirane et al., 2024), although some evidence suggests they may outperform encoder-decoder methods in certain contexts (Gao et al., 2022; Silva et al., 2024). In general, language models exhibit consistent degradation on LRLs relative to HRLs (Robinson et al., 2023), caused by unbalanced training distributions (Lankford et al., 2021), tokenization biases, and limited exposure to linguistic diversity (Shen et al., 2024), underscoring the need for targeted data augmentation, domain-specific adaptation, and specialized fine-tuning to narrow the performance gap (Elsner et al., 2024; Li et al., 2025b).

Costly, slow gigantism Furthermore, because translation is a highly common and high-frequency use case across both industry and individual users, inference with very large models (e.g., ChatGPT-scale systems) is often impractical for academic or industrial deployment due to cost and latency constraints; however, for Small Language Models (SLMs), encountering LRL inputs substantially increases hallucination rates, rendering them not only unreliable for translation but also broadly un-

[1]Tuned models are openly available https://anonymous.4open.science/r/mt_luxembourgish-408D

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 1–26
March 28, 2026

suitable for other applications that contain LRL content. Drawing inspiration from recent work on grammars versus parallel data (Aycock et al., 2025), which investigates grammar learning in the context of extremely low-resource translation, the authors conclude that nearly all models' understanding of low-resource languages stems primarily from parallel corpora rather than from grammatical descriptions or related sources. In this paper, the following research questions are formulated to empirically validate and begin to address SLMs in LRLs: **(RQ1)** How effectively can decoder-only language models address low-resource machine translation, and what performance gaps emerge across different model scales and languages? **(RQ2)** To what degree does distillation from monolingual low-resource data translate into measurable improvements in smaller large language models (LLMs) translation quality? **(RQ3)** How do varying supervised fine-tuning (SFT) configurations affect translation quality in low-resource languages, and do these configurations compromise broader model capabilities or instead yield consistent improvements across diverse LRLs?

2 LRLs' deficiencies

2.1 Situation of Language Support

Recent investigations have revealed that although LLMs are increasingly advertised as multilingual, their effective support in languages is often limited to a subset of HRLs. Moreover, systematic evaluations of language-specific performance remain scarce (for example (Lai et al., 2024; Marchisio et al., 2024; Lifewire, 2024; Ahuja et al., 2024)). Table 1 summarizes several models included in our experiments, their approximate parameter sizes, and the estimated number of languages they reportedly support. These figures are derived from official model documentation, benchmarking reports, and recent academic studies.

Despite these encouraging multilingual claims, the existing literature reveals that rigorous language-specific performance evaluations, especially for low-resource languages, are insufficient. Most current research focuses on high-resource benchmarks, leaving open critical questions about fairness and the accessibility of LLMs for diverse linguistic communities.

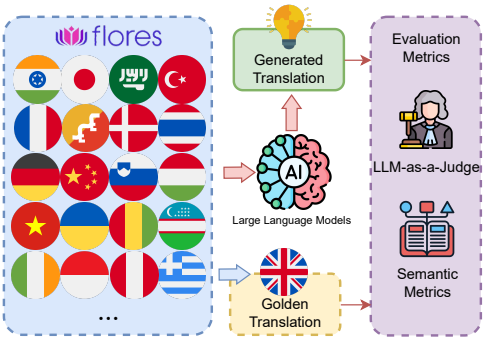


Figure 1: Evaluation pipeline

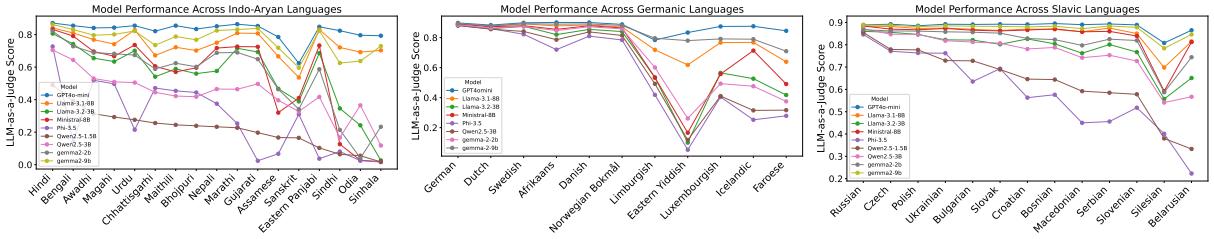
Model	Size	Languages	Date
GPT-4o-mini	—	∼25	Jul. 2024
Llama-3.1-8B-it	8B/3B	∼30	Jul. 2024
Llama-3.2-3B-it	3B	∼20	Sept. 2024
Mistral-8B-Instruct-2410	8B	∼25	Oct. 2024
Phi-3-mini-4k-instruct	4B	∼20	Apr. 2024
Phi-3.5-mini-instruct	4B	∼20	Aug. 2024
Qwen2.5 Instruct	1.5B/3B	∼25	Sept. 2024
Gemma2 Instruct	2B/9B	∼20	Jul. 2024

Table 1: Multilingual Support of LLMs

2.2 Evaluating LRLs translation Ability

We use the **FLORES-200** benchmark to systematically assess the performance of LLMs in multilingual machine translation tasks (Costa-jussa et al., 2022; Goyal et al., 2021a; Guzmán et al., 2019). FLORES-200 offers rigorously curated human-validated translation datasets across 200 languages that span diverse linguistic families and writing systems, making it highly effective for evaluating translation quality in high-resource and low-resource linguistic contexts. Our experiments leverage the full FLORES-200 dataset to comprehensively evaluate translation quality across as many languages as possible, emphasizing translations from various source languages into English.

In addition to traditional metrics, we evaluated translation quality using the **LLM-As-A-Judge** (LLMaaJ) scores (Niklaus et al., 2025), which uses a large LLM to score translations from 0 to 1 based on semantic equivalence and naturalness. A score of 1.0 denotes a perfect translation and 0.0 a totally incorrect one. In practice, we consider a score ≥ 0.8 as indicative of a good translation. Research has shown that LLMaaJ tolerates synonyms, paraphrases, and cross-linguistic structural variations, enabling it to better assess translation quality when there are multiple valid phrasings or when grammatical and typological differences (e.g., omitted



(a) Indo-Aryan (b) Germanic (c) Slavic

Figure 2: LLMaaJ scores of SLMs on Indo-Aryan/Germanic/Slavic to-English translation

pronouns) are acceptable(Zheng et al., 2023; Piergentili et al., 2025).

Regarding the LLMs investigated, as shown in Figure 1, we systematically traversed prominent proprietary APIs and open source models (refer to Table 1), presenting results using LLMaaJ metrics with quantitative semantic evaluations. Detailed LLMaaJ and BLEU scores for all source-to-English translations are provided in the Appendix Table 8 and the Appendix Table 9, and an imprecise map reveals the geographic distribution provided in the Appendix Figure 8.

2.3 Models performance in FLORES-200

We present the performance distribution in Figure 3, which visualizes more precisely the performance gap of languages across our evaluation set by linguistic family and script, thereby addressing RQ1, and complement this with the regional distribution shown in Figure 7 for finer-grained regional insights. Each bar length is calculated based on the average score, explicitly excluding the GPT4o-mini model's score to identify which LRLs are included in our experiments and how they are situated in the broader typological space.

Each bar in Figure 3 represents one language, grouped by its primary family, with bar length corresponding to the average LLMaaJ score. The figure reveals that LRLs are not evenly distributed across families: many under-resourced African, Austronesian, and Indigenous American languages cluster toward the lower end of the performance spectrum, while certain Indo-European LRLs (e.g., Luxembourgish, Maltese) perform moderately better, likely due to greater data availability or proximity to high-resource relatives.

The circular layout also highlights structural gaps in the evaluation set. Languages absent from FLORES-200—such as many North American Indigenous languages—do not appear here, not because models perform well on them, but because no evaluation data exist. This is particularly relevant for languages with small speaker populations or those concentrated in politically marginalized communities, which remain invisible in current multilingual benchmarks.

Consistent with previous work (Nekoto et al., 2020; Joshi et al., 2020), the lowest scores are observed for many Niger–Congo, Austronesian, and smaller Afro-Asiatic languages, reflecting the severe data scarcity. In contrast, LRLs in Eastern Europe and South/Southeast Asia—such as Macedonian or Sinhala—achieve slightly higher average scores, possibly benefiting from historical ties to better-supported high-resource languages. However, the overall pattern remains unchanged: LRLs across all families systematically lag behind high-resource languages, underscoring the need for targeted data collection, typologically diverse benchmarks, and bias mitigation strategies to ensure equitable progress in multilingual NLP.

2.4 Gap between Dwarf(Smaller) and Giant LLMs

Small Language Models are consistently bad in LRLs Across the Indo-Aryan, Germanic, and Slavic branches in Figure 2 (panels (a)–(c)), we observe a consistent pattern: smaller LLMs suffer a substantially larger performance drop on LRLs than on high-resource ones, while larger LLMs degrade far less. Concretely, LRLs such as Sinhala (Indo-Aryan), Luxembourgish (Germanic), and Silesian (Slavic) exhibit steep declines in smaller models but remain comparatively competitive in larger models, as visualized in Figure 2. This disparity indicates a systematic bias in current systems—particularly pronounced in smaller models—toward high-resource languages.

Solving requires training but lacks exploration Addressing this gap calls for better LRL data curation, knowledge distillation from larger LLMs, inclusive evaluation suites, and bias-

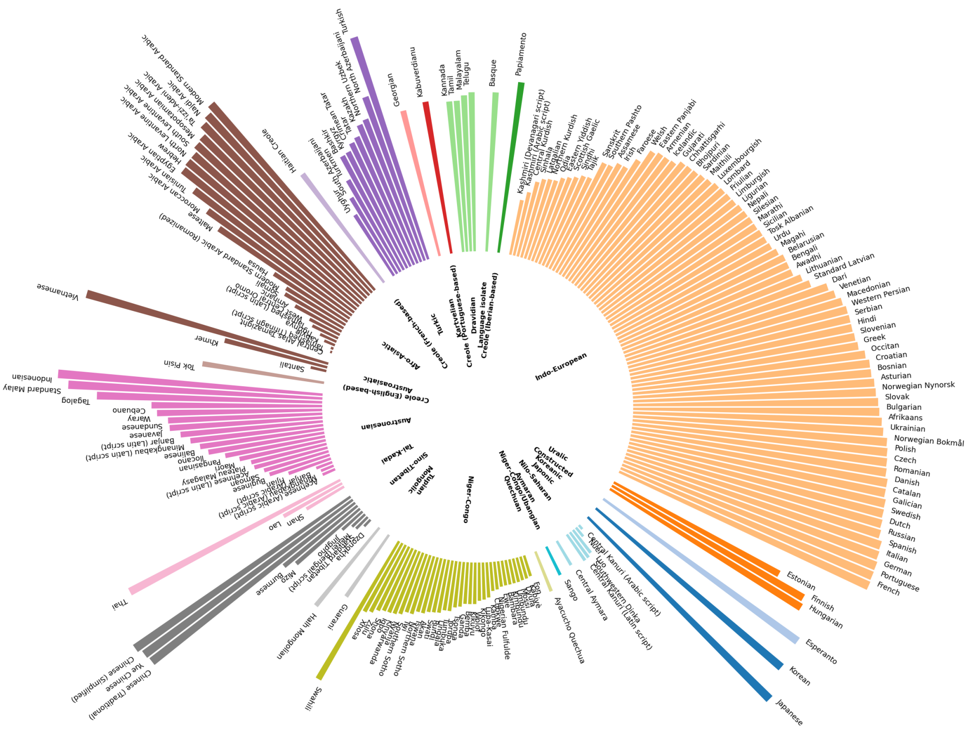


Figure 3: This circle plot illustrates the "Low-Resource" linguistic performance gap across language families. A complementary view of the geographical distribution of linguistic results is presented in Figure 8, which provides a clearer reference for cross-linguistic comparison.

mitigation strategies to ensure NLP benefits all language communities. According to the Universal Approximation Theorem (Hornik, 1991), if neural translation is viewed as a linear mapping between semantic spaces, small networks struggle to capture complex patterns and are more vulnerable to interference from HRL data. Thus, fine-tuning on high-quality paired data becomes especially crucial for smaller models, yet there remains a lack of comprehensive research on LRLs in SLMs.

3 Fine-tuning on LRLs – Taking Luxembourgish as a Key Example

3.1 Background and language selection

As highlighted in the previous section, several low-resource languages, such as Luxembourgish and Assamese (Figure 2), show a substantial translation quality gap among between large and small models. In this article, Luxembourgish serves as a representative case. Although officially recognized, it lacks sufficient high-quality corpora resources, leading to poor performance in SLMs. Its blend of Germanic roots and French influence adds complexity to NLP tasks. While larger LLMs handle Germanic languages reasonably well, they struggle with LRLs like Luxembourgish. Previous efforts to address this include LuxemBERT (Lothritz et al., 2022), LuxT5 (Plum et al., 2024), and LetzTranslate (Song et al., 2023), a low-resource translation system based on OPUS-MT.

To examine generalizability, we additionally include Ukrainian, Assamese and Khasi (an endangered language), both exhibiting similar linguistic and resource profiles, as supplementary tasks to broaden the scope of the analysis. Furthermore, generating LRL from English is more challenging for LLMs than in the reverse direction of previous research (Howcroft and Gkatzia, 2022). Regarding translation performance, LLMs exhibit a certain degree of fluent translation from LRL to English, but not vice versa (Gao et al., 2020). This asymmetry is also reflected to some extent in the hallucination issues observed when generating Luxembourgish, more details can be found in the appendix E.2.

3.2 Distillations and Soft-Target Quality

In our scenario, having only a Luxembourgish corpus without English translations rules out conventional parallel-corpus training approaches, accurately reflecting the typical data situation and model generation of LRLs. To bridge the gap between comprehension and generation in this low-resource scenario, we propose a distillation-based approach. Using a teacher model that demonstrates a robust understanding of Luxembourgish, we can distill its knowledge into a student model using the available

LRL single-side corpus. This process is expected to enhance the generation capabilities of the student model, enabling it to produce high-quality Luxembourgish output despite the limited data, and thus address the core challenge of low-resource language translation. According to further human labeling of our GPT-4o distillation dataset in Luxembourgish to English translation, **92%** of our samples were marked as fully correct.

3.3 Data Collection and Augmentations

For the training data set, we constructed a Luxembourg data set using multiple sources, including the LuxemBERT corpus, example sentences in the Luxembourg Online Dictionary (LOD) dataset[2], and additional news articles collected from previous research published data on RTL Ltzebuerg[3], following the LuxemBERT work.

Previous research has demonstrated that integrating dictionary entries can effectively enrich low-resource translation systems by providing explicit lexical alignments and clarifying semantic nuances. For example, Ghazvininejad's work improved translation fidelity in settings where parallel data is scarce (Ghazvininejad et al., 2023). Inspired by these findings, we also explore how the addition group of datasets with dictionary checks using LOD can complement our distillation approach as shown in Figure 4. Details of using the dictionary usage in the Appendix C.

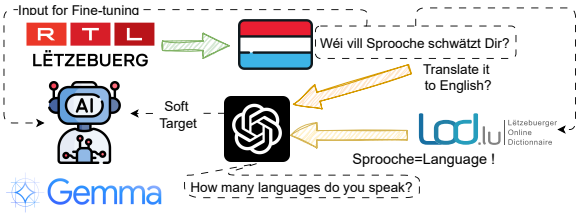


Figure 4: Pipeline of data augmentation

4 Experiments

4.1 Models and Datasets

Models The latest open-source models are used as benchmark models, and their instruction-tuned versions are utilized to leverage their general capabilities in generating dialogues and answering questions. Based on the current leaderboard for Luxembourgish proficiency in LLMs (Lothritz and Cabot, 2025), combined with the experimental results for the Germanic language group in Section 2, we select the top two base tiny models, which are Llama-3.2-3B-Instruct from Meta and Gemma-2-2b-it from Google.

Input Prompts The design of the training input templates is considered crucial. In order to prevent the model from losing its general communication and generalization abilities after instruction tuning, it is necessary for prompts to be designed in alignment with chat templates that can be understood by the model. Based on this, basic prompt testing is conducted to identify the most suitable prompt for the model. Chat-based models have been observed to be prone to losing their communication capabilities after SFT, leading to the generation of endless content and a significant increase in the likelihood of hallucinations. Therefore, in the design of the questions, the corresponding starting prompts are set at the beginning of the model responses, such as "Here is the translation: ". Through this linguistic guidance, the probability of hallucination is reduced and the model is also able to learn when to stop.

Distilled from LRL side For the training data set, the LRL monolingual corpus is used primarily as the base material, from which the LRL-to-English mapping capability is distilled from larger models. As described in Section 3.3, publicly available press datasets and dictionary example sentences are utilized as the monolingual corpus, and distillation is performed using various teacher models. Finally, the correct word-to-word mapping capability is reinforced through the lemma search to verify the dictionary content. We classify fake targets distilled into four categories: fake targets obtained by distilling facebook/nllb-200-3.3B (**D**istill-**N**LLB, DN), the fake targets obtained by distilling meta-llama/Llama-3.3-70B-Instruct. (**D**istill-**L**lama, DL), the fake targets obtained by distilling GPT-4o-mini (**D**istill-**G**PT4O, DG), and the fake targets obtained after performing dictionary checking (**D**istill-**G**PT-**D**ict-**C**hecking, DGDC). Each category contains 621,033 data samples used for model training, all having the same LRL side texts, while the corresponding fake targets are generated by different teacher models. For the validation set, the latest 300 press data entries (**Val 300**) from 2024 are used as monolingual corpus data, and the corresponding LRL entities are identified for the English mappings, thus preventing biases that may arise from the model having been trained on the

[2]https://data.public.lu/en/datasets/letzebuerger-online-dictionnaire-lod-linguistesch-daten/

[3]https://www.rtl.lu/

validation dataset. And we also do a manual check for English translations. Furthermore, we utilize the FLORES-200 benchmark as an additional validation test set.

4.2 Metrics

We evaluate translation quality using three common MT metrics (Lo et al., 2023): spBLEU, ChrF++, and the Jaccard index. spBLEU provides standardized subword-level BLEU scores using SentencePiece tokenization, enabling consistent comparison across languages. ChrF++ (Popović, 2015) combines character- and word-level n-grams and correlates well with human judgments. The Jaccard index (da F. Costa, 2021) offers a simple set-based similarity measure that is easy to interpret. For LLM-as-a-Judge evaluations, we use `google/gemma-3-27b-it` as the scoring model throughout.

4.3 Results

4.3.1 Can Small Language Models Learn?

The results in Table 2 clearly demonstrate that fine-tuning in both translation directions is highly effective. For example, the baseline EN→LB models exhibit spBLEU scores around 30, but after fine-tuning, these scores increase to nearly 38–40 values approaching our threshold for high-quality translations (spBLEU > 40). In contrast, LB→EN translations consistently score above 40, yet generating fluent Luxembourgish in the EN→LB direction remains a significant challenge. Furthermore, our experiments indicate that even a 3B model, when effectively distilled, can rival or even surpass larger models in low-resource language translation tasks. Our results indicate that GPT-4o-based distillation methods, in particular, produce substantial improvements in translation quality, confirming that parallel corpora generated by LLM represent a viable and promising strategy for supporting LRL translation tasks. In order to validate the model translation performance, we also extracted a portion of the data and asked Luxembourgers who are at least bilingual in Luxembourgish and English to label it as ground truth for data quality validation. The spBLEU score achieved with this labeled data was 51.08 on our fine-tuned Gemma-2-2b-it, showing a comparable score calculated using GPT-generated data as ground truth. Regarding the LLMaaJ score of the model, we obtained performance evaluation results and trends that are largely consistent with those of the spBLEU parameter, further cross-validating the feasibility of LLMaaJ. However, since LLMs are black-box models with limited interpretability, the scores produced by LLMaaJ can only serve as a reference and do not guarantee accuracy or validity.

Moreover, it is worth noting that DN underperforms DG by approximately 5–15 percentage points overall, and, interestingly, the **"sudden stop"** phenomenon observed in Nllb-200-3.3B (Section § E.4) is faithfully inherited by the student model, which directly explains the comparatively lower post-fine-tuning performance; accordingly, selecting a teacher of the same decoder-only family during fine-tuning helps avoid this issue. **To address RQ2**, fine-tuning with data distillation yields highly significant gains: for both evaluated models, improvements are reflected in spBLEU scores that surpass those of certain expert translation systems. Furthermore, the enhancement in the EN→LB direction exceeds that of the reverse direction, further strengthening the model's Luxembourgish generation ability. Therefore, data distillation can substantially improve translation capacity for low-resource languages, enabling even smaller models to achieve promising results.

Table 4: Impact of LoRA Rank on spBLEU During Fine-Tuning, Evaluated Across Three Rank Values

EN-LB	Rank (LoRA)	Val 300 spBLEU	FLORES 200 spBLEU
Llama-3.2-3B-Instruct	**Base Model**	6.46	4.80
	32	12.95	9.46
	64	13.05	9.23
	128	13.32	9.27
Gemma-2-2b-it	**Base Model**	5.82	4.61
	32	13.07	8.88
	64	13.17	9.12
	128	13.31	9.21

4.3.2 Unlocking LRLs within SLMs?

Can we do LoRA? We also carried out experiments using the same data to assess how the LoRA rank parameter influences training performance in translation tasks involving Luxembourgish and English. Specifically, we evaluated the ranks 32, 64 and 128 in our models. The results, presented in Table 4 and 6, indicate that variations in the LoRA rank parameter have a minimal influence on the overall translation performance, with differences typically within 1 to 2 spBLEU points. More importantly, models fine-tuned using LoRA consistently underperformed compared to their fully fine-tuned counterparts, achieving notably lower performance in Table 2. Moreover, after LoRA-based SFT, we

MT Direction	Models	Methods	Val 300				FLORES-200			
			spBLEU	ChrF++	Jaccard	LLMaaJ	spBLEU	ChrF++	Jaccard	LLMaaJ
EN-LB	Nllb-200-3.3B	**BM**	19.97	37.03	0.27	0.75	31.14	49.62	0.35	0.85
	Llama-3.3-70B-Instruct		24.35	46.58	0.27	0.87	22.55	43.08	0.26	0.83
	Llama-3.2-3B-Instruct	**BM**	6.46	26.78	0.12	0.36	4.80	22.10	0.09	0.36
		DN	37.98	55.41	0.37	0.82	14.61	38.04	0.19	0.51
		DL	40.71	57.37	0.40	0.79	20.93	41.51	0.22	0.52
		DG	42.01	57.89	0.41	0.88	22.80	42.26	0.25	0.70
		DGDC	42.16	57.87	0.42	0.89	23.40	42.90	0.26	**0.83**
	Gemma-2-2b-it	**BM**	5.82	22.71	0.10	0.50	4.61	20.78	0.07	0.51
		DN	41.77	57.71	0.42	0.89	20.41	41.21	0.25	0.78
		DL	43.78	59.02	0.44	0.87	23.03	**42.95**	**0.28**	0.79
		DG	**44.58**	**59.73**	**0.45**	0.87	23.47	42.72	**0.28**	0.76
		DGDC	44.12	59.10	**0.45**	**0.90**	**23.50**	42.49	**0.28**	0.82
LB-EN	Nllb-200-3.3B	**BM**	40.51	56.81	0.48	0.81	**48.45**	**65.03**	**0.56**	0.85
	Llama-3.3-70B-Instruct		54.14	74.24	0.57	0.89	33.96	58.02	0.41	**0.86**
	Llama-3.2-3B-Instruct	**BM**	26.31	45.98	0.33	0.58	17.62	36.79	0.26	0.46
		DN	42.78	59.33	0.48	0.82	29.37	53.88	0.38	0.79
		DL	54.64	70.98	0.57	0.82	31.72	56.50	0.41	0.79
		DG	59.88	74.97	0.63	**0.90**	32.78	57.69	0.42	0.81
		DGDC	57.88	73.46	0.60	0.89	32.56	57.60	0.41	0.85
	Gemma-2-2b-it	**BM**	27.11	47.44	0.34	0.60	14.99	37.77	0.26	0.45
		DN	41.58	57.63	0.49	0.83	42.46	60.55	0.51	0.83
		DL	58.95	72.15	0.62	0.83	41.47	60.33	0.50	0.82
		DG	**65.44**	**76.96**	**0.68**	0.86	42.67	61.30	0.51	**0.86**
		DGDC	62.75	75.13	0.65	0.89	42.73	61.25	0.51	0.85

Table 2: This table presents the performance results obtained from training on datasets generated using different distillation models and methods. We report experimental results on two datasets, VAL 300 and FLORES 200. Additionally, we evaluated the performance of Nllb-200-3.3B and Llama-3.3-70B-Instruct on the same datasets, which strongly validate the effectiveness of our training approach. BM refers to the Base Model without any SFT. LLMaaJ refers to LLM-as-a-Judge, which gives a score from 0.0 to 1.0 with a granularity of 0.1.

MT Direction	Model	BOOLQ	CB	COPA	MULTIRC	RECORD	RTE	WIC	WSC	AVG
BM(Base Model)	Llama-3.2-3B-Instruct	0.62	0.55	0.71	0.52	0.41	0.64	0.51	0.28	0.53
	Gemma-2-2b-it	0.73	0.55	0.86	0.81	0.56	0.82	0.49	0.56	0.67
En-LB	Llama-3.2-3B-Instruct-FT	0.64	0.39	0.60	0.52	0.39	0.60	0.48	0.11	0.47
	Gemma-2-2b-it-FT	0.71	0.52	0.89	0.75	0.41	0.72	0.51	0.49	0.62
LB-EN	Llama-3.2-3B-Instruct-FT	0.64	0.30	0.69	0.51	0.46	0.62	0.52	0.24	0.50
	Gemma-2-2b-it-FT	0.69	0.25	0.90	0.76	0.45	0.73	0.51	0.43	0.59

Table 3: Variations in overall performance on the SuperGLUE benchmark before and after distillation training, evaluating whether fine-tuning on LRLs induces catastrophic forgetting. The model names appended with the suffix "-FT" denote the models after applying the proposed distillation fine-tuning method.

also observed an increased tendency toward hallucination. Due to the consistently lower performance and negligible differences observed among the varying LoRA ranks, we do not to recommend to use LoRA fine-tuning in LRLs translation tasks. These findings suggest that, while LoRA provides computational efficiency, its limited parameter updates are insufficient to capture the nuanced linguistic features required for effective translation of LRLs and may even be harmful.

Does data size really matter? Figure 6 illustrates the strong influence of the size of the data set on the quality of the translation in both directions (English⇔Luxembourgish), more detailed data in the Appendix Table 7. Even using as little as 1% of the available data yields modest improvements over the base model, yet the most substantial gains emerge only at higher data ratios. For example, increasing the data from 25% to 100% nearly doubles spBLEU in the EN→LB direction for both Llama-3.2-3B-Instruct and Gemma-2-2b-it. Notably, Gemma-2-2b-it seems to learn faster in the lower data regimes, but shows some performance attenuation beyond the 50% threshold.

Catastrophic forgetting? Because SLMs are expected to handle diverse tasks, an important question is whether LRL-oriented fine-tuning harms their general abilities. To assess this, we com-

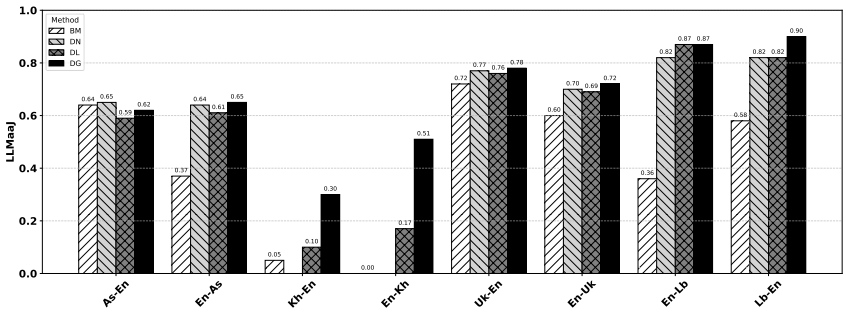


Figure 5: This figure compares the performance of four LRL pairs under the base model (Llama-3.2-3B-it) and knowledge distillation from different teacher models, evaluated using the LLMaaJ metric. "As" denotes Assamese, "Kh" denotes Khasi, and "Uk" denotes Ukrainian. Notably, the Kh—En and En-–Kh directions lack results for the DN setting (i.e., using NLLB-200-3.3B as the teacher model), as NLLB does not provide support for Khasi.

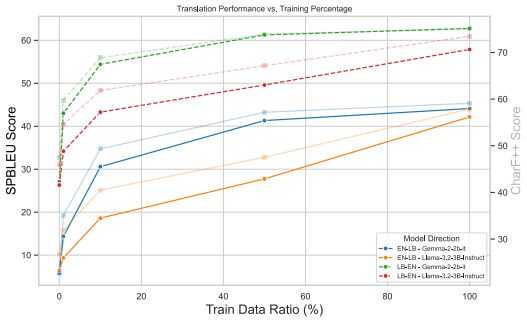


Figure 6: Performance vs. training data ratio; dashed lines show ChrF++ trends, solid lines show spBLEU; x-axis is data proportion.

pared SuperGLUE performance (Sarlin et al., 2020) before and after training. As shown in Table 3, translation-focused fine-tuning leads to only minor changes, indicating that the model retains its broader capabilities. These results suggest that distillation improves LRL translation without inducing catastrophic forgetting.

How about other LRLs? To assess generality beyond Luxembourgish, we applied the same monolingual distillation pipeline to 10,000 sentences each from Khasi, Assamese, and Ukrainian (with 1,000 validation pairs) using three teacher models: NLLB, Llama 3.3–70B, and GPT-4o-mini. We fine-tuned Llama-3.2-3B-Instruct under the same settings and evaluated it on the provided ground-truth annotations. As shown in Figure 5 and Table 10, distillation yields substantial gains in directions with low initial performance (En–As, En–Kh, En–Lb, Lb–En), while improvements are naturally smaller when the base model is already strong (e.g., As–En). Overall, these results confirm that distilled monolingual data can effectively transfer knowledge of resource-scarce languages to SLMs with minimal impact on general capabilities.

5 Conclusion

This work provides a systematic examination of small language models for low-resource language translation and shows that performance disparities across languages remain substantial. By combining monolingual corpora with knowledge distillation from high-capacity teacher models, we demonstrate that SLMs can achieve consistent and often significant improvements, in some cases surpassing much larger models. Compared to full fine-tuning, parameter-efficient approaches such as LoRA offer only limited gains, while our experiments indicate that targeted LRL training does not induce catastrophic forgetting in other capabilities. Although SLMs are not yet universally reliable for all LRL scenarios, this paper provides one of the first systematic validations of monolingual distillation for improving translation quality in LRLs.

Practical Takeaways.

1. Baseline SLM performance on LRLs is highly uneven, with large variation across language families and typologies.
2. Monolingual data combined with knowledge distillation enables small models (e.g., 3B parameters) to match or exceed models up to 70B parameters on several LRL directions.
3. LoRA-based fine-tuning is not well-suited for LRL translation; high-quality data and decoder-only teacher models provide the strongest improvements.
4. Fine-tuning on LRLs does not cause catastrophic forgetting, supporting its use in multilingual or agent-style applications involving low-resource languages.

Acknowledgments

The author Cedric Lothritz is supported by the LLMs4EU project, funded by the European Union through the Digital Europe Programme (DIGITAL) under the grant agreement 10119847.

Limitations

Distillation for synthetic data training is not new, but comprehensive training on SLMs for low-resource languages remains underexplored. From our research, with appropriate training, small models can also learn to handle very challenging low-resource languages. However, this approach relies on powerful pretrained models for knowledge distillation, which may not always be available in extremely low-resource settings. Standard metrics such as BLEU cannot fully capture linguistic or cultural accuracy, so other evaluation metrics such as CometKiwi (Rei et al., 2022) and human evaluation are still necessary to better validate the results. Another concern is the lack of interpretability in neural translation, as it is unclear whether models truly understand LRLs, highlighting the need for more work on explainability.

Ethics Statement

All models and resources developed in this work are strictly intended for research and educational purposes according to OpenAI usage guidelines; no model weights or derivatives are used — or will be used — for any commercial application. We exclusively utilize publicly available corpora or datasets for which explicit authorization has been obtained from the original data providers. All license terms have been reviewed to ensure full compliance with copyright, attribution, and sharing requirements.

No personally identifiable information (PII) is collected during this research. All data processing, storage, and retention policies are fully aligned with the EU General Data Protection Regulation (GDPR). The dataset of LOD.lu is under the CC0 license. As most of RTL datasets are based on articles from RTL, we cannot publish them, but we make them available to researchers on request.

All code, models, and processed data artifacts will be released under an open-source, research-oriented license (e.g., CC BY-NC), accompanied by comprehensive documentation and bias-analysis methodology to promote transparency and reproducibility. We commit to ongoing ethical oversight through periodic reevaluation of datasets and model outputs, prompt updates in response to emerging concerns, and consultation with interdisciplinary advisory boards to ensure adherence to the highest ethical standards.

Reproducibility Statement

All experiments were implemented and evaluated on four NVIDIA H100 GPUs with a per-device batch size of 8 using the TRL library for training. The complete codebase, configuration files, and training/evaluation scripts are available in the anonymous repository: `https://anonymous.4open.science/r/mt_luxembourgish-408D`. Pretrained checkpoints and selected fine-tuned models are released to facilitate independent verification and reuse. The repository includes environment specifications, dependency pins, and command-line recipes that enable end-to-end reproduction of the reported results.

References

Sanchit Ahuja, Divyanshu Aggarwal, Varun Gumma, Ishaan Watts, Ashutosh Sathe, Millicent Ochieng, Rishav Hada, Prachi Jain, Mohamed Ahmed, Kalika Bali, et al. 2024. Megaverse: Benchmarking large language models across languages, modalities, models and tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2598–2637.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the association for computational linguistics*, 7:597–610.

Seth Aycock, David Stap, Di Wu, Christof Monz, and Khalil Sima'an. 2025. Can llms really learn to translate a low-resource language from one grammar book? *Preprint*, arXiv:2409.19151.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kenza Benkirane, Laura Gongas, Shahar Pelles, Naomi Fuchs, Joshua Darmon, Pontus Stenetorp, David Adelani, and Eduardo Sánchez. 2024. Machine translation hallucination detection for low and high resource languages using large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9647–9665.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind

Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Marta Costa-jussa, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loïc Barrault, Gabriel Gonzalez, Prangthip Hansanti, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Luciano da F. Costa. 2021. Further generalizations of the jaccard index. *CoRR*, abs/2110.09619.

Micha Elsner et al. 2024. Shortcomings of llms for low-resource translation: Retrieval and understanding are both the problem. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1332–1354.

Luyu Gao, Xinyi Wang, and Graham Neubig. 2020. Improving target-side lexical transfer in multilingual neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3560–3566.

Yingbo Gao, Christian Herold, Zijian Yang, and Hermann Ney. 2022. Is encoder-decoder redundant for neural machine translation? In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 562–574.

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*.

Google. 2024. Gemma-2-2b-it model card.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2021a. The flores-101 evaluation benchmark for low-resource and multilingual machine translation.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2021b. The FLORES-101 evaluation benchmark for low-resource and multilingual machine translation. *CoRR*, abs/2106.03193.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. Two new evaluation datasets for low-resource machine translation: Nepali-english and sinhala-english.

Md. Arid Hasan, Prerona Tarannum, Krishno Dey, Imran Razzak, and Usman Naseem. 2024. Do large language models speak all languages equally? a comparative study in low-resource settings. *Preprint*, arXiv:2408.02237.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *Preprint*, arXiv:2302.09210.

Kurt Hornik. 1991. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257.

David M Howcroft and Dimitra Gkatzia. 2022. Most nlg is low-resource: here's what we can do about it. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 336–350.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293. Association for Computational Linguistics.

Wen Lai, Mohsen Mesgar, and Alexander Fraser. 2024. Llms beyond english: Scaling the multilingual capability of llms with cross-lingual feedback. *arXiv preprint arXiv:2406.01771*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations (ICLR)*.

Séamus Lankford, Haithem Alfi, and Andy Way. 2021. Transformers for low-resource languages: Is féidir linn! In *Proceedings of Machine Translation Summit XVIII: Research Track*, pages 48–60.

Lujun Li, Lama Sleem, Niccolo' Gentile, Geoffrey Nichil, and Radu State. 2025a. Exploring the impact of temperature on large language models: Hot or cold? *Procedia Computer Science*, 264:242–251. International Neural Network Society Workshop on Deep Learning Innovations and Applications 2025.

Zihao Li, Yucheng Shi, Zirui Liu, Fan Yang, Ali Payani, Ninghao Liu, and Mengnan Du. 2025b. Language ranker: A metric for quantifying llm performance across high and low-resource languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 28186–28194.

Lifewire. 2024. Llama 3 vs. llama 2: Why the newest model leaves its predecessor in the dust. Accessed: 2025-02-03.

Meta Llama. 2024. Llama-3.2-3b-instruct model card.

Chi-kiu Lo, Rebecca Knowles, and Cyril Goutte. 2023. Beyond correlation: Making sense of the score differences of new MT evaluation metrics. In *Proceedings of Machine Translation Summit XIX, Vol. 1: Research Track*, pages 186–199, Macau SAR, China. Asia-Pacific Association for Machine Translation.

Cedric Lothritz and Jordi Cabot. 2025. Testing low-resource language support in llms using language proficiency exams: the case of luxembourgish. *arXiv preprint arXiv:2504.01667*.

Cedric Lothritz, Bertrand Lebichot, Kevin Allix, Lisa Veiber, Tegawendé François D Assise Bissyande, Jacques Klein, Andrey Boytsov, Anne Goujon, and Clément Lefebvre. 2022. Luxembert: Simple and practical data augmentation in language model pre-training for luxembourgish. In *13th Language Resources and Evaluation Conference (LREC 2022)*.

Kelly Marchisio, Wei-Yin Ko, Alexandre Bérard, Théo Dehaze, and Sebastian Ruder. 2024. Understanding and mitigating language confusion in llms. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6653–6677.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2024. A comprehensive overview of large language models. *Preprint*, arXiv:2307.06435.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Tajudeen Kolawole, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, et al. 2020. Participatory research for low-resourced machine translation: A case study in african languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160. Association for Computational Linguistics.

Hellina Hailu Nigatu, Atnafu Lambebo Tonja, Benjamin Rosman, Thamar Solorio, and Monojit Choudhury. 2024. The zeno's paradox of 'low-resource' languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17753–17774. Association for Computational Linguistics.

Joel Niklaus, Jakob Merane, Luka Nenadic, Sina Ahmadi, Yingqiang Gao, Cyrill A. H. Chevalley, Claude Humbel, Christophe Gösken, Lorenzo Tanzi, Thomas Lüthi, Stefan Palombo, Spencer Poff, Boling Yang, Nan Wu, Matthew Guillod, Robin Mamié, Daniel Brunner, Julio Pereyra, and Niko Grupen. 2025. Swiltra-bench: The swiss legal translation benchmark. *Preprint*, arXiv:2503.01372.

Chinasa T Okolo and Marie Tano. 2024. Closing the gap: A call for more inclusive language technologies.

Andrea Piergentili, Beatrice Savoldi, Matteo Negri, and Luisa Bentivogli. 2025. An llm-as-a-judge approach for scalable gender-neutral translation evaluation. *arXiv preprint arXiv:2504.11934*.

Alistair Plum, Tharindu Ranasinghe, and Christoph Purschke. 2024. Text generation models for luxembourgish with limited data: A balanced multilingual strategy. *arXiv preprint arXiv:2412.09415*.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

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, Luisa Coheur, and André F. T. Martins. 2022. Cometkiwi: Ist-unbabel 2022 submission for the quality estimation shared task. *Preprint*, arXiv:2209.06243.

Nathaniel Robinson, Perez Ogayo, David R Mortensen, and Graham Neubig. 2023. Chatgpt mt: Competitive for high-(but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *ACL*, pages 86–96. Association for Computational Linguistics.

Lingfeng Shen, Weiting Tan, Sihao Chen, Yunmo Chen, Jingyu Zhang, Haoran Xu, Boyuan Zheng, Philipp Koehn, and Daniel Khashabi. 2024. The language barrier: Dissecting safety challenges of llms in multilingual contexts. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 2668–2680.

Ana Silva, Nikit Srivastava, Tatiana Moteu Ngoli, Michael Röder, Diego Moussallem, and Axel-Cyrille Ngonga Ngomo. 2024. Benchmarking low-resource machine translation systems. In *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, pages 175–185, Bangkok, Thailand. Association for Computational Linguistics.

Yewei Song, Saad Ezzini, Jacques Klein, Tegawende Bissyande, Clément Lefebvre, and Anne Goujon. 2023. Letz translate: Low-resource machine translation for luxembourgish. In *2023 5th International*

Conference on Natural Language Processing (ICNLP), pages 165–170. IEEE.

Jörg Tiedemann and Santhosh Thottingal. 2020. Opus-mt–building open translation services for the world. In *Annual Conference of the European Association for Machine Translation*, pages 479–480. European Association for Machine Translation.

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.

Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2023. Transformer: A general framework from machine translation to others. *Machine Intelligence Research*, 20(4):514–538.

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. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Tianyang Zhong, Zhenyuan Yang, Zhengliang Liu, Ruidong Zhang, Yiheng Liu, Haiyang Sun, Yi Pan, Yiwei Li, Yifan Zhou, Hanqi Jiang, Junhao Chen, and Tianming Liu. 2024. Opportunities and challenges of large language models for low-resource languages in humanities research. *Preprint*, arXiv:2412.04497.

Appendix

A Data Processing

Dataset selection directly impacts the reliability and generalizability of experimental results. Our criteria include having enough test samples, providing reference responses, and minimizing potential biases from overlap with pre-training data.

FLORES-200 (Costa-jussa et al., 2022) is a benchmark dataset specifically designed for low-resource and multilingual machine translation, serving as an extended version of FLORES-101 (Goyal et al., 2021b). It covers 200 languages and consists of sentences extracted from 842 web articles, with an average length of approximately 21 words. These sentences are divided into three datasets: dev, devtest, and a hidden test set. Since we require additional evaluation metrics, we use devtest as our set of tests in this study. In our paper, we primarily evaluate the translation performance of all 200 languages into English. However, in the subsequent model training, we focus solely on the Luxembourgish-English language pair for training and testing.

The VAL 300 validation set was constructed using 300 pieces of official news content from July 2024 as the source data. The corresponding ground truth in Luxembourg was generated using ChatGPT, followed by dictionary-based verification to ensure validity. Furthermore, we extracted 30 samples from the dataset and engaged Luxembourgish-English bilingual speakers to perform a quality assessment.

B Experiments settings

In our experiments, we used primarily two distinct models for supervised fine-tuning (SFT) to evaluate performance and optimization strategies. To ensure an effective training process, several hyperparameters and model configurations were meticulously selected. Specifically, the warm-up ratio was set to 0.5, facilitating a gradual increase in the learning rate during the initial training phase for improved convergence stability. The maximum gradient norm was restricted to 0.3, serving as a mechanism to prevent excessively large parameter updates and promote stable optimization dynamics. Furthermore, the input sequence length was capped at 512 tokens, ensuring that all processed data adhered to this fixed-length constraint. A weight decay of 0.01 was applied to regularize the model parameters and mitigate the risk of overfitting. It is worth noting that all of our models were trained for only one epoch. This decision was based on our observation that evaluation metrics reached their optimal performance after a single epoch, while additional epochs amplified the influence of noisy data without bringing performance gains. Moreover, we observed an increased likelihood of hallucinations and the re-emergence of uncontrolled generation, suggesting that the dialogue capability of the model after instruction fine-tuning may deteriorate due to overtraining across multiple epochs. **Therefore, we recommend employing only one epoch for translation training of LRLs on SLMs, as this constitutes a valuable training insight that warrants careful consideration.**

To ensure reproducibility across experiments, a fixed random seed of 3407 was utilized. For model architecture selection, two distinct approaches were considered: standard fine-tuning and LoRA. In cases where LoRA was employed, specific layers were targeted for adaptation, including "q_proj," "k_proj," "v_proj," "o_proj," "gate_proj," "up_proj," and "down_proj." The LoRA alpha pa-

rameter was configured to a value of 8, while the dropout rate for LoRA layers was set to 0, indicating that no dropout-based regularization was applied to these low-rank adaptation layers.

For tokenization and input preparation, a standardized procedure was adopted to ensure consistency in sequence length across the examples. The tokenizer processed each input field by truncating sequences exceeding the maximum length of 512 tokens and padding shorter sequences to this fixed length. This was achieved using the ‘padding="max_length"‘ option, thereby guaranteeing uniformity in input representation prior to model training. During the inference stage, we set the temperature parameter to 0.1 (close to 0), which has been shown to help achieve optimal machine translation performance (Li et al., 2025a). In addition, we set `max_new_tokens` to 512, enable `do_sample = True`, and set `top_p = 0.9`.

Model	Reference	SFT Methods
Llama-3.2-3B-Instruct	(Llama, 2024)	FS/ LoRA SFT
Gemma-2-2b-it	(Google, 2024)	FS/ LoRA SFT

Table 5: Various models and their SFT methods. "FS/ Lora SFT" refers to full-size and "Lora SFT" denotes Low-Rank Adaptation SFT only.

C Dictionary Processing

In our approach to enhancing translation accuracy, particularly for Luxembourgish, we developed a retrieval pipeline using Haystack 2.0. The pipeline utilizes a BM25 retriever to identify relevant dictionary entries that align closely with the input text. The retrieved dictionary entries are then incorporated directly into the prompt provided to GPT-4O, offering multiple lexical choices that help clarify ambiguous terms.

This method operates as follows: first, the BM25 retriever ranks and returns the most relevant dictionary entries based on the Luxembourgish input. These entries serve as additional context within the prompt, guiding GPT-4o toward more accurate translations. Subsequently, the original Luxembourgish sentence and the relevant dictionary context are submitted to GPT-4o for translation. By explicitly integrating these dictionary options into the prompt, GPT-4o is better equipped to resolve lexical ambiguities and correct potential translation errors, enhancing translation accuracy and coherence.

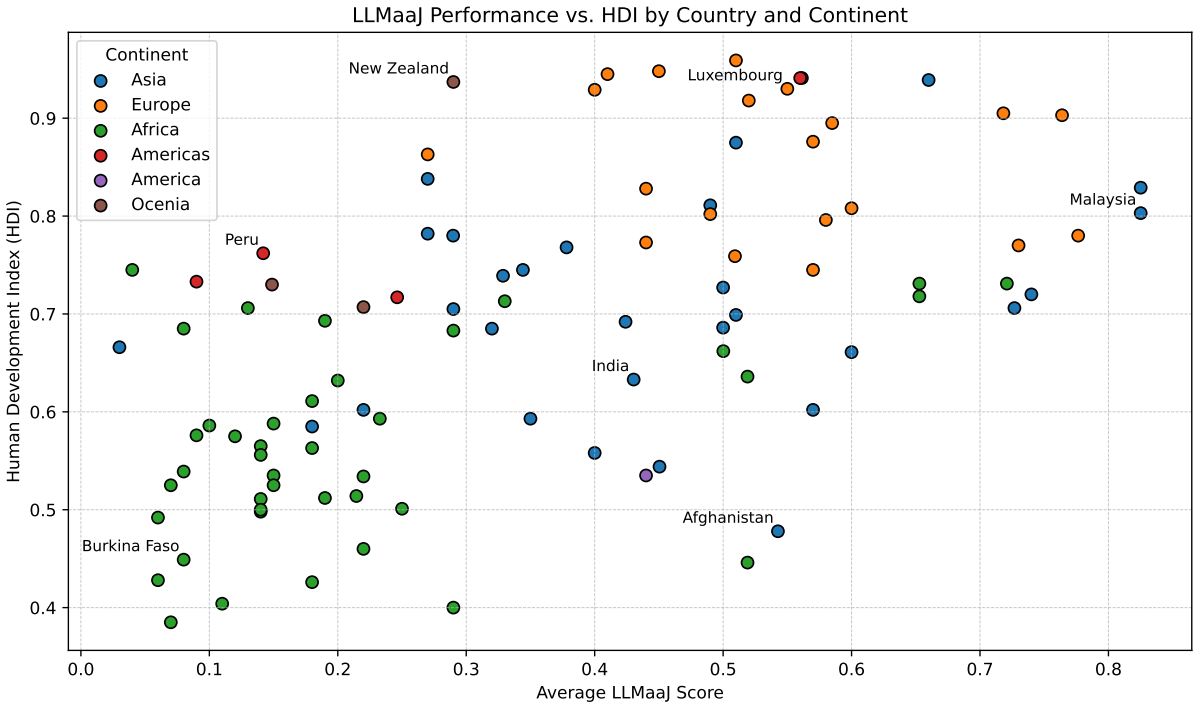


Figure 7: Scatter Plot of LLMaaJ Score and HDI Relation for LRLs

Table 6: Impact of LoRA Rank on Performance During Fine-Tuning, Evaluated Across Three Rank Values

EN-LB	Rank (LoRA)	Val 300			FLORES 200		
		spBLEU	ChrF++	Jaccard	spBLEU	ChrF++	Jaccard
Llama-3.2-3B-Instruct	**Base Model**	6.46	26.78	0.12	4.80	22.10	0.09
	r = 32	12.95	33.09	0.19	9.46	29.64	0.14
	r = 64	13.05	33.59	0.19	9.23	28.93	0.14
	r = 128	13.32	34.09	0.20	9.27	29.16	0.14
Gemma-2-2b-it	**Base Model**	5.82	22.71	0.10	4.61	20.78	0.07
	r = 32	13.07	33.36	0.21	8.88	27.93	**0.16**
	r = 64	13.17	33.35	0.21	9.12	28.06	0.16
	r = 128	13.31	**33.69**	0.21	9.21	28.20	0.16

D Dataset Size Influence

Table 7 in the appendix presents a comprehensive analysis of how dataset size influences translation performance in our low-resource Luxembourgish-English setting. We experimented with dataset sizes ranging from as small as 1% to the full dataset (100%). The results demonstrate a clear, positive correlation between the amount of data utilized during fine-tuning and the subsequent translation quality, as measured by BLEU scores.

In both translation directions (EN→LB and LB→EN), we observed that even very small datasets (e.g., 1%–5%) provide measurable improvements over baseline models, indicating that the models begin acquiring beneficial linguistic patterns early in the fine-tuning process. However, substantial performance gains occur predominantly when increasing the dataset size beyond 25%. For instance, moving from 25% to 100% dataset size nearly doubles the spBLEU scores for the EN→LB direction, clearly highlighting the significance of sufficient data availability for generating fluent, accurate translations in low-resource languages.

Interestingly, the Gemma-2-2b-it model displayed a relatively faster learning trajectory compared to the Llama-3.2-3B-Instruct model in smaller data regimes (below 50%). Nevertheless, Gemma-2-2b-it exhibited a notable attenuation in performance improvements beyond the 50% data threshold, suggesting a diminishing return effect when datasets grow larger. Conversely, the Llama-3.2-3B-Instruct model showed steadier improvements without significant attenuation up to the full dataset size, potentially indicating better scalability of linguistic capabilities with increased training data.

Table 7: Impact of Dataset Size on the Performance of Fine-Tuning

English to Luxembourgish	**Dataset Ratio**	**Val 300**			**FLORES 200**		
		spBLEU	**ChrF++**	**Jaccard**	**spBLEU**	**ChrF++**	**Jaccard**
Llama-3.2-3B-Instruct	**0%**	6.46	26.78	0.12	4.80	22.10	0.09
	1%	9.36	31.88	0.16	6.53	26.31	0.10
	10%	18.61	40.51	0.23	9.79	30.65	0.14
	50%	**27.75**	**47.52**	**0.30**	**13.39**	**34.67**	**0.17**
	100%	**42.16**	57.87	**0.42**	**23.40**	**42.90**	**0.26**
Gemma-2-2b-it	**0%**	5.82	22.71	0.10	4.61	20.78	**0.07**
	1%	14.36	35.06	0.21	9.01	27.99	**0.15**
	10%	30.58	**49.32**	0.34	15.99	36.12	0.22
	50%	41.32	**57.18**	**0.42**	22.30	**41.69**	**0.27**
	100%	44.12	59.10	**0.45**	**23.50**	42.49	**0.28**
Luxembourgish to English		**Val 300**			**FLORES 200**		
		spBLEU	**ChrF++**	**Jaccard**	**spBLEU**	**ChrF++**	**Jaccard**
Llama-3.2-3B-Instruct	**0%**	26.31	45.98	0.33	17.62	36.79	0.26
	1%	34.18	54.63	0.4	22.68	45.98	0.32
	10%	43.28	61.86	0.48	26.11	50.51	0.36
	50%	**49.60**	**67.15**	**0.53**	**29.18**	**54.35**	**0.39**
	100%	57.88	73.46	0.60	32.56	57.60	0.41
Gemma-2-2b-it	**0%**	27.11	47.44	0.34	14.99	37.77	0.26
	1%	43.00	59.80	0.47	29.25	49.15	**0.38**
	10%	54.41	68.86	0.58	36.14	55.67	0.45
	50%	61.26	**73.91**	**0.64**	41.06	**59.94**	**0.49**
	100%	62.75	75.13	0.65	**42.73**	61.25	**0.51**

E Case Study

This section mainly presents several interesting text cases encountered during the pre-trained LLM generation process, which were identified through our manual quality checks.

E.1 Uncontrollable output

A notable translation issue arises from unnecessary explanatory notes appended by the model, which negatively affects automated evaluation metrics. Consider the following example, where the original English input was:

> "He added that “they should not, however, be asked to take on obligations that go beyond their development stage, responsibility and capabilities.”"

The model produced:

> "Dëi gouf och gesäftlech gesëtt datt “si si och net an Obergrenzen vum ieren Entwéckelungsschatz, Responsabilität a Fähigkeeten uerecht bestallt.” (Note: Please keep in mind that Luxembourgish is a complex and nuanced language, and translations may vary depending on regional dialects and contexts. This translation is provided in standard Luxembourgish)"

The corresponding ground truth translation in Luxembourgish is:

> "Hien huet bäigefüügt, datt “se awer net opgefuerdert sollte ginn, Verflichtungen ze iwwerhuelen, déi iwwer hiren Entwécklungsniveau, hir Verantwortung a Fäegkeeten erausginn.”"

The inclusion of this extraneous note, likely originating from instruction-tuned general translation settings or cautious model outputs designed for conversational contexts, significantly impacts BLEU and related metrics. Such notes introduce irrelevant lexical content and length discrepancies, lowering similarity scores and thereby inaccurately reflecting the model's genuine translation ability.

To mitigate this issue, explicit prompt engineering or fine-tuning with specific stopping criteria and clearer output formatting instructions is essential. This strategy helps ensure that model-generated translations remain concise, relevant, and aligned closely with evaluation metric expectations.

E.2 Repetition Problem

During pre-training, we frequently encountered the issue of model hallucination, characterized by repetitive generation. This problem was notably less severe in the Gemma model compared to the Llama model. The repetitive generation primarily occurred in the English-to-Luxembourgish translation direction, largely due to the model's insufficient understanding of Luxembourgish. Consequently, the model frequently confused Luxembourgish with German, exacerbating this confusion further.

Below is an example of input:

> Translate the following English input text into Luxembourgish. Do not include any additional information or unrelated content. Input: The United States Strategic Command of the U.S. Department of Defense office is tracking the debris.

Below is an illustrative output that presents repetitive generation:

> Here is the translation: D'United States Strategic Command vum ustrategeschen vum ustrategeschen Ufuerderungsbüro vum ustrategeschen Ufuerderungsbüro vum ustrategeschen Ufuerderungsbüro vum ustrategeschen Ufuerderungsbüro vum ustrategeschen Ufuerderungsbüro vum ustrategeschen ...

We also observed that when distillation datasets contained a small amount of hallucinated data—such as infinitely repeating Luxembourgish content (particularly prominent in Llama 3.3 70B)—the subsequently fine-tuned models strongly inherited these hallucination patterns, resulting in substantial performance degradation. Therefore, maintaining high-quality data for fine-tuning is crucial. In comparison, data

distilled from ChatGPT rarely exhibited infinite repetition issues, enabling the model to learn appropriate stopping criteria during generation. Additionally, we found that incorporating certain generation prompts during training, such as "Here is the translation: ", could partially mitigate the inherited infinite generation problems originating from distilled models of Llama.

E.3 Hallucination to German

Given the English input sentence:

"At first glance, it looks slightly chaotic at the arrival, the children make a game of clapping hands with everyone who arrives, but that also contributes to the special charm of the event. The Widdebierg is not as high as one might assume from the logo, but it is still 386 meters, which is not nothing,"

Model produced the following output:

"Bei der Ankunft blickt man zunächst etwas chaotisch aus, die Kinder machen ein Spiel mit der Klatsch der Hände mit jedem, der kommt, aber das tragen auch zur besonderen Charme vun der Ävent vun der Éieren. Dëi Widdebiergëstcher Ääschten, well dass méi méi vun der Logo, mä si méi och 386 Méter, wéi déi éis gët nach Léck, wat déiweis méi och net."

In contrast the Luxembourgish ground truth is

"Op den éischte Bléck gesäit et op der Arrivée liicht chaotesch aus, d'Kanner maache sech e Spaass draus, jidderengem, deen ukënnt, an d'Hand ze klatschen, mä och dat dréit zum spezielle Charme vun der Manifestatioun bäi. De Widdebierg ass wuel net esou héich wéi een dat um Logo kéint unhuelen, mä ëmmerhi sinn et 386 Meter, dat ass net grad näischt."

This incorrect translation output primarily results from excessive usage of German vocabulary rather than proper Luxembourgish expressions. This phenomenon likely arises due to several factors:

- **Data Sparsity and Language Proximity:** Luxembourgish and German share considerable lexical and syntactic similarities. In conditions of limited Luxembourgish-specific training data, the model might unintentionally rely heavily on its knowledge of German, leading to significant linguistic interference.
- **Pretraining Corpus Bias:** The predominance of German texts over Luxembourgish in multilingual pretraining datasets likely reinforces German lexical and structural patterns, especially under resource-constrained fine-tuning conditions.
- **Limited Distinctive Training Examples:** Insufficient distinct Luxembourgish examples during fine-tuning might not effectively guide the model away from Germanic lexical choices, resulting in mixed-language outputs or incorrect lexical selections.

Addressing this issue effectively requires either extensive additional training data or targeted linguistic resources explicitly designed to emphasize lexical and grammatical distinctions between closely related languages such as Luxembourgish and German.

E.4 Sudden Stop From NLLB models distillation

We observed an intriguing phenomenon when using NLLB models: regardless of size (3.3B or the 700M distilled variant), the model would sometimes abruptly stop translating longer passages without warning, as if refusing to continue, and this occurred randomly. During subsequent training, it proved difficult to detect which outputs were complete versus incomplete translations; moreover, data curated with NLLB exhibited a severe failure to stop generation at the correct endpoint. Introducing explicit tags and an

end-of-translation marker (e.g., “End of Translation”) resolved the non-stopping generation issue; however, the abrupt early-stop behavior from NLLB was fully inherited by downstream models, as shown below.

English Source Sentences

The government warns against fraudsters selling fake tickets for events such as concerts or sporting events. “Be extra cautious when purchasing tickets online” - that is the government’s warning as the Olympic Summer Games and the European Championship are about to start and the festival season is also approaching. Sellers are therefore required to provide all essential information, such as the price, category, and seating location in the hall or stadium.

Translation results for one model distilled from GPT4o-mini

D’Regierung warnt virun Bedruchsbetriber, déi falsch Tickete fir Evenementer wéi Concerten oder Sport-Evenementer verkafen. “Extra virsiichteg sinn, wann een Ticketen online kafe wëll” - dat ass d’Warnung vun der Regierung, well d’Olympesch Summerspiller an d’Europameeschterschaft untrieden an och d’Festival-Saison untrëtt. D’Vendeuren mussen deemno all wichteg Informatiounen, wéi de Präis, d’Kategorie an d’Sëtzplaz am Sall oder am Stadion, matginn.

Translation results for one model distilled from NLLB-3.3B

D’Regierung warnt virun Betrüger, déi gefälschte Ticketen fir Evenementer wéi Concerten oder Sportveranstaltungen verkafen. “Sidd extra virsiichteg beim Ticketkaaf online” - dat ass d’Warnung vun der Regierung, well d’Olympesch Summerspiller an d’Europameeschterschaft ufänken an d’Festivalsaison och no kënnt. **[.....MISSING.....]**

F Prompt Design for LLM

F.1 Prompt for LLM-as-a-Judge

For the prompt, we mainly adopt the previous legal translation prompt structure (Niklaus et al., 2025) but customize it simply for only the transation needs without any domain emphasis specification. In this paper, we primarily employ google/gemma-3-27b-it as the evaluation model to assess translation quality, given its strong instruction-following capabilities and competitive performance among open-weight LLMs. For efficient model inference, we adopt SGLang as the serving framework, which enables streamlined deployment and low-latency response for both evaluation and generation tasks.

Your task is to assess the accuracy, clarity, and fidelity of the model’s translation to the golden translation.

You will be provided the golden translation, and the model’s translation. Your task is to judge how correct the model’s translation is based on the golden translation, and then give a correctness score. The correctness score should be one of the below numbers: 0.0 (totally wrong), 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, or 1.0 (totally right). You should give the correctness score directly. The correctness score must strictly follow this format: "[[score]]", e.g., "The correctness score: [[0.5]]. Golden Translation: **{Golden Translation}**

Model Translation: **{Model’s Translation}**

F.2 Prompt for SFT

We primarily adopt the classical SFT approach, where the model is trained to predict the next token by minimizing the cross-entropy loss. Consequently, training data typically consist of input-output pairs,

such as question-answer or instruction-response formats. The input is usually referred to as the prompt and the output as the answer. During training, the prompt and answer are concatenated and fed into the model, with the objective of guiding the model to generate the answer portion. In this work, we employ the following training template.

> Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.
>
> ### Instruction:
> Translate the following English input text into Luxembourgish. Do not include any additional information or unrelated content.
>
> ### Input:
> **{The sentence to be translated}**
>
> ### Response:
> **{The translated sentence}**

G Language Ability On LLMs

G.1 Translation Performance and Human Development Disparities

In this analysis, LRLs are operationally defined as those that comprise less than 0.1% of web content (according to W3Techs statistics[4]). The average *LLMaaJ* scores were calculated exclusively for the selected LRLs that also exist in the FLORES-200 dataset. Country - LRLs pairs were identified based on a mapping that utilizes Wikipedia-derived estimates of language speaker distribution.

Figure 7 reveals a clear positive correlation between a country's human development level (HDI) and the translation quality of its low-resource languages as judged by LLMs. Each point in the scatter represents a FLORES-200 language linked to a country's HDI, and the overall trend slopes upward – higher-HDI countries tend to have languages with higher LLMaaJ translation scores. This suggests that socioeconomic factors underpin disparities in LLM translation coverage, echoing the "digital language divide" observed in AI research (Okolo and Tano, 2024). In other words, languages from more developed regions generally receive far better support in large multilingual models than those from less developed regions.

When grouping languages by development tiers, the performance gap is stark. Languages from Very High HDI countries (HDI ≥ 0.80) achieve an average LLMaaJ score of around 0.54, more than double the 0.22 average for languages from Low HDI countries (HDI < 0.55). Median scores likewise jump from only 0.15 in low-HDI settings to 0.53 in very-high-HDI settings. This means a typical low-resource language in a highly developed society enjoys significantly better machine translation quality than one in a low-development context. Crucially, it is not simply the number of speakers but the socioeconomic context and digital resources that dictate how well a language is served by AI. For instance, Hindi (with over 500 million speakers) has historically been treated as "low-resource" for NLP, whereas a smaller language like Dutch (with a fraction of the speakers, but backed by a high-HDI country) is well-supported. The greater availability of data and funding in high-HDI environments allows LLMs to achieve markedly better translations for those languages.

Geographic disparities are especially pronounced. Nearly all African languages in the study cluster toward the lower-left of Figure 7, indicating both low HDI and poor translation performance. In fact, none of the African languages evaluated approach the top tier of LLMaaJ scores – a finding consistent with reports that even state-of-the-art multilingual models still lag on African languages due to limited training

[4] https://w3techs.com/technologies/overview/content_language

data and quality. By contrast, European languages (from countries with generally high HDI) occupy the upper range of the plot; these languages achieve some of the highest scores (e.g. minority languages like Occitan in France reach LLMaaJ ≈ 0.76). Several Asian languages spoken in high-HDI regions likewise perform strongly – for example, Standard Malay (Malaysia/Brunei) attains average scores above 0.80 in our data. Meanwhile, many languages of low-HDI countries remain at the bottom: Dzongkha of Bhutan (medium HDI) has one of the lowest scores (LLMaaJ ≈ 0.03), and numerous Sub-Saharan African languages (e.g. Tigrinya of Eritrea) register below 0.10. These patterns suggest that languages benefiting from a robust digital infrastructure or from close linguistic ties to well-resourced tongues (as Occitan does to French) see far better outcomes, whereas languages in impoverished or isolated settings are left behind.

Overall, the strong HDI-performance correlation highlights a systemic inequality in LLM coverage. The correlation coefficient score between HDI and LLMaaJ average score is 0.566, indicating a medium-high correlation. Communities in low-development regions face a double disadvantage: they are underserved by technology on top of existing socio-economic challenges. Indeed, globally fewer than 1% of languages have sufficient data to be considered high-resource, leaving speakers of the other 99% "essentially cut off from global technological progress". This lack of access to quality translation and language tools can hinder information access, education, and opportunities, thereby exacerbating the digital divide and reinforcing global inequalities. Our findings underscore that current multilingual AI models, despite their broad reach, de facto offer far stronger support for languages of wealthy, high-HDI communities than for those of poorer regions. Addressing this gap will require concerted efforts to bring truly inclusive language coverage to the forefront, rather than merely adding more languages without improving quality for the most disadvantaged.

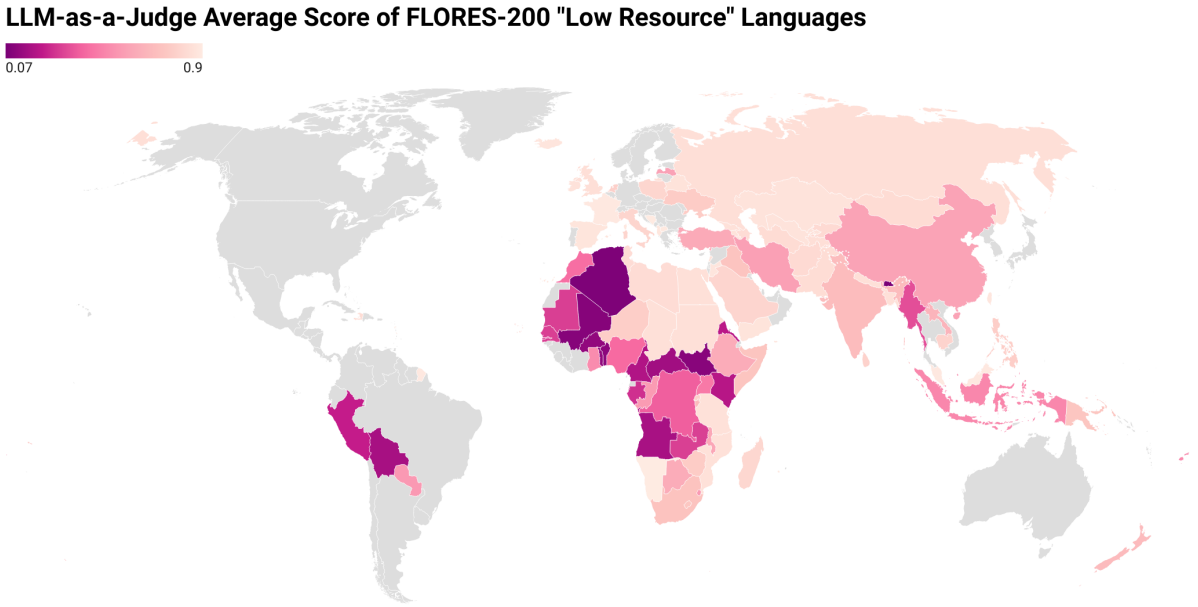


Figure 8: Linguistic geographical distribution results. Some countries might not be accurate because the official languages are European, especially in Africa. English major countries will only consider the minority language if in the dataset.

G.2 Result Tables

Table 8: The LLMaaJ results on the FLORES-200 dataset are derived from evaluations of 10 distinct large language models. Population estimates are based on heterogeneous sources, and the reported population are not guaranteed to be accurate. Therefore, they should be interpreted with appropriate caution.

Language Name	Language Branch	Population	GPT4o mini	Llama-3.1-8B	Llama-3.2-3B	Ministral-8B	Phi-3	Phi-3.5	Qwen2.5-1.5B	Qwen2.5-3B	gemma2-2b	gemma2-9b
Central Atlas Tamazight	Berber	3-4 million	0.017	0.008	0.006	0.008	0.007	0.014	0.006	0.01	0.011	0.014
Kabyle		5 million	0.078	0.054	0.027	0.025	0.02	0.038	0.02	0.042	0.028	0.08
Tamasheq (Latin script)		500,000	0.143	0.101	0.067	0.082	0.088	0.093	0.061	0.09	0.096	0.142
Tamasheq (Tifinagh script)		500,000	0.021	0.009	0.007	0.009	0.008	0.022	0.005	0.013	0.016	0.018
Hausa	Chadic	40 million	0.774	0.534	0.166	0.132	0.089	0.101	0.082	0.11	0.228	0.656
Somali	Cushitic	20 million	0.735	0.257	0.112	0.143	0.077	0.121	0.063	0.107	0.112	0.5
West Central Oromo		10 million	0.617	0.079	0.067	0.047	0.028	0.051	0.023	0.07	0.035	0.121
Amharic	Semitic	32 million	0.627	0.254	0.015	0.024	0.008	0.013	0.018	0.054	0.148	0.59
Hebrew		9 million	0.892	0.859	0.587	0.853	0.464	0.599	0.578	0.757	0.802	0.874
Maltese		520,000	0.892	0.793	0.551	0.428	0.237	0.261	0.202	0.311	0.627	0.855
Modern Standard Arabic		335 millions	0.881	0.858	0.792	0.847	0.573	0.799	0.771	0.832	0.814	0.863
Tigrinya		9 million	0.209	0.066	0.006	0.02	0.016	0.017	0.007	0.026	0.041	0.211
Egyptian Arabic		60 million	0.851	0.807	0.701	0.776	0.451	0.68	0.658	0.753	0.718	0.815
Mesopotamian Arabic		15 million	0.862	0.839	0.715	0.794	0.497	0.713	0.686	0.774	0.751	0.83
Moroccan Arabic		30 million	0.816	0.659	0.529	0.596	0.316	0.508	0.491	0.58	0.555	0.736
Najdi Arabic		10 million	0.861	0.868	0.772	0.826	0.542	0.775	0.751	0.817	0.788	0.842
North Levantine Arabic		20 million	0.869	0.813	0.706	0.774	0.461	0.677	0.654	0.757	0.735	0.823
South Levantine Arabic		24 million	0.875	0.824	0.714	0.788	0.485	0.715	0.673	0.767	0.743	0.831
Ta,Aôizzi-Adeni Arabic		11 million	0.869	0.857	0.748	0.816	0.525	0.75	0.725	0.802	0.783	0.842
Tunisian Arabic		11 million	0.837	0.724	0.611	0.686	0.418	0.611	0.57	0.667	0.631	0.773
Khmer	Khmer	16 million	0.797	0.718	0.415	0.08	0.061	0.082	0.117	0.259	0.233	0.699
Santali	Munda	7.5 million	0.018	0.073	0.007	0.002	0.004	0.005	0.001	0.01	0.052	0.387
Vietnamese	Vietic	76 million	0.881	0.867	0.839	0.856	0.623	0.676	0.833	0.854	0.849	0.875
Acehnese (Arabic script)	Malayo-Polynesian	3.5 million	0.141	0.054	0.025	0.042	0.005	0.03	0.014	0.049	0.021	0.097
Acehnese (Latin script)		3.5 million	0.394	0.309	0.195	0.213	0.169	0.219	0.157	0.235	0.209	0.385
Balinese		3.3 million	0.652	0.542	0.375	0.322	0.274	0.298	0.249	0.35	0.383	0.624
Banjar (Arabic script)		4 million	0.179	0.083	0.039	0.054	0.008	0.045	0.019	0.05	0.021	0.093
Banjar (Latin script)		4 million	0.688	0.604	0.459	0.436	0.282	0.297	0.302	0.422	0.47	0.69
Buginese		4 million	0.346	0.228	0.161	0.172	0.161	0.188	0.133	0.194	0.198	0.296
Cebuano		21 million	0.877	0.743	0.496	0.538	0.379	0.38	0.287	0.414	0.614	0.819
Ilocano		8 million	0.765	0.526	0.33	0.265	0.239	0.245	0.162	0.255	0.372	0.672
Indonesian		43 million L1	0.894	0.883	0.859	0.871	0.814	0.815	0.841	0.869	0.869	0.889
Javanese		82 million	0.837	0.7	0.489	0.376	0.256	0.308	0.286	0.436	0.527	0.767
Minangkabau (Arabic script)		6.5 million	0.157	0.057	0.03	0.037	0.006	0.044	0.012	0.038	0.018	0.081
Minangkabau (Latin script)		6.5 million	0.671	0.618	0.422	0.365	0.251	0.265	0.26	0.383	0.416	0.704
Pangasinan		1.5 million	0.487	0.38	0.282	0.291	0.292	0.298	0.206	0.269	0.319	0.492
Plateau Malagasy		5 million	0.813	0.313	0.126	0.289	0.069	0.098	0.074	0.129	0.13	0.504
Standard Malay		18 million L1	0.889	0.872	0.829	0.858	0.742	0.728	0.769	0.83	0.853	0.881
Sundanese		42 million	0.854	0.687	0.464	0.414	0.286	0.325	0.324	0.45	0.47	0.748
Tagalog		28 million	0.889	0.846	0.751	0.798	0.667	0.621	0.428	0.624	0.816	0.876
Waray		3.7 million	0.856	0.679	0.447	0.552	0.386	0.408	0.297	0.403	0.553	0.79
Fijian		330,000	0.501	0.146	0.072	0.094	0.084	0.108	0.057	0.097	0.103	0.226
Maori		185,000 (L2)	0.689	0.412	0.176	0.295	0.166	0.192	0.102	0.2	0.183	0.471
Samoan		500,000	0.728	0.313	0.117	0.118	0.09	0.121	0.076	0.121	0.126	0.4
Central Aymara	Aymara	2 million	0.168	0.085	0.074	0.083	0.072	0.092	0.061	0.093	0.087	0.126
Esperanto	Constructed	2 million (est.)	0.89	0.869	0.798	0.865	0.714	0.707	0.574	0.708	0.807	0.878
Tok Pisin	(English-based)	4 million	0.739	0.529	0.279	0.356	0.299	0.306	0.163	0.249	0.369	0.721
Haitian Creole	(French-based)	10 million	0.839	0.615	0.381	0.443	0.24	0.281	0.169	0.304	0.406	0.739
Papiamento	(Iberian-based)	340,000	0.831	0.702	0.505	0.536	0.426	0.439	0.352	0.504	0.499	0.783
Kabuverdianu	(Portuguese-based)	1.2 million	0.786	0.587	0.436	0.496	0.38	0.412	0.319	0.459	0.454	0.672
Kannada	Dravidian	44 million	0.825	0.77	0.663	0.775	0.016	0.026	0.081	0.314	0.624	0.816
Malayalam		38 million	0.845	0.797	0.664	0.777	0.015	0.027	0.102	0.341	0.663	0.844
Tamil		75 million	0.821	0.799	0.675	0.739	0.053	0.093	0.061	0.19	0.669	0.814
Telugu		81 million	0.846	0.802	0.731	0.772	0.031	0.045	0.108	0.337	0.667	0.831
Tosk Albanian	Albanian	3 million	0.884	0.828	0.655	0.806	0.263	0.288	0.213	0.365	0.622	0.836
Armenian	Armenian	6.7 million	0.867	0.835	0.569	0.838	0.086	0.124	0.078	0.22	0.634	0.841
Latgalian	Baltic	150,000	0.581	0.361	0.182	0.276	0.138	0.173	0.115	0.218	0.233	0.442
Lithuanian		3 million	0.877	0.815	0.668	0.801	0.297	0.292	0.326	0.541	0.787	0.864
Standard Latvian		1.75 million	0.886	0.822	0.665	0.812	0.322	0.35	0.353	0.59	0.785	0.872
Welsh	Celtic	875,000 (L2)	0.896	0.816	0.577	0.749	0.136	0.183	0.118	0.285	0.419	0.813
Irish		1.2 million (L2)	0.86	0.731	0.428	0.58	0.107	0.137	0.082	0.21	0.249	0.72
Scottish Gaelic		60,000	0.8	0.567	0.276	0.249	0.098	0.134	0.073	0.174	0.144	0.564
Afrikaans	Germanic	7 million	0.901	0.878	0.82	0.855	0.684	0.72	0.687	0.786	0.847	0.89
Danish		5.8 million	0.901	0.884	0.855	0.879	0.767	0.81	0.756	0.838	0.873	0.891
German		95 million (L1)	0.898	0.89	0.88	0.891	0.887	0.884	0.863	0.881	0.885	0.894
Limburgish		1.3 million	0.784	0.719	0.535	0.533	0.381	0.418	0.354	0.492	0.601	0.796
Eastern Yiddish		1 million	0.834	0.618	0.1	0.166	0.039	0.053	0.017	0.117	0.261	0.78
Faroese		70,000	0.845	0.639	0.417	0.491	0.254	0.279	0.183	0.317	0.375	0.709
Icelandic		350,000	0.876	0.768	0.526	0.714	0.241	0.252	0.173	0.315	0.476	0.789
Norwegian Bokmal		4 million	0.888	0.87	0.84	0.865	0.748	0.784	0.726	0.814	0.858	0.881
Norwegian Nynorsk		750,000	0.89	0.864	0.816	0.86	0.65	0.687	0.637	0.756	0.838	0.88
Swedish		10 million	0.899	0.892	0.875	0.879	0.791	0.822	0.777	0.841	0.874	0.893
Dutch		24 million	0.883	0.874	0.859	0.873	0.81	0.86	0.828	0.856	0.864	0.878
Luxembourgish		400,000	0.874	0.767	0.565	0.557	0.396	0.404	0.281	0.41	0.493	0.792
Greek	Greek	13 million	0.88	0.854	0.791	0.852	0.604	0.635	0.475	0.672	0.82	0.868
Assamese	Indo-Aryan	15 million	0.785	0.666	0.467	0.32	0.035	0.067	0.167	0.396	0.464	0.719
Awadhi		38 million	0.841	0.769	0.655	0.696	0.243	0.519	0.313	0.53	0.689	0.796
Bengali		265 million	0.855	0.81	0.742	0.791	0.097	0.14	0.392	0.644	0.728	0.831
Bhojpuri		50 million	0.834	0.702	0.56	0.596	0.191	0.444	0.239	0.418	0.602	0.768
Chhattisgarhi		16 million	0.821	0.672	0.541	0.605	0.191	0.471	0.256	0.445	0.589	0.735
Eastern Panjabi		33 million	0.848	0.831	0.686	0.733	0.017	0.037	0.103	0.417	0.587	0.824
Gujarati		55 million	0.853	0.807	0.693	0.725	0.012	0.024	0.197	0.497	0.649	0.838
Hindi		600 million (L2)	0.871	0.841	0.806	0.832	0.408	0.727	0.49	0.705	0.822	0.862
Magahi		14 million	0.843	0.741	0.634	0.667	0.242	0.497	0.293	0.509	0.682	0.801
Maithili		35 million	0.855	0.722	0.589	0.57	0.191	0.454	0.245	0.422	0.624	0.788
Marathi		83 million	0.864	0.809	0.716	0.726	0.131	0.253	0.227	0.464	0.69	0.831
Nepali		25 million	0.851	0.75	0.576	0.717	0.205	0.375	0.233	0.465	0.688	0.825
Odia		37 million	0.796	0.692	0.242	0.027	0.014	0.025	0.055	0.365	0.041	0.637
Sanskrit		14000+	0.624	0.536	0.389	0.41	0.18	0.31	0.165	0.327	0.341	0.596
Sindhi		32 million	0.824	0.721	0.346	0.126	0.042	0.081	0.064	0.167	0.214	0.625

Sinhala		17 million	0.793	0.703	0.026	0.019	0.011	0.016	0.017	0.118	0.233	0.729
Urdu		100+ million L2	0.855	0.828	0.701	0.736	0.188	0.215	0.276	0.505	0.674	0.822
Kashmiri (Arabic script)		7 million	0.497	0.315	0.17	0.221	0.051	0.089	0.062	0.145	0.202	0.383
Kashmiri (Devanagari script)		7 million	0.411	0.213	0.146	0.191	0.069	0.132	0.073	0.144	0.16	0.299
Central Kurdish	Iranian	6 million	0.594	0.763	0.224	0.071	0.014	0.026	0.033	0.099	0.127	0.574
Dari		10-12 million	0.86	0.873	0.745	0.793	0.405	0.415	0.561	0.684	0.775	0.84
Northern Kurdish		15 million	0.615	0.454	0.187	0.455	0.078	0.114	0.1	0.16	0.131	0.447
Southern Pashto		20 million	0.792	0.725	0.395	0.601	0.077	0.12	0.127	0.241	0.234	0.588
Tajik		8-9 million	0.848	0.766	0.212	0.178	0.05	0.1	0.075	0.193	0.141	0.682
Western Persian		55 million	0.873	0.894	0.804	0.839	0.438	0.463	0.601	0.741	0.822	0.864
Catalan	Romance	4 million	0.895	0.885	0.851	0.88	0.781	0.792	0.785	0.843	0.859	0.886
French		80+ million (L1)	0.896	0.891	0.885	0.892	0.892	0.889	0.881	0.887	0.886	0.894
Friulian		600,000	0.796	0.689	0.501	0.577	0.45	0.46	0.376	0.504	0.492	0.751
Galician		2.4 million	0.893	0.869	0.84	0.875	0.832	0.827	0.804	0.85	0.853	0.883
Italian		65 million	0.891	0.882	0.872	0.887	0.884	0.879	0.863	0.875	0.878	0.889
Ligurian		500,000	0.759	0.65	0.493	0.581	0.499	0.498	0.394	0.538	0.522	0.731
Lombard		3.5 million (est.)	0.817	0.663	0.49	0.597	0.447	0.458	0.348	0.503	0.504	0.747
Occitan		2 million	0.889	0.847	0.765	0.806	0.698	0.692	0.622	0.731	0.73	0.858
Portuguese		230 million	0.899	0.891	0.879	0.892	0.888	0.884	0.873	0.883	0.886	0.892
Romanian		24 million	0.898	0.889	0.867	0.873	0.729	0.77	0.754	0.829	0.867	0.893
Sardinian		1 million	0.758	0.68	0.505	0.538	0.426	0.426	0.34	0.476	0.51	0.746
Spanish		483 million L1	0.887	0.877	0.866	0.883	0.877	0.876	0.863	0.875	0.877	0.885
Venetian		2 million	0.858	0.792	0.677	0.772	0.614	0.612	0.542	0.695	0.703	0.842
Asturian		400,000	0.864	0.844	0.78	0.814	0.727	0.73	0.677	0.749	0.797	0.861
Sicilian		4.7 million	0.829	0.704	0.537	0.628	0.419	0.454	0.343	0.509	0.544	0.782
Belarusian	Slavic	6.5 million	0.865	0.815	0.651	0.812	0.171	0.223	0.333	0.567	0.744	0.846
Russian		150 million (L1)	0.889	0.883	0.86	0.884	0.791	0.846	0.855	0.872	0.867	0.888
Ukrainian		35 million	0.892	0.875	0.822	0.873	0.616	0.762	0.729	0.818	0.858	0.885
Bosnian		3 million	0.895	0.869	0.804	0.871	0.612	0.576	0.644	0.788	0.823	0.883
Bulgarian		8 million	0.891	0.869	0.821	0.865	0.624	0.635	0.728	0.812	0.856	0.883
Croatian		5.6 million	0.891	0.87	0.826	0.866	0.595	0.563	0.646	0.781	0.828	0.88
Macedonian		2 million	0.89	0.858	0.762	0.858	0.432	0.45	0.592	0.742	0.797	0.872
Serbian		6.5 million	0.893	0.875	0.801	0.86	0.423	0.456	0.585	0.753	0.825	0.884
Slovenian		2.1 million	0.889	0.85	0.767	0.839	0.531	0.518	0.578	0.727	0.819	0.878
Czech		10.5 million	0.892	0.882	0.856	0.87	0.697	0.771	0.779	0.847	0.862	0.887
Polish		38 million	0.885	0.873	0.846	0.867	0.714	0.763	0.777	0.847	0.861	0.881
Silesian		1 million	0.808	0.698	0.557	0.592	0.362	0.401	0.38	0.541	0.587	0.784
Slovak		5.2 million	0.892	0.864	0.802	0.862	0.602	0.693	0.689	0.807	0.852	0.882
Japanese	Japonic	125 million	0.878	0.858	0.825	0.851	0.761	0.819	0.799	0.846	0.833	0.869
Georgian	South Caucasian	4 million	0.856	0.776	0.449	0.801	0.104	0.138	0.137	0.273	0.541	0.794
Korean	Koreanic	81 million	0.875	0.843	0.786	0.842	0.573	0.766	0.76	0.823	0.792	0.861
Basque	Isolate	750,000	0.865	0.79	0.563	0.786	0.184	0.233	0.128	0.24	0.558	0.832
Halh Mongolian	Eastern Mongolic	3 million	0.834	0.699	0.151	0.514	0.042	0.084	0.065	0.136	0.147	0.613
Wolof	Atlantic	10 million	0.3	0.141	0.088	0.109	0.107	0.147	0.08	0.12	0.11	0.173
Nigerian Fulfulde		14 million	0.191	0.105	0.061	0.072	0.075	0.092	0.05	0.085	0.081	0.128
Bemba	Bantu	4 million	0.302	0.13	0.092	0.107	0.098	0.11	0.068	0.103	0.124	0.249
Chokwe		1.3 million	0.147	0.096	0.071	0.077	0.075	0.117	0.062	0.092	0.098	0.136
Ganda		7 million	0.45	0.156	0.091	0.107	0.08	0.092	0.065	0.097	0.099	0.247
Kamba		4 million	0.202	0.126	0.087	0.095	0.098	0.118	0.068	0.108	0.101	0.171
Kikongo		7 million	0.267	0.118	0.074	0.103	0.101	0.11	0.076	0.12	0.112	0.189
Kikuyu		8 million	0.239	0.158	0.095	0.116	0.112	0.139	0.085	0.119	0.122	0.199
Kimbundu		3 million	0.133	0.077	0.056	0.075	0.071	0.087	0.054	0.077	0.082	0.125
Kinyarwanda		12 million	0.788	0.296	0.096	0.098	0.071	0.091	0.068	0.115	0.114	0.494
Lingala		8-10 million	0.554	0.156	0.095	0.134	0.117	0.135	0.094	0.141	0.118	0.225
Luba-Kasai		6.5 million	0.201	0.1	0.083	0.115	0.104	0.125	0.087	0.112	0.121	0.188
Northern Sotho		5 million	0.632	0.205	0.104	0.117	0.103	0.124	0.092	0.148	0.118	0.38
Nyanja		12 million	0.7	0.215	0.11	0.129	0.101	0.127	0.086	0.133	0.166	0.436
Rundi		9 million	0.679	0.194	0.083	0.083	0.07	0.086	0.062	0.113	0.101	0.322
Shona		11 million	0.764	0.208	0.103	0.149	0.095	0.124	0.086	0.123	0.143	0.531
Southern Sotho		5.6 million	0.744	0.196	0.095	0.1	0.089	0.111	0.087	0.136	0.125	0.461
Swahili		100+ million L2	0.857	0.768	0.665	0.602	0.212	0.233	0.09	0.188	0.736	0.839
Swati		2.5 million	0.55	0.168	0.111	0.112	0.081	0.103	0.073	0.122	0.116	0.382
Tsonga		3 million	0.525	0.15	0.081	0.095	0.082	0.108	0.057	0.092	0.096	0.242
Tswana		5 million	0.624	0.193	0.092	0.104	0.088	0.111	0.075	0.122	0.113	0.377
Tumbuka		2 million	0.504	0.166	0.094	0.105	0.089	0.114	0.069	0.114	0.125	0.284
Umbundu		6 million	0.135	0.076	0.063	0.069	0.064	0.086	0.045	0.078	0.087	0.122
Xhosa		8.2 million	0.776	0.248	0.124	0.154	0.103	0.132	0.077	0.139	0.192	0.612
Zulu		12 million	0.799	0.264	0.101	0.111	0.082	0.107	0.095	0.127	0.168	0.619
Fon	Gbe	1.7 million	0.108	0.075	0.054	0.065	0.068	0.079	0.041	0.062	0.075	0.107
Ewe		7 million	0.138	0.097	0.071	0.08	0.068	0.083	0.054	0.074	0.077	0.124
Kabiye	Gur	1.2 million	0.099	0.101	0.065	0.072	0.051	0.074	0.035	0.061	0.078	0.138
Mossi		7.5 million	0.124	0.076	0.064	0.077	0.066	0.081	0.057	0.076	0.077	0.117
Akan	Kwa	11 million	0.511	0.201	0.109	0.127	0.128	0.148	0.088	0.135	0.147	0.306
Twi		17 million	0.504	0.226	0.133	0.14	0.129	0.161	0.09	0.143	0.158	0.341
Bambara	Mande	14 million	0.119	0.086	0.067	0.076	0.069	0.094	0.051	0.077	0.084	0.12
Dyula		3 million	0.12	0.066	0.054	0.073	0.076	0.097	0.051	0.074	0.073	0.105
Igbo	Volta	27 million	0.691	0.397	0.137	0.091	0.074	0.092	0.063	0.078	0.148	0.483
Yoruba		28 million	0.579	0.216	0.087	0.081	0.068	0.097	0.059	0.077	0.088	0.311
Sango	Ubangian	5 million (L2)	0.154	0.101	0.076	0.091	0.098	0.113	0.074	0.096	0.108	0.145
Luo	Nilotic	4.2 million	0.169	0.087	0.068	0.08	0.094	0.1	0.066	0.078	0.086	0.139
Nuer		1.4 million	0.065	0.038	0.033	0.036	0.023	0.037	0.02	0.05	0.038	0.065
Southwestern Dinka		2 million	0.134	0.111	0.089	0.096	0.096	0.11	0.072	0.098	0.107	0.136
Central Kanuri (Arabic script)	Saharan	4 million	0.043	0.02	0.01	0.019	0.017	0.027	0.011	0.017	0.015	0.026
Central Kanuri (Latin script)		4 million	0.153	0.1	0.073	0.092	0.112	0.12	0.074	0.104	0.087	0.143
Ayacucho Quechua	Quechua II	1 million	0.232	0.182	0.109	0.112	0.113	0.139	0.084	0.129	0.126	0.194
Chinese (Simplified)	Sinitic	920 million (L1)	0.884	0.872	0.847	0.871	0.775	0.829	0.859	0.868	0.855	0.878
Chinese (Traditional)		31 million	0.881	0.861	0.825	0.857	0.714	0.807	0.847	0.855	0.842	0.871
Yue Chinese		60 million	0.884	0.896	0.828	0.858	0.724	0.8	0.84	0.862	0.846	0.873
Burmese	Tibeto-Burman	33 million	0.748	0.672	0.075	0.616	0.021	0.033	0.033	0.094	0.178	0.638
Dzongkha		700,000	0.068	0.11	0.004	0.007	0.004	0.008	0.001	0.005	0.006	0.119
Jingpho		900,000	0.131	0.093	0.075	0.08	0.084	0.106	0.065	0.097	0.072	0.111
Meitei (Bengali script)		1.8 million	0.155	0.065	0.046	0.061	0.012	0.031	0.02	0.052	0.043	0.129
Mizo		900,000	0.334	0.325	0.203	0.185	0.189	0.217	0.158	0.219	0.328	0.593
Standard Tibetan		1.2 million	0.103	0.185	0.011	0.007	0.012	0.014	0.01	0.015	0.018	0.191
Shan	Tai	3 million	0.128	0.417	0.085	0.092	0.107	0.132	0.08	0.1	0.118	0.191
Lao		7.5 million	0.658	0.384	0.073	0.081	0.069	0.093	0.071	0.132	0.125	0.521
Thai		36 million	0.879	0.868	0.819	0.828	0.451	0.591	0.773	0.831	0.818	0.872
Guarani	Tupi	6-7 million	0.547	0.269	0.186	0.181	0.182	0.221	0.14	0.198	0.207	0.331

Northern Uzbek	Karluk	27 million	0.866	0.765	0.539	0.733	0.115	0.151	0.168	0.349	0.501	0.787
Uyghur		10 million	0.773	0.674	0.157	0.12	0.011	0.032	0.023	0.11	0.026	0.44
Bashkir	Kipchak	1.2 million	0.837	0.762	0.311	0.463	0.128	0.192	0.143	0.243	0.384	0.746
Crimean Tatar		300,000	0.765	0.609	0.42	0.518	0.175	0.257	0.215	0.366	0.418	0.705
Kazakh		13 million	0.868	0.788	0.399	0.755	0.102	0.149	0.187	0.325	0.498	0.808
Kyrgyz		4.5 million	0.827	0.731	0.333	0.655	0.086	0.15	0.162	0.278	0.308	0.709
Tatar		5 million	0.863	0.776	0.376	0.715	0.112	0.177	0.158	0.266	0.375	0.739
North Azerbaijani	Oghuz	9-10 million	0.837	0.776	0.618	0.749	0.21	0.262	0.267	0.491	0.636	0.804
South Azerbaijani		15-20 million	0.572	0.437	0.236	0.413	0.065	0.117	0.094	0.146	0.273	0.546
Turkish		75 million	0.884	0.857	0.809	0.82	0.497	0.614	0.625	0.775	0.825	0.878
Turkmen		7 million	0.834	0.538	0.289	0.287	0.102	0.153	0.115	0.211	0.257	0.656
Estonian	Finnic	1.1 million	0.89	0.838	0.708	0.811	0.175	0.222	0.314	0.531	0.777	0.869
Finnish		5.4 million	0.89	0.867	0.805	0.843	0.453	0.606	0.42	0.61	0.821	0.881
Hungarian	Ugric	13 million	0.887	0.871	0.839	0.852	0.486	0.641	0.399	0.61	0.829	0.879

Table 9: The Corpus BLEU results on the FLORES-200 dataset are derived from evaluations of 10 distinct large language models. Population estimates are based on heterogeneous sources, and the reported population are not guaranteed to be accurate. Therefore, they should be interpreted with appropriate caution.

Language Name	Language Branch	Population	GPT4o Mini	Llama 3.1 8B	Llama 3.2 3B	Ministral 8B	Phi-3	Phi-3.5	Qwen2.5 1.5B	Qwen2.5 3B	gemma-2 2B	gemma-2 9B
Central Atlas Tamazight	**Berber**	3-4 million	1.4	0.4	0.4	0.2	1.0	0.8	0.2	0.8	0.4	1.4
Kabyle		5 million	4.0	3.3	1.4	0.9	1.7	0.7	0.5	1.5	1.4	4.3
Tamasheq (Latin script)		500,000	5.2	3.9	2.7	1.9	4.3	1.7	1.0	3.4	3.3	4.9
Tamasheq (Tifinagh script)		500,000	1.3	0.4	0.3	0.2	1.0	0.7	0.1	0.5	0.6	1.1
Hausa	Chadic	40 million	30.4	20.0	7.5	2.9	3.9	1.6	1.5	4.5	8.9	25.9
Somali	Cushitic	20 million	26.6	10.8	5.3	3.2	4.0	1.3	1.9	4.0	4.2	19.1
West Central Oromo		10 million	17.2	3.5	1.9	0.9	1.7	0.7	0.3	1.5	1.1	4.2
Amharic	Semitic	32 million	18.0	8.4	1.1	0.4	1.0	0.8	0.6	2.7	4.8	19.1
Hebrew		9 million	43.6	36.4	21.2	36.9	18.1	9.3	22.3	31.7	33.1	42.6
Maltese		520,000	51.8	41.1	26.1	16.8	9.1	3.6	4.4	12.2	28.3	49.4
Modern Standard Arabic		330 million	39.2	30.1	29.5	33.9	19.0	16.0	27.2	32.6	31.3	38.6
Modern Standard Arabic (Romanized)		330 million	25.1	10.1	4.5	4.8	2.9	1.3	1.3	6.3	2.2	14.2
Tigrinya		9 million	4.7	1.8	0.7	0.3	0.7	0.7	0.2	1.3	1.1	5.5
Egyptian Arabic		60 million	30.9	11.6	21.6	24.9	13.0	10.5	18.4	23.6	21.7	29.5
Mesopotamian Arabic		15 million	33.8	12.2	23.0	26.7	14.9	12.5	20.8	25.9	24.7	31.9
Moroccan Arabic		30 million	29.1	13.7	17.0	18.1	9.9	7.3	13.2	18.4	16.3	25.7
Najdi Arabic		10 million	38.5	19.3	29.0	32.5	17.8	19.6	25.7	31.1	30.1	37.4
North Levantine Arabic		20 million	37.5	15.9	25.0	27.8	15.1	12.5	21.2	27.4	25.0	34.4
South Levantine Arabic		24 million	40.5	15.5	27.1	31.3	17.3	12.7	23.7	30.3	28.1	37.3
Ta'izzi-Adeni Arabic		11 million	35.6	11.2	25.6	29.2	16.3	15.7	23.3	28.0	27.3	33.9
Tunisian Arabic		11 million	30.7	15.3	19.9	22.2	12.8	10.0	17.5	21.8	19.9	28.1
Khmer	Khmer	16 million	25.3	17.4	12.5	2.0	3.1	1.7	3.5	9.2	6.3	22.3
Santali	**Munda**	7.5 million	0.7	3.9	0.5	0.1	0.4	0.3	0.1	0.1	2.1	12.7
Vietnamese	Vietic	76 million	35.8	33.4	30.0	31.4	19.7	12.5	28.6	32.1	29.7	36.6
Acehnese (Arabic script)	Malayo-Polynesian	3.5 million	4.8	1.5	1.0	0.9	0.6	0.5	0.4	1.6	0.5	3.1
Acehnese (Latin script)		3.5 million	12.7	10.7	6.9	5.4	6.1	2.8	2.7	6.2	6.2	13.5
Balinese		3.3 million	22.9	17.9	12.4	8.0	8.5	3.6	4.9	10.1	11.9	22.4
Banjar (Arabic script)		4 million	6.2	1.4	1.2	0.8	0.6	0.5	0.4	1.9	0.5	3.1
Banjar (Latin script)		4 million	24.9	22.4	15.9	12.7	10.0	4.7	7.3	14.4	15.8	27.1
Buginese		4 million	10.2	6.7	5.2	4.5	5.1	2.6	2.7	5.9	6.0	9.4
Cebuano		21 million	42.8	32.6	20.7	19.4	14.3	5.6	9.3	16.3	24.1	39.2
Ilocano		8 million	29.2	20.5	13.6	7.2	8.4	3.8	4.1	9.3	12.6	26.5
Indonesian		43 million L1	44.4	40.9	37.0	38.0	32.4	22.9	33.5	37.3	38.0	44.9
Javanese		82 million	37.7	27.2	18.1	10.3	8.3	3.0	6.7	14.2	18.1	33.4
Minangkabau (Arabic script)		6.5 million	5.7	1.3	0.8	0.7	0.6	0.5	0.3	1.3	0.3	2.9
Minangkabau (Latin script)		6.5 million	24.9	23.1	16.0	9.8	8.9	4.3	6.9	12.4	13.4	27.8
Pangasinan		1.5 million	17.8	14.7	11.7	9.7	10.6	5.4	5.8	10.3	11.0	18.1
Plateau Malagasy		5 million	27.4	11.0	5.2	9.5	3.7	1.5	1.5	3.9	4.5	17.1
Standard Malay		18 million L1	44.5	38.6	34.9	37.7	28.4	17.1	30.1	35.3	36.7	44.5
Sundanese		42 million	35.7	23.5	15.0	10.2	8.0	3.0	6.8	13.6	14.6	29.2
Tagalog		28 million	45.4	40.2	32.5	32.7	24.9	17.8	14.6	26.1	34.7	44.9
Waray		3.7 million	43.3	30.2	18.8	21.4	13.0	6.0	8.5	17.1	21.4	38.1
Fijian		330,000	13.3	5.9	3.5	3.0	3.7	1.5	1.5	3.7	3.6	8.9
Maori		50,000 L1	23.1	14.5	7.8	9.5	7.5	1.4	3.8	8.2	7.1	16.8
Samoan		500,000	26.2	12.5	5.9	3.9	4.5	1.3	1.9	4.6	4.4	16.0
Central Aymara	**Aymara**	2 million	5.7	2.8	2.8	2.3	3.5	1.5	1.0	2.8	2.6	4.8
Esperanto	N/A		45.1	40.3	35.2	40.6	30.2	14.0	23.7	30.5	35.1	44.3
Tok Pisin	(English-based)	120,000 L1	19.8	15.2	9.9	11.4	10.4	2.9	3.7	8.0	11.2	22.6
Haitian Creole	(French-based)	10 million	37.8	24.7	15.3	15.7	8.5	1.9	4.2	11.3	14.9	32.2
Papiamento	(Iberian-based)	340,000	42.1	32.1	21.1	19.2	15.7	5.0	10.3	19.2	18.0	38.9
Kabuverdianu	(Portuguese-based)	1.2 million	39.6	24.2	17.3	18.1	14.8	5.9	9.3	17.7	16.4	31.1
Kannada	South Dravidian	44 million	29.1	17.8	19.2	23.0	1.2	1.3	2.1	8.6	16.3	28.8
Malayalam		38 million	30.8	21.6	18.6	22.7	1.4	0.9	2.3	8.8	18.1	31.4
Tamil		75 million	27.7	16.0	19.3	21.3	2.5	1.8	1.9	6.8	17.4	29.0
Telugu	South-Central Dravidian	81 million	34.8	25.0	23.9	25.0	2.2	1.9	3.0	9.5	19.5	33.5
Tosk Albanian	Albanian	3 million	39.1	28.9	22.8	31.5	8.7	3.0	5.6	12.1	21.1	36.3
Armenian	Armenian	6.7 million	37.6	28.7	18.6	31.9	3.1	1.3	2.8	8.2	20.9	35.3
Latgalian	Baltic	150,000	19.5	11.3	6.3	6.9	3.9	1.4	2.1	5.9	5.5	14.7
Lithuanian		3 million	33.7	28.0	20.2	26.1	8.6	3.9	8.7	16.7	25.7	33.9
Standard Latvian		1.75 million	36.1	28.0	20.1	27.8	8.5	3.0	9.2	18.3	27.0	35.0
Welsh	Celtic	875,000	55.0	45.4	29.5	37.8	7.4	2.2	5.5	14.7	19.5	47.0
Irish	Celtic (Goidelic)	170k L1	37.1	27.8	16.0	20.9	5.6	2.0	3.5	10.2	10.0	30.2
Scottish Gaelic		60,000	30.6	19.6	10.5	8.6	4.4	1.2	2.8	7.1	5.8	21.0
Afrikaans	Germanic	7 million	56.7	52.7	47.2	50.4	36.0	18.6	36.1	45.0	48.7	56.5
Danish		5.8 million	48.3	45.0	40.3	44.1	35.0	30.4	34.2	40.7	43.6	48.5
German		95 million (L1)	44.0	41.3	38.7	41.3	40.0	34.9	35.6	38.4	40.4	44.1
Limburgish		1.3 million	36.4	32.9	23.2	21.6	14.8	6.3	13.1	20.7	25.5	38.2
Eastern Yiddish		1 million	49.5	25.9	7.5	9.1	3.8	1.0	0.5	7.0	14.0	45.9
Faroese		70,000	36.9	25.8	16.5	17.9	10.4	3.9	5.9	12.5	14.0	29.9
Icelandic		350,000	35.2	27.0	17.5	24.4	9.6	4.0	6.9	12.4	16.5	30.0

Norwegian Bokmål		4 million	43.5	40.6	36.8	40.1	30.6	23.8	30.3	36.3	39.2	44.0
Norwegian Nynorsk		750,000	45.0	41.1	37.2	40.5	26.4	14.4	26.4	34.0	39.4	45.0
Swedish		10 million	48.1	46.0	42.9	43.5	35.6	31.2	36.1	40.9	43.0	48.6
Dutch		24 million	31.6	29.7	28.5	29.7	25.8	25.0	25.6	28.6	29.9	32.1
Luxembourgish		400,000	46.6	34.4	23.7	22.5	14.0	5.7	7.0	15.4	19.0	38.6
Greek	Greek	13 million	35.5	32.4	28.2	31.2	19.3	13.9	15.5	23.7	29.8	35.8
Assamese		15 million	26.3	15.5	12.7	6.9	1.8	1.2	4.2	9.2	11.6	23.3
Awadhi		38 million	33.0	6.0	18.6	19.0	6.8	6.1	7.7	13.7	19.1	29.3
Bengali		265 million	33.0	22.6	24.0	24.3	3.8	2.0	10.8	19.1	21.8	31.7
Bhojpuri		50 million	26.5	13.8	14.0	13.4	5.6	3.8	5.0	9.7	14.1	22.7
Chhattisgarhi		16 million	36.6	12.7	17.0	16.7	5.7	5.1	5.5	13.2	17.6	29.3
Eastern Panjabi		33 million	34.8	12.2	23.7	23.9	1.3	0.7	2.9	12.6	18.0	34.5
Gujarati		55 million	36.0	18.8	23.5	22.6	1.3	1.0	5.1	15.2	19.9	35.0
Hindi		600 million	38.8	33.2	29.9	30.6	12.5	16.3	13.8	23.2	30.1	39.1
Magahi		14 million	38.2	14.1	20.9	19.7	7.0	6.1	7.2	13.9	22.1	33.7
Maithili	Indo-Aryan	35 million	36.9	12.0	16.1	12.6	5.1	3.3	4.9	9.4	15.3	28.4
Marathi		83 million	34.1	21.0	21.9	20.1	3.7	2.2	4.9	12.7	19.9	33.3
Nepali		25 million	37.6	24.0	17.1	22.4	5.8	4.6	5.3	13.3	20.4	34.9
Odia		37 million	27.3	21.2	5.7	0.6	1.4	1.1	1.9	9.5	1.1	18.9
Sanskrit		Few thousand L1	15.7	12.7	8.6	7.3	4.3	1.9	2.8	6.7	6.5	15.4
Sindhi		32 million	35.9	8.2	11.6	2.8	1.9	0.9	1.6	4.9	5.7	24.4
Sinhala		17 million	25.8	20.0	1.0	0.4	1.0	0.6	0.6	3.7	5.3	23.1
Urdu		70 million L1	33.3	8.8	22.7	22.7	5.5	2.6	7.4	14.9	20.4	32.2
Kashmiri (Arabic script)		7 million	14.2	6.4	4.9	3.0	2.3	1.1	1.2	3.8	4.3	10.3
Kashmiri (Devanagari script)		7 million	11.3	5.1	3.9	3.0	3.4	2.0	1.2	4.0	3.5	8.1
Central Kurdish		6 million	19.3	5.9	8.1	2.2	1.1	0.6	1.1	3.3	4.1	19.7
Dari		10-12 million	37.0	10.1	27.7	29.7	12.6	4.6	17.5	24.2	28.4	36.8
Northern Kurdish	Iranian	15 million	19.3	14.5	6.3	13.2	3.2	1.2	1.4	3.9	4.0	15.5
Southern Pashto		20 million	29.0	9.0	12.2	17.3	2.9	1.1	3.6	7.0	5.8	19.9
Tajik		8-9 million	30.9	11.4	6.1	4.2	2.2	1.0	1.7	5.4	3.7	23.1
Western Persian		55 million	34.8	15.6	27.8	29.7	12.6	3.7	17.5	24.6	28.4	35.8
Catalan		4 million	46.4	43.2	39.6	42.3	33.1	25.0	32.8	38.9	40.6	46.6
French		80+ million (L1)	45.2	42.9	39.9	42.6	41.6	37.3	38.2	41.3	42.1	45.5
Friulian		600,000	33.7	28.2	19.3	20.1	14.8	5.0	12.2	17.5	16.9	31.8
Galician		2.4 million	41.4	37.0	33.5	36.7	33.8	24.2	30.9	34.6	36.0	40.5
Italian		65 million	32.9	31.2	29.8	31.8	30.6	27.4	27.6	30.5	31.4	34.2
Ligurian		500,000	35.1	28.3	20.3	22.6	19.2	7.0	13.1	21.0	20.7	33.7
Lombard		3.5 million (est.)	35.8	25.9	19.6	22.4	16.1	5.9	10.4	18.8	19.2	32.2
Occitan	Romance	2 million	52.1	46.1	38.5	40.5	31.6	11.3	25.8	35.9	34.4	47.7
Portuguese		230 million	49.8	47.3	44.1	46.7	45.0	41.5	42.0	45.1	46.1	49.9
Romanian		24 million	43.1	40.0	36.9	37.9	27.5	15.9	29.4	34.8	38.6	43.9
Sardinian		1 million	34.4	31.2	22.0	21.6	15.6	6.1	11.8	19.1	20.7	35.7
Spanish		483 million L1	30.9	28.4	27.0	29.7	28.5	23.8	26.2	27.9	29.3	31.1
Venetian		2 million	40.0	34.7	27.0	31.7	23.9	6.6	18.6	28.3	28.8	40.5
Asturian		400,000	39.8	37.5	32.9	34.7	29.2	14.9	26.0	29.7	33.1	40.1
Sicilian		4.7 million	35.5	28.9	21.7	24.4	15.3	3.8	11.4	19.1	20.1	34.4
Belarusian		6.5 million	20.8	16.5	13.1	17.4	4.7	2.6	6.3	11.7	15.3	20.2
Russian	Slavic (East)	150 million (L1)	35.9	33.0	30.5	33.2	26.6	24.3	28.7	31.5	32.4	35.9
Ukrainian		35 million	39.7	36.2	31.2	35.3	22.1	21.6	24.7	31.1	34.3	39.9
Bosnian		3 million	42.5	38.1	32.0	37.1	22.5	12.2	23.9	31.9	33.6	42.2
Bulgarian		8 million	40.9	37.3	33.2	35.6	22.2	17.9	25.5	31.9	35.2	41.3
Croatian	Slavic (South)	5.6 million	37.7	34.9	31.3	33.4	20.4	12.0	22.3	29.0	30.7	37.8
Macedonian		2 million	42.0	37.7	30.7	36.1	16.0	7.9	21.3	30.3	32.0	41.7
Serbian		6.5 million	43.3	39.7	33.0	36.9	15.7	7.7	21.1	30.6	34.4	42.8
Slovenian		2.1 million	35.9	30.9	26.5	29.2	17.0	9.3	17.2	24.5	28.4	35.4
Czech		10.5 million	40.2	37.8	34.2	35.5	24.6	23.1	27.2	33.8	35.1	40.4
Polish	Slavic (West)	38 million	30.1	27.5	25.3	26.6	19.9	14.1	21.9	25.2	27.0	30.5
Silesian		<1 million	36.1	27.4	22.5	21.9	13.0	6.0	13.5	20.7	21.7	35.2
Slovak		5.2 million	39.7	34.6	30.1	34.2	20.5	14.6	23.6	30.5	33.6	39.3
Japanese	Japonic	125 million	26.5	23.2	20.5	21.9	17.8	16.6	18.9	22.4	21.7	26.3
Georgian	South Caucasian	4 million	27.5	20.3	11.3	21.5	3.2	1.4	3.0	7.0	12.1	24.4
Korean	Koreanic	81 million	29.3	25.1	21.1	24.4	13.9	16.5	19.4	23.8	20.9	29.0
Basque	N/A	750,000	30.1	24.7	15.3	23.6	4.9	1.6	2.8	6.2	15.3	28.8
Halh Mongolian	Eastern Mongolic	3 million	28.1	8.9	4.4	12.1	1.6	0.9	1.2	4.3	3.5	17.6
Wolof	**Atlantic**	10 million	10.2	5.7	3.9	2.9	4.4	1.4	2.0	5.0	3.5	6.7
Nigerian Fulfulde	**Atlantic–Fula**	14 million	6.8	4.1	2.5	2.5	3.9	1.6	1.3	3.5	2.6	5.3
Bemba		4 million	10.4	6.1	4.3	3.9	5.5	2.1	1.8	4.5	5.1	9.9
Chokwe		1.3 million	5.7	3.5	2.9	1.9	4.0	1.6	1.5	3.1	3.2	5.0
Ganda		7 million	15.0	7.1	4.5	3.0	4.6	1.7	1.9	4.1	4.2	10.1
Kamba		4 million	7.6	5.8	4.3	2.9	4.9	1.6	1.5	4.2	3.4	6.9
Kikongo		7 million	8.8	4.4	3.2	2.6	4.4	1.7	1.4	4.4	3.5	6.0
Kikuyu		8 million	8.2	5.7	3.3	3.2	4.8	1.9	1.3	3.8	3.8	6.5
Kimbundu		3 million	6.0	3.3	2.6	2.3	3.6	1.4	1.2	3.5	3.4	5.5
Kinyarwanda		12 million	27.7	11.3	4.6	3.5	4.1	1.2	1.4	3.8	4.6	17.9
Lingala		8-10 million	16.0	5.8	4.2	3.9	4.9	1.5	1.9	4.7	3.7	7.8
Luba-Kasai		6.5 million	7.7	3.8	2.7	2.9	4.1	2.0	1.8	4.4	3.9	6.8
Northern Sotho		5 million	27.9	9.9	5.4	3.6	4.7	1.8	1.3	5.0	4.4	18.0
Nyanja	Bantu	12 million	21.9	8.7	4.4	3.8	4.7	1.5	2.3	5.4	6.1	15.3
Rundi		9 million	18.0	6.8	3.6	2.4	3.2	1.4	1.3	3.4	3.1	10.3
Shona		11 million	23.7	8.7	4.6	3.4	4.9	1.7	1.5	5.3	5.4	17.7
Southern Sotho		5.6 million	29.0	9.3	5.0	3.3	5.0	1.6	1.2	4.9	4.4	18.5
Swahili		16 million L1	43.1	35.0	28.8	23.8	8.5	1.5	3.4	9.2	29.5	42.3
Swati		2.5 million	18.2	7.3	4.2	3.3	4.0	1.7	1.6	4.6	3.6	14.1
Tsonga		3 million	18.6	7.3	4.3	3.0	4.7	1.7	1.7	4.1	3.5	9.9
Tswana		5 million	19.5	7.5	4.4	2.7	4.2	1.6	1.0	4.1	4.1	12.9
Tumbuka		2 million	11.7	6.2	3.7	3.2	4.3	1.5	1.4	4.1	4.4	8.6
Umbundu		6 million	5.5	3.0	2.7	2.2	3.6	1.3	1.0	3.1	3.0	5.0
Xhosa		8.2 million	31.8	10.5	5.4	4.6	5.1	1.5	1.6	5.6	6.8	25.0
Zulu		12 million	33.4	11.1	4.6	3.2	4.2	1.5	1.4	4.7	5.1	24.7
Fon	**Gbe**	1.7 million	3.7	2.4	1.7	1.4	2.8	1.2	0.9	2.3	2.2	3.5
Ewe		7 million	5.1	2.9	2.5	2.1	3.3	1.3	0.8	2.4	2.2	4.3
Kabiyè	**Gur**	1.2 million	3.8	3.1	1.9	1.6	2.7	1.2	0.5	2.2	2.2	4.5
Mossi		7.5 million	4.5	2.7	2.3	2.4	3.3	1.1	1.4	3.0	2.9	4.5
Akan	Kwa	11 million	13.4	7.5	5.0	3.6	5.9	2.2	1.5	5.2	5.3	10.4
Twi		17 million	14.6	9.0	5.4	3.4	5.8	2.3	1.6	5.4	5.6	11.8
Bambara	Mande	14 million	5.8	3.0	2.6	2.4	3.9	1.1	1.0	3.7	3.0	5.0
Dyula		3 million	4.2	2.0	1.6	1.8	3.0	1.0	0.8	2.6	2.6	3.6
Igbo	Volta–Niger	27 million	24.0	14.2	5.7	1.6	3.5	1.6	0.9	3.7	5.7	17.6

Yoruba		28 million	17.3	8.6	3.9	2.8	3.5	1.2	1.7	4.4	3.4	11.0
Sango	Creolized Ubangian	400,000 L1	4.7	3.0	2.3	2.4	3.6	1.1	1.4	3.3	2.7	4.1
Luo	**Nilotic**	4.2 million	6.3	3.6	3.3	2.9	3.9	1.6	1.7	3.9	3.2	5.3
Nuer		1.4 million	3.4	2.0	1.8	1.1	2.2	0.9	0.6	1.7	1.8	3.0
Southwestern Dinka		2 million	6.1	5.0	3.8	3.5	5.0	2.0	1.8	4.0	4.5	6.0
Central Kanuri (Arabic script)	**Saharan**	4 million	2.2	1.1	0.7	0.6	0.9	0.6	0.3	1.3	0.5	1.4
Central Kanuri (Latin script)		4 million	5.9	3.1	2.8	2.9	4.9	2.3	1.2	4.0	2.6	5.3
Ayacucho Quechua	**Quechua**	1 million	6.3	5.6	3.7	2.7	4.3	2.0	1.2	3.6	3.4	5.5
Chinese (Simplified)	Sinitic	920 million	28.8	25.4	23.9	24.8	19.8	19.7	24.5	26.4	24.5	28.6
Chinese (Traditional)		31 million	27.4	23.8	21.8	23.4	17.3	16.5	22.5	25.0	22.0	27.3
Yue Chinese		60 million	29.6	14.8	23.5	25.7	19.6	15.7	24.6	26.7	23.6	29.5
Burmese	**Tibeto-Burman**	33 million	21.5	12.1	2.1	14.3	1.3	0.9	1.3	4.2	4.0	17.7
Dzongkha		700,000	0.8	1.5	0.1	0.0	0.1	0.1	0.0	0.3	0.1	1.6
Jingpho		900,000	4.0	2.5	1.8	1.8	2.7	1.4	0.9	2.5	2.3	3.9
Meitei (Bengali script)		1.8 million	4.4	1.9	1.8	1.0	0.8	0.7	0.3	1.8	0.9	4.1
Mizo		900,000	9.3	8.6	6.8	5.2	5.9	3.1	2.7	5.4	8.3	14.2
Standard Tibetan		1.2 million	1.9	3.5	0.4	0.1	0.6	0.5	0.3	0.7	0.5	3.8
Shan	**Southwestern Tai**	3 million	4.0	6.0	1.7	1.1	2.4	1.7	0.7	1.6	3.2	5.1
Lao	Tai	7.5 million	20.1	10.3	2.2	2.1	3.5	2.5	1.8	6.3	3.7	17.8
Thai		36 million	29.6	21.0	23.6	23.0	11.4	10.6	20.1	25.1	23.7	30.6
Guarani	Tupi–Guarani	6-7 million	16.1	8.9	5.6	4.3	5.6	1.8	2.0	5.5	5.7	10.4
Northern Uzbek	Karluk	27 million	32.2	21.5	14.0	21.0	3.3	1.0	3.7	8.7	12.0	28.5
Uyghur		10 million	20.3	7.3	4.4	3.0	0.8	0.4	0.6	2.9	1.5	11.0
Bashkir	Kipchak	1.2 million	27.4	16.3	7.9	10.2	3.5	1.2	2.6	6.0	8.7	23.1
Crimean Tatar		300,000	24.6	16.9	11.7	13.8	5.6	2.4	4.9	9.7	11.3	23.0
Kazakh		13 million	33.8	19.6	11.6	20.9	3.1	1.5	4.5	9.3	12.3	28.6
Kyrgyz		4.5 million	22.6	11.1	7.6	13.9	2.5	1.1	3.1	6.4	6.6	17.9
Tatar		5 million	29.1	13.9	10.2	19.1	3.5	1.4	3.0	7.2	8.8	23.3
North Azerbaijani	Oghuz	9-10 million	22.8	13.2	13.9	17.2	5.0	2.5	5.0	10.3	13.3	21.7
South Azerbaijani		15-20 million	14.7	5.4	5.6	8.9	2.3	0.9	1.3	3.7	5.5	14.4
Turkish		75 million	37.9	33.4	27.3	28.9	12.8	9.3	18.5	26.0	28.4	37.9
Turkmen		7 million	29.2	15.5	8.7	6.7	3.2	1.6	2.1	5.6	5.9	21.3
Estonian	Finnic	1.1 million	38.2	31.3	23.2	28.7	6.2	2.4	8.9	17.5	26.6	36.6
Finnish		5.4 million	35.0	30.5	26.0	28.5	12.2	10.0	11.8	19.6	26.6	34.0
Hungarian	Ugric	13 million	35.5	31.7	28.4	29.3	13.8	11.5	11.3	19.6	28.3	35.5

Table 10: Performance testing after SFT on Corresponding Validation Dataset (#1000 samples)

Language Pair	Methods	spBLEU	ChrF++	Jaccard	LLMaaJ
As-En	BM	8.75	22.72	0.16	0.64
	DN	9.00	23.03	0.16	0.65
	DL	8.87	23.04	0.16	0.59
	DG	9.43	23.69	0.16	0.62
En-As	BM	2.27	10.84	0.03	0.37
	DN	8.75	22.72	0.16	0.64
	DL	8.09	29.03	0.18	0.61
	DG	8.07	29.23	0.18	0.65
Kh-En	BM	0.63	14.66	0.06	0.05
	DN	NA	NA	NA	NA
	DL	2.79	18.66	0.10	0.10
	DG	4.81	23.43	0.14	0.30
En-Kh	BM	0.22	0.50	0.00	0.00
	DN	NA	NA	NA	NA
	DL	4.81	16.95	0.15	0.17
	DG	11.58	29.19	0.23	0.51
Uk-En	BM	22.50	41.35	0.30	0.72
	DN	25.34	44.06	0.33	0.77
	DL	25.29	44.08	0.33	0.76
	DG	24.81	43.76	0.32	0.78
En-Uk	BM	13.57	30.19	0.15	0.60
	DN	17.87	34.83	0.18	0.70
	DL	17.97	34.83	0.19	0.69
	DG	18.10	34.97	0.19	0.72
En-Lb	BM	6.46	26.78	0.12	0.36
	DN	37.98	55.41	0.37	0.82
	DL	40.71	59.02	0.44	0.87
	DG	44.58	59.73	0.45	0.87
Lb-En	BM	26.31	45.98	0.33	0.58
	DN	42.78	59.33	0.48	0.82
	DL	54.64	70.98	0.57	0.82
	DG	59.88	74.97	0.63	0.90

Tao–Filipino Neural Machine Translation: Strategies for Ultra–Low-Resource Settings

Adrian Denzel Macayan*, **Luis Andrew Madridijo***,
Ellexandrei Esponilla, Zachary Mitchell Francisco
De La Salle University-Manila
Manila, Philippines
{adrian_macayan, luis_madridijo}@dlsu.edu.ph
{ellexandrei_esponilla, zachary_francisco}@dlsu.edu.ph

Abstract

Neural Machine Translation (NMT) performance degrades significantly in ultra-low-resource settings, particularly for endangered languages like Tao (Yami) which lack extensive parallel corpora. This study investigates strategies to bootstrap a Tao-Tagalog translation system using the NLLB-200 (600 million parameter) model under extremely limited supervision. We propose a multi-faceted approach combining domain-specific fine-tuning, synthetic data augmentation, and cross-lingual transfer learning. Specifically, we leverage the phylogenetic proximity of Ivatan, a related Batanic language, to pre-train the model, and utilize dictionary-based generation to construct synthetic conversational data. Our results demonstrate that transfer learning from Ivatan improves translation quality on in-domain religious texts, achieving a BLEU score of 34.85. Conversely, incorporating synthetic data enhances the model's ability to generalize to conversational contexts, mitigating the domain bias often inherent in religious corpora. These findings highlight the effectiveness of exploiting linguistic typology and structured lexical resources to develop functional NMT systems for under-represented Austronesian languages.

1 Introduction

Neural Machine Translation (NMT) systems have achieved substantial success for high-resource language pairs, largely due to the availability of extensive parallel and monolingual corpora, which often contain millions of sentence-aligned examples (Bahdanau et al., 2015; Vaswani et al., 2017). The abundance of such resources enables NMT models to process large datasets, learn robust cross-lingual representations, and produce high-quality translations across diverse domains. However, the majority of the world's languages—beyond English, French, Chinese, and a few others—lack such extensive parallel data. In particular, endangered and low-resource languages often possess only a few thousand computer-accessible sentence pairs, severely limiting the effectiveness of standard NMT training pipelines (Haddow et al., 2022). Recent multilingual and massively pre-trained sequence-to-sequence models have broadened NMT capabilities to low-resource settings. By sharing parameters across hundreds of languages, these models induce partially language-agnostic representations that enable zero-shot generalization (Costa-jussà et al., 2024). Nevertheless, performance remains uneven and inconsistent for many low-resource languages. Ultra-low-resource languages—those with limited corpora, no presence in pre-training data, or distinctive morphological characteristics—still experience degraded translation quality. This gap highlights persistent inequities in access to translation tools and underscores the need for improved strategies that go beyond traditional data scaling.

At the same time, many low-resource or "data-poor" languages possess extensive linguistic documentation accumulated through long-term fieldwork, community initiatives, and academic research, which can be leveraged computationally. Such resources—grammars, dictionaries, religious translations, and transcribed oral traditions—contain structured linguistic knowledge not captured by conventional corpora. Tao (Yami), a Batanic language spoken on Orchid Island, Taiwan, exemplifies this scenario. Although Tao lacks the large-scale parallel corpora typical of mainstream NMT approaches, it maintains rich lexical resources, ranging from Biblical translations to community-authored texts. These circumstances raise an important research question: can structured linguistic resources meaningfully compensate for data scarcity when developing NMT systems for endangered languages? To address this question, the present work investigates ultra-low-resource strategies for bootstrapping a Tao–English transla-

*Equal contribution

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 27–36

tion system under extremely limited supervision (fewer than 5,000 parallel sentences). Instead of relying solely on data volume, we focus on three complementary approaches that leverage linguistic structure and other indicators to reduce degradation and improve translation quality. Ultimately, this study proposes a scalable framework for transforming static linguistic archives into functional translation models, offering a roadmap for other under-represented Austronesian language.

Primary Corpus Construction We compile, digitize, and standardize a parallel Tao–English corpus drawn from diverse domains. Our primary sources include religious texts (specifically the Tao translation of the New Testament), educational publications, and community-authored narratives.

Synthetic Data Augmentation To address the scarcity of authentic parallel data, we employ two complementary data augmentation strategies: **Dictionary-Assisted Generation:** We generate synthetic sentence pairs by employing the comprehensive Tao–Chinese–English dictionary and morphological rules provided by Rau and Dong (2006). We map high-frequency lexical items and grammatical constructions to their target equivalents, expanding the coverage of the training data to include morphological patterns commonly found in conversational Tao that is not fully represented in the Bible corpus. **Pivot-Based Augmentation:** We supplement the limited authentic corpora by utilizing Mandarin and Tagalog as pivot languages. We utilize commercial translation systems (e.g., Google Translate) to translate Mandarin resources into Tagalog and English, creating pseudo-parallel pairs that align with our Tao data. This approach increases data diversity and introduces semantic variations that help the model generalize beyond the specific domains of the primary corpus.

Transfer Learning Strategy We adopt a transfer learning approach to leverage the phylogenetic relationships within the Austronesian language family. We initialize our Neural Machine Translation (NMT) models using weights pre-trained on high-resource languages. We specifically fine-tune on typologically related Batanic languages (such as Ivatan) and regionally dominant languages (such as Tagalog) before adapting to Tao. This curriculum learning strategy allows the model to leverage shared morphological features and cognates, with the intent to significantly enhance translation performance in ultra-low-resource conditions.

2 Related Works

Multilingual and Transfer-Learning Approaches for Low-Resource NMT: A well-established strategy for improving NMT for low-resource languages is leveraging multilingual and transfer-learning techniques. Early foundational work demonstrated that NMT models trained on high-resource language pairs can provide robust parameter initializations that improve translation quality when fine-tuned on low-resource pairs (Zoph and Knight, 2016). Sharing sub-word vocabulary and morphology between source and target languages further boosts performance when the languages are related, based heavily on the principle of linguistic proximity (Nguyen and Chiang, 2017). Recent studies explore meta-learning approaches for adaptation to low-resource languages, demonstrating that models can achieve competitive BLEU scores with as few as 600 parallel sentences (Gu et al., 2018). However, survey work confirms that these methods struggle when the target language is unseen in pre-training or is typologically distant from high-resource languages (Haddow et al., 2022; Costa-jussà et al., 2024).

Domain Divergence and "Auxiliary Fine-Tuning: A significant, yet often overlooked, challenge in this transfer learning paradigm is Domain Divergence. In ultra-low-resource settings, researchers are often forced to rely on auxiliary domains—most commonly religious texts— that diverge significantly from the target application of daily conversation. Ranathunga et al. (2024) explicitly addressed this in Exploiting Domain-Specific Parallel Data, investigating the impact of domain mismatch on Multilingual Sequence-to-Sequence Language Models (msLMs). Their study confirmed that while msLMs provide strong initialization, they fail to generalize when fine-tuned solely on divergent auxiliary data due to substantial lexical distribution mismatches. Crucially, they propose "auxiliary fine-tuning"—adapting the model to a high-resource related language in the target domain (e.g., Tagalog conversational text) prior to the final low-resource adaptation—as a mechanism to bridge this semantic gap.

Ultra-Low-Resource NMT: In scenarios where both parallel and monolingual corpora are extremely limited, semi-supervised and unsupervised approaches such as back-translation or noise-augmented self-training are often applied (Sennrich

et al., 2016). However, these methods are constrained by the scarcity of usable monolingual data, especially in endangered or under-documented languages. Studies indicate that ultra-low-resource settings (fewer than 5,000 sentence pairs) require additional techniques beyond standard semi-supervised NMT to achieve reliable translation quality (Aharoni et al., 2019).

The Shift to Large Language Models (LLMs): The period from 2024 to 2026 has witnessed a paradigm shift from specialized encoder-decoder architectures (like NLLB) to general-purpose Large Language Models (LLMs). The Lin et al. (2025) study established the first comprehensive benchmark for Formosan languages (Atayal, Seediq, Paiwan), which share significant phylogenetic proximity to Tao (Yami). Their findings reveal a complex landscape: while off-the-shelf LLMs initially struggle with the VSO word order and focus systems typical of Austronesian languages, they exhibit remarkable few-shot learning capabilities when prompted with high-quality dictionary definitions. This suggests that the future of ultra-low-resource NMT may lie not in training from scratch, but in Parameter-Efficient Fine-Tuning (PEFT) of massive pre-trained models. Techniques such as Low-Rank Adaptation (LoRA) allow for the adaptation of large models (7B+ parameters) on consumer hardware by updating less than 1% of the parameters, effectively mitigating the risk of "catastrophic forgetting" often observed when over-training on tiny datasets.

Dictionary-Based Data Augmentation: Dictionary-based augmentation has emerged as a viable strategy for extremely low-resource MT. By leveraging bilingual lexica, morphological patterns, or rich wordlists, synthetic parallel sentence pairs can be generated to improve coverage of rare vocabulary and grammatical constructions (García et al., 2019; Zhang et al., 2023). These approaches are particularly useful when monolingual corpora are insufficient for self-supervised methods, and they complement transfer learning from related languages. Furthermore, recent scholarship argues that quantity of data is secondary to cultural and semantic fidelity. Lovenia et al. (2024) introduced SEACrowd, a collaborative data hub specifically for Southeast Asian languages. Unlike global datasets which often suffer from "translationese," SEACrowd standardizes corpora across nearly 1,000 indigenous languages, explicitly addressing the "cultural misrepresentation" prevalent in Western-centric models. This aligns with the release of Oepen et al. (2025), which employs advanced language identification filters to recover usable monolingual data for languages previously discarded as noise. For Tao, this implies that augmenting training data with "culturally aware" synthetic sentences—derived from folklore or community narratives rather than generic web text—is essential for preserving semantic nuance.

MT for Endangered, Indigenous, and Austronesian Languages: Recent work highlights translation efforts for endangered and indigenous languages, including South American, North American, and Austronesian languages (Cardoso et al., 2022; Rodríguez et al., 2023). Such studies emphasize the importance of domain-specific corpora (e.g., religious or educational texts) and careful selection of primary data for transfer learning. While Austronesian languages such as Hawaiian, Māori, and Ivatan have been examined in the context of MT, few studies explicitly address the Yami (Tao) language. However, significant foundational work in Yami computational linguistics exists. Yang et al. (2010) proposed a model for constructing a Yami WordNet, creating a crucial lexical database aligned with English semantic concepts. Subsequent studies expanded this into ontological resources, including an integrated semantic network for *ka-* verbs (Yang et al., 2011) and a computational analysis of Yami emotion phrases (Yang et al., 2012). These computational lexical resources provide a structured basis that can be leveraged and expanded on to enhance machine translation performance for the Tao language.

Critical Synthesis and Limitations of Existing Methodologies: While the literature provides robust individual strategies for low-resource NMT, a critical synthesis reveals a distinct gap in their application to Batanic languages. Existing transfer learning approaches predominantly focus on high-resource pivots (like English or Spanish), often neglecting the potential of "phylogenetic transfer" from closely related, yet still low-resource, sister languages (such as Ivatan). Furthermore, while dictionary augmentation is widely proposed, there is a lack of empirical research on how this specifically interacts with "Liturgical Bias"—the tendency of models trained on Bible corpora to hallucinate archaic formality in casual settings. Current methodologies largely fail to address the "Domain-Register Incongruence" that occurs when a model pre-trained on modern web text (Tagalog) is fine-

tuned on archaic religious text (Tao Bible), and then expected to translate daily conversation. This study aims to address this specific intersection: bridging the gap between phylogenetic transfer learning and domain-adaptive synthetic data generation to construct a functional NMT system for an ultra-low-resource, endangered Austronesian language.

3 Language, Data and General Setup

The Yami Language: Yami, autonymically known as Tao, is an Austronesian language spoken by the indigenous Tao people of Orchid Island (Lanyu), located off the southeastern coast of Taiwan. Despite its geographic proximity to the main island of Taiwan, Yami is phylogenetically classified within the Batanic branch of the Malayo-Polynesian family, rather than the Formosan languages (Smith, 2017). Based on lexical innovations and archaeological evidence, Smith (2017) groups Batanic languages with the Northern Luzon and Greater Central Philippine languages, distinguishing them from other Malayo-Polynesian branches.Typologically, Yami exhibits the structural characteristics of a Philippine-type language (Rau and Dong, 2006). It features a dominant Verb-Initial (VSO) word order and a rich agglutinative morphology. A defining grammatical feature is its complex focus system, where verbal affixes—including prefixes, infixes, and circumfixes—signal the semantic relationship (Actor, Patient, Locative, or Instrumental) between the verb and the focused noun phrase. Currently, the language is classified as endangered, with inter-generational transmission threatened by the widespread adoption of Mandarin Chinese.

Dictionary: The primary lexical resource for the language is the *Yami Texts with Reference Grammar and Dictionary*, compiled by Rau and Dong (2006). This comprehensive volume provides Yami lexical entries with definitions in both English and Chinese, capturing the nuances of the language as spoken on Orchid Island.

Parallel Corpus: As Yami is a ultra-low-resource language, large-scale parallel corpora are not readily available. The parallel data used in this study primarily consists of the texts collected by Rau and Dong (2006), which include cultural narratives and daily conversations aligned with Tagalog and Chinese translations. Furthermore, we utilize the Yami translation of the New Testament Bible, which provides a substantial number of aligned sentence pairs. These sources were manually cleaned and aligned to create a Yami-Tagalog parallel dataset suitable for training and validation.

Model: To address the data scarcity of the Yami language, we employ a transfer learning approach using NLLB-200 (No Language Left Behind), a multilingual machine translation model developed by NLLB Team et al. (2022). Unlike general-purpose Large Language Models (LLMs), NLLB-200 utilizes a Transformer-based encoder-decoder architecture explicitly optimized for translation across 200+ languages. We utilize the 600-million parameter variant as our foundational base. The model is particularly well-suited for this task as it was pre-trained on a massive dataset including several Austronesian and Philippine-type languages (e.g., Tagalog, Ilokano, Cebuano, and Pangasinan) that share typological and genealogical features with Yami. We fine-tune the model on the Yami-English parallel corpus, leveraging the model's pre-existing cross-lingual representations to improve alignment and translation generation quality in an ultra-low-resource setting.

Evaluation Set: We establish a hybrid evaluation benchmark to assess performance on the Yami-Tagalog language pair. The set consists of (1) a Cognate Challenge Set: 20 manually curated sentence pairs derived from Rau and Dong (2006), where Yami terms are mapped to Tagalog equivalents and verified via morphological analysis of shared Austronesian cognates; and (2) a Religion Domain Set: 200 verse-pairs randomly sampled from the parallel Bible corpus. We have made sure that the parallel corpus and the evaluation set do not overlap.

Evaluation Metrics We evaluate the quality of our translation models using a combination of lexical, morphological, and semantic metrics. First, we report BLEU (Papineni et al., 2002), the standard metric for n-gram overlap, to provide a baseline comparable to existing literature. Second, given the rich morphology of the Yami language, we employ chrF++ (Popović, 2015), a character n-gram F-score. Unlike word-level metrics, chrF++ is less sensitive to tokenization errors and captures morphological accuracy in languages with complex affixation. Finally, to assess semantic fidelity beyond lexical overlap, we utilize SBERT (Sentence-BERT) (Reimers and Gurevych, 2019). We compute the cosine similarity between the embeddings of the generated translation and the reference text, which credits translations that are semantically correct even if they diverge lexically from the standard.

4 Methodology

4.1 Extracting the Bible Parallel Corpus

To extract the text required for machine translation of Yami to other languages, we decide to source the text from the various Philippine language translations of the Bible. As a key religious text, it is readily available as computer-readable data and sufficient size for a ultra-low-resource NMT model. Table 1 contains select Philippine languages and the respective source Bible edition that serves as the basis of the primary corpus for each language.

Table 1: List of Languages and Bible Versions

Language	Bible Version / Source
Bikolano	Marahay na Bareta Biblia
Ilokano	Ti Baro a Naimbag a Damag Biblia
Ivatan	VTSP (Bible.com)
Pangasinan	Maung A Balita Biblia
Tagalog	Ang Biblia (2001)
Yami	Seysyo No Tao

4.2 Selecting a Target Language for Yami Machine Translation

We select Tagalog as the target language of the Yami translation to strengthen general Filipino-based translation models and provide a versatile real-world use case, given its dominance in the country's capital, major urban and suburban areas, namely Metro Manila and the CALABARZON region. Furthermore, the presence of multilingual translation model, with regards to the Philippine branch, is centered on Tagalog—one of the most common representations on NLP platforms such as HuggingFace (Hugging Face, 2025).

4.3 Data Preprocessing

The parallel corpus for improving improve and fine-tuning the selected model consisted of aligned Bible verse pairs for Yami-Tagalog. In accordance with standard best practices in machine translation and text normalization, all alphabetic characters are converted to lowercase to reduce vocabulary sparsity and simplify the model's learning space. Whitespace inconsistencies were also standardized to avoid unintended tokens that could negatively affect tokenization quality. After normalization, all verse pairs were examined for alignment completeness. Verses in one language that did not have a corresponding verse in the paired language were removed to maintain a strictly parallel corpora, ensuring that every training example consists of a clean and fully aligned source–target pair. After filtering, the textual data was passed through a tokenizer, with each verse constrained to a maximum length of 128 tokens. This constraint follows common practice in transformer-based MT systems, which typically limit input lengths for efficient batch processing (Ahmad et al., 2024). Any verse exceeding 128 tokens was truncated to preserve uniformity in sequence length and computational feasibility.

4.4 Creating a Synthetic Yami-Tagalog Parallel Corpus

By using only the Bible parallel corpus to train the base NLLB-200 Model, there is a risk of the model being biased towards the often archaic sentence structure and vocabulary of Bible verses. In order to prevent the model from being limited to performance in translating Bible verses, we elect to obtain parallel sentences in different context outside of faith-based circumstances. While direct Yami-to-Tagalog human translation of sentences are severely lacking in resource, Yami-to-Chinese parallel corpora are more readily available, such as in the work of (Indigenous Languages Research and Development Foundation, 2020). For our synthetic corpus, we chose the parallel sentences provided by the dictionary because it contained conversational sentences, such as those common in classroom instruction and family settings. For the purposes of creating a parallel corpus, having Chinese as one of the languages allows effective use of other MT tools, such as Google Translate to translate the Chinese translation of Yami sentences to Tagalog. The tool is comparatively an improvement in terms translation quality over the local model of NLLB-200. Examples of NLLB-200's poor Chinese-to-Tagalog translation quality are shown in Table 2.

4.5 Improving Yami-to-Tagalog translation using Synthetic Data

Initially, only the Bible parallel corpus trained our first Yami-to-Tagalog model (Model 1). To further improve the initial model, we supplement the training data from the Bible parallel corpus with the synthetic data Yami-Tagalog parallel corpus created from the Yami-Chinese parallel corpus sourced from Indigenous Languages Research and Development Foundation (2020). With this new training, NLLB-200 model trained trained using the same initial hyperparameters to set comparison with the previous model against Model 2.

Table 2: Comparison of Yami Translations

Yami	Chinese	English	Tagalog (NLLB-200 600M)	Tagalog (Google Translate)
kokey kamo papoen tana	大家好。我始了。	Hello everyone. Here we go.	Bigyang mabuti... tayo ay nagsisimula...	Hello sa lahat. Magsimula tayo.
tanek kamo	起立! 敬!	Please stand up! Salute!	起立! 敬礼!	Mangyaring tumayo! Salute!
akokey a sinsi namen	老好。	Hello.	Teacher mabuti.	Hello, guro.
lisna kamo	坐下。	Please sit down.	坐下。	Mangyaring umupo.

However, considering that the synthetic data, being only 3966 in sentence count, is outnumbered by the real data (10498). To prevent the Bible data from dominating the conversational and cultural data in the synthetic parallel corpus, we duplicated the synthetic data three times, i.e. 3966 original sentences $\times$ 3 = 11898 total synthetic sentences. We chose the $\times$ 3 multiplier because it roughly equalize the count between the real and synthetic data. Moreover, further shuffling of the synthetic data and the Bible verses was done (Xu et al., 2019) to prevent catastrophic interference caused by feeding one data source followed by a new and separate data source (van de Ven et al., 2024).

4.6 Improving Yami-to-Tagalog translation using a Pretrained Model

Additionally, we performed pretraining of the base NLLB-200 model on Ivatan-to-Tagalog because Ivatan is another Batanic language. Although the NLB-200 model has training on some Philippine languages, it does not have any training on a Batanic language. By training the initial model on Ivatan-To-Tagalog, the model will have some initial representation of a closely related Batanic language before being introduced to Yami, which may improve the translation quality of the final model (Model 3). After training the model on Ivatan, we produce another fine-tuning it towards Yami using the Yami-to-Tagalog Bible parallel corpus and the synthetic Yami-Tagalog corpus.

5 Analysis of Performance

Table 3 presents the quantitative results across our three experimental configurations. We assess performance using BLEU, chrF++, and SBERT metrics on two distinct test sets: the in-domain *Bible Test Set* (200 pairs) and the out-of-domain *Validation Set* (20 pairs), which consists of conversational and daily-life vocabulary Yami text.

The Effectiveness of Language Transfer (Model 3): Our results provide strong empirical evidence for the utility of phylogenetic transfer in ultra-low-resource settings. Model 3, which utilizes pre-training on the closely related Ivatan language before fine-tuning on Yami, achieved the highest performance on the in-domain Bible dataset, securing a **BLEU score of 34.85** and a **chrF score of 58.36**. This represents a marked improvement over the baseline Model 1, which achieved a BLEU score of 32.00. We attribute this gain to the specific genealogical relationship between Ivatan and Yami, both of which belong to the Batanic branch of the Malayo-Polynesian family. Unlike general multilingual pre-training, initializing weights with Ivatan allows the model to "warm start" with a high degree of relevant morphosyntactic alignment. The significantly higher chrF score (58.36 vs 56.26) is particularly telling; since chrF operates at the character n-gram level, it suggests that Model 3 is far more successful at generating the correct agglutinative affixes and reduplication patterns shared by Batanic languages, even when exact lexical matches are unavailable. This confirms that in the absence of massive parallel corpora, exploiting the structural priors of a sister language is a potent strategy for stabilizing the decoder.

Generalization vs. Domain Specialization (Model 2): The performance of Model 2, which augments the training set with synthetic dictionary-based data, illustrates a critical trade-off between domain specialization and robust generalization. On the specific Bible test set, Model 2 experienced a slight degradation in performance, with the BLEU score dropping to **30.97** compared to the baseline's 32.00. This dip is attributable to the dilution of the model's probability distribution; by introducing conversational data, the model is

Table 3: Performance Comparison: Initial vs. Fine-Tuned vs. Pretrained (Ivv) Models

	Model 1		Model 2		Model 3	
Metric	**Bible**	**Val.**	**Bible**	**Val.**	**Bible**	**Validation**
BLEU	32.00	3.12	30.97	6.67	34.85	3.64
chrF	56.26	28.28	55.16	34.01	58.36	32.29
Median SBERT	92.26	82.77	92.55	79.80	90.54	77.94

less over fitted to the specific idiolect and archaic sentence structures of the religious text. However, this minor trade-off in the source domain yielded disproportionate gains in generalization. On the Validation Dataset—which mimics real-world usage—Model 2 effectively doubled the translation quality, achieving a **BLEU score of 6.61** compared to the baseline's 3.12. Furthermore, it achieved the highest chrF score on the validation set (**34.01**) among all models. This indicates that the inclusion of synthetic conversational pairs successfully mitigates the bias, allowing the model to recover basic conversational structures (e.g., greetings, imperatives) that are statistically rare in the corpus but remain essential for a functional translation system.

Semantic Divergence and Formality Bias (SBERT Analysis): The SBERT (Sentence-BERT) semantic similarity scores reveal a nuanced limitation of pure transfer learning regarding register and formality. While Model 3 was the strongest structural translator (highest BLEU), it recorded the *lowest* semantic similarity score on the conversational Validation set (**SBERT 77.94**), significantly lower than Model 1 (82.77). We postulate that this is driven by a "Formality Bias." Both the Ivatan pre-training data and the Yami Bible data consist of high-register, formal, and often archaic language. Consequently, when presented with a casual input from the Validation set, Model 3 is predisposed to generate a formally rigid or archaic Tagalog equivalent. While these translations may be grammatically sound, they diverge semantically from the modern, casual Tagalog references used in the validation set, resulting in lower embedding similarity. In contrast, Model 2, which was exposed to synthetic conversational pairs sourced from dictionary examples, maintained a higher SBERT score, suggesting that dictionary-based augmentation acts as a necessary "bridge" to modern semantics, preventing the model from becoming locked into an exclusively liturgical register.

6 Conclusion

This study presented a systematic investigation into bootstrapping Neural Machine Translation for Tao (Yami), an endangered ultra-low-resource language, by leveraging the NLLB-200 architecture. Our findings demonstrate that in settings where parallel data is fewer than 5,000 sentences, relying solely on data quantity is insufficient; instead, exploiting linguistic structures and rich lexical resources is key in translating low-resource languages. We establish that phylogenetic transfer learning—specifically pre-training on the closely related Ivatan language—is the most effective strategy for maximizing translation fidelity. This approach yielded the highest in-domain performance (BLEU 34.85), confirming that shared morphological and syntactic features between Batanic languages can be effectively leveraged to stabilize the decoder. However, our analysis also revealed a critical "liturgical bias" in models trained exclusively on religious texts. To address this, we demonstrate the effectiveness of dictionary-assisted synthetic data augmentation. While this approach incurred a minor trade-off in in-domain precision, it significantly enhances the model's generalization capabilities, improving the BLEU score on conversational out-of-domain data. This suggests a complementary framework for future work: utilizing transfer learning to establish the grammatical blueprint of the language, while employing synthetic augmentation to expand the semantic system required for modern communication. Ultimately, this work provides a reproducible blueprint for other underrepresented ultra-low-resource Austronesian languages. The study highlights that even in the absence of large-scale corpora, functional translation systems can be constructed by intelligently combining digitized linguistic heritage—such as grammars and dictionaries—with pre-trained SOTA multilingual transfer learning. Such efforts are essential for fostering an inclusive technological landscape the meets the needs of marginalized languagess.

7 Recommendations

Based on the on the analysis of model performance and the limitations observed during evaluation and configuration for the training, the following recommendations are proposed for future system improvements and further research. While basic text normalization was used, such as lowercasing and character filtering, was effectively used in this study. Implementing a more advanced normalization, like true-casing, punctuation utilization, and Unicode aware normalization, may be able to improve the model's performance. Utilizing better hardware will also enable future researchers to configure the training process to be more optimized by using larger batch sizes, training for more epochs, or experimenting with the latest and larger token models. In addition, future work can look into using a more sophisticated tokenization approaches, like sub-word modeling, to improve the handling of rare words and terms, and to also handle the rich morphological structure of the Tao language. Lastly, we recommended that future ultra-low-resource research involves collaborating with native speakers and language experts to verify the quality of translation of the model and to help promote research interest on the indigenous languages of the Philippines.

Limitations

Although Bible-based corpus is a widely available source in various languages in the Philippines, the different Bible translations, due to translators having varying source-versions, interpretations, faithfulness to the source text, and target use cases for their translations, may differ significantly in wording and structure from one language to another. Additionally, Bible verses are not as effective representations of the typical sentence structure or vocabulary of a language.

Moreover, our paper only focuses on one ultra-low-resource Austronesian language in the Yami language of Orchid Island. The machine translation findings presented in this paper are not representative of all Austronesian low-resource languages, nor are these results representative of low-resource languages globally. Likewise, these results and findings are not representative of the effectiveness of the various machine translation improvement techniques presented in our paper, given that we only focused on one source-to-target language pair, namely Yami to Tagalog. Future researchers will benefit from incorporating more low-resource languages in their works to better understand the effectiveness of pre-training using related languages, augmentation using synthetic data, and other techniques not implemented in this paper, to improve low-resource machine translation.

The Yami-Tagalog parallel corpus presented in this paper can not be used for validation of machine translation quality because it was created synthetically from a Yami-Chinese parallel corpus. For this task, human translations are still preferred because of the various biases and inaccuracies of current machine translation models. However, this parallel corpus can be used to augment real Yami-Tagalog parallel data, such as the Bible parallel corpus, for fine-tuning or training machine translation models, given the scarcity of quality parallel corpora incorporating the Yami language.

The neural machine translation model used in this paper, namely Meta's No Language Left Behind model (600 million parameters) was initially selected for VRAM, memory and speed considerations. In lieu, models with more parameters may likely produce higher quality translations than the specific model used in this paper. Additionally, we were limited to a limit of 30 hours of GPU accelerator usage on Kaggle, which further underscored our choice of model. Future researchers will benefit from using improved hardware, latest models, or more efficient fine-tuning techniques for their neural machine translation models. Furthermore, we also used the default tokenizer of NLBB-200 in consideration towards the limited time frame of our study. Lastly, our evaluation of machine translation quality did not incorporate human evaluation. Although the metrics we used to measure translation quality allow for the models to be compared to other machine translation models, these metrics do not capture translations of figure speech and ambiguous sentences. Additionally, these metrics also fail to detect biases in translations (e.g., unexpected gender bias when translating a sentence without gendered pronouns).

Acknowledgments

We thank Nathaniel Oco for introducing and educating us on NLP and NMT. We further acknowledge the use of Gemini to assist with the LaTeX formatting of the technical manuscript. All scientific claims, experimental results, and manuscript text were done and verified by the authors.

Author Contributions

Adrian Denzel Macayan designed the transfer learning methodology, conducted the formal analysis and review of related literature, and contributed to both the original draft and the final review and editing.
Luis Andrew Madridijo led the conceptualization and methodology design, developed the software for model training, and contributed to the writing of the original draft.
Ellexandrei Esponilla contributed to the the data curation and pre-processing of the Bible corpus, performed data validation, and contributed to the writing of the original draft.
Zachary Mitchell Francisco was responsible for the curation and development of synthetic data and contributed to the writing of the original draft.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Mahmoud Ahmad, Auwal Khalid, Lukman Aliyu, Babangida Sani, and Mariya Abdullahi. 2024. Arewa NLP's participation at WMT24. In *Proceedings of the Ninth Conference on Machine Translation*, pages 829–832, Miami, Florida, USA. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA.

Filipe Cardoso, Rui Silva, and Luis Gomes. 2022. Improving neural MT of indigenous languages with multilingual transfer learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 299–307, Dublin, Ireland. Association for Computational Linguistics.

Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846.

Víctor M. García, Jeroni Suárez, Marta R. Costa-jussà, and José A. R. Fonollosa. 2019. Context-aware monolingual repair for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 389–395, Florence, Italy. Association for Computational Linguistics.

Jiatao Gu, Yong Wang, Yun Chen, Kyunghyun Cho, and Victor O.K. Li. 2018. Meta-learning for low-resource neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3622–3631, Brussels, Belgium. Association for Computational Linguistics.

Barry Haddow, Rachel Bawden, Antonio Valerio Miceli Barone, Jindřich Helcl, and Alexandra Birch. 2022. Survey of low-resource machine translation. *Computational Linguistics*, 48(3):673–732.

Hugging Face. 2025. Hugging face model hub: Models (tagalog). Accessed: 2025-12-05.

Indigenous Languages Research and Development Foundation. 2020. e (indigenous language e-paradise). Accessed: 2025-12-05.

Kaiying Kevin Lin, Hsiyu Chen, and Haopeng Zhang. 2025. FormosanBench: Benchmarking low-resource Austronesian languages in the era of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 16527–16539, Online. Association for Computational Linguistics.

Holy Lovenia, Rahmad Mahendra, Salsabil Maulana Akbar, Lester James V. Miranda, Jennifer Santoso, Elyanah Aco, Akhdan Fadhilah, Jonibek Mansurov, Joseph Marvin Imperial, Onno P. Kampman, Joel Ruben Antony Moniz, Muhammad Ravi Shulthan Habibi, Frederikus Hudi, Railey Montalan, Ryan Ignatius, and 1 others. 2024. SEACrowd: A multilingual multimodal data hub and benchmark suite for Southeast Asian languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5155–5203, Miami, Florida, USA. Association for Computational Linguistics.

Toan Nguyen and David Chiang. 2017. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Stephan Oepen, Nikolay Arefyev, Ona de Gibert, Andrey Kutuzov, Sampo Pyysalo, Jörg Tiedemann, and 1 others. 2025. HPLT 3.0: Very large-scale multilingual resources for LLM and MT. *Preprint*, arXiv:2511.01066. arXiv preprint arXiv:2511.01066.

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*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Surangika Ranathunga, Shravan Nayak, Shih-Ting Cindy Huang, Yanke Mao, Tong Su, Yun-Hsiang Ray Chan, Songchen Yuan, Anthony Rinaldi, and Annie En-Shiun Lee. 2024. Exploiting domain-specific parallel data on multilingual language models for low-resource language translation. *Preprint*, arXiv:2412.19522. arXiv preprint arXiv:2412.19522.

D. Victoria Rau and Maa-Neu Dong. 2006. *Yami Texts with Reference Grammar and Dictionary*. Institute of Linguistics, Academia Sinica, Taipei.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Ana Rodríguez and 1 others. 2023. Train global, tailor local: Minimalist multilingual translation into endangered languages. In *Proceedings of the 6th Workshop on Technologies for MT of Low Resource Languages (LoResMT 2023)*. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Alexander D Smith. 2017. The western malayo-polynesian problem. *Oceanic Linguistics*, 56(2):435–490.

Gido M van de Ven, Nicholas Soures, and Dhireesha Kudithipudi. 2024. Continual learning and catastrophic forgetting. *arXiv preprint arXiv:2403.05175*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 5998–6008.

Nuo Xu, Yinqiao Li, Chen Xu, Yanyang Li, Bei Li, Tong Xiao, and Jingbo Zhu. 2019. Analysis of back-translation methods for low-resource neural machine translation. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 466–475. Springer.

Meng-Chien Yang, Si-Wei Huang, and D. Victoria Rau. 2011. Two ontological approaches to building an integrated semantic network for yami *ka-* verbs. In *2011 International Conference on Asian Language Processing (IALP)*, pages 1–4. IEEE.

Meng-Chien Yang, D. Victoria Rau, and Ann Hui-Huan Chang. 2010. A proposed model for constructing a yami wordnet. In *2010 International Conference on Asian Language Processing (IALP)*, pages 1–4. IEEE.

Meng-Chien Yang, D. Victoria Rau, and Yi-Hsin Wu. 2012. Analyzing and classifying the yami emotion phrases using ontological structure and computation. In *2012 International Conference on Asian Language Processing (IALP)*, pages 45–48. IEEE.

Ling Zhang and 1 others. 2023. Bilex Rx: Lexical data augmentation for massively multilingual Machine Translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15745–15763, Toronto, Canada. Association for Computational Linguistics.

Barret Zoph and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

Text Filter Based on Automatically Acquired Vocabularies for Multilingual Machine Translation

Kenji Imamura and Masao Utiyama
National Institute of Information and Communications Technology
Hikari-dai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
{kenji.imamura, mutiyama}@nict.go.jp

Abstract

In this paper, we propose a text filter designed to support multiple languages. The method simply aggregates vocabulary from a monolingual corpus and compares it against the input. Despite its simplicity, the approach proves highly effective in removing code-mixed text. When combined with existing language identification techniques, our method can enhance the purity of the corpus in the target language. Consequently, applying it to parallel corpora for machine translation has the potential to improve translation quality. Additionally, the proposed method supports the incremental addition of new languages without the need to retrain those already learned. This feature easily enables our method to be applied to low-resource languages.

1 Introduction

Multilingual models are becoming increasingly common because neural models can process multiple languages using a single model (Johnson et al., 2017). For example, encoder-based models include multilingual BERT (mBERT; Devlin et al., 2019) and XLM-RoBERTa (XLM-R; Conneau et al., 2020), while encoder-decoder models feature multilingual BART (including mBART-50; Liu et al., 2020; Tang et al., 2020). In addition, decoder-only models, which are commonly referred to as large language models (LLMs), such as Llama 3, Qwen, and GPT-oss also support multiple languages.

However, the multilingual corpora used to train these models are inherently noisy, as they are often acquired automatically from web sources. For instance, Briakou et al. (2023) reported that even in well-cleaned monolingual training data for LLMs, approximately 1.4% contained a mixture of other languages. To remove such undesirable data, filtering is essential. However, corpora often include languages that even the dataset creators cannot comprehend, and automatic processing is necessary.

Language identification is an effective approach for excluding non-target languages from corpora (see Section 2.2 for details). However, conventional methods often fail to remove texts that contain segments of other languages, which are referred to as code-mixed texts in this paper, when the proportion of those languages is relatively small.

In this paper, we propose a monolingual filtering method designed to support multiple languages. The proposed approach performs token-level identification in a straightforward manner by matching against automatically acquired vocabulary. We apply this method to parallel corpora that include alignment scores or have been pre-filtered using alternative techniques, and demonstrate that it improves machine translation quality.

Our proposed method is particularly effective in removing code-mixed texts. By using corpora with fewer code-mixed instances for model training, systems can generate more consistent language, thereby improving machine translation quality.

Furthermore, new languages can be added incrementally, as the approach only requires tokenizing monolingual corpora of the target language and aggregating tokens to construct the vocabulary. This eliminates the need to retrain previously learned languages. For low-resource languages, an independent filtering method is particularly valuable, since major language identifiers may not reliably support them.

The remainder of this paper is organized as follows. Section 2 introduces related work, including multilingual corpus filtering and language identification. Section 3 provides a detailed description of the proposed method. Section 4 presents experiments on parallel corpus filtering and machine translation evaluation to validate the effectiveness of the proposed approach. Finally, Section 5 concludes the paper.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 37–48
March 28, 2026

2 Related Work

2.1 Multilingual Corpus Filtering

Many multilingual corpora are collected from the web. Data obtained from sources other than Wikipedia are typically filtered using language identification and other techniques. In this section, we first summarize methods for filtering multilingual corpora.

2.1.1 Collection of Monolingual Corpora

CC-100 is a collection of monolingual corpora covering 100 languages (Conneau et al., 2020). It was constructed following the procedure used for building the CCNet corpus (Wenzek et al., 2020). The procedure can be summarized as follows:

1. The collected web pages are deduplicated at the paragraph level.

2. A fastText-based language identifier (Joulin et al., 2016; Bojanowski et al., 2017) is used to detect the language of each page, and pages with low identification scores are removed.

3. For 48 high-resource languages, language models are trained for each language to compute paragraph-level perplexity (PPL). Paragraphs with high PPL values are discarded, with thresholds determined individually for each language based on its PPL distribution.

To summarize, filtering is performed based on language identification scores at the page level and perplexity derived from language models at the paragraph level. Because both scores reflect the overall unit, texts containing code-mixing are not removed if the proportion of other languages is minimal.

2.1.2 Parallel Corpora

CCMatrix (Schwenk et al., 2021b) and WikiMatrix (Schwenk et al., 2021a), both multilingual parallel corpora, were constructed following nearly identical procedures.

1. Monolingual filtering was carried out in the following steps: 1) paragraphs were extracted from the CCNet corpus, 2) sentences were segmented, 3) duplicate sentences were removed, and 4) languages were identified using fastText (Grave et al., 2018) and langid.py (Lui and Baldwin, 2012).

2. Next, sentence alignment was carried out to build the parallel corpus. In CCMatrix, sentence embeddings were generated with the LASER model (Artetxe and Schwenk, 2019), and cross-lingual matches were identified using FAISS index search (Douze et al., 2025).

The NLLB corpus (NLLB Team et al., 2022) was constructed in a manner similar to CCMatrix. Monolingual filtering involved 1) language identification, 2) sentence segmentation, and 3) deduplication. Sentence alignment was then performed using scores derived from the LASER 3 model. In addition, heuristic filters based on sentence length and the proportion of symbols or numbers were applied.

While sentence alignment plays a key role in constructing parallel corpora, language identification is central to processing monolingual data. In this paper, we focus on monolingual filtering.

2.1.3 Parallel Corpus Filtering / Data Curation Task in WMT

The Conference on Machine Translation (WMT) organized shared tasks on parallel corpus filtering from 2018 to 2020 (Koehn et al., 2018, 2019, 2020). In these tasks, participants filtered noisy parallel corpora provided by the organizers and trained Transformer-based encoder-decoder models (Vaswani et al., 2023). Translation quality was then evaluated by the organizers. The target languages varied by year: in 2018, the focus was on a high-resource pair, German-English (De-En), while in 2019 and 2020, the focus shifted to low-resource pairs such as Nepali-English (Ne-En), Sinhala-English (Si-En), Pashto-English (Ps-En), and Khmer-English (Km-En).

The approaches adopted by participants in the 2020 shared task can be summarized as follows (Koehn et al., 2020): For monolingual processing, filtering involved 1) pre-filtering based on sentence length and character types, 2) language identification, and 3) language model scoring. For parallel processing, filtering relied on 4) LASER scores, 5) bidirectional cross-entropy between source and target languages, and 6) word-level translation scores.

In the WMT-2023 parallel data curation task (Sloto et al., 2023), Steingrimsson (2023) employed three language identification tools and discarded sentences where fewer than two out of three agreed.

To summarize, language identification and language model scores are central to monolingual fil-

Unit Type	Name	#Langs.	Description	Note
Sentence / Text	fastText[1]	176	Multiclass classifier based on skip-gram (Mikolov et al., 2013).	Grave et al. (2018); Joulin et al. (2016); Bojanowski et al. (2017)
	langid.py[2]	97	Naïve Bayes classifier based on a deterministic finite automaton	Lui and Baldwin (2012)
Token / Character	CMX	100	Multiclass classifier based on a feed-forward neural network; the language is identified at the token level	Zhang et al. (2018)
	CLD3[3]	107	Multiclass classifier based on a feed-forward neural network that averages character n-gram inputs	Google Chrome browser plugin.
	LanideNN	131	Bidirectional RNN classifier that identifies the language character by character from character embeddings	Kocmi and Bojar (2017)
	Equilid[4]	70	Three-layer neural network that identifies the language token by token	Jurgens et al. (2017)

Table 1: Summary of language identification.

tering. In practice, several language identifiers can be used in combination.

2.2 Language Identification

As noted in the previous section, language identification plays a central role in monolingual filtering.

Table 1 provides an overview of language identification methods. Almost all of the proposed approaches are learning-based and do not rely on manually crafted rules or dictionaries. These methods can be broadly categorized into two types: 1) sentence- or text-level identification and 2) token- or character-level identification.[5]

When sentence- or text-level language identifiers are used, code-mixed texts may be misclassified as the target language if the proportion of mixed content is small. In contrast, token-level identifiers determine the language at the token level, enabling filtering based on aggregated token-level results (Zhang et al., 2018). Since our proposed method adopts token-based identification, it is effective for handling code-mixed texts.

[1] https://fasttext.cc/docs/en/language-identification.html

[2] https://github.com/saffsd/langid.py

[3] https://github.com/google/cld3

[4] https://github.com/davidjurgens/equilid

[5] Unfortunately, we were unable to validate the performance of the token-level identifiers, as they were either difficult to obtain or no longer functional due to outdated implementations.

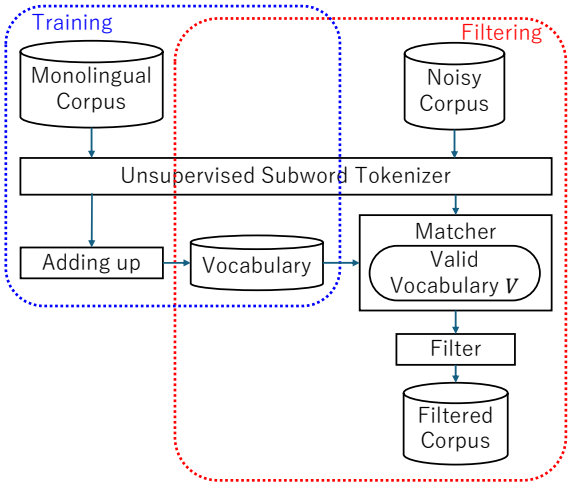


Figure 1: Structure of the proposed method.

3 Proposed Method

The proposed approach employs a binary classifier for each language. It automatically extracts vocabulary from a monolingual corpus, performs token-level identification by matching input tokens to the vocabulary, and applies filtering to remove invalid sentences. The overall architecture is illustrated in Figure 1. Training and filtering follow the steps outlined below.

Training

1. The monolingual corpus is tokenized into subwords. Although it is preferable for the corpus to contain only the target language, a certain

level of noise is acceptable, as discussed later. Therefore, web texts automatically collected using other language identifiers can be utilized.

For subword tokenization, we employ SentencePiece (Kudo and Richardson, 2018), an unsupervised tokenizer trained on the corpus.

2. Subword tokens are collected, sorted by frequency, and stored as a vocabulary.

Filtering

3. Load the vocabulary constructed in Step 2. To ensure coverage within the vocabulary limit VL, retain only the most frequent subwords and define them as the valid vocabulary V. Low-frequency subwords are excluded, as they typically represent noise.

4. The noisy corpus to be filtered is tokenized into subwords using the same tokenizer as in Step 1.

5. Match tokens against the valid vocabulary.

6. Sentences whose valid token ratio, the proportion of tokens contained in the valid vocabulary, is below the threshold TR are discarded as noise, while the remaining sentences are retained.

   Specifically, sentences satisfying the condition of the filter function $\mathrm{vocabFilter}(W)$ are output as the filtering result:

$$\begin{aligned} \mathrm{vocabFilter}(W) &= \frac{\sum_i \mathrm{match}(w_i)}{|W|} \\ &\geq TR, \\ \mathrm{match}(w) &= \begin{cases} 1 & \text{if } w \in V \\ 0 & \text{else} \end{cases}, \end{aligned}$$

   where W and w_i denote the sentence to be identified and the i-th token in W, respectively.

The proposed method has the following characteristics.

- In the overall framework, the proposed method effectively filters code-mixed texts by aggregating token-level identifications. This can be achieved by setting the threshold TR to a relatively high value.
- The proposed approach employs a language-specific binary classifier. In contrast to multi-class classifiers, which may require retraining on all languages when adding a new one, our method only constructs a binary classifier for the target language. This enables incremental addition of languages without reprocessing existing ones.
- Regarding the valid vocabulary, even if the monolingual corpus used for vocabulary acquisition contains noise, language identification remains feasible because function words dominate the top of the vocabulary hierarchy, content words occupy the middle, and noise tends to appear at the bottom (Table 2).
- The proposed method resembles filtering based on unigram language models. However, language models often assign low probabilities to content words, even in grammatically correct sentences. In contrast, our approach benefits from subword-level identification, allowing recognition of content words even when the full word is absent from the valid vocabulary.

Table 2 presents examples of valid vocabularies for German and Pashto obtained from the experiments described in the next section (English and Khmer vocabularies are provided in Appendix A). Although the total vocabulary size differs substantially across languages, two common patterns emerge: 1) symbols and function words, which constitute a large portion of each language, dominate the upper ranks; and 2) the lowest ranks consist almost entirely of noise, often including tokens from other languages. Consequently, removing the lower-ranked subwords yields the valid vocabulary for each language.

The vocabulary limit (VL) is defined as the threshold that separates valid subwords from invalid ones. In this study, VL was determined based on the cumulative coverage ratio, set to 99.5%. The resulting valid vocabulary sizes are 20,927 for English, 21,342 for German, 6,210 for Pashto, and 12,602 for Khmer. Subwords near this threshold often correspond to fragments of content words, making their classification inherently uncertain. To address this, we relax the token ratio threshold (TR) during the final filtering step, allowing for minor inconsistencies. In this work, TR is set to 0.9.

Order	Subword	Frequency	Cumulo-coverage
1	.	597M	3.71%
2	,	566M	7.23%
3	▁und	276M	8.95%
4	▁die	245M	10.47%
5	en	243M	11.98%
:			
21340	Wikipedia	13,937	99.49%
21341	teis	13,936	99.49%
21342	▁oxid	13,936	99.49%
21343	▁Malo	13,935	99.50%
21344	408	13,935	99.50%
21345	ARS	13,934	99.50%
:			
110927	Ấ	1	100.00%
110928	높	1	100.00%
110929	받	1	100.00%

(a) German vocabulary

Valid Vocabulary

Order	Subword	Frequency	Cumulo-coverage
1	د▁	6,789K	6.51%
2	په▁	3,282K	9.66%
3	او▁	2,198K	11.77%
4	.	2,131K	13.81%
5	کې▁	1,767K	15.50%
:			
6208	▁زهرا	266	99.49%
6209	کے	265	99.49%
6210	ئل	265	99.49%
6211	▁گرا	265	99.50%
6212	▁تنګ	265	99.50%
6213	▁فرشت	265	99.50%
:			
33473	▁	1	100.00%
33474	ټ	1	100.00%
33475	ã	1	100.00%

(b) Pashto vocabulary

Table 2: Examples of German and Pashto vocabularies obtained from the experiments in Section 4. Subwords contributing to a cumulative coverage of 99.5% were retained as the valid vocabulary.

4 Experiments

In this study, we refer to the proposed approach as 'vocabFilter'. Its effectiveness is evaluated based on machine translation quality using a filtered corpus.

4.1 Experimental Settings

4.1.1 vocabFilter Settings

We used CC-100 (Conneau et al., 2020) as the monolingual corpus for vocabulary acquisition.

We employed SentencePiece (Kudo and Richardson, 2018) as the tokenizer. The tokenizer model used in this study is the same as that adopted in the pretrained models mBART (Liu et al., 2020) and XLM-R (Conneau et al., 2020). This is the Unigram model of SentencePiece, which supports 100 languages and contains a vocabulary of approximately 250K subwords. Imamura and Utiyama (2024) reported that this model yields a low rate of unknown (UNK) tokens.

We set the hyperparameters to $VL = 0.995$ and $TR = 0.9$.

4.1.2 Parallel Corpus Filtering Task

We evaluated our approach following the parallel corpus filtering task of WMT. For this task, the organizers provided a noisy bilingual corpus together with sentence-level alignment scores (Table 3).

Lang. Pair	Noisy Corpus #Sents.	Noisy Corpus #Tokens	Selected Corpus #Sents.	Selected Corpus #Tokens
De-En	104M	1.0B	-	100M
Ps-En	1.02M	11M	-	5.0M
Km-En	4.17M	58M	-	5.0M

Table 3: Parallel corpora used in the experiment.

- In 2018, the target language pair was German-English, which is considered high-resource.
- In 2020, the targets were Pashto-English and Khmer-English, both regarded as low-resource pairs.
- We excluded the 2019 tasks from our evaluation because sentence alignment scores were not provided, and the objective of this study is monolingual corpus filtering.

The evaluation was carried out using the following procedure.

1. We filtered the noisy parallel corpus using the proposed method and comparative approaches.

2. After sorting the filtered results by alignment score, we selected the top sentences to reach a fixed number of tokens, counted on the English side. The thresholds were 100 million tokens for German and 5 million tokens for Pashto and Khmer (Table 3). Thus, we assumed that the amount of information in the parallel corpus remained constant regardless of the filtering method.

3. We trained the translation model using FairSeq (Ott et al., 2019), following the WMT shared task setup. Details of the hyperparameters are provided in Appendix B.

4. Finally, we evaluated translation quality on the test sets provided by WMT. For German, we used the devtest set, while for Pashto and Khmer, we combined the devtest and test sets. Translation quality was measured using sacreBLEU (Post, 2018) with the tokenizer for Flores-200 (NLLB Team et al., 2022; Goyal et al., 2022). BLEU was chosen because accurate surface translation was important in this study.

4.1.3 Comparative Methods

Following the baseline of the WMT-2020 shared task, we adopted fastText filtering as the baseline and combined it with the proposed method. Filtering was applied separately to the source and target languages.

4.2 Results

4.2.1 Translation Quality

Table 4 presents the BLEU scores obtained when each filter was applied to the source and target languages.

The effect of the proposed method, vocabFilter, varied across languages. For German (De $\leftrightarrow$ En), vocabFilter alone yielded lower translation quality than fastText alone; however, combining both methods improved the BLEU score over the baseline (i.e., fastText only).

Among the low-resource languages, Pashto (Ps $\leftrightarrow$ En) showed a smaller effect compared to German. However, translation quality improved when vocabFilter was applied to the source side in addition to fastText.

By contrast, for Khmer, both vocabFilter alone and its combination with fastText improved BLEU scores over the baseline, except when all filters were applied in the En $\rightarrow$ Km direction.

To summarize, translation quality tended to improve when vocabFilter was used in combination with fastText.

4.2.2 Number of Sentences Filtered out and Remaining

Table 5 presents the number of parallel sentences when language identification filtering was applied to both the source and target sides. 'Filtered' denotes the total number of sentences remaining after filtering, and 'Selected' denotes the number of sentences after selecting a fixed number of tokens.

First, focusing on 'Selected', we observe that although the number of English tokens is fixed, German tends to favor shorter sentences, as vocabFilter retained more sentences than fastText. In contrast, Pashto and Khmer show a slight decrease in sentence count, indicating a preference for longer sentences.

Next, focusing on 'Filtered', we can estimate the number of sentences for which fastText and vocabFilter produced different identification results. For example, in German, the difference between applying both filters and applying only fastText was 5.8M sentences (31.3M $-$ 25.5M), representing sentences accepted only by fastText. Similarly, sentences accepted only by vocabFilter totaled 20.6M (46.1M $-$ 25.5M), meaning that about 25% of the entire noisy corpus yielded different identification outcomes.

These differences suggest that fastText and vocabFilter evaluate sentences from different perspectives. Therefore, using them together can enhance language purity.

4.2.3 Example Sentences Filtered by vocabFilter

Table 6 provides example sentences accepted by fastText but rejected by vocabFilter. The sentences were tokenized using SentencePiece, and tokens shown in red indicate those that failed to match the valid vocabulary. Token ratio refers to the proportion of valid tokens; sentences with a token ratio below 0.9 were filtered out.

Regardless of language, these examples show that most code-mixed texts accepted by fastText were appropriately filtered out by vocabFilter. However, numeric tokens often caused identification failures (cf. No. 4, 7, and 8) because long numbers were frequently absent from the valid vocabulary. In addition, some tokens did not match the vocabulary even in sentences written in the cor-

fastText		vocabFilter		XX→En			En→XX		
Source	Target	Source	Target	De→En	Ps→En	Km→En	En→De	En→Ps	En→Km
✓	✓			29.5	8.8	7.3	27.5	10.7	14.8
		✓	✓	27.7 (-)	8.6	**8.1** (+)	25.4 (-)	10.6	**15.1** (+)
✓	✓	✓		**30.9** (+)	**9.1** (+)	**8.1** (+)	**28.6** (+)	**11.0** (+)	**15.1** (+)
✓	✓		✓	30.2 (+)	8.9	**8.1** (+)	28.1 (+)	**11.0** (+)	14.9
✓	✓	✓	✓	30.8 (+)	8.8	7.7 (+)	28.4 (+)	10.7	14.2 (-)

Table 4: BLEU scores when language identification was applied to each language. Bold indicates the highest score for each translation direction, and underlining denotes the second highest. The (+) and (-) symbols represent significant improvements or degradations, respectively, compared with fastText only (first row of data), based on bootstrap resampling with sacreBLEU ($p < 0.05$).

fastText	vocabFilter	De ↔ En			Ps ↔ En			Km ↔ En		
(both side)	(both side)	Noisy	Filtered	Selected	Noisy	Filtered	Selected	Noisy	Filtered	Selected
✓			31.3M	8.7M		560K	226K		2.27M	241K
	✓	104M	46.1M	12.3M	1.02M	593K	214K	4.17M	2.67M	221K
✓	✓		25.5M	7.64M		415K	214K		1.92M	215K

Table 5: Number of parallel sentences before and after filtering for each language identification.

Language	No.	Tokenized and matched example sentence	Token ratio
English	1	__" Ma hl zeit ", __cho reo graph er , __Theater __am __Wall , __War endorf	0.875
	2	__ -__ ي __رخصت __ملي __Em ba ssy __of __Afghanistan __in __Ott awa	0.750
	3	__ FOL LOW __US __ON __SOCIAL !	0.857
German	4	__Kontakt __B Berlin 1 __2015 -08- 26 T 17 :00 :26 +00:00	0.833
	5	__T sche chi en , __Li šov	0.857
	6	__Homepage __ | __Druck en __ | __Nach __oben	0.778
Pashto	7	__ 446 #__ 5__ __وړاندې __وړځي __مياشت و __7__by __Dari us s ssss	0.692
	8	5- رح __ __امريکا __شمالي __C ____ __د __اور پک FSX __& __P 3 D __2.5	0.867
	9	__Chang zhou __Daily s ____ __مح صو الت __يام لر ني __د Co . ، __Ltd	0.800
Khmer	10	__ផ្នែក __Put z meister __(__25 __)	0.857
	11	__English , __У к р а ї н с ь к а , __Français , __Español ... __សួម __មើល __ផ្នែក __ខ ន __បន្ថែម __ទៀត	0.800
	12	__Ma un fac turer __ ស្បែក ជើង បុរស សុវត្ថិភាព អ្នក	0.800

Table 6: Example sentences accepted by fastText but rejected by vocabFilter. Tokens shown in red indicate those that failed to match the valid vocabulary.

rect target language (cf. No. 3 and 5).

Conversely, even in languages that do not typically use Latin script, such as “Co. Ltd” in Pashto (No. 9) or “English” in Khmer (No. 11), frequently occurring words are recognized as valid tokens. Because vocabFilter does not rely solely on character-based decisions, high-frequency foreign words are not subject to filtering.

Although the proposed method could not perfectly filter all sentences, it offers a significant advantage by automatically removing code-mixed texts.

4.3 Influence of Hyperparameters

The proposed method involves two hyperparameters: 1) VL, the threshold for valid vocabulary, and 2) TR, the token ratio used to identify sentences. In this section, we examine how changes in these parameters affect translation quality.

Specifically, we measured translation quality by varying $VL \in \{0.95, 0.99, 0.995, 1.0\}$ and $TR \in \{0.8, 0.9, 0.95\}$. The results are presented in Table 7. This table reports the average BLEU scores across three language directions (De↔En, Ps↔En, and Km↔En) to illustrate overall trends. Detailed

TR \VL	0.95	0.99	0.995	1.0
0.8	**16.3**	15.9	15.8	15.5
0.9	15.1	<u>16.1</u>	15.8	15.5
0.95	12.5	<u>16.1</u>	16.0	15.5

(a) Average of XX → En.

TR \VL	0.95	0.99	0.995	1.0
0.8	**18.2**	**18.2**	18.0	17.8
0.9	17.4	18.1	18.1	17.6
0.95	12.5	17.9	17.9	17.8

(b) Average of En → XX.

Table 7: BLEU scores of vocabFilter under various settings of the hyperparameters VL and TR. Bold indicates the highest score, and underline indicates the second highest.

results for each language are provided in Appendix C.

Scores were noticeably lower for $VL = 1.0$ and for $TR \in \{0.9, 0.95\}$ when $VL = 0.95$. The remaining scores were similar, indicating that the proposed method performs well unless extreme hyperparameter settings are used.

5 Conclusion

In this paper, we proposed a simple monolingual filtering method. The method is a binary classifier that matches input tokens against an automatically acquired vocabulary. Because it operates at the subword level, it can handle unknown words relatively well. Adding a new language requires only tokenizing and aggregating a monolingual corpus, eliminating the need to retrain existing language models and enabling incremental language expansion.

We applied the proposed method to the filtering of noisy corpora and demonstrated improvements in machine translation quality. The method was particularly effective in removing code-mixed texts. When combined with other language identification techniques, it enables the creation of a cleaner and more consistent corpus.

We plan to release the program, along with the acquired vocabulary, after expanding the supported languages.

Limitations

Training the proposed method requires a monolingual corpus for the target language, although, as noted in Section 3, some degree of noise is acceptable.

The proposed method aims to improve corpus purity by removing code-mixed texts. However, this introduces a trade-off, as it reduces linguistic variety, particularly with respect to vocabulary. For example, user-generated content on social media often contains useful code-mixed expressions, yet the proposed method attempts to filter these out as well. In our experiments, we adopted BLEU as the evaluation metric, emphasizing surface-level similarity; however, it is also necessary to evaluate the method from additional perspectives, such as robustness to diverse inputs.

Ethics Considerations

Since the vocabulary used in this approach is automatically extracted from monolingual corpora, it may include problematic subwords if the corpora contain erroneous or inappropriate texts. Moreover, the filtering process cannot remove texts with problematic content.

References

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Preprint*, arXiv:1607.04606.

Eleftheria Briakou, Colin Cherry, and George Foster. 2023. Searching for needles in a haystack: On the role of incidental bilingualism in PaLM's translation capability. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9432–9452, Toronto, Canada. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2025. The faiss library. *Preprint*, arXiv:2401.08281.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. *Preprint*, arXiv:1802.06893.

Kenji Imamura and Masao Utiyama. 2024. An empirical study of multilingual vocabulary for neural machine translation models. In *Proceedings of the Eleventh Workshop on Asian Translation (WAT 2024)*, pages 22–35, Miami, Florida, USA. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *Preprint*, arXiv:1607.01759.

David Jurgens, Yulia Tsvetkov, and Dan Jurafsky. 2017. Incorporating dialectal variability for socially equitable language identification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 51–57, Vancouver, Canada. Association for Computational Linguistics.

Tom Kocmi and Ondřej Bojar. 2017. LanideNN: Multilingual language identification on character window. *Preprint*, arXiv:1701.03338.

Philipp Koehn, Vishrav Chaudhary, Ahmed El-Kishky, Naman Goyal, Peng-Jen Chen, and Francisco Guzmán. 2020. Findings of the WMT 2020 shared task on parallel corpus filtering and alignment. In *Proceedings of the Fifth Conference on Machine Translation*, pages 726–742, Online. Association for Computational Linguistics.

Philipp Koehn, Francisco Guzmán, Vishrav Chaudhary, and Juan Pino. 2019. Findings of the WMT 2019 shared task on parallel corpus filtering for low-resource conditions. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 54–72, Florence, Italy. Association for Computational Linguistics.

Philipp Koehn, Huda Khayrallah, Kenneth Heafield, and Mikel L. Forcada. 2018. Findings of the WMT 2018 shared task on parallel corpus filtering. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 726–739, Belgium, Brussels. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30, Jeju Island, Korea. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on*

Machine Translation: Research Papers, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021a. 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: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021b. CCMatrix: Mining billions of high-quality parallel sentences on the web. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6490–6500, Online. Association for Computational Linguistics.

Steve Sloto, Brian Thompson, Huda Khayrallah, Tobias Domhan, Thamme Gowda, and Philipp Koehn. 2023. Findings of the WMT 2023 shared task on parallel data curation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 95–102, Singapore. Association for Computational Linguistics.

Steinthor Steingrimsson. 2023. A sentence alignment approach to document alignment and multi-faceted filtering for curating parallel sentence pairs from web-crawled data. In *Proceedings of the Eighth Conference on Machine Translation*, pages 366–374, Singapore. Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *Preprint*, arXiv:2008.00401.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need. *Preprint*, arXiv:1706.03762.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Yuan Zhang, Jason Riesa, Daniel Gillick, Anton Bakalov, Jason Baldridge, and David Weiss. 2018. A fast, compact, accurate model for language identification of codemixed text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 328–337, Brussels, Belgium. Association for Computational Linguistics.

A Examples of Vocabulary in English and Khmer

Table 8 presents a subset of the valid vocabularies for English and Khmer obtained in the experimental setup described in Section 4. The vocabularies for German and Pashto are shown in Table 2.

B Hyperparameters of Machine Translator during Filtering Experiment

In the filtering experiments described in Section 4.1.2, we trained translation models using the hyperparameters listed in Table 9.

C Details of Influence of Hyperparameters

Table 7 shows the change in average translation quality across all languages when hyperparameters VL and TR are varied. Table 10 provides the corresponding language-specific details.

Order	Subword	Frequency	Cumulo-coverage
1	.	2,733M	3.60%
2	,	2,537M	6.94%
3	▁the	2,393M	10.08%
4	s	1,989M	12.70%
5	▁to	1,676M	14.91%
:			
20925	▁Pune	71,375	99.49%
20926	▁235	71,339	99.49%
20927	▁Edwin	71,338	99.49%
20928	bare	71,316	99.50%
20929	bana	71,316	99.50%
20930	▁Nou	71,307	99.50%
:			
169752	봤	1	100.00%
169753	雰	1	100.00%
169754	踴	1	100.00%

(a) English vocabulary

Valid Vocabulary

Order	Subword	Frequency	Cumulo-coverage
1	▁	13,362K	9.72%
2	។	1,013,K	10.46%
3	▁km▁	1,000,K	11.19%
4	▁ន៊	934K	11.87%
5	▁ា	931K	12.55%
:			
12600	tone	99	99.49%
12601	▁Plu	99	99.49%
12602	▁Aja	99	99.49%
12603	DV	99	99.50%
12604	Ali	99	99.50%
12605	18)	99	99.50%
:			
54417	Ư	1	100.00%
54418	ƙ	1	100.00%
54419	왔	1	100.00%

(b) Khmer vocabulary

Table 8: Examples of valid vocabularies for English and Khmer. Subwords covering up to 99.5% cumulatively were considered part of the valid vocabulary.

Model structure	
Architecture	Transformer
# of layers	5
Embedding dimension	512
FFN inner dimension	2,048
Attention heads	2
Other model settings	Share all embeddings Normalize before
Training	
Dropout	0.4
Attention dropout	0.2
ReLU dropout	0.2
Loss function	Label smoothed cross-entropy
Label smoothing	$\epsilon = 0.2$
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.98$)
Learning rate	1e-3
LR scheduler	Inverse square root
Warm-up steps	4,000
Global batch size	Roughly 16,000 tokens
Training epochs	100
Translation	
Beam width	5
Length penalty	1.2

Table 9: Hyperparameters for Model Training and Translation

	De → En				Ps → En				Km → En			
TR \VL	0.95	0.99	0.995	1.0	0.95	0.99	0.995	1.0	0.95	0.99	0.995	1.0
0.8	31.5	30.6	30.4	29.7	8.9	8.8	9.0	9.0	8.4	8.3	8.1	7.7
0.9	30.8	31.4	30.8	29.7	8.2	8.5	8.5	9.1	6.2	8.3	8.1	7.7
0.95	26.1	31.2	31.2	29.7	7.5	8.6	8.6	9.1	3.9	8.4	8.2	7.6

(a) XX → En directions.

	En → De				En → Ps				En → Km			
TR \VL	0.95	0.99	0.995	1.0	0.95	0.99	0.995	1.0	0.95	0.99	0.995	1.0
0.8	29.5	28.4	27.9	27.6	9.8	10.8	11.3	10.9	15.3	15.3	14.9	14.9
0.9	29.1	29.1	28.5	27.4	8.9	10.1	10.8	10.6	14.1	15.2	15.0	14.9
0.95	26.2	29.3	29.1	27.7	8.9	10.1	10.8	10.6	4.5	15.3	15.5	14.9

(b) En → XX directions.

Table 10: Performance of vocabFilter under different settings of hyperparameters VL and TR.

Comparing LLM-Based Translation Approaches for Extremely Low-Resource Languages

Jared Coleman[1], **Ruben Rosales**[2], **Kira Toal**[2], **Diego Cuadros**[1]
Nicholas Leeds[1], **Bhaskar Krishnamachari**[3], **Khalil Iskarous**[3]
[1]Loyola Marymount University, [2]Independent Researcher, [3]University of Southern California
jared.coleman@lmu.edu, ruben.rosales@ieee.org, kira.toal@lmu.edu, dcuadros@lion.lmu.edu
nleeds@lion.lmu.edu, bkrishna@usc.edu, kiskarou@usc.edu

Abstract

We present a comprehensive evaluation and extension of the LLM-Assisted Rule-Based Machine Translation (LLM-RBMT) paradigm, an approach that combines the strengths of rule-based methods and Large Language Models (LLMs) to support translation in no-resource settings. We present a robust new implementation (the Pipeline Translator) that generalizes the LLM-RBMT approach and enables flexible adaptation to novel constructions. We benchmark it against four alternatives (Builder, Instructions, RAG, and Fine-tuned translators) on a curated dataset of 150 English sentences, and compare them across translation quality and runtime. The Pipeline Translator consistently achieves the best overall performance. The LLM-RBMT methods (Pipeline and Builder) also offer an important advantage: they naturally align with evaluation strategies that prioritize grammaticality and semantic fidelity over surface-form overlap, which is critical for endangered languages where mistranslation carries high risk.

1 Introduction

Recent advances in Large Language Models (LLMs) like OpenAI's GPT series have led to dramatic improvements in machine translation. However, these models still perform poorly on languages with little or no representation in their training data (Chowdhery et al., 2022; Robinson et al., 2023). This is a critical limitation, especially for endangered language revitalization efforts where reliable tools are urgently needed but parallel corpora are unavailable.

To address this, recent work introduced a new paradigm called **LLM-Assisted Rule-Based Machine Translation (LLM-RBMT)** (Coleman et al., 2024), which enables translation into a no-resource language using zero weight training or fine-tuning. Instead, it combines linguistic knowledge with LLM capabilities to guide sentence construction via grammar-aware tooling. This approach was successfully used to develop the first English-to-Owens Valley Paiute (OVP, a critically endangered Indigenous language) translator. Unlike well-resourced languages (and even many other low-resource settings) OVP has *no* publicly available parallel corpora and only a handful of fluent speakers. Figure 1 illustrates the LLM-RBMT pipeline: the English input is decomposed into simpler, translatable sentences using an LLM-based segmenting tool. Each simple sentence is translated into OVP using sentence-building tools, and then translated back into English to evaluate semantic fidelity. This process guarantees grammatical correctness while maintaining flexibility through LLM support.

In this work, we significantly expand the capabilities of the original LLM-RBMT system. We redesign the Pipeline Translator to support more complex sentence structures, introduce a more modular architecture, and improve transparency through structured validation. These enhancements enable broader grammatical coverage while preserving the core strengths of the LLM-RBMT paradigm. We also provide the first systematic evaluation of this framework by benchmarking the Pipeline translator against four alternative approaches: a more autonomous LLM-RBMT system (Builder), a prompt-only translator that relies solely on in-context instructions (Instructions), a Retrieval-Augmented Generation approach (RAG), and a fine-tuned LLM trained on a small parallel corpus (Fine-tuned). Our results show that the Pipeline translator achieves the highest overall translation quality, though at greater implementation cost. The Builder and Instructions approaches offer more lightweight and flexible solutions, trading off some accuracy for ease of use. In contrast, both the Fine-tuned and RAG methods consistently underperform, suggesting that in extremely low-resource scenarios, structured rule-based strategies may outperform more data-driven ones.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 49–68

The techniques proposed in this paper are designed for endangered languages like OVP, where the constraints differ fundamentally from mainstream MT research and even from settings typically described as "low-resource." Evaluation datasets must be constructed through careful, time-intensive collaboration with community members, and the pool of speakers available to verify translations is extremely limited. As a result, evaluation scale is necessarily much smaller than in well-resourced settings, and smaller still than in many "low-resource" scenarios where hundreds or thousands of parallel sentences may still exist. Our work demonstrates that rigorous, comparative MT research is both possible and valuable under these extreme constraints, providing a replicable framework for other endangered language efforts.

To our knowledge, this is the first work to rigorously evaluate the LLM-RBMT machine translation paradigm. We also introduce an evaluation framework aligned with the unique needs of endangered language communities, where grammatical correctness and semantic fidelity is often more important than literal, word-for-word accuracy. In settings where fluent speakers are scarce, even minor mistranslations or ungrammatical outputs can erode trust and misinformation can quickly propagate, undermining revitalization efforts. Our evaluation strategy prioritizes grammaticality and semantic fidelity over surface-level lexical overlap, rewarding translations that preserve intended meaning even if they involve simplification or restructuring.

A key advantage of the LLM-RBMT paradigm is that it guarantees grammatically correct output, even when vocabulary gaps require placeholder substitutions. This is critical for endangered language communities: with few fluent speakers available to catch and correct errors, mistranslations risk propagating through educational materials and learner communities, potentially undermining the very linguistic heritage we hope to preserve.

1.1 Contributions

We present the first systematic evaluation and extension of LLM-Assisted Rule-Based Machine Translation (LLM-RBMT). Specifically, we implement and compare five English-to-Owens Valley Paiute (OVP) translators using OpenAI's gpt-4o and gpt-4o-mini models:

1. **Pipeline**: A reengineered LLM-RBMT system with richer grammatical support.
2. **Builder**: An LLM-RBMT variant that constructs sentences step-by-step within a constrained vocabulary.
3. **Instructions**: A prompt-only baseline using in-context grammar (Tanzer et al., 2024).
4. **Fine-tuned**: An LLM fine-tuned on a small parallel corpus generated via LLM-RBMT.
5. **RAG**: A retrieval-augmented system grounded in dictionary lookups.

Our evaluation methodology is designed to be rigorous under the constraints of endangered language research. All translations were verified for grammaticality by intermediate-level speakers (native speakers being extremely rare), and back-translations for the LLM-RBMT systems were produced by these same speakers. Rather than relying on large-scale human evaluation (which is infeasible given speaker scarcity) we propose an evaluation framework based on automatic semantic similarity metrics, baseline contextualization, and qualitative examples that together provide meaningful insight into system behavior across a range of sentence complexities. This methodology offers a practical template for future work in settings where traditional evaluation approaches are impossible.

We evaluate these systems on a curated dataset of 150 English sentences spanning six distinct sentence types and compare them across translation quality and runtime. Our analysis combines both automatic metrics and qualitative observations, surfacing tradeoffs in structure, reliability, cost, and ease of implementation. While the evaluation is focused on a single endangered language, the sentence set was carefully designed to cover a broad range of grammatical constructions, allowing us to draw meaningful and generalizable conclusions.

2 Related Work

There is decades of research in machine translation for low-resource languages (Ranathunga et al., 2023), but these methods often do not perform well for endangered or "no-resource" languages, which typically lack sufficient parallel corpora for any training. As a result, traditional paradigms like rule-based machine translation (RBMT) and statistical machine translation (SMT) are still relevant for endangered languages (Torregrosa et al., 2019; Khanna et al., 2021; Pirinen, 2019), despite their reliance on expert-crafted rules. The LLM-RBMT

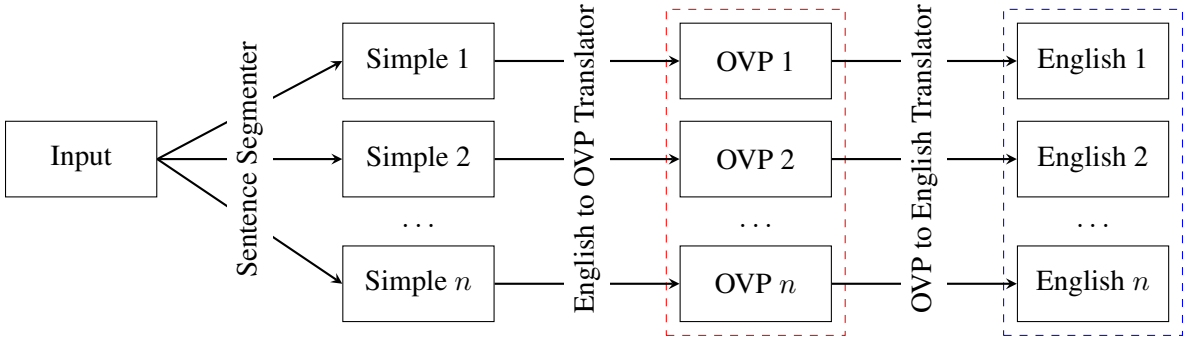


Figure 1: The "Pipeline" English to OVP translation process originally proposed in (Coleman et al., 2024). The box with a red, dashed border indicates the set of sentences in Owens Valley Paiute (the target language) and the box with a blue, dashed border indicates the set of English sentences they translate to. Ideally, the input sentence, simple sentences, and English output sentences will have high semantic similarity.

paradigm prototyped in (Coleman et al., 2024) offers a novel alternative that combines the determinism of RBMT with the flexibility of modern LLMs.

LLM-RBMT relies on two common in-context learning techniques: *prompt engineering* and *few-shot learning*, which supply task-relevant information without modifying model weights (Brown et al., 2020). There is growing interest in using such techniques for low-resource machine translation (Zhang et al., 2024; Court and Elsner, 2024; Elsner and Needle, 2023; Tanzer et al., 2024; Aycock et al., 2024), although these approaches are not rule-based and rely on LLMs to generate translations directly from in-context linguistic hints. Recent work, however, suggests state-of-the-art LLMs still struggle in these settings (Court and Elsner, 2024). One of the translators presented in this paper is based on the "Machine Translation from One Book" method (Tanzer et al., 2024).

Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) is another emerging strategy in which models access external tools or corpora at inference time. RAG has been applied to low-resource translation by incorporating dictionaries or example sentences and has gained traction more broadly as a technique for enhancing LLMs without retraining (Bubeck et al., 2023; Frieder et al., 2023; Park et al., 2023; Wang et al., 2023). We implement a RAG-style translator using OpenAI's function-calling capabilities to ground translations in trusted linguistic sources.

Fine-tuning has improved LLM performance across many tasks (Han et al., 2024), including translation for moderately low-resource languages (Lankford et al., 2023; Cruz and Cheng, 2019). However, in extremely low-resource settings, fine-tuning on limited data is unlikely to match the reliability of rule-based or prompt-based methods. We include a fine-tuned translator trained on a small parallel corpus to directly compare this strategy with LLM-RBMT and prompt-based alternatives.

To our knowledge, this is the first work to directly compare LLM-RBMT, prompt-only, RAG, and fine-tuned translation strategies in an extremely low-resource, endangered language setting.

3 Approaches

We implemented five translators: (1) Pipeline, (2) Builder, (3) Instructions, (4) Fine-tuned, and (5) RAG. In this section, we describe each translator in more detail.

3.1 Pipeline Translator

The Pipeline Translator builds on the LLM-Assisted Rule-Based Machine Translation (LLM-RBMT) paradigm introduced in (Coleman et al., 2024), but with a rearchitected implementation to support richer linguistic structures and greater modularity. At its core is a redesigned sentence builder that leverages OpenAI's structured output functionality and a formal schema defined using `pydantic`. This allows the system to capture fine-grained grammatical features such as **proximity**, **person**, **plurality**, **inclusivity**, and **reflexivity** with structured validation.

Each sentence is a structured object with three components: a **subject** (pronoun or noun, optionally verb-derived, annotated with proximity, plurality, and possessive markers), a **verb** (marked for tense and aspect, optionally prefixed with an object pronoun), and an optional **object** (noun or

nominalized verb with possessive determiners and agreement suffixes). This supports constructions like "sawa-dü-ii hukaw̃ia-doka ui-buni-ku" ("The cook saw the walkers"), where both subject and object are nominalized verbs. Newly supported features include agentive nominalization, possessive determiners (including reflexive forms marked differently in OVP), and pronoun disambiguation distinguishing "you and I" (`taa`) from inclusive/exclusive "we." The validation-first design improves robustness and transparency. For vocabulary, logic, and implementation details, see Appendix 6.

3.2 Builder Translator

The Builder Translator is a new LLM-RBMT approach that gives the LLM access to *only* the sentence-building tool and asks it to build sentences one choice at a time. This is unlike the Pipeline Translator, which uses various tools (see Figure 1) to build sentences. While the Pipeline approach never interacts with the target language, the Builder approach is given words in the target language and their English translations at each step of the sentence-building process. The potential advantage of this approach is that it is more aware of the vocabulary available and relies on fewer expert-designed tools. For this reason, the Builder approach might be more flexible and easier to implement for new languages and sentence constructions. We suspect it will, however, be slower and more expensive since it requires more interactions with the LLM (making one call to the model for every word it selects). The prompts and examples used for the Builder Translator can be found in Appendix 7.

3.3 Instructions Translator

The Instructions Translator is a new approach inspired by (Tanzer et al., 2024). In this approach, the LLM has no access to any tools or sentence-building capabilities. Instead, it is given a set of instructions on how to build sentences in the target language (i.e., a small "grammar book") and asked to use them to generate translations. Because this approach requires no tooling, it is the easiest to implement. Unlike the Pipeline and Builder approach, however, the translations it produces are not guaranteed to be grammatically correct. The prompts and examples used for the Instructions Translator can be found in Appendix 8.

3.4 Fine-tuned Translator

The Fine-tuned Translator is built using a standard fine-tuning approach. We fine-tune the gpt-4o and gpt-4o-mini models on 393 parallel sentences generated using the sentence builder tool described in Section 3.1. To ensure the dataset is representative of the vocabulary and sentence constructions supported by the other translators, we follow a structured approach to generate the parallel data. We start by selecting subjects and verbs from the vocabulary to guarantee that every verb, noun, and pronoun is represented in the dataset at least three times. Other parts of speech (objects, suffixes, etc.) are then randomly selected to complete each sentence. This results in 243 unique sentence pairs.

We also included an additional 125 sentence pairs with unknown vocabulary, to allow the model to learn to handle partial translations. We generated sentences with either subjects, objects, verbs, subjects and objects, or all three (subjects, objects, and verbs) that are missing from the vocabulary. The last 25 sentence pairs represent examples where the translator must simplify the sentence before translating. For example, the English sentence "She laughed quietly at his silly joke" should be simplified to something like "He told a joke. She laughed." before translating since the dataset contains no examples of prepositional phrases. Similarly the other translators are not given the information or tools to handle complex sentence structures like this and so must rely on simplifying the sentence before translating. The prompts and examples used for the Fine-tuned Translator can be found in Appendix 9.

3.5 RAG Translator

The RAG Translator uses a Retrieval-Augmented Generation approach to assist the LLM in generating translations grounded in real language data. Specifically, it leverages OpenAI's function-calling capabilities to interface with an external OVP dictionary. When translating, the system identifies key content words (e.g., nouns, verbs) from the English input and issues semantic search queries to the dictionary API. The returned results include matching OVP words, definitions, and usage examples. These are injected into the prompt as context, which the LLM then uses to produce a translation.

This approach requires no prior training or prompt-specific tuning for the target language. Instead, the dictionary serves as the only source of linguistic information, and the model is prompted

to adhere as closely as possible to the forms and grammar evident in the retrieved examples. The advantage of this method is that it can work with any language for which even a small dictionary and set of example sentences exist. However, because it does not use structured tools like LLM-RBMT or explicit grammar rules like the Instructions Translator, its output can vary in grammaticality and fluency depending on the coverage and quality of the retrieved material. The prompts and examples used for the RAG Translator can be found in Appendix 10.

4 Evaluation

We evaluate the translators using a dataset of 150 sentence pairs, evenly distributed across six sentence types: subject-verb, subject-verb-object, two-verb, two-clause, complex, and nominalization.

4.1 Evaluation Metrics

Translation quality is assessed using five metrics, which fall into two categories: lexical overlap metrics and semantic similarity metrics.

Lexical Metrics. We report BLEU and chrF++ scores for completeness, though these metrics are known to be limited in settings like ours. They rely on surface-level overlap with a reference and penalize semantically faithful but lexically divergent translations.

Semantic Metrics. To better evaluate meaning preservation, we also report:

- **BERTScore** (Zhang et al., 2020), which compares contextualized embeddings of sentences to assess alignment at the token level.
- **COMET** (Rei et al., 2020), a learned metric trained to estimate human judgments of translation quality.
- **Semantic Similarity (MiniLM)**, computed as the cosine similarity between `all-MiniLM-L6-v2` model (Reimers and Gurevych, 2020) sentence embeddings.

Although all five metrics are reported, the main results in this paper focus on the MiniLM-based semantic similarity score, as it most effectively captures the intended behavior of our translation system: preserving meaning, even if surface forms differ. Results for BLEU, chrF++, BERTScore, and COMET are consistent with Semantic Similarity and are included in the appendices.

4.2 Contextualizing Evaluation Scores

Because our translation task emphasizes semantic alignment (and because traditional ground-truth references are unavailable) we contextualize the evaluation scores using a baseline analysis. Specifically, we compute each metric over all 11,175 distinct pairs of unrelated sentences in the dataset. These scores represent a background distribution of unrelated sentence comparisons and serve as a proxy for what a "typical" low similarity score looks like in our setting.

Table 1 shows the mean and standard deviation for each metric under this null hypothesis. This allows us to define conservative quality thresholds. For example, a translation that yields a Semantic Similarity score above $\mu + 3\sigma$ (e.g., > 0.746) is very unlikely to be unrelated to the input sentence and therefore likely preserves core meaning.

Table 1: Baseline scores computed over all pairwise comparisons of unrelated sentences in the dataset.

Metric	Mean	Standard Deviation
Semantic Similarity	0.569	0.059
BLEU Score	0.041	0.022
chrF++ Score	13.571	6.007
BERTScore (F1)	0.887	0.018
COMET Score	0.452	0.093

Baseline histograms for all metrics are provided in Appendix 11. We believe that this kind of contextual analysis is especially valuable in low/no-resource translation, where traditional reference-based evaluation is impractical. We encourage other researchers working in this space to adopt similar baselining strategies to interpret metric scores more meaningfully.

To apply these metrics in the absence of ground-truth references, we follow the indirect evaluation strategy from (Coleman et al., 2024). For each translation, we compute similarity scores between the original English input and two English reconstructions: the *backwards translation*, generated by translating the output OVP sentence back into English, and the *comparator translation*, which is the backwards translation with any vocabulary not available to the LLM-RBMT translators replaced by placeholders (e.g., `[VERB]`, `[OBJECT]`). The comparator is important because the backwards evaluation does not penalize the LLM-RBMT sys-

tems for using English placeholders.

A good translation, then, should score high on both the backwards and comparator translations. The following translation, for example, is perfect:

They are climbing.

Target (translator/model: Instructions/gpt-4o-mini)
Uhuw̃a tsibui-ti.
Backwards
They are climbing. 1.00
Comparator
They are climbing. 1.00

Another interesting case, though, is when the backwards translation is good but the comparator translation is bad, as in the following example:

The king wore a crown.

Target (translator/model: Pipeline/gpt-4o)
[crown]-neika [king]-uu ma-[wear]-ku.
Backwards
The king wore the crown. 0.995
Comparator
[SUBJECT] [VERB]-ed [OBJECT]. 0.541

This is a case where the translation is correct, but the LLM-RBMT translators do not have the vocabulary for an exact translation. When the backwards *and* comparator translations are bad, the translation is likely incorrect. For example:

My brother and I went hiking.

Target (translator/model: Pipeline/gpt-4o)
mia-ku [brother]-ii. nüü mia-ku.
Backwards
My brother went. I went. 0.837
Comparator
[SUBJECT] went. I went. 0.634

This is a case where the Pipeline translator performed poorly: a lot of information was lost in simplifying the input sentence.

Because the Pipeline and Builder Translators are based on the LLM-RBMT paradigm, they always produce grammatically correct sentences. This is not the case, however, for the Instructions and Fine-tuned Translators. Whenever the target sentence was ungrammatical, we assigned backwards and comparator semantic similarity scores of 0.0. For example, the following translation is not grammatical. Subject pronouns like "uhu" (he/she/it, distal) cannot take subject determiner suffixes like "-uu":

He works.

Target (translator/model: Instructions/gpt-4o-mini)
waakü-dü uhu-uu.
Backwards
N/A 0.00
Comparator
N/A 0.00

4.3 Results and Discussion

In this section, we present the results of evaluating the four translators on the dataset of 150 sentences. Figures 2 and 3 show the backwards and comparator semantic similarity scores, respectively. Some interesting observations can be made from these results. First, the Pipeline Translator is the most consistent across sentence types and models, likely due to it having the most structure and tooling to guide the translation process. Interestingly, the gpt-4o Instructions Translator performs relatively well, especially given its ease of implementation and quick translation time. It does have much higher variance, however, which suggests that it isn't as reliable as the Pipeline and Builder Translators. The Fine-tuned Translator performs the worst, likely due to the small number of parallel sentences used to train the model. It's important to note that both the Instructions and Fine-tuned Translators' performance depends greatly on prompt design and the quality of the parallel data used to train the model (respectively). Evaluating different versions of these translators with different prompts and training data is an important area for future work. Perhaps surprisingly, the RAG Translator also performs poorly, despite having access to significantly more vocabulary and example sentences than the other systems. A closer inspection reveals the reason for this: many of its translations are *nearly* correct, but are ungrammatical. For example:

Romeo wrote a letter.

Target (translator/model: RAG/gpt-4o)
Romeo i-dü-mui-pü piponibü.
Backwards
N/A 0.000
Comparator
N/A 0.000

This sentence attempts to translate "Romeo wrote a letter" by composing "i-dü-mui-pü" (something like "has written me") with "piponibü" ("paper"). While semantically aligned with the input, the sen-

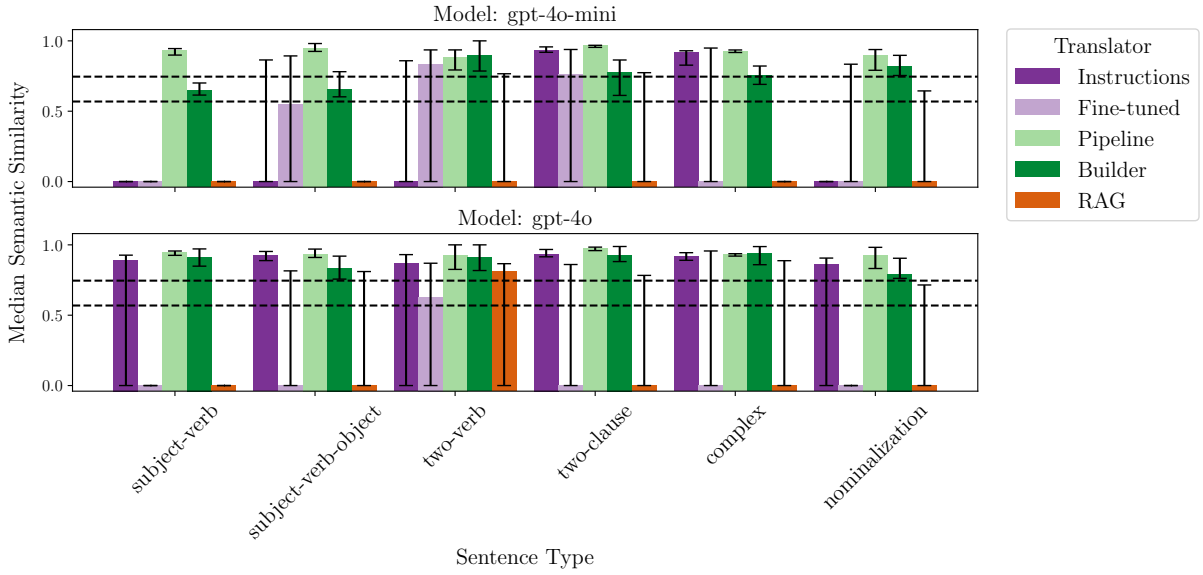


Figure 2: Backwards similarity scores: semantic similarity between target and backwards sentences. The colored bars indicate the median for each translator and error bars indicate the 25th and 75th percentiles. The dashed lines represent the baseline semantic similarity scores for unrelated sentences μ and $\mu + 3\sigma$ (see Section 4).

tence is not grammatically correct, and so cannot be reliably back-translated. Thus, it receives a semantic similarity score of 0 under our evaluation framework. This suggests that the RAG Translator may benefit from more explicit guidance such as access to grammatical rules, partial sentence builders, or controlled generation mechanisms to turn these "almost right" outputs into correct and reliable translations. Exploring such hybrid strategies is a promising direction for future work.

The two LLM-RBMT approaches (Pipeline and Builder) are particularly interesting to compare. The Pipeline Translator generally performs better than the Builder Translator on the backwards metric, but the Builder Translator performs equally well and sometimes better on the comparator metric. In Section 3.2, we hypothesized that the Builder Translator might be more capable of producing good translations when the exact vocabulary used in the input sentence is not available. The results suggest this is sometimes the case. Consider the following translation:

The chef prepared a meal.

Target (translator/model: Builder/gpt-4o)
[chef]-uu tuunapi-neika a-zawa-ku.
Backwards
The chef cooked the food. 0.926
Comparator
[SUBJECT] cooked food. 0.766

and the same sentence by the Pipeline Translator:

The chef prepared a meal.

Target (translator/model: Pipeline/gpt-4o)
[meal]-neika [chef]-uu a-[prepare]-ku.
Backwards
The chef prepared the meal. 0.997
Comparator
[SUBJECT] [VERB]-ed [OBJECT]. 0.529

Because the Builder Translator is aware of the vocabulary available in the system, it chooses to use the words "cook" and "food" (available in the vocabulary) instead of "prepare" and "meal" (not available in the vocabulary) to produce a translation that doesn't rely on so many English placeholders.

Table 2 shows average cost per translation. The Instructions and Fine-tuned translators are cheapest since they make only one model call; Builder is most expensive due to many calls. Pipeline is fairly inexpensive and balances cost with quality well. RAG is cheaper than Builder but costlier than Pipeline, likely due to function-calling overhead and larger context windows.

Table 3 shows average translation time per sentence, with results consistent with cost: Instructions and Fine-tuned are fastest, Builder is slowest (due to multiple model calls), and RAG falls between Builder and Pipeline.

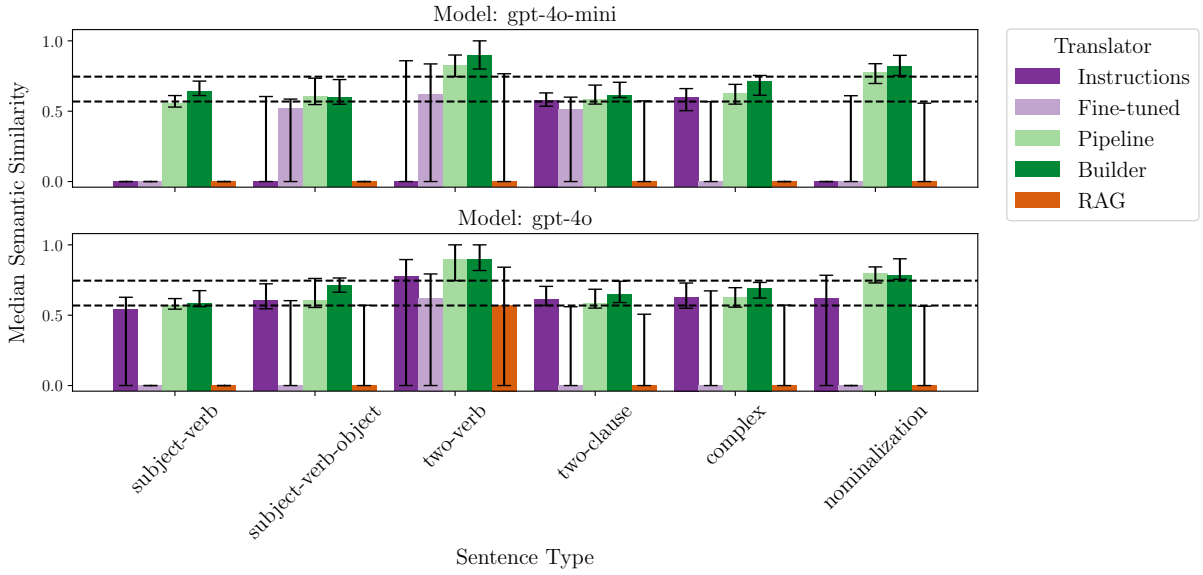


Figure 3: Comparator similarity scores: semantic similarity between target and comparator sentences. The colored bars indicate the median for each translator and error bars indicate the 25th and 75th percentiles. The dashed lines represent the baseline semantic similarity scores for unrelated sentences μ and $\mu + 3\sigma$ (see Section 4).

Table 2: Avg. Cost/Translation and Standard Deviation

Translator	Model	Avg. Cost ($)	Std. Dev. ($)
Builder	4o	0.581	0.186
Builder	4o-mini	0.024	0.011
Fine-tuned	4o	0.001	0.000
Fine-tuned	4o-mini	0.000	0.000
Instructions	4o	0.003	0.000
Instructions	4o-mini	0.000	0.000
Pipeline	4o	0.015	0.001
Pipeline	4o-mini	0.001	0.000
RAG	4o	0.107	0.003
RAG	4o-mini	0.006	0.000

Table 3: Avg. Translation Time and Standard Deviation

Translator	Model	Avg. Time (s)	Std. Dev. (s)
Builder	4o	18.855	8.066
Builder	4o-mini	10.991	5.264
Fine-tuned	4o	1.317	2.411
Fine-tuned	4o-mini	2.047	10.797
Instructions	4o	1.262	0.383
Instructions	4o-mini	0.848	1.347
Pipeline	4o	9.391	49.176
Pipeline	4o-mini	5.862	2.498
RAG	4o	7.288	1.956
RAG	4o-mini	10.677	3.908

5 Conclusion

We presented and evaluated five machine translation approaches for extremely low-resource languages. The LLM-Assisted Rule-Based Machine Translation (LLM-RBMT) paradigm, particularly the Pipeline Translator, achieved the best overall performance, balancing grammaticality and semantic fidelity. The Builder Translator handled lexical gaps well but was more costly and the Instructions and Fine-tuned translators were faster and cheaper but less reliable. Surprisingly, the RAG Translator underperformed despite having access to more vocabulary and sentence examples, suggesting that unconstrained retrieval has limited benefit in low-resource settings. We also presented an evaluation framework that prioritizes grammaticality and semantic alignment over word-for-word accuracy.

We believe this work opens many avenues for impactful future research. One direction is to extend LLM-RBMT to support even more advanced constructions. Another is to combine LLM-RBMT with retrieval, more careful prompt-engineering, and fine-tuning. Ablation studies could also shed light on which components (tools, prompts, training data, etc.) most influence performance. Finally, there is a need for standardized benchmarks tailored to extremely low-resource languages. We view this work as a foundation toward that broader goal.

References

Seth Aycock, David Stap, Di Wu, Christof Monz, and Khalil Sima'an. 2024. Can LLMs Really Learn to Translate a Low-Resource Language from One Grammar Book?

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott M. Lundberg, Harsha Nori, Hamid Palangi, Marco Túlio Ribeiro, and Yi Zhang. 2023. Sparks of artificial general intelligence: Early experiments with GPT-4. *CoRR*, abs/2303.12712.

Aakanksha Chowdhery and 1 others. 2022. PaLM: Scaling Language Modeling with Pathways.

Jared Coleman, Bhaskar Krishnamachari, Ruben Rosales, and Khalil Iskarous. 2024. LLM-assisted rule based machine translation for low/no-resource languages. In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 67–87, Mexico City, Mexico. Association for Computational Linguistics.

Sara Court and Micha Elsner. 2024. Shortcomings of llms for low-resource translation: Retrieval and understanding are both the problem. *CoRR*, abs/2406.15625.

Jan Christian Blaise Cruz and Charibeth Cheng. 2019. Evaluating language model finetuning techniques for low-resource languages. *CoRR*, abs/1907.00409.

Micha Elsner and Jordan Needle. 2023. Translating a low-resource language using GPT-3 and a human-readable dictionary. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology, SIGMORPHON@ACL 2023, Toronto, Canada, 14 July 2023*, pages 1–13. Association for Computational Linguistics.

Simon Frieder, Luca Pinchetti, Alexis Chevalier, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Petersen, and Julius Berner. 2023. Mathematical capabilities of chatgpt. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. 2024. Parameter-efficient finetuning for large models: A comprehensive survey. *CoRR*, abs/2403.14608.

Tanmai Khanna, Jonathan N. Washington, Francis M. Tyers, Sevilay Bayatlı, Daniel G. Swanson, Tommi A. Pirinen, Irene Tang, and Hèctor Alòs i Font. 2021. Recent advances in Apertium, a free/open-source rule-based machine translation platform for low-resource languages. *Machine Translation*, 35(4):475–502.

Séamus Lankford, Haithem Afli, and Andy Way. 2023. adaptmllm: Finetuning multilingual language models on low-resource languages with integrated LLM playgrounds. *Inf.*, 14(12):638.

Patrick S. H. 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 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Joon Sung Park, Joseph C. O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, UIST 2023, San Francisco, CA, USA, 29 October 2023- 1 November 2023*, pages 2:1–2:22. ACM.

Tommi A Pirinen. 2019. Workflows for kickstarting RBMT in virtually no-resource situation. In *Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages*, pages 11–16, Dublin, Ireland. European Association for Machine Translation.

Surangika Ranathunga, En-Shiun Annie Lee, Marjana Prifti Skenduli, Ravi Shekhar, Mehreen Alam, and Rishemjit Kaur. 2023. Neural machine translation for low-resource languages: A survey. *ACM Comput. Surv.*, 55(11):229:1–229:37.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nathaniel R. Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for High- (but not Low-) Resource Languages.

Garrett Tanzer, Mirac Suzgun, Eline Visser, Dan Jurafsky, and Luke Melas-Kyriazi. 2024. A benchmark for learning to translate a new language from one grammar book. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Daniel Torregrosa, Nivranshu Pasricha, Maraim Masoud, Bharathi Raja Chakravarthi, Juan A. Alonso, Noe Casas, and Mihael Arcan. 2019. Leveraging rule-based machine translation knowledge for under-resourced neural machine translation models. In *Proceedings of Machine Translation Summit XVII Volume 2: Translator, Project and User Tracks, MT-Summit 2019, Dublin, Ireland, August 19-23, 2019*, pages 125–133. European Association for Machine Translation.

Xuena Wang, Xueting Li, Zi Yin, Yue Wu, and Jia Liu. 2023. Emotional intelligence of large language models. *Journal of Pacific Rim Psychology*, 17:18344909231213958.

Chen Zhang, Xiao Liu, Jiuheng Lin, and Yansong Feng. 2024. Teaching large language models an unseen language on the fly. *CoRR*, abs/2402.19167.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. *CoRR*, abs/1904.09675.

6 Pipeline Translator

The Pipeline Translator is a modular system that breaks down the translation process into three main stages: sentence simplification, structured translation, and backwards translation. Each stage is handled by an LLM guided via system prompts, examples, and structured outputs. This architecture enables flexibility, transparency, and better control over how grammar and vocabulary are applied.

6.1 Step 1: Sentence Simplification

The first step converts the original English sentence into one or more simple SV/SVO constructions. These simplified sentences are expressed in a structured JSON format that includes the subject, verb, object (if applicable), and associated grammatical features such as tense, aspect, and proximity.

The system prompt for this step instructs the LLM to split input sentences into semantically equivalent simple clauses, avoiding extra modifiers and maintaining grammatical simplicity:

```
You are an assistant that splits user input sentences into a set of simple SVO or SV sentences. The set of simple sentences should be as semantically equivalent as possible to the user input sentence. No adjectives, adverbs, prepositions, or conjunctions should be added to the simple sentences. Indirect objects and objects of prepositions should not be included in the simple sentences. Subjects and objects can be verbs (via nominalization) (e.g., "run" → "the runner", "the one who ran", "the one who will run").
```

For example, given the input sentence:

```
I saw two men walking their dogs while drinking coffee.
```

The model might return:

```
{ "sentences": [ {"subject": "I", "verb": "see", "verb_tense": "past", "object": "man"}, {"subject": "man", "verb": "walk", "verb_tense": "past_continuous", "object": "dog"}, {"subject": "man", "verb": "drink", "verb_tense": "past_continuous", "object": "coffee"} ] }
```

These structured representations serve as the intermediate form that feeds into the next stage.

6.2 Step 2: Structured Translation to OVP

Each simplified sentence is then translated into a grammatical OVP sentence using a deterministic set of sentence-building functions. These functions rely on explicit mappings between English lemmas and available Paiute roots (for nouns and verbs), combined with correct suffixes for tense, aspect, plurality, and proximity.

The model uses custom logic to convert structured inputs into Paiute phrases, considering:

- **Pronoun resolution**: e.g., `first-person singular → nüü`.
- **Tense and aspect suffixes**: e.g., `past + completive → -ku`, `present continuous → -ti`.
- **Object pronoun prefixes**: added before the verb for transitive verbs.
- **Word lookup and fallback**: If a word is not found in the lexicon, it is left in brackets (e.g., `[crown]`).

The resulting Paiute sentence components are ordered based on grammatical rules: subject-first for pronouns, verb-first for noun subjects.

6.3 Step 3: Backwards Translation

To enable evaluation, the structured Paiute output is reverse-engineered back into English. This step uses the same grammar logic and vocabulary definitions used during construction, and represents each sentence as a list of parts of speech (e.g., subject, object, verb) and their definitions.

These parts are passed to the LLM, which is prompted to return a natural English translation:

```
You are an assistant for translating structured sentences into natural English sentences.
```

An example input to this prompt might look like:

```
[ {"part_of_speech": "subject", "positional": "proximal", "word": "wood"}, {"part_of_speech": "object", "positional": "proximal", "word": "dog"}, {"part_of_speech": "verb", "tense": "present ongoing (-ing)", "word": "see"} ]
```

The model would respond with:

```
This wood is seeing this dog.
```

This translation is then used for semantic comparison with the original English sentence.

6.4 Comparator Sentences

In addition to the backwards translation, we also compute a *comparator translation*, in which any word not available to the translator is replaced with a placeholder (e.g., `[OBJECT]`, `[VERB]`). This lets us measure how well the system did using only known vocabulary, and is particularly useful for LLM-RBMT systems that transparently insert English placeholders for out-of-vocabulary words. It helps ensure that semantic similarity is not inflated by untranslated content.

For full implementation details, including example prompts and models used, we refer the reader to the Data and Code Appendix.

7 Builder Translator

The system prompt for the Pipeline Translator is as follows:

```
You are an assistant trying to build a sentence in Paiute. The user will provide you with parts of speech and vocabulary options one at a time. Make choices to best approximate the meaning of Input Sentence. Do not make choices that are not provided by the user. This may mean you can't build the sentence you want, but that's okay. Whenever you've chosen enough parts of speech and vocabulary to form a grammatically correct sentence, the user will ask you if you want to continue. If you're happy with the sentence you've built, you can choose to stop. If not, continue selecting optional parts of speech and vocabulary until you're satisfied.
```

The model is them prompted with messages like:

```
Input Sentence: Jared will eat an apple.
Current Translation: .
Current Choices: { 'subject_noun': None, 'subject_suffix': None, 'subject_noun_nominalizer': None, 'verb': None, 'verb_tense': None, 'object_pronoun': None, 'object_noun': None, 'object_noun_nominalizer': None, 'object_suffix': None }
Please select a required part of speech: subject_noun, verb
```

and should respond with choices like:

```
subject_noun
```

, to which the translator will then prompt the model again with something like:

```
Input Sentence: Jared will eat an apple.
Current Translation: .
Current Choices: { 'subject_noun': None, 'subject_suffix': None, 'subject_noun_nominalizer': None, 'verb': None, 'verb_tense': None, 'object_pronoun': None, 'object_noun': None, 'object_noun_nominalizer': None, 'object_suffix': None }
Please select a word for subject_noun: nüü (I), uhu (he/she/it), uhuña (they), mahu (he/she/it), mahuña (they), ihi (this), ihiña (these), taa (you and I), nüügwa (we (exclusive)), taagwa (we (inclusive)), üü (you), üügwa (you (plural)), isha' (coyote), isha'pugu (dog), kidi' (cat), pugu (horse), wai (rice), tüba (pinenuts), maishibü (corn), paya (water), payahuupü (river), katünu (chair), toyabi (mountain), tuunapi (food), pasohobü (tree), nobi (house), toni (wickiup), apo (cup), küna (wood), tübbi (rock), tabuutsi' (cottontail), kamü (jackrabbit), aaponu' (apple), tüsüga (weasle), mukita (lizard), wo'ada (mosquito), wükada (bird snake), wo'abi (worm), aingwü (squirrel), tsiipa (bird), tüwoobü (earth), koopi' (coffee), pahabichi (bear), pagwi (fish), kwadzi (tail), tüka (eat), puni (see), hibi (drink), naka (hear), kwana (smell), kwati (hit), yadohi (talk to), naki (chase), tsibui (climb), sawa (cook), tama'i (find), nia (read), mui (write), nobini (visit), katü (sit), üwi (sleep), kwisha'i (sneeze), poyoha (run), mia (go), hukañia (walk), wünü (stand), habi (lie down), yadoha (talk), kwatsa'i (fall), waakü (work), wükihaa (smile), hubiadu (sing), nishua'i (laugh), tübinohi (play), yotsi (fly), nüga (dance), pahabi (swim), tünia (read), tümui (write), tsiipe'i (chirp)
Because this is a subject_noun word, you can also choose to use a wildcard by putting the word in brackets. For example: [wildcard]
```

The model is then prompted to select a word for the subject noun, and the process continues until the sentence is complete. When the sentence is grammatically correct, the model will be asked if it wants to continue or stop:

```
Input Sentence: Jared will eat an apple.
Current Translation: [Jared]-ii tüka-wei.
Enter one of the following choices:
c: Continue building the last sentence
n: Add and build a new Paiute sentence for this translation
t: Terminate and return the current translation.
```

For more information on the specific prompts, models, examples, and more, we refer the reader to the Data and Code Appendix.

8 Instructions Translator

The system prompt for the Instructions Translator contains all of the grammar rules and vocabulary available to the model and is as follows (split into two parts for readability):

```
You use the following grammar rules to translate user input sentences from English to Owens Valley Paiute. Use the vocabulary and sentence structures available to translate the input sentence as best as possible. It doesn't need to be perfect and you can leave English words untranslated if necessary.
# Vocabulary
## Nouns: isha' (coyote), isha'pugu (dog), kidi' (cat), pugu (horse), wai (rice), tüba (pinenuts), maishibü (corn), paya (water), payahuupü (river), katünu (chair), toyabi (mountain), tuunapi (food), pasohobü (tree), nobi (house), toni (wickiup), apo (cup), küna (wood), tübbi (rock), tabuutsi' (cottontail), kamü (jackrabbit), aaponu' (apple), tüsüga (weasle), mukita (lizard), wo'ada (mosquito), wükada (bird snake), wo'abi (worm), aingwü (squirrel), tsiipa (bird), tüwoobü (earth), koopi' (coffee), pahabichi (bear), pagwi (fish), kwadzi (tail)
## Transitive Verbs: tüka (eat), puni (see), hibi (drink), naka (hear), kwana (smell), kwati (hit), yadohi (talk to), naki (chase), tsibui (climb), sawa (cook), tama'i (find), nia (read), mui (write), nobini (visit)
## Intransitive Verbs: katü (sit), üwi (sleep), kwisha'i (sneeze), poyoha (run), mia (go), hukañia (walk), wünü (stand), habi (lie down), yadoha (talk), kwatsa'i (fall), waakü (work), wükihaa (smile), hubiadu (sing), nishua'i (laugh), tsibui (climb), tübinohi (play), yotsi (fly), nüga (dance), pahabi (swim), tünia (read), tümui (write), tsiipe'i (chirp)
## Object Suffixes: eika (proximal), oka (distal)
## Object Pronouns: i (me), u (him/her/it (distal)), ui (them (distal)), ma (him/her/it (proximal)), mai (them (proximal)), a (him/her/it (proximal)), ai (them (proximal)), ni (us (plural, exclusive)), tei (us (plural, inclusive)), ta (us (dual), you and I), ü (you (singular)), üi (you (plural), you all)
```

```
## Subject Suffixes: ii (proximal), uu (distal)
## Subject Pronouns: nüü (I), uhu (he/she/it), uhuwña (they), mahu (he/she/it), mahuwña (they), ihi (this), ihiwña (these), taa (you and I), nüügwa (we (exclusive)), taagwa (we (inclusive)), üü (you), üügwa (you (plural))
## Verb Nominalizer Tenses: dü (present), pü (have x-ed, am x-ed), weidü (future (will)),
# Sentence Structure
## Simple Sentence Structure:
Subject-Object-Verb: [object noun]-[object suffix] [subject noun]-[subject suffix] [object pronoun]-[verb]-[verb tense]
Subject Pronoun-Object-Verb: [object noun]-[object suffix] [subject pronoun] [object pronoun]-[verb]-[verb tense]
Subject-Verb: [verb]-[verb tense] [subject noun]-[subject suffix]
## Verb Nominalization Sentence Structure:
Subject Nominalizer: [verb]-[verb nominalizer tense]-[subject suffix] [verb nominalizer]-[verb nominalizer tense]
Object Nominalizer: [verb]-[verb nominalizer tense]-[object suffix] [subject noun]-[subject suffix] [object pronoun]-[verb]-[verb tense]
Subject&Object Nominalizer: [verb]-[verb nominalizer tense]-[object suffix] [verb nominalizer]-[verb nominalizer tense]-[subject suffix] [subject noun]-[subject suffix] [object pronoun]-[verb]-[verb tense]
# Fortis/Lenis Transformations p->b, t->d, k->g, s->z, m->w̃
```

The model is then prompted with messages like:

```
TThis cook saw the ones who walked by the house.
```

and should respond with:

```
sawa-dü-ii hukaw̃ia-doka ui-buni-ku.
```

We use few-shot examples to demonstrate how the grammar rules and vocabulary should be used to translate the input sentence. For more information on the specific prompts, models, examples, and more, we refer the reader to the Data and Code Appendix.

9 Fine-tuned Translator

The fine-tuned translator is built using a standard fine-tuning approach (the dataset used for fine-tuning is described in Section 3.4). The system prompt for the Fine-tuned Translator is as follows:

```
You are a translator for translating text from English to OVP. For any word that does not have an equivalent in OVP, leave the word untranslated and place it inside brackets.
```

The model is then prompted with messages like:

```
The jackrabbit is eating the bread.
```

and should respond with:

```
kamü-ii tüka-wei [bread]-neika.
```

Fine-tuning was conducted using the OpenAI API, targeting two base models: `gpt-4o-2024-08-06` and `gpt-4o-mini-2024-07-18`. Each model was trained on 193,920 tokens using a dataset automatically compiled from seven translation subsets (e.g., "random_good_translations.csv," "random_no_subject_noun.csv," etc.). The data was transformed into a chat-completion format, with one user-assistant message pair per row. The training was conducted with the following hyperparameters:

- **Epochs:** 3
- **Batch size:** 1
- **Learning rate multiplier:** 1.8 (for `gpt-4o-mini`), 2.0 (for `gpt-4o`)
- **Seed:** 1296158545 (for `gpt-4o-mini`), 236502696 (for `gpt-4o`)

No validation split was used during training. The final models are identified as `ft:gpt-4o-2024-08-06:kubishi::AInyiTpj` and `ft:gpt-4o-mini-2024-07-18:kubishi::AInrzzLW`, respectively.

For more information on the specific prompts, models, examples, and implementation details, we refer the reader to the Data and Code Appendix.

10 RAG Translator

The RAG (Retrieval-Augmented Generation) Translator uses an LLM augmented with retrieval capabilities to translate English to Owens Valley Paiute. It is capable of calling external tools to supplement its knowledge during translation.

10.1 Available Tools

The model has access to the following tools:

- `search_english`: retrieves dictionary definitions, glosses, and example usages for English words.
- `search_sentences`: retrieves example sentence pairs from a parallel English–Paiute dataset.

10.2 Workflow

The translator proceeds in an iterative dialogue:

1. The English sentence is passed to the model.
2. The model may invoke tools to look up word meanings or similar sentences.
3. Retrieved results are inserted into the conversation.
4. The model generates a Paiute translation, optionally using retrieved examples.

All interactions and retrieval steps are included in the translation metadata.

10.3 Few-Shot Tool Use Instruction

The model is guided to use tools through few-shot prompting. Prior examples are included in the message history that demonstrate:

- Calling `search_sentences` on full input sentences to retrieve relevant phrasal translations.
- Decomposing complex inputs and calling `search_english` on key content words (e.g., "sit," "chair," "in").
- Handling tense and argument structure by querying expressions like "present continuous" or "reflexive possessive pronoun."

This few-shot strategy "teaches" the model to reason over the tools available and to apply them appropriately even on novel inputs.

For code and configuration details, see the Data and Code Appendix.

11 Evaluation Metric Baselines

To establish reference baselines for each evaluation metric, we computed pairwise scores between all distinct sentence pairs in the dataset (11,175 total comparisons). These comparisons reflect how each metric scores unrelated or semantically distant sentences, and help contextualize what constitutes a "high" or "low" value for a given metric.

Figure 4 shows the distribution of cosine similarities between sentence embeddings generated by the all-MiniLM-L6-v2 model. As reported in the main text, the mean similarity between unrelated sentences is approximately 0.569 with a standard deviation of 0.059. Similar histograms for the other metrics are provided below. These baseline distributions are used throughout the paper to interpret evaluation results and define thresholds for semantic alignment.

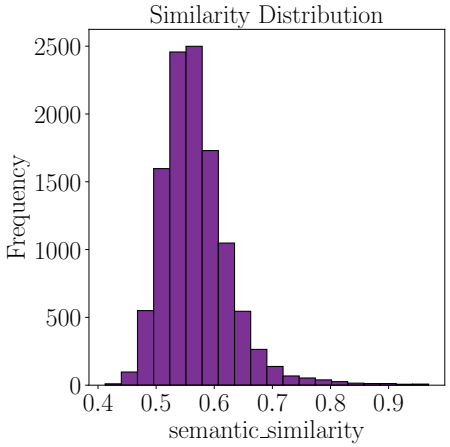


Figure 4: Distribution of MiniLM-based semantic similarity between all pairs of unrelated sentences in the dataset.

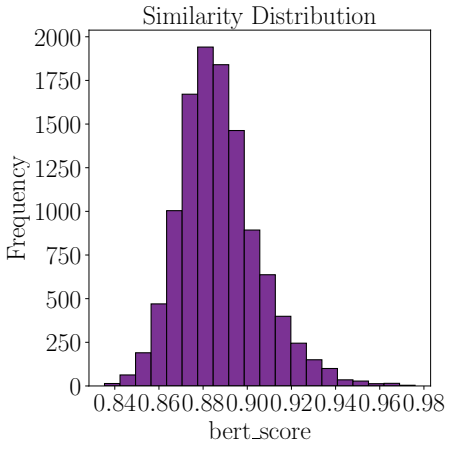


Figure 5: Distribution of BERTScore (F1) between all pairs of unrelated sentences in the dataset.

12 Vocabulary

Table 4 (on the next page) shows the vocabulary available to the Pipeline and Builder Translators.

tüka	eat
puni	see
hibi	drink
naka	hear
kwana	smell
kwati	hit
yadohi	talk to
naki	chase
tsibui	climb
sawa	cook
tama'i	find
nia	read
mui	write
nobini	visit

(a) Transitive Verbs

katü	sit
üwi	sleep
kwisha'i	sneeze
poyoha	run
mia	go
hukawĩa	walk
wünü	stand
habi	lie down
yadoha	talk
kwatsa'i	fall
waakü	work
wükihaa	smile
hubiadu	sing
nishua'i	laugh
tsibui	climb
tübinohi	play
yotsi	fly
nüga	dance
pahabi	swim
tünia	read
tümui	write
tsiipe'i	chirp

(b) Intransitive Verbs

isha'	coyote
isha'pugu	dog
kidi'	cat
pugu	horse
wai	rice
tüba	pinenuts
maishibü	corn
paya	water
payahuupü	river
katünu	chair
toyabi	mountain
tuunapi	food
pasohobü	tree
nobi	house
toni	wickiup
apo	cup
küna	wood
tübbi	rock
tabuutsi'	cottontail
kamü	jackrabbit
aaponu'	apple
tüsüga	weasle
mukita	lizard
wo'ada	mosquito
wükada	bird snake
wo'abi	worm
aingwü	squirrel
tsiipa	bird
tüwoobü	earth
koopi'	coffee
pahabichi	bear
pagwi	fish
kwadzi	tail

(c) Nouns

ku	completive (past)
ti	present ongoing (-ing)
dü	present
wei	future (will)
gaa-wei	future (going to)
pü	have x-ed, am x-ed

(d) Object Suffixes

nüü	I
uhu	he/she/it
uhuwĩa	they
mahu	he/she/it
mahuwĩa	they
ihi	this
ihiwĩa	these
taa	you and I
nüügwa	we (exclusive)
taagwa	we (inclusive)
üü	you
üügwa	you (plural)

(e) Subject Pronouns

ii	(proximal)
uu	(distal)

(f) Subject Suffixes

i	me
u	him/her/it (distal)
ui	them (distal)
ma	him/her/it (proximal)
mai	them (proximal)
a	him/her/it (proximal)
ai	them (proximal)
ni	us (plural, exclusive)
tei	us (plural, inclusive)
ta	us (dual), you and I
ü	you (singular)
üi	you (plural), you all

(g) Object Pronouns

eika	(proximal)
oka	(distal)

(h) Object Suffixes

Table 4: Vocabulary available in sentence building system.

13 BLEU and chrF++ Scores

We provide the BLEU, chrF++, BERTScore, and COMET results of the input sentences against the backwards and comparator translations (starting on the next page).

14 Reproducibility Checklist

Unless specified otherwise, each answer refers to material provided either in the main paper, in the appendix, or in the public source code and data archive (data_code_appendix.zip). That archive includes:

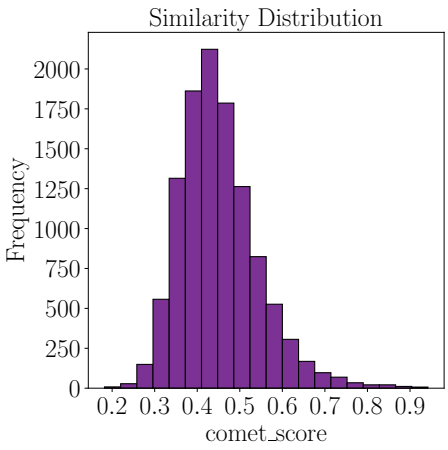


Figure 6: Distribution of COMET scores between all pairs of unrelated sentences in the dataset.

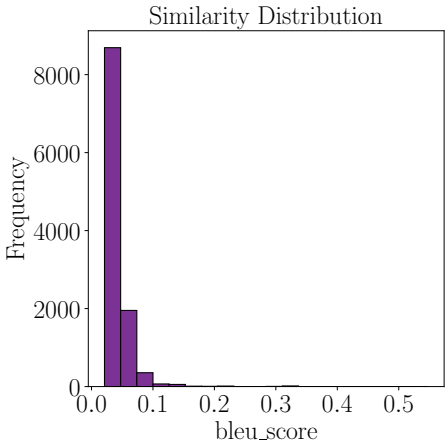


Figure 7: Distribution of BLEU scores between all pairs of unrelated sentences in the dataset.

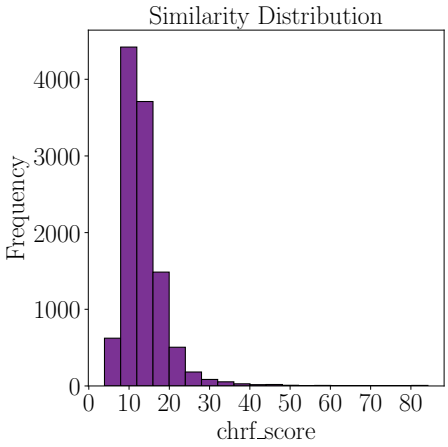


Figure 8: Distribution of chrF++ scores between all pairs of unrelated sentences in the dataset.

- Full source code for all translators and evaluation scripts
- All datasets used in experiments
- A README.md with detailed reproduction instructions

Conceptual and Descriptive Clarity

- Includes a conceptual outline and/or pseudocode description of AI methods introduced: **Yes**
- Clearly delineates statements that are opinions, hypotheses, and speculation from objective facts and results: **Yes**
- Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper: **Yes**

Theoretical Contributions

- Does this paper make theoretical contributions? **No**

Datasets

- Does this paper rely on one or more datasets? **Yes**
- A motivation is given for why the experiments are conducted on the selected datasets: **Yes**
- All novel datasets introduced in this paper are included in a data appendix: **Yes** (included in data_code_appendix.zip)
- All novel datasets will be made publicly available upon publication with a license that allows free research use: **Yes**
- All datasets drawn from the existing literature are accompanied by appropriate citations: **Yes**
- All datasets drawn from the existing literature are publicly available: **Yes**
- All datasets that are not publicly available are described in detail: **N/A**

Computational Experiments

- This paper states the number and range of values tried per (hyper-)parameter during development, along with the criterion used for selecting the final setting: **Yes** (see Appendix 9)
- Any code required for preprocessing data is included in the appendix: **Yes**

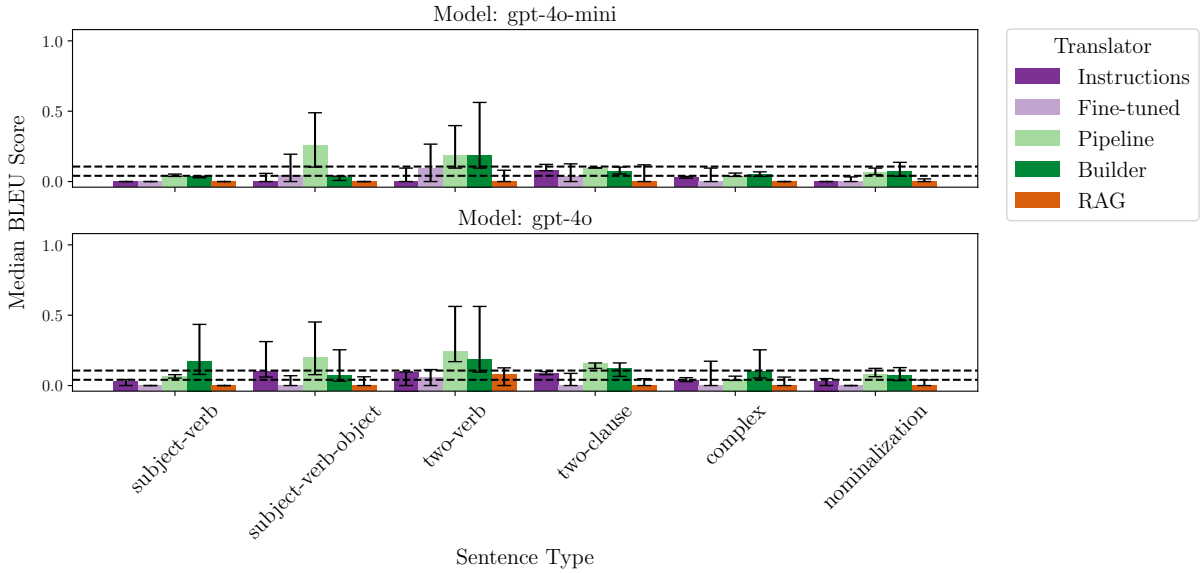


Figure 9: BLEU scores of the input sentences against the backwards translations.

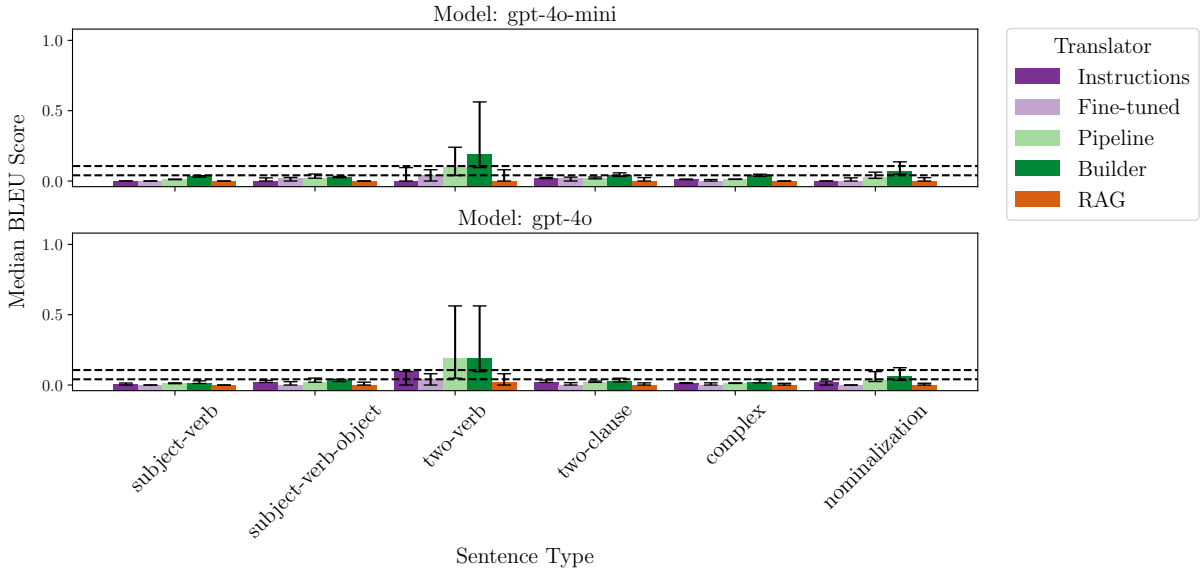


Figure 10: BLEU scores of the input sentences against the comparator translations.

- All source code for conducting and analyzing the experiments is included in a code appendix: **Yes** (data_code_appendix.zip)
- All source code will be made publicly available upon publication with a license that allows free research usage: **Yes**
- All source code implementing new methods has comments detailing the implementation, with references to the paper where applicable: **Partial** (in the Readme.md, not in the source code itself)
- If an algorithm depends on randomness, the method used for setting seeds is described: **Partial** (OpenAI API responses are nondeterministic and seeds are not controlled)
- This paper specifies the computing infrastructure used for running experiments (hardware, software libraries, OS, etc.): **Yes** (see Appendix 6)
- This paper formally describes evaluation metrics used and explains the motivation for choosing them: **Yes** (Section 4)
- This paper states the number of algorithm runs used to compute each reported result: **Yes**

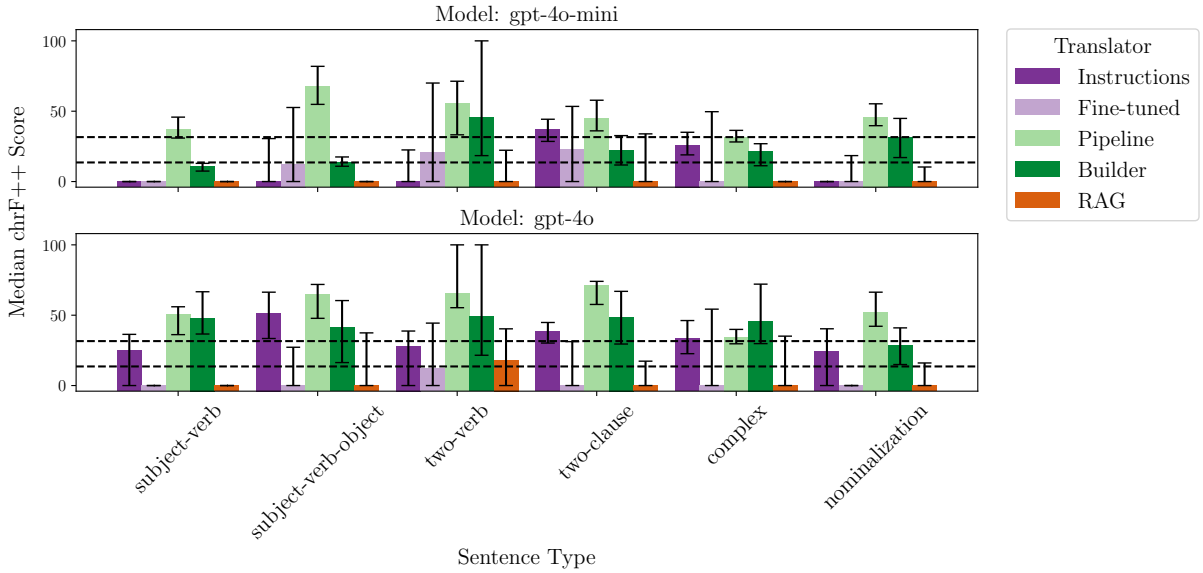


Figure 11: chrF++ scores of the input sentences against the backwards translations.

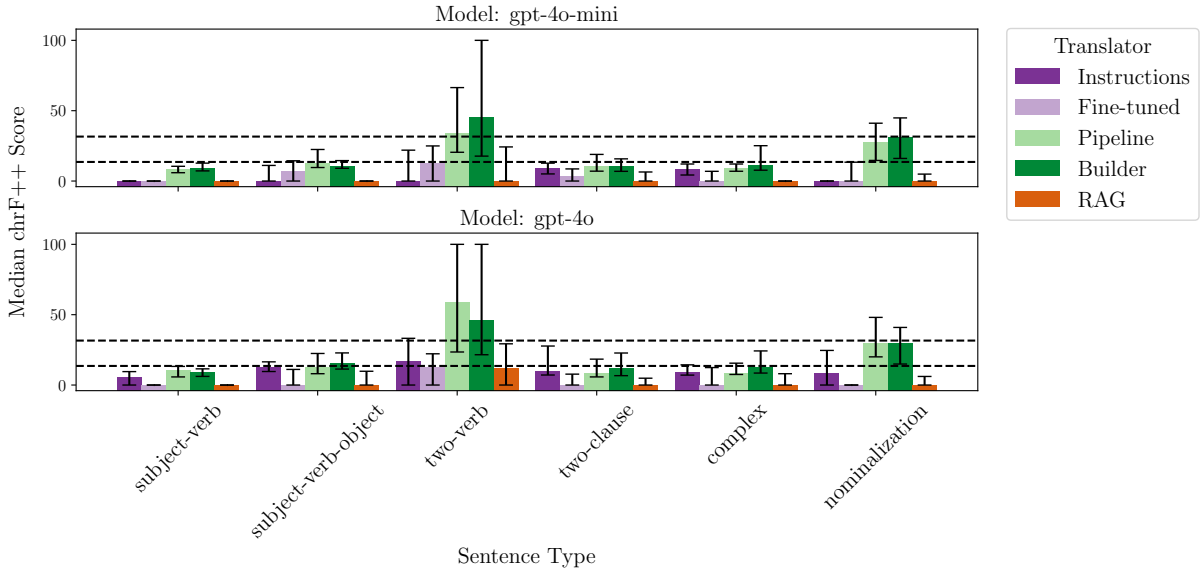


Figure 12: chrF++ scores of the input sentences against the comparator translations.

- Analysis of experiments includes measures of variation or other distributional information (e.g., error bars): **Yes**
- The significance of any improvement or decrease in performance is judged using appropriate statistical tests: **No**
- This paper lists all final (hyper-)parameters used for each model/algorithm: **Yes** (see Appendix 9)

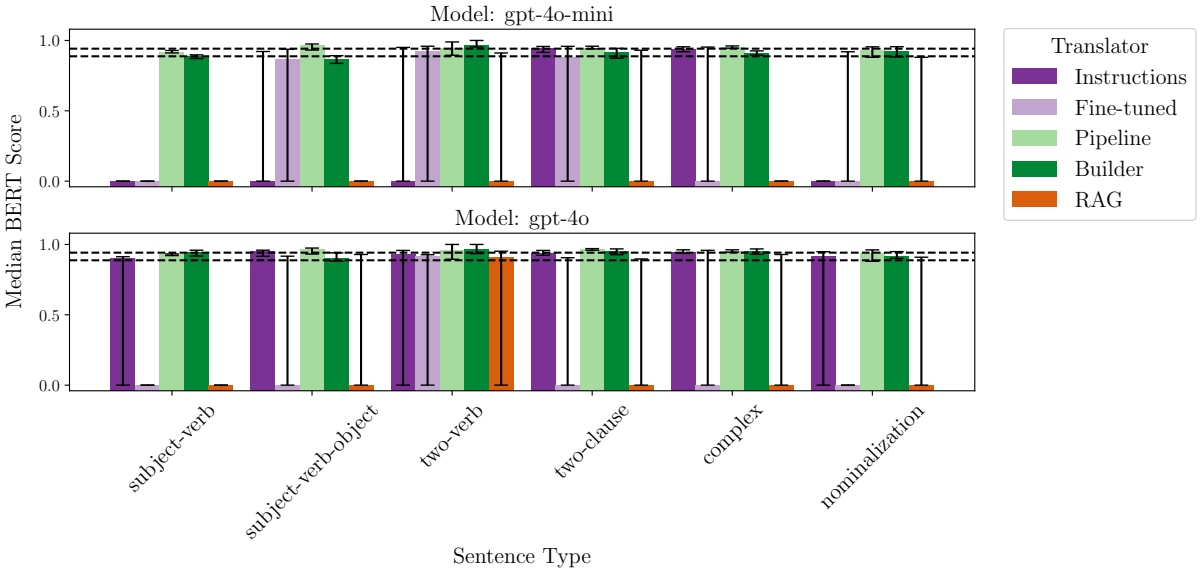


Figure 13: BERTScore of the input sentences against the backwards translations.

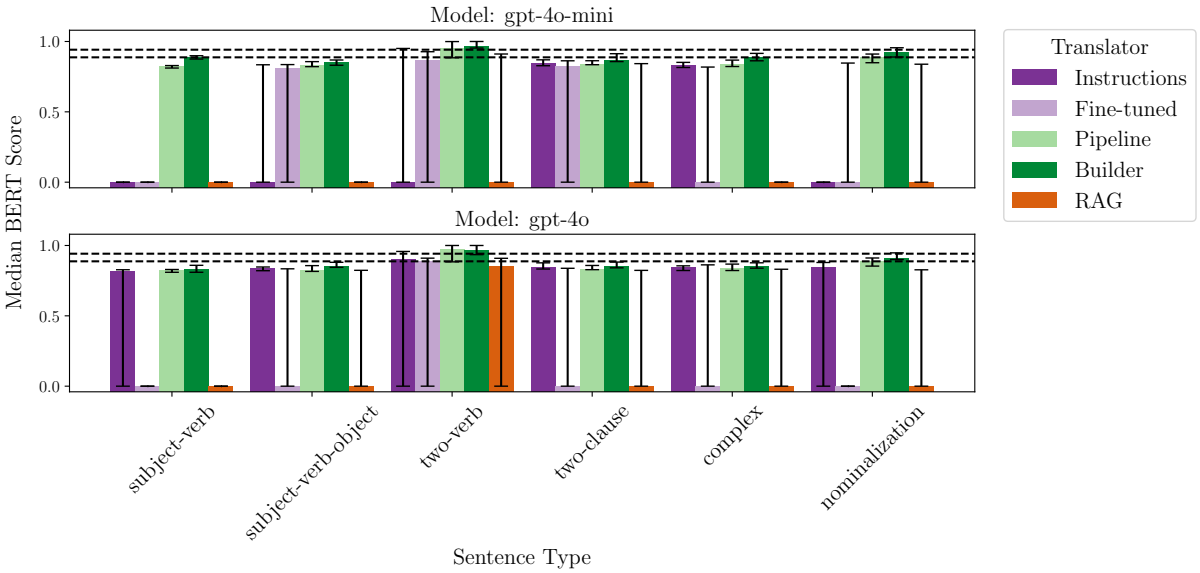


Figure 14: BERTScore of the input sentences against the comparator translations.

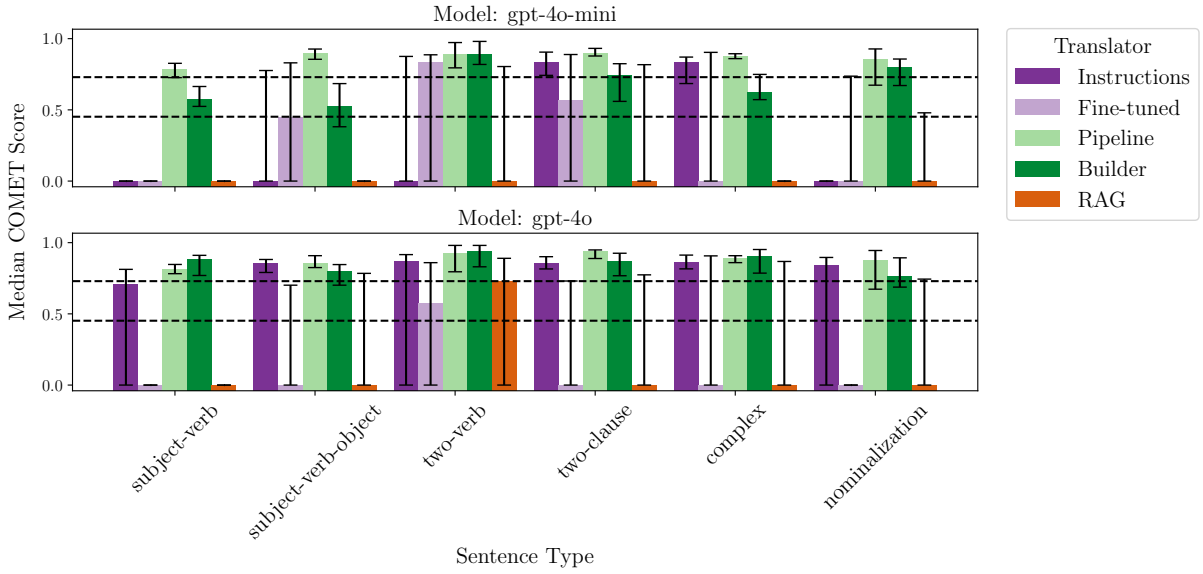


Figure 15: COMET scores of the input sentences against the backwards translations.

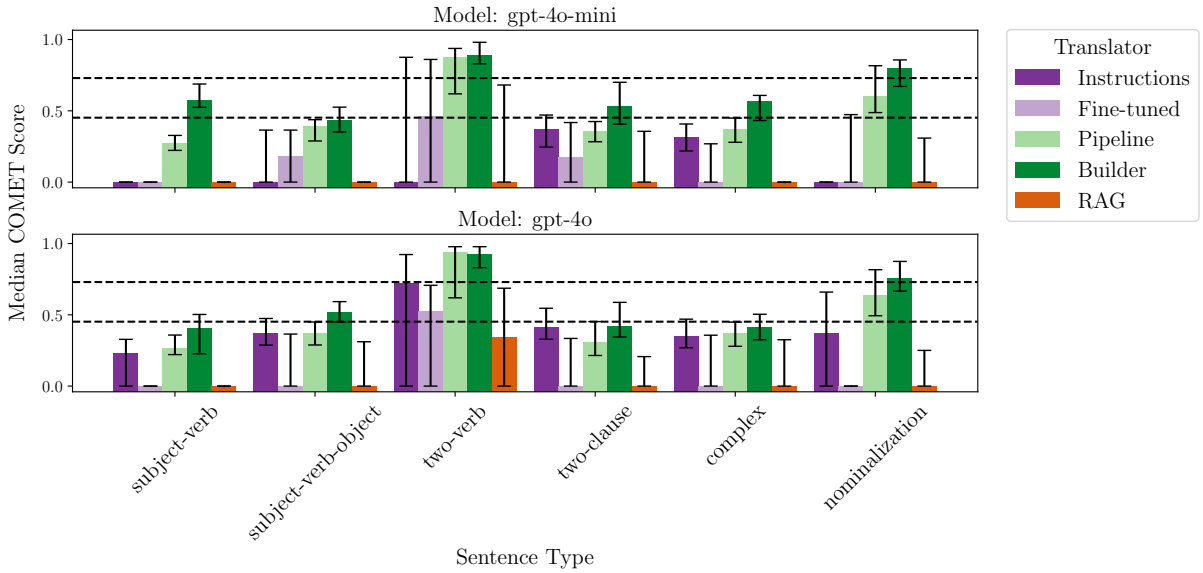


Figure 16: COMET scores of the input sentences against the comparator translations.

Can LLMs Translate Italy's Language Varieties?

Edoardo Signoroni
NLP Centre, Faculty of Informatics
Masaryk University
Botanická 68a, 602 00 Brno, Czechia

Pavel Rychlý
NLP Centre, Faculty of Informatics
Masaryk University
Botanická 68a, 602 00 Brno, Czechia

Abstract

We evaluate the capabilities of several small large language models (LLMs) to translate between Italian and six low-resource language varieties from Italy (Friulan, Ligurian, Lombard, Sicilian, Sardinian, and Venetian). Using recent benchmark datasets, such as FLORES+ and OLDI-Seed, we compare prompting and fine-tuning approaches for downstream translation, evaluated with CHRF scores. Our findings confirm that these LLMs struggle to translate into and from these low-resource language varieties. Pretraining and fine-tuning a small LLM did not yield improvements over a zero-shot baseline. These results underscore the need for further NLP research on Italy's low-resource language varieties. As the digital divide continues to threaten the conservation of this diverse linguistic landscape, greater engagement with speaker communities to create better and more representative datasets is essential to boost the translation performance of current LLMs.

1 Introduction

Recent advances in Large Language Models (LLMs) have significantly improved multilingual natural language processing, enabling high-quality machine translation and other downstream tasks for many major world languages. However, the benefits of these models are unevenly distributed, with low-resource languages remaining underrepresented in both training data and model capabilities.

Italy is home to a uniquely diverse linguistic landscape, featuring regional and minority language varieties alongside Standard Italian. While Standard Italian achieved official status with the country's unification, it became widespread only after the birth of mass media, with most of the population speaking a local language variety in everyday life. Over time, most local varieties declined, having faced not only marginalization, but also legal ambiguity and social stigma. Even if some scattered interest is re-emerging, even in NLP, more than 30 language varieties are endangered, as recognized by UNESCO. Without systematic evaluation and targeted development, speakers of Italy's local varieties are excluded from the advantages of modern language technology, contributing to cultural loss.

This work aims to address these challenges by assessing the performance of several LLMs in translating between Italian and six low-resource varieties: Friulan, Ligurian, Lombard, Sicilian, Sardinian, and Venetian. We evaluate both zero-shot, few-shot, continued pretraining, and fine-tuning approaches using recent machine translation benchmark datasets.

Our objective is to measure the translation quality of small sized LLMs for Italy's language varieties, in order to confirm current limitations of LLMs and highlight avenues for future research and resource development.

The remainder of the paper is organized as follows. Section 2 provides an overview of Italy's language varieties and their sociolinguistic context. Section 3 reviews related work in low-resource machine translation, and for the language varieties of Italy. Section 4 details our experimental setup, datasets, and models. Section 5 presents and discusses the results, followed by conclusions and future directions in Section 6.

2 Language Varieties of Italy

Italy presents one of the most diverse language landscapes in Europe (Maiden and Parry, 1997). Standard Italian was adopted as the national language only relatively recently, after the unification in 1861, at a time when it was spoken only by less than 10% of the population, mostly in Rome and Tuscany. In fact, Italian itself was a development of a literary language based on Vulgar Latin and from the Tuscan variety spoken by the Florentine upper class. In the first phases of the nation's his-

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 69–77
March 28, 2026

tory it was important for the state to standardize the local varieties to ensure the smooth operations of officials and teachers. With illiteracy remaining widespread for most of the 1800s and 1900s, the rise of education and mass media cemented the widespread use of Standard Italian, with detriment to the use of local languages.

During the Fascist era, local and minority languages were suppressed, under the pretext of them undermining central authority. A severe linguistic policy of italianization was enforced throughout the nation and its occupied territories (van der Jeught, 2016; Ramponi, 2024). Local language varieties were banned even in everyday life, teaching was allowed only in Italian, toponyms and even surnames were changed to Italian-sounding forms. For this reason, local language varieties are stigmatized as a sign of ignorance and lack of integration, and denoted with the negative-charged and linguistically improper connotation of the term *dialetti* (dialects), implying a derivative status as "dialects of Standard Italian". As Ramponi (2024) points out, the term *language varieties* is a more politically neutral denomination, preventing judgment on the prestige and status of each language.

The republican constitution of 1946 provides for the protection of linguistic minorities, but the specific implementation, through the Law 482/1999, is demanded to the local level and covers only a handful of language varieties (Albanian, Catalan, Germanic, Greek, Slovenian, Croatian, French, Franco-provençal, Friulian, Ladin, Occitan, and Sardinian). Others are protected by regional laws (Arbëreshë Albanian in Apulia and Calabria; Algherese Catalan, Gallurese, Sardinian, and Sassarese in Sardinia; German in the Walser-speaking Valle del Lys (Aosta Valley); Cimbrian, Ladin, and Mòcheno in Trentino; Calabrian Greek and Occitan (i.e., Vivaro-Alpine Gardiol) in Calabria; Francoprovençal (i.e., Faetar) and Griko in Apulia) or only recognized (Lombard, Piedmontese, and Sicilian in Lombardy, Piedmont, and Sicily, respectively; Promoted: Friulian and Slovenian in Friuli-Venezia Giulia, and Francoprovençal, French, Occitan, and Walser in Piedmont; Both: Venetian in Veneto and Ligurian Tabarchino in Sardinia) (Ramponi, 2024). This also reflects in the current status of the varieties: some are used only by very small communities of elders, while others are co-official with Italian and used in schools and education in their specific region, e.g. German and French. There exist cultural institutes that promote initiatives on language and culture for small language groups, but most often the non-officially recognized varieties are supported by politically-motivated and polarized groups.

In 2017, the ISTAT (*Istituto Nazionale di Statistica*, National Institute of Statistics) reported that 45.9% of the population mainly speak Italian at home, 32.3% use Italian and a local language, and 14.1% mostly speak a local language (ISTAT, 2017). Most local language varieties live in *dialalìa*[1] (Ramponi, 2024) with Italian, which is used in all formal and official settings, whereas the local variety are more and more confined to informal situations, overlapping with Italian even in these domains. For this reason, even if some historical literary traditions exist, most of the language varieties are primarily used in spoken and informal settings, and lack a codified written form, with the speakers improvising writing "as the words sound". When the speakers write their variety, it is often in code-switching.

Of the many language varieties present in Italy, more than 30 are endangered according to UNESCO (Moseley, 2010; Ramponi, 2024). According to Joshi et al. (2020), 10 endangered varieties are in the second-to-last position of the scale, while the rest are last (Ramponi, 2024). For consistency and ease of access, we limit our exploratory evaluation to the language varieties that appear in the OLDI-Seed (Costa-jussà et al., 2024) dataset, summarized in Table 1.

3 Related Work

In this Section, we first report on related work regarding prompting LLMs for low-resource machine translation (3.1) and then focus on specific machine translation resources for the languages we experiment with (3.2).

3.1 LLMs for Low-Resource Machine Translation

Some recent work explores and leverages LLMs prompting and adaptation for low-resource machine translation. Robinson et al. (2023) evaluates ChatGPT translation on the FLORES-200, the previous version of FLORES+. Thy find that for high-resource languages, it performs on par or ex-

[1]This is a particular form of *diglossia*, where there is not a clear functional separation between "high" and "low" varieties. The high variety, e.g. Italian, is used in formal contexts, but also encroaches in everyday informal communication, which in *diglossia* would be reserved only for the "low" variety, e.g. Lombard.

ID	Name	Branch	Conservation	Status	Standard	Speakers
fur	Friulan	Romance (Rhaeto-Romance)	DE	P, p	Y	0.6M
lij	Ligurian	Romance (Gallo-Italic)	DE	p*	Y	0.5M
lmo	Lombard	Romance (Gallo-Italic)	DE	r	N	3.5M
scn	Sicilian	Romance (Italo-Romance)	V	r	N	4.7M
srd	Sardinian	Romance (Sardinian)	DE	P, p**	Y	1.0M
vec	Venetian	Romance (^{o})	V	p	Y	3.9M

Table 1: Table of the language varieties in our experiments. For each one, the table reports its ISO-639-3 code, its name, its genealogy, the conservation status according to Moseley (2010) (Definitely Endangered [DE], Vulnerable [V], its status (Protected [P] varieties are those so by national law, whereas Recognition [r] and Promotion [p] are at regional level. For these varieties, Promotion implies recognition), if an official orthography exists, and the number of speakers (Ramponi, 2024). *Only *Tabarchino* in Sardinia **In its Gallurese and Sassarese varieties. ^{o}Disputed, either Gallo-Italic, or Italo-Dalmatian

ceeds traditional MT models, but struggles for low-resource languages, confirming that a language's resource level is the most important feature in determining ChatGPT's relative ability to translate it.

Shu et al. (2024) employ a keyword translation and retrieval method to augment translation for low-resource languages. They test their approach with GPT-4o and LLaMA 3.1 405B, which struggle in a zero-shot low-resource scenario, and outperform the baselines.

Merx et al. (2024) test LlaMA 2 70b, Mixtral 8x7B, and GPT-4 on English-Mambai translation. They explore few-shot prompting with a novel corpus prepared by a language manual and supplemented with sentences from a native speaker. They include dictionary entries, sentences, and semantic embeddings in the prompts significantly improves the translation. However, their experiments also show wide fluctuations in BLEU score across different domains.

Guo et al. (2024) devise a way to teach LLMs to translate low-resource languages by guiding them with a textbook-like approach. They assess this method on FLORES+ with ChatGPT and BLOOMZ and achieve better performance than zero-shot baselines by enhancing the models' knowledge to generate accurate and fluent sentences.

Tanzer et al. (2024) introduce a method for learning English-Kalamang translation using several pages of field linguistics reference materials, thus enabling a model to learn from a human-readable grammar book. While the experiments are promising, they still fall short of human performance. Hus and Anastasopoulos (2024) reuse the same method, leveraging the long context of GPT-4 and improve the performance of machine translation for 16 low-resource languages. Aycock et al. (2025) however, deconstruct these experiments on Nepali and Guarani, showing that the biggest contribution is from task relevant data in the book, that is parallel examples for translation, and grammatical information for linguistic tasks.

3.2 Machine Translation for the Varieties of Italy

Some work was done for the language varieties of Italy. Here we focus on datasets and tools related to machine translation for the languages we cover in this paper. For a wider survey of NLP for the language varieties of Italy, we direct the reader to Ramponi (2024).

Delmonte et al. (2009) investigate Venetian-English machine translation, between other NLP tools needed to train their statistical model, such as a PoS tagger and a rule-based system to translate Italian text into Venetian.

Wdowiak (2022) develop a Sicilian-Italian and Italian-Sicilian neural machine translation system using small subword vocabularies to train Transformers models with a high dropout. Using further techniques, such as backtranslation and multilingual translation, and the incorporation of theoretical knowledge, they manage to reach BLEU scores in the 30s.

Tyers et al. (2017) create the first machine translation system from Italian to Sardinian, using a rule-based approach. Their system achieves the stated objective of generating Sardinian text ready for post-editing. Fronteddu et al. (2017) presents a Catalan-Sardinian translator.

Signoroni (2022) describes a human-evaluated, revised, and corrected Lombard-Italian parallel cor-

pus destined to train machine translation systems. With the help of bilingual annotators, they audit an automatically aligned Wikipedia corpus from OPUS (Tiedemann, 2009).

Haberland et al. (2024) presents a Ligurian-Italian (and partially English) parallel corpus, specifically tailored for the cultural environment of Ligurian speakers. They find that using this corpus improves the performance of translation models compared to NLLB-3.3B.

Multilingual language models such as mBERT (Devlin et al., 2019) includes a tiny percentage of data from Lombard, Piedmontese, and Sicilian. However, these data are often indiscriminately collected from Wikipedia, regardless of linguistic quality or representativeness.

4 Methodology

In the following Section, we first describe the datasets we use (4.1) and then the methodology of our experiments (4.2).

4.1 Data

In our experiments, we use FLORES+ (dev and devtest) and OLDI-Seed (train) (Costa-jussà et al., 2024). Both benchmark datasets comprise professionally translated sentences (1k for FLORES+ and 6k for OLDI-Seed) from Wikipedia. In our experiment on pretraining (see Section 4.2), we use the target portion of the OLDI-Seed as monolingual pretraining data. For multilingual pretraining, we combine the data for all languages.

4.2 Experiments

Our goal is to evaluate the performance of existing small-to-mid sized LLMs when prompted for translation between Italian and several local languages of Italy. We chose to translate to and from Italian because this would be the most probable real-life application of this technology.

Prompting For prompting, we use Ollama.[2] We prompt three instruct models, Mistral NeMo (Mistral-AI, 2024), EuroLLM-9B-Instruct (Martins et al., 2024), and Qwen2.5-14B-Instruct (Qwen et al., 2025); and two reasoning models, Deepseek-R1-14B (DeepSeek-AI et al., 2025) and Phi-4-Reasoning (Abdin et al., 2024). We experiment with 0-shot, 1-shot, and 3-shots prompts both in English and Italian. Examples for the few-shot prompts are sampled from the *dev* split of FLORES+. We use default setting, but set the temperature at 0.7.

Pretraining and Fine-Tuning To investigate the effects of continued pretraining and fine-tuning fully, we limit our experiments to a small LLM, Qwen2.5-05.B-Instruct (Qwen et al., 2025), due to hardware and compute constraints. We test several pretraining and fine-tuning combinations, summarized in Table 2. We then prompt these models for translation into the low-resource varieties. For both experimental phases, we use the LitGPT library.[3]. We use a local server with a combination of NVIDIA A40 and A100. We pretrain the models on a single GPU, for 10 epochs with an learning rate of 1e-4 and a batch size of 2048 tokens. For fine-tuning, we train the models for 3 epochs, with a learning rate of 1e-4, and a batch size of 16.

To evaluate the results we compute BLEU (Papineni et al., 2002), CHRF (Popović, 2015), and Comet (Rei et al., 2020) with HuggingFace `evaluate` library. Due to the low-resource nature of the language in the experiment, we prefer CHRF over the other metrics, since Comet is not reliable for unseen languages.

5 Results and Discussion

Below, we discuss the results for prompting (5.1 and fine-tuning experiments (5.2).

5.1 Prompting

General Observations As expected, translation into Italian is ∼10-18 CHRF better than the other direction. Less expected is the sub-par performance of the *reasoning* models, trailing behind of an average ∼13 CHRF points. For both type of models and directions, *Venetian* was the easiest language for the models.

Instruct Models Among the *instruct* models, `eurollm-9b-instruct` (eurollm) is the best performing for each language and direction. Interestingly, when using English prompts, it loses performance the more shots are given (39.6 > 37.5 CHRF). Conversely, with an Italian prompt, the model retains quality, but does not improve with more shots (40.2-5 CHRF). Using more shots for `mistral-nemo:12b` (mistral) degrades its performance with both prompts, from ∼39 to ∼36 CHRF.

[2]https://github.com/ollama/ollama

[3]https://github.com/Lightning-AI/litgpt

Pretrain	Pretrain Data	Fine-Tune	Fine-Tune Data	Name
N	-	N	-	no-pt_no-ft
Y	monolingual	N	-	mono-pretrain
Y	multilingual	N	-	multi-pretrain
N	-	full	monodirectional	full-mono
Y	multilingual	full	monodirectional	multi-pretrain_full-mono
Y	monolingual	full	monodirectional	mono-pretrain_full-mono
Y	multilingual	full	multidirectional	multi-pretrain_full-multi
N	-	qlora	monodirectional	qlora-mono

Table 2: Combinations of pretraining and fine-tuning for `qwen2.5:14b-instruct`.

Figure 1: Average CHRF score for each model and language, when translating from and into Italian.

Conversely, `qwen2.5:14b-instruct` (qwen) benefits from few-shot prompts, from ∼37 to ∼39 CHRF. For the *instruct* models, Lombard is the hardest language to translate into. When translating into Italian, Sardinian is the harder to process.

Reasoning Models *Reasoning* models did not perform better than *instruct* ones. `deepseek-r1:14b` (deepseek) performs comparably to `mistral` and `phi4-reasoning:14b` (phi) is the worst performing model overall, with roughly half of the CHRF score of the other models (∼20 CHRF on average). `deepseek`'s performance stays stable with few-shot and Italian prompts, while `phi` get worse the more examples it is given. Using Italian prompts does not change its behavior.

5.2 Pretraining and Fine-Tuning

Among all the tested models, zero-shotting `qwen2.5:0.5b-instruct` gives the best results. Despite the huge difference in size, `qwen` performs almost as well as the other models for most of the languages, and even reaching `mistral` and `deepseek` on Lombard and Venetian. Just behind the baselines, we find *mono-pretrain*, that is continued pretraining with monolingual target data. Future work is warranted to see if more and more diverse data could push the model above the baseline for all languages, not only for Friulan. Full finetuning with monodirectional data is effective only for Lombard, and achieves worse performance than pretraining with other directions. When paired with pretraining, full finetuning is detrimental, as is multilingual and multidirectional data. QLoRA performs the worst overall, with `mono-pretrain_full-mono` under-performing it only for Ligurian and Venetian.

6 Conclusions

Italy has a rich and diverse linguistic landscape. However, most of its language varieties are endangered and lack modern NLP resources and tools. LLMs are being leveraged to fill the gap between high-resource and low-resource languages also for machine translation. We tested six such models for six language varieties of Italy. Our results show that LLMs alone are not enough to close the digital divide between Italian and local low-resource language varieties. These unseen languages, unlike high-resourced Italian, pose a challenge for both instruct and newer reasoning small-to-mid-sized models, leading to poor translation performance. Comparison with a smaller 0.5B model shows that bigger is not always better. Regardless of size, LLMs cannot translate what they do not see, and more high-quality monolingual data could ameliorate the issue.

Limitations

Our work is intended as a starting point for work on Italy's underresourced language variants. We do not engage in extensive parameter tuning, nor in data collection or augmentation to improve the quality of the translation. These tasks are often language-specific and thus are left to future work.

Another limitation has to do with the data. Due to the complex landscape of Italy's language varieties, the data for at least some of the languages involved, as acknowledged also by Costa-jussà et al. (2024), may not be representative of all speakers of the state variety or language, such as the case of Lombard, for example, which does not have a standardized orthography. Moreover, the data we used covered only a small fraction of the language varieties present in Italy, and only in some specific form that may or may not be completely mirrored in everyday usage.

For the same reasons, a deep qualitative evaluation was not possible due to widespread differences in orthography and lexicon.

Ethics Statement

Prompting and fine-tuning LLMs is an energy- and data-hungry endeavor. We estimate the carbon footprint of our experiments at 98.82 $kgCO_2eq$ (Lacoste et al., 2019).

Although we do not intend to deploy the systems we trained for actual, real-word usage, we note that in such cases they may produce unreliable, biased, incorrect, or not representative output, also for the complex language issues mentioned in Sections 2 and 6.

Acknowledgments

We would like to thank the reviewers for their useful comments. This work has been partly supported by the Ministry of Education, Youth and Sports of the Czech Republic within the LINDAT-CLARIAH-CZ project LM2023062.

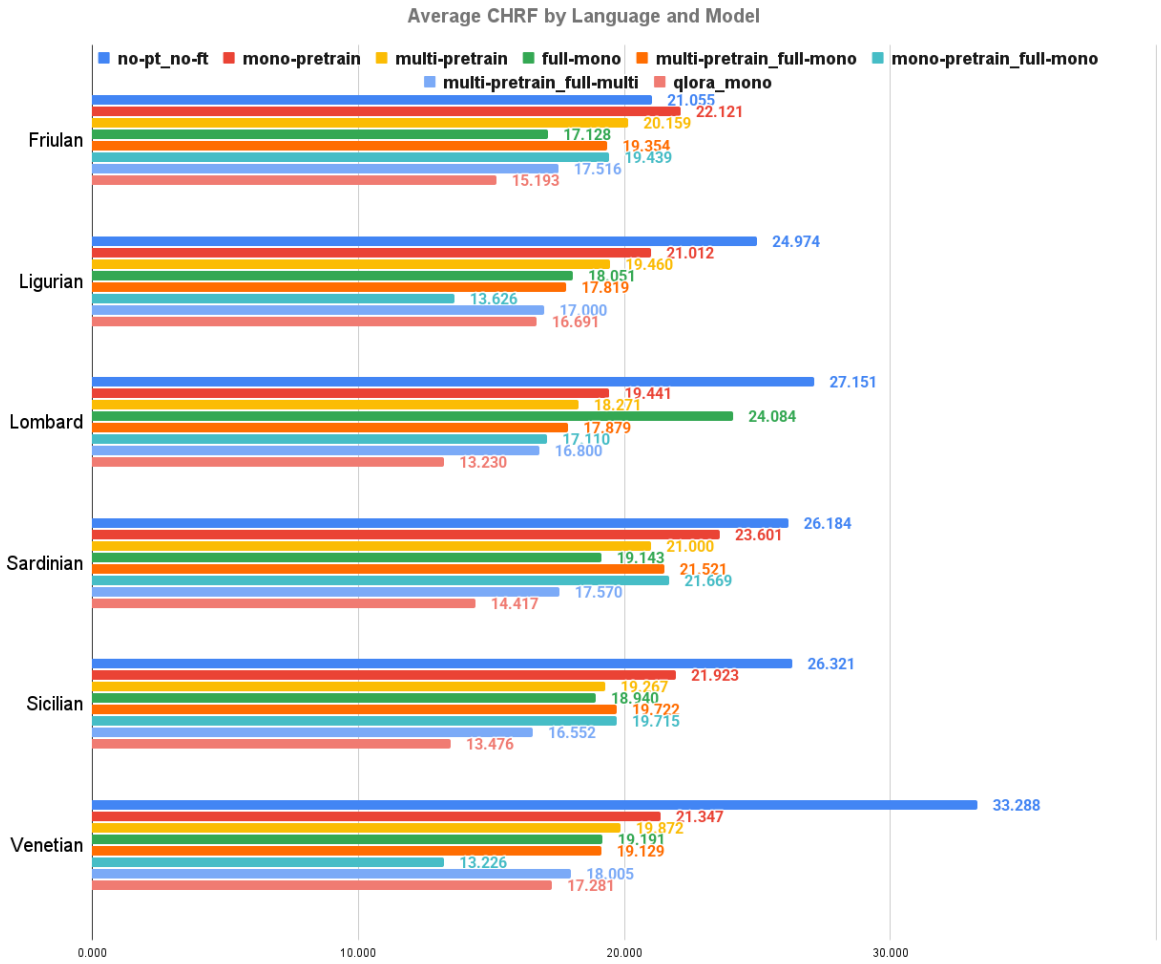


Figure 2: Average CHRF for pretrained and fine-tuned `qwen2.5:0.5b-instruct` models. *no-pt_no-ft* is the baseline model. All the models were prompted without additional examples.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. *Preprint*, arXiv:2412.08905.

Seth Aycock, David Stap, Di Wu, Christof Monz, and Khalil Sima'an. 2025. Can LLMs really learn to translate a low-resource language from one grammar book? In *The Thirteenth International Conference on Learning Representations*.

Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, and 20 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Rodolfo Delmonte, Antonella Bristot, Sara Tonelli, and Emanuele Pianta. 2009. English/veneto resource poor machine translation with stilven. In *Proceedings of ISMTCL*, pages 82–89, Besançon, France. Presses Universitaires de Franche-Comté.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Gianfranco Fronteddu, Hèctor Alòs i Font, and Francis M. Tyers. 2017. Una eina per a una llengua en procés d'estandardització: El traductor automàtic català-sard. *Linguamática*, 9(2):3–20.

Ping Guo, Yubing Ren, Yue Hu, Yunpeng Li, Jiarui Zhang, Xingsheng Zhang, and Heyan Huang. 2024. Teaching large language models to translate on low-resource languages with textbook prompting. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15685–15697, Torino, Italia. ELRA and ICCL.

Christopher R. Haberland, Jean Maillard, and Stefano Lusito. 2024. Italian-Ligurian machine translation in its cultural context. In *Proceedings of the 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages @ LREC-COLING 2024*, pages 168–176, Torino, Italia. ELRA and ICCL.

Jonathan Hus and Antonios Anastasopoulos. 2024. Back to school: Translation using grammar books. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20207–20219, Miami, Florida, USA. Association for Computational Linguistics.

ISTAT. 2017. L'uso della lingua italiana, dei dialetti e di altre lingue in italia. `https://www.istat.it/it/archivio/207961`. Accessed: 2025-08-07.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.

Martin Maiden and Mair Parry. 1997. *The Dialects of Italy*. Routledge.

Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M. Guerreiro, Ricardo Rei, Duarte M. Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, Pierre Colombo, Barry Haddow, José G. C. de Souza, Alexandra Birch, and André F. T. Martins. 2024. Eurollm: Multilingual language models for europe. *Preprint*, arXiv:2409.16235.

Raphaël Merx, Aso Mahmudi, Katrina Langford, Leo Alberto de Araujo, and Ekaterina Vylomova. 2024. Low-resource machine translation through retrieval-augmented LLM prompting: A study on the Mambai language. In *Proceedings of the 2nd Workshop on Resources and Technologies for Indigenous, Endangered and Lesser-resourced Languages in Eurasia (EURALI) @ LREC-COLING 2024*, pages 1–11, Torino, Italia. ELRA and ICCL.

Mistral-AI. 2024. Mistral nemo. `https://mistral.ai/news/mistral-nemo`. Accessed: 2025-08-07.

Christopher Moseley. 2010. *Atlas of the World's Languages in Danger*. UNESCO Publishing.

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*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Alan Ramponi. 2024. Language varieties of Italy: Technology challenges and opportunities. *Transactions of the Association for Computational Linguistics*, 12:19–38.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore. Association for Computational Linguistics.

Peng Shu, Junhao Chen, Zhengliang Liu, Hui Wang, Zihao Wu, Tianyang Zhong, Yiwei Li, Huaqin Zhao, Hanqi Jiang, Yi Pan, Yifan Zhou, Constance Owl, Xiaoming Zhai, Ninghao Liu, Claudio Saunt, and Tianming Liu. 2024. Transcending language boundaries: Harnessing llms for low-resource language translation. *Preprint*, arXiv:2411.11295.

Edoardo Signoroni. 2022. Piötòst ché nièn, mèi piötòst-a manually revised lombard-italian parallel corpus. *RASLAN 2022 Recent Advances in Slavonic Natural Language Processing*, page 105.

Garrett Tanzer, Mirac Suzgun, Eline Visser, Dan Jurafsky, and Luke Melas-Kyriazi. 2024. A benchmark for learning to translate a new language from one grammar book. *Preprint*, arXiv:2309.16575.

Jörg Tiedemann. 2009. *News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces*, volume V, pages 237–248.

Francis M. Tyers, Hèctor Alòs i Font, Gianfranco Fronteddu, and Adrià Martín-Mor. 2017. Rule-based machine translation for the italian–sardinian language pair. *The Prague Bulletin of Mathematical Linguistics*, 108:221–232.

Stefaan van der Jeught. 2016. The protection of linguistic minorities in italy: A clean break with the past. *Journal on Ethnopolitics and Minority Issues in Europe*.

Eryk Wdowiak. 2022. A recipe for low-resource nmt. In *Intelligent Computing (SAI 2022)*, pages 739–746, Cham, Switzerland. Springer International Publishing.

Balancing Fluency and Adherence: Hybrid Fallback Term Injection in Low-Resource Terminology Translation

Kurt Abela[1] **Marc Tanti**[2] **Claudia Borg**[1]

[1]Department of Artificial Intelligence, University of Malta
[2]Institute of Linguistics and Language Technology, University of Malta
{kurt.abela, marc.tanti, claudia.borg}@um.edu.mt

Abstract

Integrating domain-specific terminology into Machine Translation systems is a persistent challenge, particularly in low-resource and morphologically-rich scenarios where models lack the robustness to handle imposed constraints. This paper investigates the trade-off between *static* dictionary-based data augmentation and *dynamic* inference constraints (Constrained Beam Search). We evaluate these methods on two high-to-low resource language pairs: English-Maltese (Semitic) and English-Slovak (Slavic). Our experiments reveal a dichotomy: while dynamic constraints achieve near-perfect Terminology Insertion Rates (TIR), they drastically degrade translation quality (BLEU) in low-resource settings, breaking the fragile fluency of the model. Conversely, static augmentation improves terminology adherence on unseen terms in Maltese (4% $\rightarrow$ 19%), but fails in the context of a highly inflected language like Slovak. To resolve this conflict, we propose **Hybrid Fallback Term Injections**, a strategy that prioritizes the fluency of static models while using dynamic constraints as a safety net. This approach recovers up to 90% of missing terms while mitigating the quality degradation of pure constraint approaches, providing a viable solution for high-fidelity translation in data-scarce environments.

1 Introduction

Specialised translation domains require strict adherence to specific terminology. A primary challenge in low-resource MT is the "data-efficiency" problem: how to integrate a newly provided terminology database into a system when domain-specific parallel sentences are unavailable or extremely scarce. This paper focuses on methods that use dictionaries to improve terminology adherence without requiring massive parallel corpora. In Machine Translation (MT), this remains an open problem, especially in a low-resource scenario. While Neural Machine Translation (NMT) systems have achieved remarkable fluency, they struggle with rare, domain-specific terms, often opting for frequent synonyms or hallucinations (Koehn and Knowles, 2017; Raunak et al., 2021). This issue is exacerbated in low-resource settings, where the model's limited exposure to diverse contexts makes it resistant to learning new vocabulary and fragile when forced to use specific terms (Hasler et al., 2018).

Current approaches to terminology integration generally fall into two categories: *training-side* (static) and *inference-side* (dynamic). Static methods, such as source-side terminology injection or inline annotation, involve embedding terminology directly into the source sentences during training to bias the model's generation (Dinu et al., 2019). Dynamic methods, such as Constrained Beam Search (CBS) or Grid Beam Search, manipulate the decoding algorithm to force the inclusion of specific tokens at runtime (Post and Vilar, 2018; Hokamp and Liu, 2017).

In high-resource scenarios, dynamic constraints are often preferred for their strict enforcement of terminology. However, in low-resource settings, these constraints can be detrimental (Dinu et al., 2019). A low-resource model, when forced to include a constraint that it does not naturally predict, often sacrifices syntactic coherence, resulting in severe disfluency that satisfies the constraint but fails as a translation (Hasler et al., 2018). Conversely, static methods preserve fluency but do not ensure term inclusion (Dinu et al., 2019), particularly when dealing with the complex morphology typical of low-resource languages (e.g., Semitic or Slavic families), where a lemma-based injection may not match the required inflection (Bergmanis and Pinnis, 2021).

In this work, we explore this stability-adherence trade-off through an evaluation of English-Maltese (ENG-MLT) and English-Slovak (ENG-SLK)

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 78–86
March 28, 2026

translation in the fisheries domain.[1] Our contributions are as follows:

1. **Data Curation via Topic Classification:** For both language pairs, we established a specialised domain baseline by aligning documents from the European Parliament's Committee on Fisheries (PECH)[2] and legislative texts from EUR-Lex.[3] For the ENG-MLT pair, we significantly expanded this corpus by using a BERT-based topic classifier (Micallef et al., 2022) to filter domain-specific documents from the generic DGT Translation Memory (Steinberger et al., 2012).

2. **Benchmarking Augmentation vs. Constraints:** We demonstrate that Acontextual Drilling is effective for Maltese (raising TIR from 5% to 55%) but struggles in the Slovak context due to morphological mismatch. Conversely, we show that while CBS achieves 100% TIR, it causes a significant regression in BLEU scores (up to -6 points in some setups), confirming the fragility of low-resource models under hard constraints.

3. **The Hybrid Fallback Solution:** We introduce a pipelined decoding strategy that attempts translation via the augmented model first, falling back to constrained decoding only when specific terminology is missing. This method achieves the "best of both worlds," recovering ∼90% of terminology while largely preserving the fluency of the static model.

The remainder of this paper is organized as follows: Section 2 reviews related work in terminology constraints and data augmentation. Section 3 details our domain adaptation pipeline and the proposed hybrid strategy. Section 4 presents the comparative experimental results and a qualitative analysis of morphological barriers. Finally, Section 5 concludes the study by listing the limitations and future work.

[1]Code can be found on Github: https://github.com/MLRS/Balancing-Fluency-and-Adherence-Hybrid-Fallback-Term-Injection

[2]Documents retrieved from the European Parliament Public Register: https://www.europarl.europa.eu/committees/en/pech/documents/

[3]Access to European Union law: https://eur-lex.europa.eu/

2 Related Work

The integration of specific terminology into NMT outputs has been approached primarily through two distinct paradigms: inference-time constraints (dynamic) and training-time data augmentation (static).

2.1 Dynamic Inference Constraints

The foundational work in forcing specific lexical constraints during decoding is Grid Beam Search (GBS) by Hokamp and Liu (2017), which extends beam search to ensure the inclusion of target tokens. While effective, GBS suffers from high computational costs. Post and Vilar (2018) improved upon this with Constrained Beam Search (CBS), using a finite-state machine to track constraint satisfaction with significantly lower overhead. This is the implementation used in our experiments.

Despite their guarantees, dynamic constraints are known to be fragile. Hasler et al. (2018) observed that applying hard constraints can degrade overall translation quality if the model is not prepared to handle them. Furthermore, they highlighted that in constrained decoding, models often produce "copying" errors or syntactic breaks when the constrained term conflicts with the model's internal language model. Our work extends these observations specifically to the low-resource regime, quantifying the "fragility" of low-resource models when subjected to such constraints.

Recent shared tasks further highlight the limitations of purely hard inference-time constraints. The WMT 2025 Terminology Translation Task (Semenov et al., 2025) reports that systems relying solely on constrained decoding often struggle with fluency and document-level consistency, motivating hybrid approaches that combine constraint-aware training with selective inference-time control. These results align with our findings that strict decoding constraints, while effective for term insertion, can be brittle when models lack sufficient domain support.

2.2 Static Data Augmentation

An alternative approach involves teaching terminology via data augmentation, often referred to as source-side injection or inline annotation. Song et al. (2019); Dinu et al. (2019) demonstrated that exposing models to terminology directly in the source sentence, either by replacing source words with target translations or appending dictionaries,

can bias the model towards correct terminology usage without altering the decoding algorithm.

More recent work has revisited training-side terminology integration in light of large-scale pre-trained and instruction-tuned models. Xu and Carpuat (2021) propose soft lexical constraints that are optimized jointly with the translation objective, allowing models to trade off constraint satisfaction and fluency during training rather than enforcing hard decisions at inference time. Similarly, Kim et al. (2024) investigate efficient terminology integration for LLM-based translation systems, showing that even strong generative models benefit from explicit terminology signals when translating specialised content. These findings reinforce the view that soft or fallback-based mechanisms are better aligned with model uncertainty, particularly in low-resource or domain-shifted scenarios.

Other approaches focus on "soft constraints" using special tags for inline annotation. Bergmanis and Pinnis (2021) and Exel et al. (2020) proposed embedding target terminology directly into the source sentence (alongside the corresponding source term) using special tokens (e.g., `<term_start> term <term_end>`). This allows the model to learn a copying mechanism or a specific translation path for marked terms. While successful in high-resource settings, the efficacy of these methods in low-resource, morphologically rich scenarios remains under-explored.

2.3 Low-Resource and Domain Adaptation

Domain adaptation in low-resource NMT is notoriously difficult due to the risk of overfitting or catastrophic forgetting (Koehn and Knowles, 2017). Williams et al. (2023) established baselines for English-Maltese using generic data, while Benkova et al. (2021) investigated similar trade-offs between general and domain-specific systems for English-Slovak. Both highlight the need for specialised data filtering.

To address data scarcity, techniques like back-translation (Sennrich et al., 2016) and transfer learning are standard. However, precise terminology adaptation often requires lexical overlap that back-translation alone cannot guarantee. Recent trends use pre-trained monolingual models to filter parallel corpora for domain specificity (Aulamo et al., 2020; Zhang et al., 2020b). This approach ensures that the limited compute budget of low-resource training is spent on high-quality, relevant samples.

3 Data and Methodology

We investigate the impact of terminology integration strategies on two low-resource language pairs: English-Maltese (ENG-MLT) and English-Slovak (ENG-SLK). Both pairs involve morphologically rich target languages (Semitic and Slavic, respectively) and present distinct challenges for constraint adherence.

In our experiments, we use baseline models for each language pair to evaluate the models' knowledge of the new domain. For our experiments, we chose the **fisheries** domain (EU legislation and reports regarding maritime policy) as the new domain. The choice is made on the basis of the availability of high-quality, domain-specific terminology in the IATE database, combined with the accessibility of parallel European Parliament Committee on Fisheries (PECH) documents, which allows for a controlled evaluation of domain adaptation in a low-resource setting.

We experiment with two key techniques: (i) Static Acontextual Augmentation and (ii) Dynamic Constraint Decoding (CBS). Based on the results, we propose a third strategy, which we call Hybrid Fallback. This uses the output of Static Acontextual Augmentation to verify the presence of the required target term. If the term is missing, it falls back to re-decoding using CBS as a safety net.

To conduct the evaluation, we automatically collate a parallel corpus of European Parliament documents related to fisheries. We split the dataset for both fine-tuning and testing.

3.1 Data Curation and Domain Adaptation

We established a specialised domain dataset for both Maltese and Slovak, with curation methods varying according to resource availability, as described below.

3.1.1 Maltese (ENG-MLT) Data Setup

Generic Baseline: We trained a baseline model using the 2.9M sentence pairs from the English-Maltese parallel corpus curated by Williams et al. (2023). The corpus is heavily skewed towards formal domains, consisting of approximately 55% legal, 32% parliamentary, and 11% health-related texts, with less than 1% representing generic or informal domains. We utilized the raw text versions of the corpus to apply our own preprocessing pipeline. This involved re-tokenizing the target side using the `MLRS/BERTu` vocabulary (Micallef

et al., 2022) and filtering sentence pairs to match the sequence length constraints of our Transformer architecture.

In-Domain Data Mining: To curate a domain-specific fine-tuning set, we used a semi-supervised filtering approach. We first fine-tuned a BERT-based Maltese model, **BERTu** (Micallef et al., 2022), on the MultiEURLEX dataset (Chalkidis et al., 2021) to predict top-level domain labels based on EUROVOC descriptors (Publications Office of the European Union, 2023). This classifier, which achieved an F1 score of 77.4 (macro-average over 21 top-level domains), was applied to the generic DGT Translation Memory (Steinberger et al., 2012). Documents classified as *Fisheries* for which both the Maltese and English documents were available were collected to create the Fine-Tuning set.

Fine-Tuning Set: The mined DGT data was combined with manually aligned legislative texts (see below), yielding a robust specialised domain corpus of **96,558** sentence pairs. From this corpus, we randomly held out 2,500 pairs for validation and 2,500 pairs for testing, leaving 91,558 pairs for fine-tuning.

3.1.2 Slovak (ENG-SLK) Data Setup

Generic Baseline: We trained a baseline model on the OPUS-100 corpus (Zhang et al., 2020a), consisting of approximately 1M sentence pairs. While multilingual pre-trained models such as Tiedemann and Thottingal (2020) are publicly available and partially trained on the same dataset, we opted to train a custom bilingual model for two primary reasons. Firstly, we are using the Fairseq toolkit, thus using the same framework throughout allows for a seamless and controlled fine-tuning pipeline. Secondly, it ensures architectural parity with our ENG-MLT setup, maintaining the same Transformer configuration to ensure that observations regarding terminology adherence are not skewed by differences in model capacity. There is a strong domain overlap with this training data and our test set, since OPUS-100 contains a lot of legal data.

In-Domain Manual Curation: Unlike the Maltese setup, where we used a classifier for data selection, we relied on manual curation rather than automated mining. We aligned documents from two primary sources:

1. **Committee Drafts:** Draft reports from the European Parliament's Committee on Fisheries (PECH), retrieved from the public register[4].
2. **Legislative Texts:** Specific fisheries regulations (e.g., CELEX:52023PC0587[5]) scraped from **EUR-Lex**[6].

Fine-Tuning Set: This manual curation resulted in a high-quality but extremely sparse specialised domain dataset of **5,736** sentence pairs. Due to the data scarcity, we reserved a smaller split of 1,000 pairs each for validation and testing, resulting in a remaining training set of 3,736 pairs.

3.1.3 Terminology Dictionaries and POS Filtering

We extract terminology from the Interactive Terminology for Europe (IATE) database[7], filtering specifically for the fisheries domain. For **ENG-MLT**, we find 3,343 unique term pairs, whilst for **ENG-SLK**, we find 1,365 unique term pairs.

To investigate the impact of grammatical category on constraint stability, we processed these dictionaries using language-specific Part-of-Speech (POS) taggers. We used `bert-base-uncased` fine-tuned for POS tagging (Devlin et al., 2018) to identify and isolate nouns within the English source terms. This allowed us to create a **Nouns-Only** subset of the dictionary to assess how results would be affected by the morphologically rich nature of both language pairs. We present the comparative results between the full dictionary and this nouns-only subset in Section 4.

Test Set Annotation (Seen vs. Unseen) To test the models' capacity for generalisation (versus memorisation), we stratified the test set terminology based on exposure during training. We cross-referenced every target term required in the test sets against the respective training corpora. Terms present in the training data were classified as *Seen*, while those appearing exclusively in the test set were classified as *Unseen*. This distinction enables the calculation of **Unseen TIR** in Section 4, to serve as a metric for determining how much a model can handle terminology constraints that it has not previously memorized from the source data.

[4] https://www.europarl.europa.eu/committees/en/pech/home/highlights

[5] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52023PC0587

[6] https://eur-lex.europa.eu/homepage.html

[7] https://iate.europa.eu/

3.2 Model Architecture and Training

All translation models are based on the Transformer architecture (Vaswani et al., 2017) and were trained using the Fairseq toolkit (Ott et al., 2019). We deliberately avoid using massive pre-trained models to ensure that any improvements in terminology adherence are a direct result of our data-efficient injection strategies rather than inherited weights from a high-resource multilingual model.

Hyperparameters We use a standard Transformer configuration suitable for low-resource settings. The models comprise 6 encoder and 6 decoder layers, with an embedding dimension of 512, a feed-forward dimension of 2,048, and 8 attention heads. For training, we use the Adam optimizer ($\beta_1 = 0.9, \beta_2 = 0.98$) with an inverse square root learning rate schedule and a warmup of 4,000 steps. We apply label smoothing ($\epsilon = 0.1$) and dropout (0.3) to mitigate overfitting.

Tokenization To ensure consistent handling of morphology, we use WordPiece tokenization. For the ENG-MLT pair, we use the `MLRS/BERTu` tokenizer (Micallef et al., 2022) for the target side, as this vocabulary was used in the pre-trained Maltese NMT baseline. For the English-Slovak pair, we use the `gerulata/slovakbert` tokenizer (Pikuliak et al., 2021) for the target side to maintain methodological symmetry. For the English source side, we use the `bert-base-cased` tokenizer.

3.3 Methods for Terminology Integration

We compare two approaches to integrating terminology into the NMT pipeline and propose a third approach referred to as Hybrid Fallback Term Injection.

3.3.1 Method 1: Static Acontextual Augmentation

We adopt a list-based data augmentation strategy, often referred to as "dictionary drilling" (Exel et al., 2020). Given a terminology dictionary D, we append each term pair directly to the fine-tuning data as a distinct training example. To ensure the model attends to these short sequences during training, we apply an oversampling factor of $K = 10$, meaning that each dictionary entry is repeated 10 times and appended to the training corpus.

3.3.2 Method 2: Dynamic Constrained Decoding (CBS)

For dynamic integration, we use CBS (Post and Vilar, 2018). During inference, we identify all source terms present in the input sentence via regex matching against the IATE dictionary. The complete list of corresponding target terms is passed to the decoder as positive constraints, forcing their inclusion in the output.

3.3.3 Method 3: Hybrid Fallback Term Injection Strategy

We propose a pipelined decoding strategy to balance fluency and adherence. The system first generates a translation using the statically augmented model (Method 1). We then automatically verify the presence of the required target terms in the output.

Verification Mechanism: The verification process uses strict string matching against the dictionary term. If the augmented model generates a term that does not exactly match the glossary form, it is flagged as missing.

- If the augmented model successfully produced the terms (exact match), the translation is accepted.
- If terms are marked as missing, the system falls back to re-decoding the specific sentence using CBS (Method 2).

This strategy uses CBS as a "safety net" for stubborn terms that the static model fails to recall.

4 Results and Discussion

We evaluate the proposed methods on the specialised fisheries test sets. We report BLEU and chrF++ for translation quality. For constraint adherence, we report **Terminology Insertion Rate (TIR)** and **Unseen TIR** (performance on terms not present in the training corpus). Statistical significance is calculated using a paired t-test ($p < 0.05$).

4.1 Main Results: The Stability-Adherence Trade-off

Table 1 (ENG-MLT) and Table 2 (ENG-SLK) present the results. When analyzing Static Augmentation, a clear dichotomy emerges between the two language pairs. For Maltese, static injection significantly improves adherence, quadrupling the Unseen TIR from 4.38% to 19.06%. In contrast,

for Slovak, the method fails to yield improvements, with TIR remaining stagnant.

Meanwhile, CBS shows a consistent pattern across both languages: they achieve perfect TIR (100%) but at the cost of translation quality (BLEU). The Hybrid method consistently bridges this gap, as it was able to maintain high BLEU with a high TIR.

4.1.1 Static Augmentation

For **ENG-MLT**, the fine-tuned baseline already shows high adherence (TIR 49.73%) due to specialised further fine-tuning. Importantly, the model maintains the high translation quality of the baseline (60.21 BLEU and 76.62 chrF++) while significantly increasing TIR to 55.47% and quadrupling performance in unseen TIR (4.38% $\rightarrow$ 19.06%)

In contrast, for **ENG-SLK**, Static Augmentation fails to make an impact, with TIR remaining stagnant at $\sim$16%. While the complex morphology of Slovak presents a challenge, we attribute this failure primarily to the scarcity of specialised training data. As detailed in Section 3, the Slovak fine-tuning set consisted of only $\sim$5,700 specialised domain pairs, compared to over 96,000 for Maltese. This limited quantity appears insufficient for the model to effectively generalize the dictionary terms injected via augmentation.

4.1.2 CBS

Dynamic CBS achieves 100% TIR across the board but incurs a significant quality penalty, confirming the fragility of low-resource models under hard constraints. For **ENG-MLT** (Table 1), applying CBS to the Static model precipitates a sharp drop in BLEU from 60.21 to 53.85.

We observe an identical pattern in **ENG-SLK** (Table 2). The Slovak model suffers a comparable degradation, dropping from 61.16 to 55.80 BLEU (-5.36 points). This consistency suggests that the brittleness of constrained decoding is not language-specific but rather a symptom of the low-resource regime. When these models are forced to include tokens they cannot probabilistically support, they sacrifice local syntactic coherence, resulting in disfluent output.

4.1.3 Hybrid Fallback Term Injection Strategy

The **Hybrid Fallback Term Injection** method navigates this trade-off between translation quality and terminology adherence by utilizing CBS only as a secondary decoding pass. This approach consistently yields higher BLEU scores than pure constrained decoding (Method 2) while maintaining high TIR.

In **ENG-MLT**, the full hybrid strategy achieves 54.14 BLEU, representing a modest improvement over the 53.85 score produced by pure CBS. A more significant improvement is observed with the *Nouns-only* hybrid variant, which reaches 59.15 BLEU. Although this configuration results in a slight reduction in adherence (85.71% vs. 90.94% Unseen TIR), it demonstrates that noun-based constraints are less disruptive to the model's target-side fluency. Furthermore, the Nouns-only approach reduces the CBS fallback frequency from 26.76% to 11.16%, indicating that the static model is more likely to generate correct nominal forms naturally. These results suggest that the quality degradation observed in pure CBS is largely driven by forcing the inclusion of non-nominal terms, which require complex Semitic morphological agreement that the low-resource model cannot reliably support.

For **ENG-SLK**, the full hybrid method similarly improves translation quality over pure CBS (57.15 vs. 55.80 BLEU). The most effective results are achieved by the *Nouns-only* hybrid variant, which reaches 63.93 BLEU, notably surpassing the specialized fine-tuning baseline of 61.48. Critically, this variant maintains 100.0% Unseen TIR. The fact that restricting constraints to nouns improves both quality and adherence in Slovak suggests that non-nominal constraints often introduce syntactic conflicts that prevent the decoder from successfully placing even valid terms. By focusing on noun-based constraints, the model achieves perfect adherence with a low fallback rate (12.60%), significantly improving both decoding speed and output quality compared to pure constrained approaches.

4.2 Qualitative Analysis

To understand the source of the BLEU degradation in fully constrained models, we performed a manual error analysis on the ENG-MLT language pair (Table 3).

Firstly, we observe that the baseline model is prone to hallucination in this low-resource setting. As shown in the first example, the model hallucinates domain-specific terms such as "merluzz" (cod) and invents years in the target output even when they are absent from the source. This confirms the instability of the unaugmented low-resource model.

Table 1: ENG-MLT Results. Comparison of Baseline, Static Augmentation, CBS, and Hybrid Fallback Term Injection. Speed is measured in sentences per second (sent./sec). (†) indicates statistical significance ($p < 0.05$) compared to the Fine-tuned Baseline.

Model / Method	**BLEU**	**chrF++**	**TIR**	**Unseen TIR**	**Speed** *(sent./sec)*
Baseline model	32.11	52.21	34.38%	5.00%	43.18
Fine-tuning on specialised corpus	59.96	67.91	49.73%	4.38%	45.80
Method 1: Static Augmentation					
Static Aug. (Acontextual Drill)	**60.21**†	**68.13**†	55.47%	19.06%	**48.99**
Method 2: Dynamic Constraints (CBS)					
Static Aug. + Dynamic CBS	53.85	62.49	**100.00%**	**100.00%**	5.91
Static Aug. + Dynamic CBS (Noun Constraints Only)	58.63	66.81	**100.00%**	**100.00%**	8.28
Method 3: Hybrid Fallback Term Injection					
Hybrid (Static → CBS)	54.14	62.69	89.85%	90.94%	5.27
Hybrid (Static → CBS, Nouns Only)	59.15	66.95	83.62%	85.71%	6.81

Table 2: ENG-SLK Results. Comparison of Baseline, Static Augmentation, CBS, and Hybrid Fallback Term Injection. Speed is measured in sentences per second (sent./sec). (†) indicates statistical significance ($p < 0.05$) compared to the Fine-tuned Baseline.

Model / Method	**BLEU**	**chrF++**	**TIR**	**Unseen TIR**	**Speed** *(sent./sec)*
Baseline model	60.95	62.88	17.28%	0.71%	22.73
Fine-tuning on specialised corpus	**61.48**	**62.99**	16.71%	2.13%	**22.68**
Method 1: Static Augmentation					
Static Aug. (Acontextual Drill)	61.16	62.79	16.43%	2.13%	22.14
Method 2: Dynamic Constraints (CBS)					
Static Aug. + Dynamic CBS	55.80†	58.38†	**100.00%**	**100.00%**	5.47
Static Aug. + Dynamic CBS (Noun Constraints Only)	60.39†	62.45†	**100.00%**	**100.00%**	8.02
Method 3: Hybrid Fallback Term Injection					
Hybrid (Static → CBS)	57.15†	59.45†	83.85%	66.67%	4.42
Hybrid (Static → CBS, Nouns Only)	63.93†	65.50†	92.50%	100.00%	5.32

However, the Dynamic (CBS) approach introduces a different error type: context blindness, where the model ignores the semantic context of the source sentence to satisfy a lexical constraint. For instance, the English term "header" (in a document context) was constrained to the fisheries translation "qtugħ ir-ras" (the physical decapitation of a fish). Similarly, the term "Draft" (as in a document draft) was forced to "Pixka" (a catch of fish). Because CBS forces these terms regardless of the surrounding context, the model sacrifices semantic logic to satisfy the constraint, leading to a drop in BLEU scores.

Finally, the Hybrid Fallback Term Injection approach demonstrates better data integrity. In cases like alphanumeric codes (e.g., "COD/03AN"), where baselines often attempt to translate the substring "COD" into the fish name "Bakkaljaw," the Hybrid Fallback Term Injection approach correctly identifies when to fallback, preserving the code exactly as required.

5 Conclusion

In this paper, we investigate the challenge of integrating domain-specific terminology into low-resource MT. Our results reveal a fundamental trade-off between *stability* (translation fluency) and *adherence* (terminology usage) in data-scarce environments.

We find that standard integration methods exhibit distinct limitations. Static Acontextual Augmentation proved effective for Maltese, where fine-tuning was performed on a concentrated, specialised domain dataset. However, the same method failed for Slovak, potentially due to the smaller specialised domain fine-tuning data setup.

In contrast, Dynamic Constraints (CBS) achieved high TIR but imposed a severe penalty on translation quality. The significant regression in BLEU scores confirms that low-resource models lack the probability mass to accommodate forced tokens without compromising syntactic coherence.

To resolve this, we introduced a Hybrid Fallback Term Injection strategy. By prioritizing the fluent output of the augmented model and using constrained decoding solely as a fallback mechanism, this approach recovered up to 90% of missing terminology without the catastrophic quality loss associated with pure constraints.

Table 3: **Qualitative Error Analysis.** Examples of hallucination and context blindness.

Method	Error Type	Description
Baseline	Hallucination	**Src:** "The Commission proposes..." **Out:** "L-istokk tal-merluzz..." **Tgt:** "Il-Kummissjoni tipproponi" *Analysis:* Hallucinates "merluzz" (cod) and repeats years erroneously.
CBS	Context Blindness	**Src:** "...header of each amendment" **Out:** "...qtugħ ir-ras..." **Tgt:** "l-intestatura ta' kull emenda" *Analysis:* Forces the fisheries term for "heading" (decapitation of fish), resulting in the "decapitation of an amendment."
CBS	Context Blindness	**Src:** "Draft opinion" **Out:** "Pixka ta' opinjoni" **Tgt:** "Abbozz ta' opinjoni" *Analysis:* Forces "Pixka" (fish/catch) instead of "Abbozz" (document draft).
Hybrid	Data Integrity	**Src:** "COD/03AN" **Out:** "COD/03AN" **Tgt:** "COD/03AN" *Analysis:* Correctly preserves alphanumeric codes where baselines often attempt to translate "COD" to "Bakkaljaw" (which is the translation of the fish "Cod").

5.1 Limitations and Future Work

Computational Cost The primary drawback of the Hybrid Fallback Term Injection implementation is inference latency. Because the Static model fails to recall terms frequently (recall $\approx$ 55%), it triggers the expensive CBS fallback for nearly half the test set. As a result, it operates at a lower speed (sent./sec) than the CBS baseline in our experiments. Future work should explore lighter-weight constraint mechanisms to improve this speed-accuracy profile.

Morphological Complexity We emphasize that our current verification mechanism uses strict surface-form matching. While this allows for rigorous testing of adherence, it penalizes valid morphological inflections. Future work proposes exploring *inflection-aware data augmentation* and morphological analyzers in the verification loop to better handle the varied grammatical cases required by the target context.

References

Mikko Aulamo, Sami Virpioja, and Jörg Tiedemann. 2020. Opusfilter: A flexible tool for filtering and combining parallel corpora. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4254–4261.

Lucia Benkova, Dasa Munkova, L'ubomír Benko, and Michal Munk. 2021. Evaluation of english–slovak neural and statistical machine translation. *Applied Sciences*, 11(7):2948.

Toms Bergmanis and Marcis Pinnis. 2021. Context-independent terminology translation with neural machine translation models. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 142–153.

Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos. 2021. MultiEURLEX - a multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6974–6996, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

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*, pages 3063–3068.

M. Exel, M. Cettolo, and P. Passban. 2020. Terminology-constrained neural machine translation training data generation. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 231–240.

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*, pages 506–512.

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*, pages 1535–1546. Association for Computational Linguistics.

Sejoon Kim, Mingi Sung, Jeonghwan Lee, Hyunkuk Lim, and Jorge Gimenez Perez. 2024. Efficient terminology integration for LLM-based translation in specialized domains. In *Proceedings of the Ninth Conference on Machine Translation*, pages 636–642, Miami, Florida, USA. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39.

Kurt Micallef, Albert Gatt, Marc Tanti, Lonneke van der Plas, and Claudia Borg. 2022. Pre-training data quality and quantity for a low-resource language: New corpus and BERT models for Maltese. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 90–101, Hybrid. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Li, Ashish Ghouri, Michael Dauphin, Michael Auli, and David Grangier. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53.

Matúš Pikuliak, Štefan Grivalský, Martin Konôpka, Miroslav Blšták, Martin Tamajka, Viktor Bachratý, Marián Šimko, Pavol Balážik, Michal Trnka, and Filip Uhlárik. 2021. Slovakbert: Slovak masked language model. *Preprint*, arXiv:2109.15254.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1314–1324. Association for Computational Linguistics.

Publications Office of the European Union. 2023. EuroVoc: the eu's multilingual thesaurus.

Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. 2021. The curious case of hallucinations in neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

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.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Kai Song, Yue Zhang, Heng Yu, Weihua Luo, Kun Wang, and Min Zhang. 2019. Code-switching for enhancing nmt with pre-specified translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 449–459.

Ralf Steinberger, Mohamed Ebrahim, Alexandros Poulis, Carrasco-Benitez Manuel, Schlüter Patrick, Przybyszewski Marek, and Gilbro Signe. 2012. DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 454–459. European Language Resources Association (ELRA).

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Aiden Williams, Kurt Abela, Rishu Kumar, Martin Bär, Hannah Billinghurst, Kurt Micallef, Ahnaf Mozib Samin, Andrea DeMarco, Lonneke van der Plas, and Claudia Borg. 2023. UM-DFKI Maltese speech translation. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 433–441, Toronto, Canada (in-person and online). Association for Computational Linguistics.

Weijia Xu and Marine Carpuat. 2021. EDITOR: An edit-based transformer with repositioning for neural machine translation with soft lexical constraints. *Transactions of the Association for Computational Linguistics*, 9:311–328.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020a. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639.

Boliang Zhang, Ajay Nagesh, and Kevin Knight. 2020b. Parallel corpus filtering via pre-trained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8545–8554, Online. Association for Computational Linguistics.

Context Volume Drives Performance: Tackling Domain Shift in Extremely Low-Resource Translation via RAG

David Samuel Setiawan, Raphaël Merx, Jey Han Lau
The University of Melbourne
{david.setiawan, raphael.merx}@student.unimelb.edu.au
laujh@unimelb.edu.au

Abstract

Neural Machine Translation (NMT) models for low-resource languages suffer significant performance degradation under domain shift. We quantify this challenge using **Dhao**, an indigenous language of Eastern Indonesia with no digital footprint beyond the New Testament (NT). When applied to the unseen Old Testament (OT), which exhibits a 3x increase in OOV rate (25.9%) and distinct thematic divergence from the New Testament (NT) training data, a standard NMT model fine-tuned on the NT drops from an **in-domain score of 36.17 chrF++** to **27.11 chrF++**. To recover this loss, we introduce a **hybrid framework** where a fine-tuned NMT model generates an initial draft, which is then refined by a Large Language Model (LLM) using Retrieval-Augmented Generation (RAG). The final system achieves **35.21 chrF++** (+8.10 recovery), effectively matching the original in-domain quality. Our analysis reveals that this performance is driven primarily by the **number of retrieved examples** rather than the choice of retrieval algorithm. Qualitative analysis confirms the LLM acts as a robust "safety net," repairing severe failures in zero-shot domains.

1 Introduction

For the majority of the world's 7,000+ languages, biblical texts often represent the only available large-scale digital resource (Ranathunga et al., 2023). However, translation efforts typically prioritize the New Testament (NT), leaving the Old Testament (OT), which constitutes 75% of the Bible, untranslated. The reason is that the NT contains the theological core and is an essential starting point for a new believer or a new church. Furthermore, the linguistic complexity and size of the OT present significant translation challenges.

As of August 2025, while 2,574 languages possess a complete NT, only 776 have a complete OT (Wycliffe Global Alliance, 2025). While leveraging available NT data to train Machine Translation (MT) models for the OT is a logical next step, this workflow presents a distinct **domain shift** challenge. Despite forming a single canon, the NT and OT diverge significantly in vocabulary and style, as they originate from different source languages (Hebrew vs. Koine Greek) and cover distinct themes (historical narrative vs. theological discourse).

Our analysis of the English source text (World English Bible) quantifies this shift: the Out-of-Vocabulary (OOV) rate relative to the NT training vocabulary increases from 8.1% on the in-domain NT validation set to 25.9% on the OT test set (see Appendix A). Consequently, NMT models trained solely on the NT generalize poorly, leading to marked performance degradation (Akerman et al., 2023).

In this work, we address this NT-to-OT shift using **Dhao**, an indigenous language of Eastern Indonesia with fewer than 5,000 speakers (SIL International, 2025). Unlike existing domain adaptation work which relies on target-domain monolingual corpora (Chu and Wang, 2020; Marashian et al., 2025), we operate under a stricter constraint: the primary training data is domain-bound (NT), with the only available general-domain signal coming from a small digitized grammar book. To address this, we introduce a hybrid **NMT + LLM Post-Editing** framework. We utilize a fine-tuned NMT model to generate an initial draft and employ a Retrieval-Augmented Generation (RAG) enhanced LLM to refine the output using context retrieved from both the grammar book and the NT.

We systematically compare **sentence-level retrieval** (whole-sentence similarity) against **word-level retrieval** (aggregated source word matches) to assess their impact on the proposed hybrid pipeline. Our results indicate that while increasing context volume consistently yields gains across all strategies, the specific choice of retrieval algorithm is secondary. The final optimized system restores

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 87–101

character-level overlap (chrF++) to in-domain levels, though a gap remains in subword-level similarity (spBLEU), likely due to the higher stylistic and lexical divergence of the Old Testament. Based on these findings, we summarize our primary contributions as follows:

Contributions

1. **Zero-Shot Domain Adaptation for Unseen Languages:** We propose a hybrid NMT+LLM framework that successfully tackles domain shift for an extremely low-resource language with no digital footprint. We demonstrate that this architecture allows an LLM to correct a language it has never seen (Dhao) by leveraging the structural priors of a fine-tuned NMT model, outperforming baselines that rely on either model individually.

2. **Context Volume as the Primary Performance Driver:** We demonstrate that context volume drives LLM post-editing performance more significantly than the choice of retrieval algorithm in low-resource RAG. Our analysis shows that distinct retrieval strategies yield comparable gains when normalized for volume.

3. **Validation of the "Safety Net" Hypothesis:** We provide qualitative evidence that LLM post-editing specifically mitigates catastrophic NMT failures, such as hallucinations and repetition loops, in zero-shot domains, validating the hybrid architecture's robustness for data-scarce settings.

2 Related Work

Domain Shift in Low-Resource MT Standard NMT models are highly lexicalized, making them brittle when applied to distributions differing from their training data (Koehn and Knowles, 2017; Hu et al., 2019; Haddow et al., 2022). This brittleness often manifests as catastrophic hallucinations or fluent but unfaithful outputs when the model encounters out-of-domain data (Müller et al., 2020; Raunak et al., 2021). Specifically, such shifts have been shown to exacerbate the model's reliance on training-domain priors, leading it to ignore source constraints in favor of frequently observed sequences from the training set (Wang and Sennrich, 2020).

LLMs and the Hybrid Solution While Large Language Models (LLMs) excel at high-resource translation, they struggle with "unseen" languages due to a lack of pre-training exposure (Robinson et al., 2023; Hendy et al., 2023). Effective translation often remains unattainable for languages with underrepresented scripts even with RAG (Lin et al., 2025). However, **In-Context Learning (ICL) with language alignment** (e.g., dictionary constraints) has been identified as a viable method to unlock LLM capabilities for these languages, significantly outperforming fine-tuning which suffers from overfitting (Li et al., 2025).

To mitigate the weaknesses of both paradigms, recent literature converges on hybrid architectures. An NMT model followed by an LLM post-editor has been shown to be an optimal recipe for low-resource translation, specifically for mitigating "lexical confusion" (Nielsen et al., 2025). However, the optimal retrieval granularity for this post-editing remains an open question. While current strategies optimize for n-gram diversity (Caswell et al., 2025) or compositional phrases (Zebaze et al., 2025), word-level retrieval has also been proposed for grammatical learning (Tanzer et al., 2024). Our work synthesizes these insights: we employ the hybrid framework validated by Nielsen et al. (2025), but we systematically compare these sentence-level versus word-level strategies to determine whether performance gains stem from choice of retrieval algorithm or simply the increased volume of in-context examples.

3 Experimental Setup

3.1 Data Construction

We utilize the Dhao language resources introduced in Section 1 to construct a zero-shot domain adaptation benchmark. As Dhao lacks a standard digital footprint, we curate our datasets from the only two available sources: a Bible translation and a digitized grammar book (Balukh, 2020).

Primary Corpus (Parallel Bible) We source the parallel biblical text from the **ebible corpus** (Akerman et al., 2023). We align the **target** Dhao translation (written in **Latin script**) against the **source** World English Bible (WEB). The primary objective of this alignment is to decompose the raw verse-level text into a sentence-level parallel corpus, thereby providing the granular signal required for effective NMT training. The data is partitioned as follows:

- **In-domain (train & eval):** The complete New Testament (NT), comprising 7,644 parallel verses. We reserve 95% for fine-tuning the NMT model and 5% for in-domain validation.

- **Out-of-domain (test):** The first 500 verses of the Book of Genesis (Old Testament). Although the Old Testament is not fully translated in Dhao, a translation of Genesis exists; we utilize this text as our **ground truth** for evaluation. It serves as a strictly **unseen domain** to evaluate the model's generalization capabilities under the lexical shift described in Section 1. We verified that none of these verses appear in the supplementary grammar book, which exclusively cites examples from the New Testament (Balukh, 2020), ensuring no data leakage occurs.

Supplementary Corpus (Grammar Extraction) To support RAG-based post-editing, we extracted a supplementary corpus from *A Grammar of Dhao* (Balukh, 2020). Using a semi-automated pipeline involving PDF segmentation and LLM extraction (detailed in Appendix B), we curated a clean dataset of **1,011 parallel sentences** and **2,377 bilingual lexicon entries**. These resources represent the only available general-domain data for the language.

3.2 Models

We employ a hybrid architecture that leverages the complementary strengths of specialized NMT and general-purpose LLMs:

- **NMT (Drafting):** We use **NLLB-200-distilled-600M** (Costa-jussà et al., 2024). We specifically select this distilled version over larger variants (e.g., 1.3B or 3.3B) to prioritize faster inference speeds. This allows for rapid iteration during experimentation while still leveraging the model's massive multilingual pre-training, which provides a robust initialization for fine-tuning on the limited Dhao NT data.

- **LLM (Post-Editing):** We utilize **Gemini 2.5 Flash** (Comanici et al., 2025) for the post-editing stage. This model was selected for its large context window (enabling the ingestion of extensive RAG examples) and its cost-effectiveness for iterative experimentation.

3.3 Evaluation Metrics

Given the low-resource nature of Dhao and the lack of standardized linguistic tools (e.g., morphological analyzers or tokenizers), we report performance using two robust metrics:

- **spBLEU:** A tokenizer-agnostic BLEU score using SentencePiece (Costa-jussà et al., 2024). Since Dhao lacks a gold-standard tokenizer, spBLEU ensures that performance is measured based on learned sub-word units rather than potentially flawed rule-based tokenization.

- **chrF++:** A character n-gram metric (Popović, 2017). We prioritize chrF++ as it is strictly more robust than word-level BLEU for low-resource languages. By measuring character-level overlap, it provides partial credit for correct stems even when the model generates incorrect affixes or spelling variations, which is critical for accurately evaluating an unseen dialect like the Old Testament.

4 Methodology

We propose a hybrid translation framework that integrates specialized NMT with LLM-based post-editing to mitigate domain shift in extremely low-resource settings. While the architecture follows a standard post-editing setup, our primary contribution lies in the systematic optimization of the RAG context, when translating an unseen language with domain shift.

4.1 The Hybrid NMT-LLM Pipeline

The translation pipeline, illustrated in Figure 1, consists of two distinct phases:

Phase 1: NMT Drafting We fine-tune the NMT model described in Section 3.2 on the in-domain (NT) corpus to generate an initial hypothesis y_{nmt}. We adopt the optimal hyperparameters from the eBible benchmark (Akerman et al., 2023), as detailed in Appendix D.

Phase 2: RAG-Enhanced Post-Editing We post-edit the translation with **Gemini 2.5 Flash** (Comanici et al., 2025). The LLM receives a structured prompt containing: (1) the original source sentence x; (2) the NMT draft y_{nmt}; (3) a set of retrieved parallel sentences (sourced from both the NT and the grammar book); and (4) a set of retrieved lexicon entries formatted as direct mappings

(e.g., *English Word (POS)* → *Dhao Word*). The composition of these retrieved contexts depends on the experimental setup. We evaluate parallel sentence retrieval and lexicon retrieval independently, but combine them in the final optimized system (see Section 5.3). The full prompt structure and integration details are provided in Appendix C. The model is instructed to compare the draft against the source and selectively correct NMT failures like hallucinations or repetition loops only when necessary rather than re-translating from scratch.

4.2 Baselines

To evaluate the proposed framework, we compare against four baseline configurations evaluated on the out-of-domain (OT) test set. These baselines rely on **static retrieval** strategies, contrasting with the dynamic retrieval methods detailed in Section 4.3.

1. **NMT-Only:** The NLLB model fine-tuned solely on the NT data, serving as the lower-bound for domain adaptation.

2. **NMT + Grammar:** The NLLB model fine-tuned on the NT data augmented with the grammar book parallel sentences.

3. **LLM Direct Translation:** Gemini 2.5 translating directly from English to Dhao in 0-shot (no context) and 5-shot (5 fixed, randomly selected NT sentences) settings

4. **LLM Post-Editing (No RAG):** The hybrid pipeline without retrieval, relying on internal LLM knowledge to correct the NMT draft in 0-shot and 5-shot settings.

4.3 Retrieval Strategies (RAG)

A core contribution of this work is investigating whether performance gains in low-resource RAG stem from retrieval strategy, on the volume of examples, or both. Unlike the baselines which use static context, these strategies dynamically retrieve examples relevant to the specific input sentence x.

4.3.1 Parallel Sentence Retrieval

We retrieve relevant parallel pairs from a combined corpus of the in-domain NT and the grammar book. All retrieval operations are performed on the source side (English), bypassing the need for retrieval models trained on Dhao. We evaluate four strategies:

Sentence-Level Approaches (Fixed k) These methods retrieve a fixed number of sentences k based on their similarity to the source input. We test $k \in \{5, ..., 100\}$.

- **BM25 (lexical):** A standard sparse retrieval method that ranks sentences based on exact keyword overlap, normalized for document length.

- **BGE Embeddings (semantic):** A semantic retrieval method using `bge-large-en-v1.5` (Xiao et al., 2023). We compute the cosine similarity between the source sentence embedding and corpus embeddings to capture semantic relevance beyond keyword matching.

- **ChrF-Counterweighted (lexical, with diversity focus):** Adapted from Caswell et al. (2025), this method promotes n-gram diversity. It iteratively selects examples with high character n-gram overlap while penalizing n-grams present in previously selected examples, ensuring the context window is not filled with redundant phrasing.

Word-Level Approach (Dynamic k) Inspired by Tanzer et al. (2024), we implement a **Fuzzy Word Matching** strategy. Instead of retrieving based on the whole sentence, we retrieve the top-n parallel sentences for each word in the source sentence. We compute token similarity using the normalized Levenshtein distance via the `rapidfuzz` library, retaining only matches with similarity ≥ 0.5. Unlike sentence-level methods, the total number of examples k is **dynamic**, scaling with the sentence length ($k \approx n \times$ sentence_length). We ablate $n \in \{1, 2, 3, 5, 10, 15, 20\}$ to determine if granular, word-level context outperforms sentence-level retrieval.

4.3.2 Lexicon Retrieval

We further augment the context with bilingual dictionary entries extracted from the grammar book. We compare two configurations:

- **Fuzzy Retrieval:** Retrieving the top-n similar lexicon entries per source word. We evaluate $n \in \{3, 5, 10, 15, 20, 25, 30, 50, 70, 100\}$.

- **Full Dictionary:** Providing the entire 2,375-entry lexicon in the context window, treating the dictionary as a static resource rather than a retrieved element.

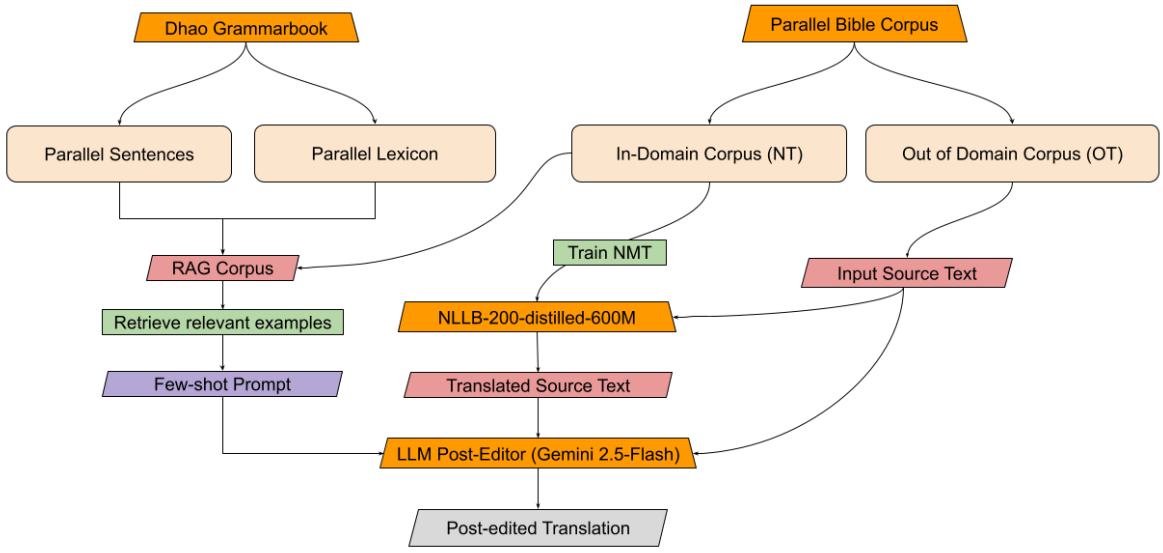


Figure 1: **Hybrid Post-Editing Architecture.** The workflow integrates two parallel streams. **Center:** The NLLB model, fine-tuned on the In-Domain (NT) corpus, generates an initial *Translated Source Text* (draft). **Left:** A Retrieval-Augmented Generation (RAG) module queries the combined corpus (Grammar Book + NT) to extract relevant examples for the *Few-shot Prompt*. **Bottom:** The LLM Post-Editor (Gemini) synthesizes the NMT draft, the original source, and the retrieved context to produce the final *Post-edited Translation*. **Legend:** Orange: Core objects (data and models) | Green: Processing steps | Red: Intermediate outputs | Purple: Prompt configuration | Gray: Final output.

5 Results

5.1 Baseline Performance & Domain Shift

We first quantify the severity of the domain shift by evaluating the fine-tuned NMT model on the out-of-domain (OT) test set. As shown in Table 1, the model suffers a performance collapse: spBLEU drops from **25.19** (in-domain NT) to **7.66** (OT), and chrF++ drops from **36.17** to **27.11**. This confirms that standard fine-tuning is insufficient for the lexical and stylistic divergence of the NT-to-OT shift.

LLMs as a Safety Net Gemini-2.5-Flash failed as a direct translator (2.98 spBLEU), confirming it possesses no prior knowledge of Dhao. However, the **Hybrid Post-Editing** framework significantly outperformed the NMT baseline (+4.88 spBLEU in the 5-shot setting). Qualitative analysis reveals that the LLM acts as a "safety net." The NMT model frequently suffers from catastrophic failures on OOV terms, such as entering infinite repetition loops. The LLM consistently identifies and truncates these loops, recovering coherent text (see Table 7).

Model	Context	spBLEU	chrF++
In-Domain Reference (NT)			
NMT (NLLB)	None	25.19	36.17
Out-of-Domain Test (OT)			
NMT (NLLB)	None	7.66	27.11
NMT + Grammar	None	7.67	26.62
LLM Direct	0-shot	2.98	18.84
LLM Direct	5-shot	7.37	22.95
LLM Post-Edit	0-shot	10.65	27.94
LLM Post-Edit	**5-shot**	**12.54**	**29.62**

Table 1: **Baseline Performance Quantification.** Comparison of NMT and LLM baselines on the out-of-domain (OT) test set. The first row shows in-domain (NT) performance as a ceiling reference. Note the severe drop when NMT is applied to the OT.

5.2 Retrieval Strategy Analysis

A core research question was whether performance gains in low-resource RAG stem from the choiceß of retrieval algorithm (e.g., semantic embeddings vs. lexical overlap) or simply the volume of in-context examples. To answer this, we decouple our analysis into two parts: first evaluating parallel sentence retrieval in isolation, and subsequently

analyzing the impact of lexicon retrieval.

5.2.1 Parallel Sentence Retrieval Analysis

Impact of Context Volume As shown in Figure 2 (Left) and detailed in Table 4 (Appendix E), all dynamic sentence-level strategies outperform the static 5-shot baseline (dashed red line) immediately, even at low k. We observe strong, consistent improvement as the context volume increases from 5 to 60, regardless of whether dense embeddings (BGE) or sparse matching (BM25) is used. However, these methods plateau around $k \approx 60$. In contrast, the Word-Level strategy (Figure 2, Right) circumvents this saturation. By retrieving granular examples, it allows the model to effectively utilize a much larger context volume, with performance continuing to scale until peaking at an effective $k \approx 137$ (see Table 5 for full numerical results).

Performance Convergence and Efficiency Trade-offs When comparing the optimal configurations of each method, we observe a convergence in peak performance. As illustrated in the bar chart in Figure 3, the maximum chrF++ scores for the Word-Level, ChrF-RAG, and BGE strategies are all within **0.5 points** of each other. The **Word-Level Fuzzy Matching** strategy achieves the absolute highest score (**35.28 chrF++**), but it outperforms the best sentence-level baseline (ChrF-RAG: 34.98) by only a small margin (+0.3).

However, efficiency analysis favors sentence-level retrieval. ChrF-RAG achieves 99% of the optimal performance with just $K = 60$ examples, whereas the word-level strategy requires nearly double the volume (≈137) for a marginal gain. This makes sentence-level retrieval the more pragmatic choice for production environments where token usage and latency are constraints.

Impact of Retrieval Corpus To validate the generalizability of our findings, we isolated the impact of the supplementary grammar data. We compared the performance of the best configuration (Word-Level, $n = 10$) using the combined corpus versus using *only* the in-domain NT for retrieval.

Results show that restricting the retrieval source to the NT corpus results in only a marginal performance drop compared to the combined corpus (from **35.28** to **35.01 chrF++**, and **18.93** to **18.47 spBLEU**). This confirms that the approach remains a viable solution for extremely low-resource languages where a Bible translation may be the *only* available digital resource, without requiring the digitization of supplementary grammar books.

Configuration	spBLEU	chrF++
In-Domain Reference (NT)	*25.19*	*36.17*
Out-of-Domain Results (Genesis)		
Baseline (NMT Only)	7.66	27.11
+ Lexicon (Full)	16.27 ↑ **8.61**	31.32 ↑ **4.21**
+ Sentences (Word-Level)	18.93 ↑ **11.27**	**35.28** ↑ **8.17**
+ Combined (Final)	**19.88** ↑ **12.22**	35.21 ↑ **8.10**

Table 2: **Experimental Results.** Comparison of component contributions. The first row provides the in-domain (NT) upper bound. Our final combined system (Word-Level Sentences + Full Lexicon) achieves **35.21 chrF++**, effectively recovering the performance lost to domain shift by nearly matching the in-domain reference of 36.17.

5.2.2 Lexicon Retrieval Analysis

Having analyzed parallel sentence retrieval, we independently evaluate the utility of the bilingual lexicon.

Impact of Dictionary Volume As shown in Figure 4, performance improves linearly with the number of entries provided, contrasting with the plateau observed in sentence retrieval. The optimal performance was achieved by providing the **Full Dictionary**, yielding **16.27 spBLEU** (+8.61) and **31.32 chrF++** (+4.21). The fuzzy matching approach with $N = 100$ closely approximated this peak (≈30.88 chrF++), suggesting that for targeted lexical information, quantity is strictly beneficial. Providing the entire lexicon maximizes the probability of retrieving precise translations for OOV terms without introducing the syntactic noise inherent in full sentences (see Table 6 in Appendix E for full results).

5.3 Final System Performance

Our final system combines the optimal parallel sentence retrieval strategy, the Word-Level Fuzzy Matching ($n = 10$, yielding an effective $k \approx 137$), with the full bilingual lexicon. As shown in Table 2, this yields the highest overall translation accuracy.

The final model achieves **35.21 chrF++**, which almost matches the in-domain performance of the NMT model (36.17 chrF++). This indicates that our RAG-enhanced post-editing framework has successfully recovered the performance lost to domain shift.

Interestingly, while the combined approach

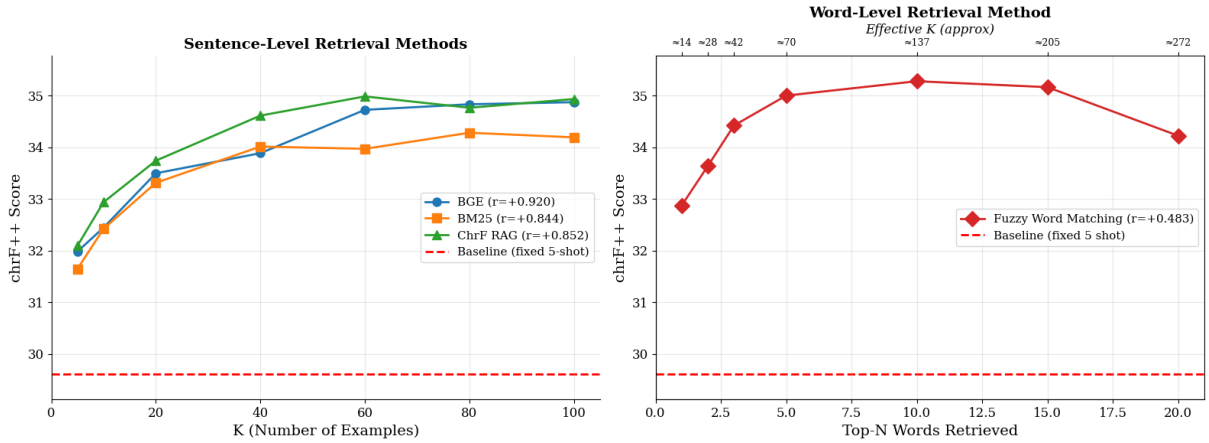


Figure 2: **Impact of Context Volume on Performance.** Comparison of absolute chrF++ scores across retrieval strategies relative to the **Fixed 5-Shot Baseline** (dashed red line, corresponding to the LLM Post-Editing baseline in Table 1). **Left:** Sentence-level methods (BGE, BM25, ChrF-RAG) show rapid initial gains but plateau at $K \approx 60$. **Right:** The Word-Level strategy allows the model to ingest a higher effective volume of examples (peaking at effective $K \approx 137$) to squeeze out marginal performance gains.

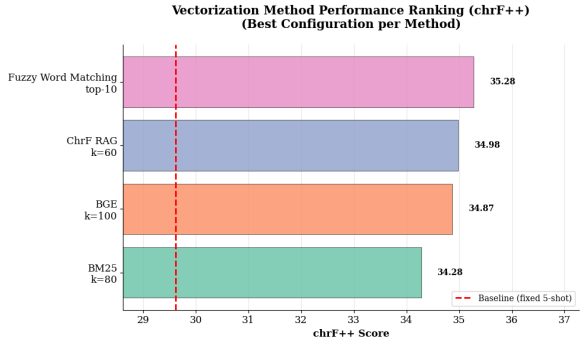


Figure 3: **Retrieval Strategy Performance Convergence.** We compare the optimal configuration of the Word-Level strategy (Fuzzy Matching, top-10) against the best configurations of three sentence-level baselines: ChrF-RAG ($k = 60$), BGE Semantic Retrieval ($k = 100$), and BM25 ($k = 80$). The bar chart displays the absolute chrF++ scores, with the dashed red line indicating the Fixed 5-shot Baseline.

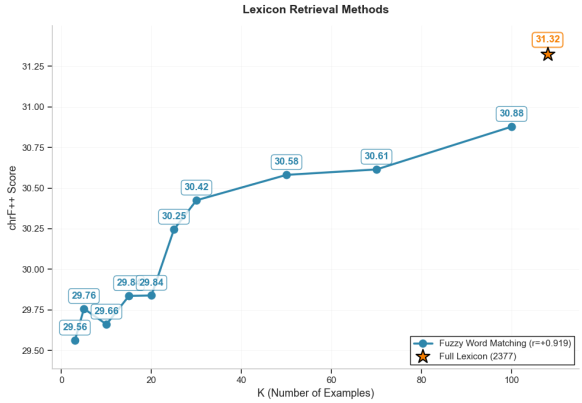


Figure 4: **Lexicon Retrieval Performance.** Impact of increasing the number of retrieved lexicon entries (K) on post-editing performance. Unlike sentence-level retrieval which plateaus, lexicon retrieval improves monotonically with volume. The highest performance is achieved by providing the **Full Lexicon** (star marker), yielding a chrF++ score of 31.32.

yields the highest spBLEU (**19.88**), the chrF++ score (**35.21**) is slightly lower than the sentence-only configuration (**35.28**). This suggests a nuanced trade-off in metric sensitivity: the lexicon provides high-precision constraints that improve exact **subword-level overlap** (boosting spBLEU), which is otherwise hindered by the 25.9% OOV rate. Conversely, the **character-level overlap** (chrF++) appears to reach saturation with sentence-level retrieval alone, nearly matching the in-domain reference of 36.17. The slight drop in chrF++ when adding the full lexicon may indicate that the increased **context volume** introduces minor syntactic noise that outweighs marginal gains in character-level accuracy. Nevertheless, the combined model remains the most robust overall system for recovering performance across both subword and character granularities.

5.4 Qualitative Analysis

To investigate the source of the performance gains, we analyzed the test set outputs. We found that the fine-tuned NMT model frequently suffers from catastrophic failures, which the RAG-enhanced LLM effectively repairs. As illustrated in Table 7 (see Appendix F), we observed three distinct failure modes:

The "Safety Net" Effect The most prevalent NMT error is the **repetition loop**, where the model gets stuck generating a single token sequence (e.g., *kahib'i-kalèbho* in GEN 10:17). The LLM demonstrates a language-agnostic ability to identify these non-sensical patterns, disregard the NMT draft, and re-translate based on the source and retrieved context.

Hallucination Correction The NMT model occasionally generates fluent but factually incorrect text. In GEN 11:5, the NMT hallucinates "King David" (*dhèu aae Daud*)—a figure frequent in the training data but absent in the source. The Post-Editor correctly identifies this mismatch against the English source ("The Lord") and the retrieved lexicon, correcting it to *Lamatua*.

Syntactic Reconstruction Finally, the LLM successfully reconstructs complex genealogical idioms that confuse the NMT. In GEN 11:17, the NMT fails to render the phrase "became the father of," likely due to the structural divergence between the literal grammar book data and the idiomatic Bible. The Post-Editor, aided by retrieved examples, correctly utilizes the Dhao idiom *matana* ('became the father of'), demonstrating the value of the word-level retrieval strategy in capturing high-frequency idiomatic patterns.

5.5 Recommendations

Based on our findings, we offer three key recommendations for practitioners working on unseen, low-resource languages:

Start with Context Volume, then look into Context Efficiency We recommend a two-step approach to RAG for low-resource MT: first, maximize the number of retrieved examples (k), as context saturation provides the most significant boost to translation quality regardless of the retrieval method. Second, tune for computational efficiency. Since our experiments show that distinct retrieval strategies converge to a similar performance band, practitioners can select the algorithm that best fits their latency constraints, switching from computationally expensive brute-force methods to faster alternatives, such as semantic search using sentence embedding models (e.g., BGE) or inverted indices (BM25), without sacrificing translation accuracy.

Consider Leaving Lexicographic Data for ICL, not Fine-Tuning Our "NMT + Grammar" baseline demonstrated that simply adding grammar book data at the fine-tuning stage can be detrimental. Performance actually degraded from 27.11 to 26.62 chrF++, likely because the rigid, pedagogical style of grammar book examples conflicts with the literary flow of the target domain. We show these resources can be best preserved as external knowledge bases for RAG, allowing the model to query specific terms dynamically without polluting the model's internal stylistic representations.

Keep an Out-of-Domain Test Set to Measure Robustness Standard practices in low-resource NMT often involve randomly splitting available corpora (e.g., the Bible) into training and test sets (Vázquez et al., 2021; Marashian et al., 2025). While many papers assume that the Bible belongs to a single, religious domain, our analysis shows a marked domain shift within this text, demonstrating that measuring out-of-domain performance is possible even when only the Bible is available as parallel corpus. Therefore, we recommend that low-resource researchers take this into consideration instead of using all verses of the Bible for both train and test, opting instead for document-level holdouts (e.g., distinct books or Testaments) to avoid inflated performance estimates (Khiu et al., 2024). This aligns with recent findings from the WMT 2025 General Translation task (Kocmi et al., 2025), which argue that evaluating on "easy" in-domain data masks model brittleness and that robust assessment requires testing on challenging, document-level out-of-domain holdouts.

6 Conclusion

This work addresses domain shift in extremely low-resource settings. We demonstrate that while standard NMT suffers catastrophic degradation on unseen domains, our proposed hybrid NMT+LLM framework functions as a robust "safety net," effectively recovering the quality lost to lexical and stylistic divergence.

Crucially, we find that context volume, rather than retrieval algorithm, is the primary driver of performance. We observe that distinct retrieval strategies (lexical vs. semantic) converge to comparable quality levels when normalized for volume. By validating these trade-offs on a language with no digital footprint, we provide a scalable blueprint for accelerating the translation of the Old Testament for thousands of low-resource languages worldwide.

7 Limitations

While this study provides a robust framework for tackling domain shift, it also highlights several limitations and clear avenues for future research:

1. **Generalizability of Language and Domain:** The experiments were conducted on a single language pair (English-to-Dhao) and a single, specific domain shift (NT-to-OT). Future work is needed to test the generalizability of this framework. It would be valuable to validate whether the superiority of word-level retrieval and the "safety net" function of the LLM post-editor hold true for other low-resource language pairs and different types of domain shifts, such as translating from religious to secular text (e.g., news or health domains).

2. **Optimizing Contextual Synergy:** As discussed in Section 5.3, our investigation into combining context types yielded mixed results. While combining the best sentence retrieval method with the full dictionary yielded the highest spBLEU score, it caused a slight decrease in the chrF++ score compared to using sentences alone. This suggests a lack of perfect synergy, likely because the large volume of combined data introduced noise. Future work should conduct finer-grained ablation studies to find the optimal balance, for instance, by combining parallel sentence retrieval with a *retrieved subset* of the lexicon rather than the full dictionary, which may reduce noise and improve both metrics.

Ethical Considerations

Data Usage and Copyright. This work utilizes data from the Dhao Alkitab (copyright ©2012 Unit Bahasa dan Budaya) and *A Grammar of Dhao* (Balukh, 2020). The Bible translation is licensed under **Creative Commons Attribution-NoDerivatives 4.0 International (CC BY-ND 4.0)**, which permits redistribution for research purposes provided the text remains unaltered. The grammar book is an **Open Access** publication (LOT Dissertation Series 570). Our use of these materials for non-commercial linguistic analysis and machine translation evaluation is consistent with these licenses and established **Fair Use** protocols for academic research. We provide full attribution to the original authors and rights holders in our citations.

Impact on Low-Resource Communities. Our primary goal is to develop technologies that support the preservation and revitalization of very low-resource languages. We recognize that AI development for indigenous languages carries the risk of extractive research practices. To mitigate this, we focus on methods that can be deployed with minimal data and computational resources, making them accessible to local stakeholders. We hope this work serves as a foundation for future community-driven language tools.

Risks of Generative Models. Neural Machine Translation and LLMs are prone to hallucinations, which poses a specific risk when handling sensitive or religious texts where accuracy is paramount. Our proposed **hybrid automated framework** (using LLMs as a post-editing safety net) is explicitly designed to identify and correct such anomalies. However, we emphasize that automated translations should always be reviewed by native speakers and community leaders before being treated as authoritative.

Acknowledgements

This research was supported by the Commonwealth through an Australian Government Research Training Program Scholarship (https://doi.org/10.82133/C42F-K220). This research was undertaken using the LIEF HPC-GPGPU Facility hosted at the University of Melbourne. This Facility was established with the assistance of LIEF Grant LE170100200. Lau was supported by the Australian Research Council under Grant LP210200917.

References

Vesa Akerman, David Baines, Damien Daspit, Ulf Hermjakob, Taeho Jang, Colin Leong, Michael Martin, Joel Mathew, Jonathan Robie, and Marcus Schwarting. 2023. The ebible corpus: Data and model benchmarks for bible translation for low-resource languages. *arXiv preprint arXiv:2304.09919*.

Jermy Immanuel Balukh. 2020. *A grammar of Dhao: An endangered Austronesian language in Eastern Indonesia*. LOT dissertation series.

Isaac Caswell, Elizabeth Nielsen, Jiaming Luo, Colin Cherry, Geza Kovacs, Hadar Shemtov, Partha Talukdar, Dinesh Tewari, Baba Mamadi Diane, Djibrila Diane, Solo Farabado Cissé, Koulako Moussa Doumbouya, Edoardo Ferrante, Alessandro Guasoni, Christopher Homan, Mamadou K. Keita, Sudhamoy

DebBarma, Ali Kuzhuget, David Anugraha, and 5 others. 2025. SMOL: Professionally translated parallel data for 115 under-represented languages. In *Proceedings of the Tenth Conference on Machine Translation*, pages 1103–1123, Suzhou, China.

Chenhui Chu and Rui Wang. 2020. A survey of domain adaptation for machine translation. *Journal of information processing*, 28:413–426.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Marta Ruiz Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, and 19 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630:841 – 846.

Barry Haddow, Rachel Bawden, Antonio Valerio Miceli Barone, Jindřich Helcl, and Alexandra Birch. 2022. Survey of low-resource machine translation. *Computational Linguistics*, 48(3):673–732.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation. *arXiv preprint*. ArXiv:2302.09210 [cs].

Junjie Hu, Mengzhou Xia, Graham Neubig, and Jaime Carbonell. 2019. Domain adaptation of neural machine translation by lexicon induction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2989–3001, Florence, Italy.

Eric Khiu, Hasti Toossi, David Anugraha, Jinyu Liu, Jiaxu Li, Juan Flores, Leandro Roman, A. Seza Doğruöz, and En-Shiun Lee. 2024. Predicting machine translation performance on low-resource languages: The role of domain similarity. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1474–1486, St. Julian's, Malta.

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 Lakougna, Jessica Lundin, Christof Monz, Kenton Murray, and 10 others. 2025. 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.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver.

Yue Li, Zhixue Zhao, and Carolina Scarton. 2025. It's all about in-context learning! teaching extremely low-resource languages to LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 29532–29547, Suzhou, China.

Dianqing Lin, Aruukhan, Hongxu Hou, Shuo Sun, Wei Chen, Yichen Yang, and Guodong Shi. 2025. Can large language models translate unseen languages in underrepresented scripts? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23148–23161, Suzhou, China.

Ali Marashian, Enora Rice, Luke Gessler, Alexis Palmer, and Katharina von der Wense. 2025. From priest to doctor: Domain adaptation for low-resource neural machine translation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7087–7098, Abu Dhabi, UAE.

Mathias Müller, Annette Rios, and Rico Sennrich. 2020. Domain robustness in neural machine translation. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 151–164, Virtual. Association for Machine Translation in the Americas.

Elizabeth Nielsen, Isaac Rayburn Caswell, Jiaming Luo, and Colin Cherry. 2025. Alligators all around: Mitigating lexical confusion in low-resource machine translation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 206–221, Albuquerque, New Mexico.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark.

Surangika Ranathunga, En-Shiun Annie Lee, Marjana Prifti Skenduli, Ravi Shekhar, Mehreen Alam, and Rishemjit Kaur. 2023. Neural machine translation for low-resource languages: A survey. *ACM Computing Surveys*, 55(11):1–37.

Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. 2021. The curious case of hallucinations in neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1172–1183, Online. Association for Computational Linguistics.

Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore.

SIL International. 2025. Ethnologue: Languages of the world. https://www.ethnologue.com/. Accessed: August 14, 2025.

Garrett Tanzer, Mirac Suzgun, Eline Visser, Dan Jurafsky, and Luke Melas-Kyriazi. 2024. A benchmark for learning to translate a new language from one grammar book. In *International Conference on Representation Learning*, volume 2024, pages 18955–18985.

Raúl Vázquez, Yves Scherrer, Sami Virpioja, and Jörg Tiedemann. 2021. The Helsinki submission to the AmericasNLP shared task. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 255–264, Online.

Chaojun Wang and Rico Sennrich. 2020. On exposure bias, hallucination and domain shift in neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3544–3552, Online. Association for Computational Linguistics.

Wycliffe Global Alliance. 2025. 2025 global scripture access. https://wycliffe.net/resources/statistics/. Accessed: 2025-12-10.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Armel Randy Zebaze, Benoît Sagot, and Rachel Bawden. 2025. Compositional translation: A novel LLM-based approach for low-resource machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 22328–22357, Suzhou, China.

A Domain Shift Analysis

To quantify the lexical divergence between the New Testament (NT) and Old Testament (OT), we analyzed the frequency of domain-specific terms and the Out-of-Vocabulary (OOV) rates.

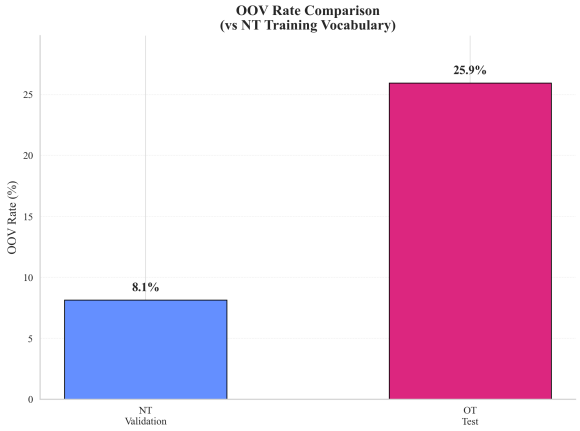


Figure 5: The Out-of-Vocabulary (OOV) rate of the in-domain NT Validation set (8.1%) versus the out-of-domain OT Test set (25.9%). All rates are calculated relative to the NT training vocabulary.

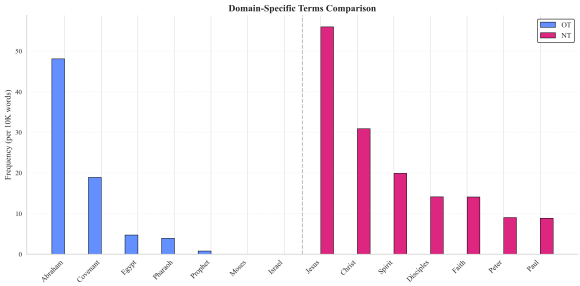


Figure 6: Domain-specific term frequencies (normalized per 10k words) comparing the Old Testament (OT) and New Testament (NT) corpora. Note the prevalence of historical terms in the OT versus theological terms in the NT.

B Data Construction Details

We illustrate the complete data processing pipeline used to curate the experimental datasets from the unstructured grammar book and the raw biblical text.

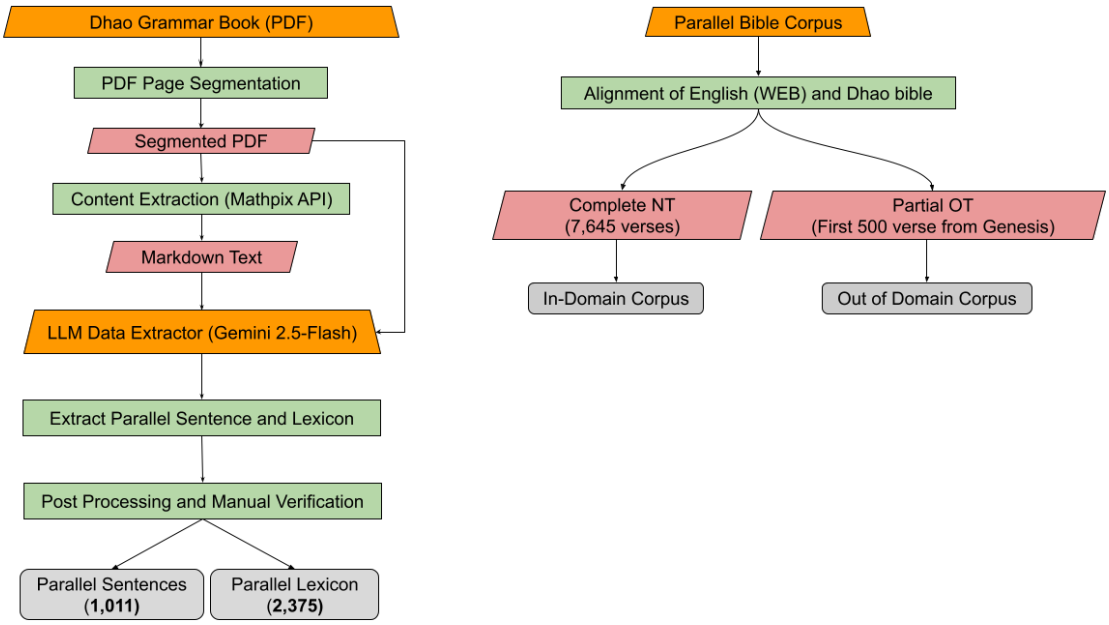


Figure 7: **Data Construction Pipeline.** The workflow processes two primary sources to create the experimental datasets. **Left:** We extract supplementary parallel sentences and lexicon entries from the unstructured Dhao Grammar Book PDF using a multi-stage pipeline involving OCR and LLM-based extraction. **Right:** The Parallel Bible Corpus is aligned and partitioned to simulate domain shift, using the New Testament (NT) as the in-domain training corpus and the Old Testament (OT) as the out-of-domain test set. **Legend:** Orange: Core objects (data and model) | Green: Processing steps | Red: Intermediate data | Gray: Final output datasets.

C Prompt Templates

To ensure reproducibility, we provide the exact prompt structures used for the Large Language Model (Gemini 2.5 Flash). We utilized two distinct prompt strategies: one for direct translation (baseline) and one for the post-editing task.

C.1 System Instructions

The following system prompts establish the persona and constraints for the LLM.

Direct Translation Prompt

System Message:
Dhao is a member of the Sumba-Flores branch of the Malayo-Polynesian language family. It is spoken in Ndao Island in the Lesser Sunda Islands in Indonesia by about 5,000 people. It is classified as a member of the Sumba branch of Malayo-Polynesian languages, but may be a Papuan language. It is also known as Ndao, Ndaonese or Ndaundau.
You are an expert Bible translator in Dhao language. Your job is to translate bible verses from English to Dhao language, providing accurate and faithful translations that maintain the meaning and context of the source text. When provided with glossary entries or example translations, use them as reference to help ensure correct translation. You must respond ONLY with your translation in Dhao - no explanations, no reasoning, no additional text.

User Template:
Source text (English): {src_text}
Translate the above text from English to Dhao:

Post-Editing Prompt

System Message:
Dhao is a member of the Sumba-Flores branch of the Malayo-Polynesian language family. It is spoken in Ndao Island in the Lesser Sunda Islands in Indonesia by about 5,000 people. It is classified as a member of the Sumba branch of Malayo-Polynesian languages, but may be a Papuan language. It is also known as Ndao, Ndaonese or Ndaundau.
You are an expert Bible translator in Dhao language. Your job is to correct and verify machine generated bible verses in Dhao language which is translated from the English language. Only make changes when necessary, ensuring that the post-edited dhao verse is aligned with the source English verse. When provided with glossary entries or example translations, use them as reference to help ensure correct translation. You must respond ONLY with the corrected translation text - no explanations, no reasoning, no additional text.

User Template:
Source text (English): {src_text}
Machine translation (Dhao): {pred_text}
Correct the machine translation if necessary:

C.2 Dynamic Context Injection

Depending on the retrieval strategy, the following blocks are dynamically appended to the User Template.

Dynamic Component: Parallel Sentence Examples

(Appended when $k > 0$ parallel sentences are retrieved)
To help with the translation, here are some example parallel sentences between Dhao and English:
Dhao: [target_example_1]
English translation: [source_example_1]
Dhao: [target_example_2]
English translation: [source_example_2] ...

Dynamic Component: Glossary Entries

(Appended when Lexicon Retrieval is enabled)
To help with the translation, here is a word list between English and Dhao in the format: English word (pos tag) -> Dhao word:

- source_word_1 (noun) -> target_word_1
- source_word_2 (verb) -> target_word_2
- source_word_3 -> target_word_3

D Hyperparameters and Training Details

All NMT models were fine-tuned using the Hugging Face `transformers` library on a single NVIDIA A100 GPU. We utilized the `facebook/nllb-200-distilled-600M` pre-trained checkpoint. Table 3 summarizes the hyperparameters used for both the baseline (NMT-Only) and the augmented (NMT + Grammar) configurations.

Note that the *NMT + Grammar* model was trained for more steps (7,000 vs 5,000) to account for the additional training data provided by the grammar book CSV.

Hyperparameter	NMT-Only	NMT + Grammar
Base Model	NLLB-200-distilled-600M	
Precision	bfloat16	
Attention Impl.	SDPA (Scaled Dot Product Attention)	
Learning Rate	2e-4	
Label Smoothing	0.2	
Warmup Steps	1,000	
Early Stopping Patience	4	
Batch Size (per device)	16	
Grad. Accumulation Steps	4	
Effective Batch Size	64	
Max Sequence Length	400	
Beam Size (Eval)	2	
Max Training Steps	**5,000**	**7,000**

Table 3: Fine-tuning hyperparameters for the NMT baselines.

E Detailed Retrieval Results

We provide the complete numerical results for the retrieval ablation studies discussed in Section 5.2. Table 4 details the performance of sentence-level strategies (BM25, BGE, ChrF-RAG) across varying context sizes (K). Table 5 details the word-level fuzzy matching strategy across varying retrieval densities (N). Finally, Table 6 presents the results for Lexicon Retrieval, comparing dynamic retrieval against the static full-dictionary baseline.

Method	K	spBLEU	chrF++
Baseline			
NMT (NLLB)	-	7.66	27.11
BGE (Semantic)			
BGE	5	15.23	31.98
BGE	10	15.71	32.45
BGE	20	16.97	33.50
BGE	40	17.34	33.88
BGE	60	18.34	34.72
BGE	80	18.32	34.83
BGE	100	18.47	34.87
BM25 (Lexical)			
BM25	5	14.76	31.65
BM25	10	15.50	32.42
BM25	20	16.53	33.31
BM25	40	17.44	34.01
BM25	60	17.34	33.97
BM25	80	17.51	34.28
BM25	100	17.25	34.19
ChrF-RAG (Diversity)			
ChrF	5	15.33	32.09
ChrF	10	16.32	32.94
ChrF	20	17.01	33.74
ChrF	40	18.05	34.61
ChrF	**60**	**18.44**	**34.98**
ChrF	80	18.12	34.76
ChrF	100	18.27	34.93

Table 4: Detailed ablation results for **Sentence-Level** retrieval methods. Note that performance gains tend to plateau around $K = 60 - 80$ for most methods.

N (per word)	Eff. K	spBLEU	chrF++
Word-Level Fuzzy Matching			
1	≈14	15.96	32.87
2	≈28	16.94	33.64
3	≈42	17.72	34.42
5	≈70	18.72	35.00
137	**≈137**	**18.93**	**35.28**
15	≈205	18.82	35.16
20	≈272	16.90	34.22

Table 5: Detailed ablation for the **Word-Level** strategy. "Eff. K" denotes the approximate effective number of sentences retrieved. Performance peaks at $N = 10$ before degrading due to context noise.

N (per word)	Eff. K	spBLEU	chrF++
Word-Level Fuzzy Matching			
3	≈77	13.84	29.56
5	≈128	14.29	29.76
10	≈256	14.18	29.66
15	≈384	14.53	29.84
20	≈512	14.85	29.84
25	≈640	15.19	30.25
30	≈768	15.19	30.42
50	≈1280	15.29	30.58
70	≈1791	15.76	30.61
100	≈2559	16.22	30.88
Static Context			
Full Dictionary	**2377**	**16.27**	**31.32**

Table 6: Detailed ablation for Lexicon Retrieval. Unlike sentence retrieval, performance scales monotonically with volume, peaking when the Full Dictionary is provided.

F Qualitative Analysis Examples

We provide concrete examples of the failure modes discussed in Section 5.4. Table 7 highlights three specific instances where the NMT baseline failed catastrophically on the Out-of-Domain test set, and how the RAG-enhanced Post-Editor recovered the correct translation.

Type	Source (English)	NMT Output (Draft)	LLM Post-Edit (Final)
Repetition Loop	*"it rained on the earth forty days and forty nights"* (GEN 7:12)	*Hèia bèli-camèd'a bèli-camèd'a...* (repeats indefinitely)	*Èj'i bhori ètu rai èpa nguru lod'o mèu-mèda.* (Correct translation)
Hallucination	*"the lord came down to see the city..."* (GEN 11:5)	*...dhèu aae **Daud** puru...* ("King **David** came down...")	***Lamatua** puru nèti dedha mai...* ("**The Lord** came down...")
Complex Syntax	*"...after he became the father of peleg"* (GEN 11:17)	*Nèti èèna ka, Eber mamuri toke d'ai... Èle èèna ka, nèti èèna ka, nèti èèna ka...* (Degenerates into repetition loop)	*...èli nèngu **matana** Peleg...* (Recovers genealogical idiom)

Table 7: Examples of NMT failures corrected by the RAG-enhanced Post-Editor. The LLM acts as a safety net, fixing repetition loops, named entity hallucinations, and recovering complex idioms.

Building and Evaluating a High Quality Parallel Corpus for English Urdu Low Resource Machine Translation

Munief Hassan Tahir, Hunain Azam, Sana Shams, Sarmad Hussain
Center for Language Engineering, Al-Khawarizmi Institute of Computer Science
University of Engineering and Technology, Lahore, Pakistan
munief.hassan@kics.edu.pk, l216202@lhr.nu.edu.pk,
sana.shams@kics.edu.pk, sarmad.hussain@kics.edu.pk

Abstract

Low-resource languages like Urdu suffer from limited high quality parallel data for machine translation. We introduce a curated English–Urdu corpus of 80,749 high-fidelity sentence pairs across 18 diverse domains, built via ethical collection, manual alignment, deduplication, and strict length-based filtering (AWCD $\leq$ 5). The corpus is converted into a bidirectional SFT dataset with bilingual (English/Urdu) instructions to enhance prompt-language robustness. Fine-tuning Llama-3.1-8B-Instruct (Llama-FT) and UrduLlama 1.1 (UrduLlama-FT) yields major gains over the baseline. sacreBLEU scores reach 24.65–25.24 (En→Ur) and 76.14–77.97 (Ur→En) for Llama-FT, with minimal sensitivity to prompt language. Blind human evaluation on 90 sentences per direction confirms substantial perceptual improvements. Results demonstrate the value of clean parallel data and bilingual instruction tuning, revealing complementary benefits of general SFT versus Urdu specific pretraining. This work provides a reproducible resource and pipeline to advance Urdu machine translation and similar low-resource languages.

1 Introduction

In an increasingly interconnected world, machine translation (MT) has become a fundamental component of natural language processing (NLP), facilitating smooth cross-lingual communication. Low-resource languages like Urdu, one of the most widely spoken languages in the world with over 230 million speakers (SIL International, 2022), remain woefully underserved, whereas high-resource language pairs like English–French or English–Chinese have profited from decades of research and enormous parallel corpora. Modern neural machine translation systems face significant obstacles due to Urdu's intricate morphology, right-to-left Nastaliq script, rich code-mixing with Persian and Arabic loanwords, and scarcity of high-quality parallel data. The public resources currently available for translating between English and Urdu are either domain-restricted, noisily aligned, or insufficient in scale, which leads to models that are culturally misaligned, have poor generalization, and have lexical sparsity. This study fills these gaps by presenting a large-scale curated English–Urdu parallel corpus that includes 80,749 high-fidelity sentence pairs from 18 standardized domains. The dataset guarantees linguistic equivalency, traceability, and suitability for neural MT architecture training and evaluation. It is constructed through a rigorous pipeline of ethical data acquisition, structural and manual sentence alignment, and multi-stage preprocessing. In addition to being larger and cleaner than previous English–Urdu resources, this corpus creates a repeatable framework for creating domain-balanced, ethically sourced datasets for other low-resource languages.

2 Related Work

FConv-NN introduces a simple yet effective method to improve Urdu–English (UR-EN) neural machine translation in low-resource settings (Israr et al., 2024). The model was trained and evaluated using FAIRSEQ, an open-source PyTorch toolkit, on a system with an Intel Core i9-9900K CPU (3.60 GHz) and an NVIDIA GeForce GTX 1650 SUPER GPU. Experiments on a 100K Urdu–English corpus (90K train, 5K validation, 5K test) showed the FConv-NN model achieved a BLEU score of 40.22, a 18.42 point gain over baseline CNN and a BLEURT score of 0.565, outperforming CNN and CNN-ADL across all metrics. It also produced more fluent translations than Google and Bing. The model enables efficient, real-time translation in low-resource settings but still trails RNN-based models, highlighting the need for improved hybrid architectures.

The English–Urdu NMT model employs a three-

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 102–110

layer LSTM encoder-decoder with a sequence-to-sequence architecture, integrating preprocessing steps (noise removal, tokenization, POS-tagging) and Word2Vec Skip-Gram embeddings to manage morphological variations and out-of-vocabulary words without segmentation (Andrabi and Wahid, 2022). Trained with the Adam optimizer and Soft-Max loss on a cleaned parallel corpus prepared via Python tokenization and Selenium web scraping, the model used a fixed sequence length of 15, achieving optimal performance at sequence length 10. It attained BLEU scores of 50.86 (training) and 47.06 (testing), showing strong context retention for short sequences. Human evaluations rated translations 53% excellent and 38% good, confirming the model's fluency and reliability for English–Urdu translation.

The English–Urdu NMT system uses a six-layer LSTM encoder-decoder with Bahdanau attention and GloVe embeddings to enhance context retention and translation quality (Kumhar et al., 2022). Extensive preprocessing—truecasing (English), Unicode normalization, non-printable character removal, and sentence padding—addresses Urdu's complex morphology and script alignment. Trained on Google Colab using the Adam optimizer and categorical cross-entropy, the system utilized a parallel corpus of 1,083,734 tokens (542,810 English, 540,924 Urdu) from religious texts, web-scraped news, and daily-use phrases, split 70:30 for training and testing. It achieved an average BLEU score of 45.83, though accuracy remains limited in specialized domains (e.g., health, tourism, business) and the system supports text-only translation, lacking speech input.

Expectation Maximization (EM) based transliteration is an unsupervised, language-independent technique proposed to enhance Urdu-to-English translation by learning transliteration patterns and handling out-of-vocabulary (OOV) words without a separate dataset (Mohy ud Din, 2019). Using the UMC005 Quran-based parallel corpus of 6,414 sentence pairs, the approach achieved BLEU score improvements of 0.63–0.91 for SMT and 1.28–2.05 for NMT, with the Transformer model outperforming LSTM (11.61 vs. 9.08). Tokenization and preprocessing further improved results by +3.5 BLEU points. However, the study was limited by the lack of a large, high-quality Urdu–English parallel corpus, emphasizing the need for better data resources and exploration of unsupervised translation methods for low-resource languages like Urdu.

A model for English to Urdu and Hindi machine translation using translation rules and artificial neural networks presents a hybrid multilingual translation framework that integrates rule-based methods with artificial neural networks (ANN) to translate English text into Urdu and Hindi (Khan and Usman, 2019). The system employs translation rules and bilingual dictionaries within an encoder-decoder architecture implemented in Java and MATLAB, where linguistic rules and tokens are numerically encoded for neural processing. Trained on approximately 465 grammar rule pairs and 9,000 bilingual word entries for each language pair, the model achieved strong results, with BLEU 0.6054, METEOR 0.8083, and F-score 0.8250 for Urdu translations. The study emphasizes that expanding grammar rules, enriching the bilingual dictionary, and refining case marking are key to enhancing accuracy in Urdu and Hindi translation.

The study "Linguistics Knowledge-Driven Multi-Task (LKMT) Neural Machine Translation for Urdu and English" (Hassan et al., 2024) presents a novel pre-trained model that enhances Urdu–English translation by integrating linguistic knowledge such as part-of-speech (POS) and dependency (DEP) features into a Transformer-based architecture. Trained on a large monolingual corpus of 5,464,575 sentences and fine-tuned using the Tanzil and religious Urdu–English parallel corpora, the model effectively captures deeper lexical and syntactic relationships. Experimental results show BLEU score improvements of +1.97 for Urdu→English and +2.42 for English→Urdu over previous models like XLM and mBART. While demonstrating strong performance for low-resource languages, the study notes that limited high-quality Urdu parallel data remains a key challenge. Future work aims to incorporate richer linguistic and semantic features to further enhance translation quality and domain adaptability.

The paper "Enriching Source for English-to-Urdu Machine Translation" (Jawaid et al., 2016) proposes adding artificial case markers (pseudo-words) to the English source text to improve phrase-based SMT performance when translating into Urdu, a morphologically rich and free word-order language. Using the Moses framework, GIZA++, and a 5-gram SRILM model, the system was trained on the "ALL" parallel corpus and supported with Urdu monolingual data. By preordering English sentences to match Urdu syntax and inserting pseudo-markers to represent grammatical roles, the

approach improved alignments and translation accuracy, achieving up to +1 BLEU gain over the baseline. However, occasional over-generation of markers caused inconsistent results, indicating a need for further refinement.

The paper "Advancing Roman Urdu to Urdu Transliteration using Machine Learning Techniques" (Ahmad and Ahmad, 2024) uses a diverse dataset of 6.5 million sentences from social media, messaging, and poetry to train and evaluate transliteration models. The authors compared Seq2Seq, RNN+LSTM, and Tensor2Tensor attention-based models, finding the Transformer-based approach achieved the best performance with a BLEU score of approximately 75. Training was conducted on an Alienware 15R2 laptop with a Core i7 processor, 32 GB RAM, and 8 GB GPU memory, using four attention layers, dropout, and subword tokenization for effective contextual understanding. While the Transformer model accurately handled complex, compound, and long sentences, limitations included occasional errors with rare or unseen words and reliance on high computational resources, highlighting the importance of data diversity, model architecture, and context-aware design in Roman Urdu transliteration.

The paper "Low-Resource Transliteration for Roman-Urdu and Urdu Using Transformer-Based Models" (Butt et al., 2025) addresses transliteration challenges between Roman-Urdu and Urdu using a transformer-based m2m100 model. The authors employ two datasets: the large-scale Roman-Urdu-Parl (RUP) corpus with 6.3 million sentence pairs and the smaller, domain-specific Dakshina dataset with 10,000 sentences. Their methodology includes Masked Language Modeling (MLM) pretraining on monolingual Roman-Urdu and Urdu text to improve subword-level generalization, followed by fine-tuning for direct transliteration with language-specific tokens. Models were trained using standard transformer settings, including batch sizes of 64–128, learning rate 1e-5, and gradient accumulation, on hardware provided by DFKI. Results show that the fine-tuned m2m100 model achieves Char-BLEU scores up to 97.44 for Roman-Urdu $\rightarrow$ Urdu and 96.37 for Urdu $\rightarrow$ Roman-Urdu, outperforming previous RNN baselines and GPT-4o Mini in zero-shot evaluation. Limitations include inherent ambiguity in Roman-Urdu spelling, trade-offs between domain adaptation and retention of previously learned patterns, and reliance on parallel corpora, highlighting potential avenues for multi-reference evaluation or semi-supervised approaches.

The paper "Urdu-to-English-Based Unsupervised Machine Translation" (Raza et al., 2024) addresses the challenge of translating between a low-resource language pair with significant structural differences, where Urdu follows a Subject-Object-Verb (SOV) order and English uses Subject-Verb-Object (SVO). To overcome the scarcity of parallel corpora, the authors proposed an unsupervised neural machine translation model trained solely on monolingual data from the Tanzil Corpus. The system employs a Transformer-based architecture with a shared encoder, leveraging cross-lingual embeddings, denoising autoencoders, and on-the-fly backtranslation to optimize translation quality. Experimental results showed that the unsupervised approach achieved a BLEU score of 5.21, while a supervised variant reached 13.85, indicating improvements but also highlighting the challenges posed by structural differences, idiomatic expressions, and morphological complexity.

The Transtech system (Masroor et al., 2019) is a rule-based Roman Urdu to English translator designed to handle variable spelling and grammar. It operates in three phases: a scanner for tokenization and spell checking, a POS tagger using an LL(1) parser and context-free grammar, and a translator module for semantic analysis and sentence generation. The system uses a self-constructed corpus of over 3,000 words and 2,000 sentences, supported by a knowledge base storing linguistic and syntactic information. Evaluation against Google Translator showed Transtech provides more accurate translations, especially for word order, tense, and pronouns. Limitations include rare/unseen words, complex sentences, and dependency on the manually curated knowledge base, with future improvements suggested through corpus expansion and machine learning integration.

3 Data Collection and Preprocessing

This section outlines the methodology for constructing a high-quality English-Urdu parallel corpus through systematic data collection, sentence-level alignment, and rigorous preprocessing. The process ensures linguistic equivalence, data integrity, and suitability for downstream NLP tasks, particularly machine translation.

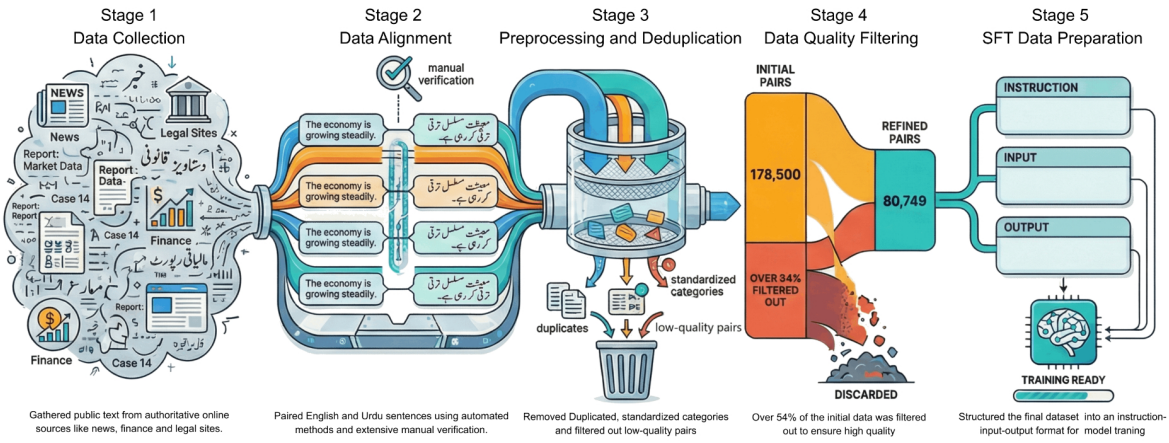


Figure 1: Data Processing Pipeline

3.1 Data Collection

We collected bilingual textual content from publicly accessible, authoritative online sources spanning diverse domains, including news outlets, agricultural extensions, financial institutions, culinary archives, telecommunications providers, legal repositories, and historical records. Priority was given to reputable platforms offering verifiable parallel content.

Using automated, non-intrusive pipelines, we extracted only publicly available textual material such as articles, reports, guidelines, and descriptive passages while avoiding restricted or login-protected sections. All retrieved content was cataloged with source metadata and stored in a structured repository, adhering to ethical standards for open-domain data acquisition.

3.2 Data Alignment

Parallel documents were pre-paired with explicit filename correspondence. Sentence-level alignment began automatically by exploiting structural cues (e.g., punctuation, paragraph breaks, and formatting markers) to generate initial English-Urdu sentence pairs, which were consolidated into a central corpus.

Subsequently, a designated annotator conducted manual pairwise verification to resolve alignment ambiguities caused by structural variations, cultural adaptations, or content omissions/additions in Urdu or English translations. A final exhaustive validation was then performed on the consolidated corpus to ensure strict one to one sentence correspondence, eliminating discrepancies and yielding high fidelity parallel pairs.

3.3 Data Preprocessing and Deduplication

The aligned corpus initially comprising 178,500 sentence pairs from verified sources underwent systematic preprocessing to eliminate noise, standardize formatting, and enhance translation quality. This step ensured the final dataset was clean, consistent, and optimized for bilingual modeling tasks.

3.3.1 Merging and Initial Analysis

The aligned sentence pairs from all sources were consolidated into a single corpus containing 178,500 parallel entries, with fields `English`, `Urdu`, and `Category`. Basic descriptive statistics including total rows, column structure, and non-null counts per field were computed. Empty rows (where every field was missing) arose occasionally during source ingestion and were removed, yielding a negligible reduction in size.

3.3.2 Category Normalization

The `Category` field, originally assigned by the data collector for each source, exhibited inconsistencies such as duplicate labels (e.g., “News” vs. “News ”), trailing whitespace, and typographical errors (e.g., “Relgion”). Normalization was essential to

Table 1: Distribution of Standardized Categories in the Final Dataset

Category Group	Count
National News	49,516
International News	8,898
Law & Judiciary	6,484
Food & Recipes	4,941
Banking	2,576
Agriculture	2,277
Sports	2,042
Telecom	1,766
Religion	615
Health	521
Business	326
Science	268
Corporate & Business	173
Anti-Drug Education	154
Showbiz	73
Daily Life & Communication	54
Weather	41
History & Arts	24

enforce a uniform taxonomy across the entire corpus, thereby enabling reliable domain-based analysis and stratified sampling. Domain identification was performed on sentences belonging to the "News" category using the Domain Identification API.[1] Similar categories were collapsed into 18 standardized groups while preserving thematic integrity. The resulting distribution, after all subsequent preprocessing steps, is reported in Table 1.

3.3.3 Identifier Assignment

To maintain traceability throughout the pipeline, unique identifiers were introduced at two stages. A `Raw_Sentence_ID` (RSID000003–RSID178450) was assigned to every entry in the merged raw corpus. After completion of the cleaning stages described in the following subsections (deduplication, filtering, and quality assurance), a `Cleaned_Sentence_ID` (CSID000003–CSID118482) was allocated to each retained high-quality pair, ensuring unambiguous reference in downstream modeling and evaluation.

3.3.4 Deduplication and Filtering

Duplicates were removed in two stages: (1) exact matches on both `English` and `Urdu` (removing 56,434 rows), and (2) duplicates on `English` with varying `Urdu` (removing 1,703 rows, retaining the first occurrence). Sentences with two or fewer words in English were filtered out (1,880 rows), as they typically represent fragments unsuitable for translation tasks.

3.4 Data Quality Filtering

Word counts were calculated for each sentence: `EWC` (English Word Count) and `UWC` (Urdu Word Count), using whitespace tokenization. The absolute difference, $\texttt{AWCD} = |\texttt{EWC} - \texttt{UWC}|$, was computed to gauge translation fidelity, assuming well-aligned translations exhibit similar lengths.

Statistics on the post-filtering dataset (80,749 pairs) include: average EWC of 16.20, average UWC of 17.83, and average AWCD of 2.24 (Table 2).

Table 2: Key Statistics of Final Dataset

Metric	Value
Total Sentence Pairs	80,749
Unique Categories	18
Average EWC	16.20
Average UWC	17.83
Average AWCD	2.24

To ensure the highest possible translation quality for downstream modeling, we adopted a conservative filtering strategy based on AWCD. Sentences with AWCD ≤ 5 were classified as high-fidelity pairs and retained without modification, as this threshold captures translations with closely matching lengths, which strongly correlates with accurate semantic alignment in parallel corpora.

Sentences with AWCD > 5 were excluded from the final dataset. While some pairs in this range may still represent valid translations (e.g., due to legitimate structural differences between English and Urdu), higher AWCD values introduce greater risk of misalignment, partial translations, or added/omitted content. Given the priority of producing a clean, high-precision corpus suitable for training and evaluating machine translation systems where even moderate misalignment can degrade performance. We opted to prioritize precision by retaining only the most reliable pairs.

This conservative approach yielded 80,749 high-quality sentence pairs for the final dataset. The distribution of retained and excluded pairs is shown in Table 3.

[1] https://tech.cle.org.pk/domainidentification

Table 3: Distribution of Translation Quality Groups in the Final Dataset

Status	Count	Percentage
Retained (AWCD ≤ 5)	80,749	60.9%
Excluded (AWCD > 5)	51,920	39.1%

This preprocessing pipeline ensures a clean, structured corpus ready for linguistic review and NLP applications.

3.5 Supervised Fine-Tuning Data Preparation

To prepare the corpus for supervised fine-tuning (SFT) of instruction tuned large language models, we transformed the high quality parallel sentence pairs into an instruction-following format consisting of three fields: `instruction`, `input`, and `output`.

We first split the retained high fidelity pairs into training and testing sets using a 90:10 ratio, ensuring stratified sampling across the standardized categories to preserve domain diversity in both splits.

To fully exploit the bidirectional nature of the parallel data, we augmented the dataset by creating separate examples for each translation direction. For each parallel pair, we generated two SFT instances sharing the same `Cleaned_Sentence_ID` (CSID) but differing in direction: one with the English sentence as source and Urdu as target, and one with Urdu as source and English as target. This effectively doubled the number of training examples while maintaining perfect alignment.

A key aspect of our preparation strategy was the use of bilingual instructions to enhance robustness and mitigate potential biases arising from instruction language. Prior work has demonstrated that the language of the instruction can significantly influence performance in multilingual LLMs (Zhu et al., 2024). To address this and promote balanced improvement across both directions, we constructed four distinct types of instruction examples:

- English instruction + English input $\rightarrow$ Urdu output
- English instruction + Urdu input $\rightarrow$ English output
- Urdu instruction + English input $\rightarrow$ Urdu output
- Urdu instruction + Urdu input $\rightarrow$ English output

For each type, we curated separate pools of manually crafted prompt templates (approximately 10-15 varied phrasings per pool) that explicitly indicate the required translation direction while varying in stylistic nuance (e.g., "Translate the following English text into Urdu:", "Provide an accurate Urdu translation of this English sentence:", and the Urdu equivalents). During dataset construction, a template was randomly sampled from the appropriate pool based on the desired instruction language and translation direction. This random selection ensures lexical and structural diversity in the instructions, which has been shown to improve generalization in instruction tuning (Wang et al., 2023; Zhang et al., 2023).

By incorporating instructions in both languages and covering all direction combinations, the resulting SFT dataset encourages the model to perform reliably regardless of whether the prompt is presented in English or Urdu. The final SFT training set comprises twice the number of original training pairs providing a rich, diverse resource for bidirectional machine translation fine-tuning.

4 Experiments

In this section, we describe the finetuning experiments conducted using the supervised fine-tuning (SFT) dataset prepared in Section 3.5. We evaluate the translation performance of the finetuned models against appropriate baselines on established benchmarks, focusing on the impact of prompt language and translation direction.

4.1 Models

We experiment with the following models:

- **Llama-3.1-8B-Instruct** (Grattafiori et al., 2024): The base instruction-tuned model from Meta, serving as our primary baseline (referred to as **Base**).
- **UrduLlama 1.1**: An Urdu adapted variant built on Llama-3.1-8B-Instruct through continued pretraining (CPT) on approximately 800 million Urdu tokens followed by instruction finetuning on 432K Urdu instructions (referred to as **UrduLlama 1.1**).

Both UrduLlama 1.1 model and Llama-3.1-8B-Instruct were finetuned on the SFT Data prepared in Section 3.5.

Table 4: BLEU scores on the held-out test set from our corpus.

Prompt Language	Direction	Base	Llama-FT	UrduLlama-FT
Urdu	En → Ur	14.22	**25.24**	23.11
English	En → Ur	13.65	**24.65**	23.82
Urdu	Ur → En	19.79	**76.14**	29.14
English	Ur → En	16.99	**77.97**	29.79

4.2 Fine-Tuning Setup

We fine-tuned Llama-3.1-8B-Instruct and UrduLlama 1.1 using the `torchtune` library (torchtune maintainers and contributors, 2024) with LoRA (Hu et al., 2021) for parameter efficiency.

LoRA adapters (rank 8, $\alpha = 16$) were applied to attention projections and MLP layers. Training used AdamW (learning rate 3×10^{-4}, weight decay 0.01), cosine scheduling with 100 warmup steps, and `bfloat16` precision. The per-device batch size was 1 with gradient accumulation of 8 steps (effective batch size 8). Activation checkpointing was enabled to reduce memory usage.

Using Nvidia A100 40GB GPU, models were trained for 3 epochs on the full SFT training set with shuffling. The same hyperparameters were used for both models to ensure a fair comparison.

4.3 Automatic Evaluation

We automatically evaluated translation quality using sacreBLEU (Post, 2018) on a held-out test set from the split of our corpus. Evaluations were conducted across the four combinations of prompt language (English or Urdu) and translation direction.

Table 4 reports BLEU scores for three models: the original Llama-3.1-8B-Instruct without further fine-tuning (Base), Llama-3.1-8B-Instruct fine-tuned on our corpus (Llama-FT), and UrduLlama 1.1 fine-tuned on the same corpus (UrduLlama-FT).

Fine-tuning on our high-quality parallel corpus yields large improvements over the Base model in both directions. For English → Urdu, Llama-FT achieves the highest scores (24.65–25.24), outperforming UrduLlama-FT by 1–2 BLEU points. In the Urdu → English direction, the gains are particularly striking: Llama-FT reaches BLEU scores above 76, far exceeding both the Base (17) and UrduLlama-FT (29). This demonstrates the value of supervised finetuning on clean, domain diverse parallel data, especially when translating from the lower-resource language.

A key observation is the robustness of Llama-FT to prompt language. The performance difference between English and Urdu prompts is negligible (e.g., 24.65 vs. 25.24 for En → Ur; 77.97 vs. 76.14 for Ur → En). In contrast, the Base and UrduLlama-FT show more noticeable drops when prompted in Urdu (up to 3 points in some cases). This stability can be attributed to our bilingual instruction strategy during SFT preparation, which exposed the model to instructions in both languages and reduced sensitivity to the prompt's language which is a common issue in multilingual LLMs. (Zhu et al., 2024)

4.4 Human Evaluation

To complement automatic metrics, we conducted a blind human evaluation on translations generated using English prompts in both directions.

We randomly sampled 90 sentences from the validation split, covering all categories. For each sentence, the three models (Llama 3.1-8B-Instruct, UrduLlama, and Llama 3.1-8B-Instruct-finetuned) produced translations, which were pooled, shuffled, and presented anonymously to two bilingual annotators fluent in English and Urdu.

Annotators assigned a single overall quality score on a 5-point Likert scale (5 = excellent, 1 = poor), assessing the translation's combined adequacy and fluency. Final scores per model and direction were obtained by averaging the ratings from both annotators. Table 5 reports the average human scores.

The human judgments align closely with the automatic BLEU scores while providing additional nuance on perceived quality. In the English → Urdu direction, the Base model receives an extremely low rating, confirming its limited ability to generate coherent Urdu text. Both fine-tuned models show dramatic improvements, with UrduLlama-FT slightly edging out Llama-FT (3.80 vs. 3.68). This small advantage for UrduLlama-FT likely stems from its prior continued pre-training on massive monolingual Urdu data, which enhances Urdu

generation capability.

Table 5: Human evaluation results

Direction	Base	UrduLlama	Fine-tuned
English → Urdu	1.05	**3.80**	3.68
Urdu → English	3.36	3.57	**3.86**

Conversely, in the Urdu → English direction, Llama-FT achieves the highest rating (3.86), outperforming UrduLlama-FT (3.57) and substantially surpassing the Base model (3.36). This mirrors the large BLEU gains observed for Llama-FT in this direction and underscores the value of high quality bidirectional parallel data for improving comprehension and translation from the lower resource language.

Overall, human evaluation corroborates the automatic metrics: supervised finetuning on clean, domain diverse parallel sentences yields major perceptual quality improvements, particularly when starting from a general-purpose base model. The complementary strengths UrduLlama-FT's edge in Urdu generation and Llama FT's superiority in Urdu to English translation highlight how different adaptation strategies (monolingual continued pretraining vs. parallel SFT) benefit distinct aspects of bidirectional performance.

5 Conclusion

In this work, we presented a carefully constructed English-Urdu parallel corpus comprising 80,749 high-quality sentence pairs across 18 diverse domains. Through systematic collection from authoritative sources, rigorous sentence level alignment, extensive preprocessing, and conservative length-based filtering (AWCD ≤ 5), we produced a clean and traceable resource suitable for low resource machine translation. By transforming this corpus into a bidirectional supervised finetuning (SFT) dataset augmented with bilingual instructions (both English and Urdu prompts), we effectively doubled the training examples while promoting robustness to prompt language.

Fine-tuning Llama-3.1-8B-Instruct (Llama-FT) and UrduLlama 1.1 (UrduLlama-FT) on this dataset resulted in substantial improvements over the unmodified Llama-3.1-8B-Instruct baseline. Automatic evaluation using sacreBLEU showed Llama-FT achieving BLEU scores of 24.65–25.24 (En→Ur) and 76.14–77.97 (Ur→En), far outperforming the baseline and demonstrating near-complete insensitivity to prompt language. UrduLlama-FT also delivered strong gains, particularly in Urdu generation. Blind human evaluation on 90 sentences per direction corroborated these findings: fine-tuned models scored 3.68–3.80 (En→Ur) and 3.57–3.86 (Ur→En) on a 5-point scale, compared to the baseline's 1.05 and 3.36.

The results underscore the value of high quality parallel data for bidirectional translation in low resource settings and show that bilingual instruction tuning effectively reduces prompt language bias. Complementary strengths emerged: Llama-FT excelled in Urdu-to-English translation, while UrduLlama-FT retained an edge in English-to-Urdu generation due to its prior Urdu specific adaptation. Together, these contributions advance open research on Urdu machine translation and provide a reproducible pipeline for similar low resource languages.

6 Limitations and Future Work

The corpus, while diverse, is heavily weighted toward news domains, potentially limiting generalization. The evaluation was primarily internal on held-out data from the same corpus, which ensures consistency but restricts the evaluation of out-of-domain performance.

Future directions include publicly releasing the full cleaned corpus. We also intend to evaluate the fine-tuned models on external benchmarks such as FLORES-200 and WAT Urdu tasks for broader comparability. To enable practical deployment on resource-constrained devices common in Urdu-speaking regions, we plan to apply the same fine-tuning recipe to smaller, efficient base models to further reduce memory footprint and inference latency while preserving translation quality.

References

Ahsan Ahmad and Mohsin Ali Ahmad. 2024. Advancing Roman Urdu to Urdu transliteration using machine learning techniques. *Asian Journal of Multidisciplinary Research & Review*, 5(2):108–127.

Syed Abdul Basit Andrabi and Abdul Wahid. 2022. Machine translation system using deep learning for English to Urdu. *Computational Intelligence and Neuroscience*, 2022:7873012.

Umer Butt, Stalin Varanasi, and Günter Neumann. 2025. Low-resource transliteration for Roman-Urdu and Urdu using transformer-based models. In *Proceedings of the Eighth Workshop on Technologies for*

Machine Translation of Low-Resource Languages (LoResMT 2025), pages 144–153. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Muhammad Naeem Ul Hassan, Zhengtao Yu, Jian Wang, Ying Li, Shengxiang Gao, Shuwan Yang, and Cunli Mao. 2024. LKMT: Linguistics knowledge-driven multi-task neural machine translation for Urdu and English. *Computers, Materials & Continua*, 81(1):1901–1923.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Huma Israr, Muhammad Khuram Shahzad, and Shahid Anwar. 2024. Improved Urdu–English neural machine translation with a fully convolutional neural network encoder. *International Journal of Mathematical, Engineering and Management Sciences*, 9(5):1067–1088.

Bushra Jawaid, Amir Kamran, and Ondřej Bojar. 2016. Enriching source for English-to-Urdu machine translation. In *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP@COLING)*, pages 54–63, Osaka, Japan. Association for Computational Linguistics.

Shahnawaz Khan and Imran Usman. 2019. A model for English to Urdu and Hindi machine translation system using translation rules and artificial neural network. *The International Arab Journal of Information Technology*, 16(1):125–131.

Sajadul Hassan Kumhar, Syed Immamul Ansarullah, Akber Abid Gardezi, Shafiq Ahmad, Abdelaty Edrees Sayed, and Muhammad Shafiq. 2022. Translation of english language into urdu language using lstm model. *Computers, Materials & Continua*, 74(2):3899–3912.

Hafsa Masroor, Muhammad Saeed, Maryam Feroz, Kamran Ahsan, and Khawar Islam. 2019. Transtech: development of a novel translator for roman urdu to english. *Heliyon*, 5(5):e01780.

Usman Mohy ud Din. 2019. Urdu–English machine transliteration using neural networks. Master's thesis, COMSATS University Islamabad, Lahore Campus.

Matt Post. 2018. A call for clarity in reporting bleu scores. *Preprint*, arXiv:1804.08771.

Ahmed Raza, Usama Ahmed, Kainat Saleem, Muhammad Azam Hussain, and Amna Sarwar. 2024. Urdu-to-english-based unsupervised machine translation. *Journal of Computer Science and Applications*, 1(2):1–10.

SIL International. 2022. Urdu. `https://www.ethnologue.com/language/urd`. Ethnologue: Languages of the World (25th edition, accessed 2025).

torchtune maintainers and contributors. 2024. torchtune: Pytorch's finetuning library.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 13484–13508.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Guoyin Wang, and 1 others. 2023. Instruction tuning for large language models: A survey. *ACM Computing Surveys*.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the association for computational linguistics: NAACL 2024*, pages 2765–2781.

Semi-Automatic construction of a Quechua-Spanish dictionary

Maximiliano Duran
Université de Franche-Comté
maximiliano.duran@yahoo.fr

Max Silberztein
Université de Franche-Comté
max.silberztein@gmail.com

Abstract

This paper presents a set of linguistic resources that describes Quechua verbs. We first present a dictionary of 1,444 fundamental Quechua verbs, associated with morpho-syntactic grammars to formalize their inflection and their derivations, that can be used to produce over 2,777,000 conjugated Quechua derived verbal forms. We aligned this list of Quechua verbal forms with the corresponding Spanish dictionary that contains 618,000 conjugated verbal forms, thus producing both a Spanish to Quechua and a Quechua to Spanish dictionary.

1 Introduction

Quechua is an endangered language, spoken by over 6 million people in Peru, Equator, Bolivia, and Argentina. Quechua has an important characteristic: its verbal morphology is robust and highly productive. Using the NooJ linguistic development platform (see Silberztein 2016),[1] we have contructed a dictionary of 1,444 Quechua verbs and formalized the corresponding inflectional and derivational morphological systems.

Quechua morphological system is based on the use of suffixes, which we have classified into four fundamental sets, see (Duran, 2014):

- Conjugation endings suffixes include two subsets: the Indefinite Tense Endings ITE = {*ni*, *nki, n, nchik, niku, nkichik, nqaku*}, and the Future Tense Endings FTE = {*saq, nki, nqa, sunchik, saqku, nkichik, nqaku*}.

- Interposition Suffixes (IPS) placed after the verb radical and the infinitive suffix "y" or the conjugation endings. IPS suffixes are divided into two very productive subsets:

- Derivation Inter Position Suffixes IPSd = {chaku, chi, chka, ykacha, ykachi, ykamu, ykapu, ykari, yku, ysi, kacha, kamu, kapu, ku, lla, mpu, mu, naya, pa, paya, pu, raya, ri, rpari, rqu, ru, tamu} (27),which will generate new verbs by derivation. For example:

asiy [to laugh] → *asiriy* [to smile]
maskay [to search] →*maskapay* [to inquire]
rimay [to talk] → *rimaykuy* [to greet]
samay [to rest] → *samarquy* [to bivouac].

- Postposition Suffixes (PPS) placed after the conjugation endings, The Postposition Suffixes (PPS), placed after the conjugation endings, allow to express modalities: declarative, interrogative, imperative, emphasis, exclamation, negative or affirmative, passive, active or conditional: PPS = {- *ch, -chá, -chik, -chiki, -chu, -chu(?), -chusina, -m, -mi, -má, -man, -ña, -pas, -puni, -qa, -raq, -s, -si, -taq, -yá* }

- Non Derivation Inter Position Suffixes IPSd, serving to express the Past Tenses (rqa, sqa, ra) and some moods Reflexive (wa), Subjunctive (sqa), and the Gerund

[1] NooJ is a free and open-source linguistic development platform linguists use to formalize various linguistic phenomena, from orthographic to semantic. NooJ allows linguists to construct electronic dictionaries, as well as regular, context-free, context-sensitive and unrestricted formal grammars. NooJ contains tools to edit, test, debug and maintain large-coverage linguistic resources. There are linguistic resources in the NooJ format for over 30 natural languages, see https://nooj.univ-fcomte.fr.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 111–118

(spa, pti), IPSd = { pti, rqa, spa, sqa, stin, wa}

- Verb Nominalizing Suffixes (VNS) derive a verb into a noun: VNS = {-i, -na, -q, -sqa }.

Some verbs accept only one nominalization suffix, whereas others may take two, three, or all four nominalization suffixes. For example:

imikuy → *miku**i*** [meal]
namikuy →*miku**na*** [feed]
qmikuy → *miku**q*** [dinner]
sqamikuy → *miku**sqa*** [what was eaten]

2 Conjugation

The basic indefinite tense Quechua conjugation scheme consists of the agglutination of the verbal root and the ITE suffix. Named as indefinite, it may simultaneously express a present, a past tenses, or a usual activity. The following table presents the conjugation of a verb in the Indefinite tense:

Pronoun	SP-EN	Verb root	ITE
ñoqa	Yo I	*VR*	*ni*
qam	Tu You		*nki*
pay	El(la) He, she		*n*
ñoqanchik	Nosotros We (inc.)		*nchik*
ñoqaiku	Nosotros We (ex.)		*niku*
qamkuna	Ustedes You		*nkichik*
paykuna	Ellos(llas) They		*nku*

Table 1: Indefinite Tense

In the same way, the conjugation paradigm that corresponds to the future tense is shown in Table 2:

Pronoun	SP-EN	Verb root	FTE
ñoqa	Yo I	*VR*	*saq*
qam	Tu You		*nki*
pay	El(la) He, she		*nqa*
ñoqanchik	Nosotros We (inc.)		*sunchik*
ñoqaiku	Nosotros We (ex.)		*saqku*
qamkuna	Ustedes You		*nkichik*
paykuna	Ellos(llas) They		*nqaku*

Table 2: Future Tense

As of now, we have listed 1,444 Quechua fundamental verbs, and have associated each of them with their corresponding conjugation paradigm.
In Quechua, many combinations of tenses, aspects, and circumstances result from the agglutination of one, two, or three IPS. For example:

mikuchkani [I am eating]
mikurqani [I have eaten]
mikuchkarqani [I was eating]
mikurani [I ate]
mikunaimi karqa [I was going to eat]
mikusaq [I will eat]
mikurqusaq [I am going to eat quickly]

3 Derivation

Fundamental verbs can be agglutinated with one or more suffixes to produce new verb meanings. Next are a few examples of fundamental verbs agglutinated with one IPS suffix:

asiy [to laugh] + *ri* ⟶ *asiriy* [to smile]
rimay [to talk] + *yku* ⟶ *rimaykuy* [to greet]
ripuy [to go away] + *ku* ⟶ *ripukuy* [to move]
suyay [to wait] + *yku* ⟶ *suyaykuy* [to wait patiently]

Most derived verbs are listed as entries in traditional dictionaries. For instance, mono-suffixed derived verbs are represented as lexical entries, as the following ones:

asipayay [to make fun of], derived from *asiy* [to laugh]
*maska*kuy [to search for oneself], derived from maskay [to search]
rakinakuy [to divorce], derived from *rakiy* [to split]
rimanakuy [to dialogue], derived from *rimay* [to talk]

but many derived verbs, such as the ones derived from the verb *rimay* [to talk], are not listed in any dictionary: *rimaymanay, rimatamuy, rimaruy, rimapuy, rimarquy, rimarpariy*. Our project is to recognize them and produce their Spanish translation automatically.

4 Formalization with NooJ

The NooJ platform contains tools that allow linguists to develop various types of grammars, from the orthographic to the semantic levels. NooJ inflectional grammars can be regular or context-free grammars. In its simplest form, a NooJ

regular grammar is just a disjunction of (suffix, property) pairs in the following form:

```
Paradigm = suffix/properties |
suffix/properties | …;
```

Typically, a suffix is added to the lexical entry to produce an inflected form, which is then associated with some linguistic property. For example, the following paradigm WALK describes the inflection of all the English verbs that take an "s" in the Present, third person singular, "ing" in the Gerund, and "ed" in the Preterit and the Past Participle:

```
WALK = <E>/Infinitive |
s/Present+3+singular | ing/Gerund |
ed/Preterit | ed/PastParticiple;
```

The first term represents the fact that if one adds the empty string (noted <E> in NooJ) to a lexical entry, one produces its infinitive form. The second term represents the fact that if one adds an "s" to the lexical entry, one produces a form in the Present tense, 3rd person, singular.
Suffixes may contain stack operators to process words, such as <B> (for "Backspace") to delete the letter at the top of the stack. For instance, the following rule represents the paradigm used to inflect the word "man":

```
MAN = <E>/singular | <B2>en/plural;
```

If one deletes the two last letters of the lexical entry "man" and then adds the suffix "en", one produces the plural form "men".
NooJ contains also "linguistic" operators specific to each language, such as the Spanish operator <Á> (add accent), the French operator <D> (duplicate letter), the Semitic operator <F> (definalize/finalize last consonant), etc.
NooJ contains a handy operator specific for Quechua: <D> which, applied to certain inanimate nouns, duplicates them to generate a new noun bearing a superlative content, such as in the following examples:

rumi [stone] + <D> → *rumirumi* [stony]
sacha [tree] + <D> → *sachasacha* [forest]
aqu [sand] + <D> → *aquaqu* [sandy]

The following rule formalizes the Quechua Indefinite Tense IT paradigm, described in Table 1 of suffixes:

```
IT = <B> (ni/+1+s | nki/+2+s |
n/+3+s | nchik/+1+pin | niku/+1+pex
| nkichik/+2+p | nku/+3+p);
```

The linguistic codes correspond to the following properties: `+1`, `+2`, `+3`: first, second, or third person; `+s`: singular; `+p`: plural; `+pin`: inclusive plural; `+pex`: exclusive plural. In NooJ Context-Free grammars, a rule can refer to another rule, using its name prefixed with the colon character (":"). Rule names are the equivalent of the auxiliary symbols in generative grammars. For example, rule IPSIPS is defined as:

```
IPSIPS = <B> (chi/FACT mu/ACENT
| chi/FACT pu/APT |
chaku/DVAL :CHKA/PROG |
chi/FACT chi/FACT);
```

Here, `:CHKA` refers to the paradigm **`CHKA`**. The Quechua dictionary uses a list of 27 semantic features (FACT, ACENT, APT, PROG, etc.). For instance: FACT (factitive), ACENT (towards oneself), PROG (Progressive), etc.
The following rule formalizes verbal derivations that use one suffix:

```
V_SIP1_INF = <B> (:CHAKU | :CHI |
:CHKA  |:YKACHA | :YKACHI | :YMANA |
:YKAMU| :YKAPU | :YKARI | :YKU |
:YSI | :KACHA | :KAMU  | :KAPU | :KU
| :LLAV | :MU | :NAYA | :PAV | :PAYA
| :PU | :RAYAV | :RIV | :RPARI |
:RQU | :RU |:TAMU) y/V;
```

Applied to the verb *rimay* [to talk], NooJ produces 27 derived verbs automatically, including the following ones:

```
rimachiy,V+FLX=V_SIP1_INF
rimarpariy,V+FLX=V_SIP1_INF
rimaykuy,V+FLX=V_SIP1_INF
```

These derived verbs correspond to the following English translations:

rimachiy [make him talk]
rimarpariy [go to talk to all of them]
rimaykuy [say your greetings]

4.1 Multi-suffixes inflections

Quechua verbs can be derived using two IPS suffixes. The Boolean matrix of table 3, presented in (Duran 2017), represents 240 valid combinations of IPS suffixes.

SIP_TR	CHAKU	CHI	CHKA	YKACHA	YKACHI	YKAMU	YKAPU	YKARI	YKU	YSI	KACHA	KAMU	KAPU	KU
CHAKU	0	1	1	0	1	1	1	1	1	1	1	0	0	0
CHI	0	1	1	1	1	1	1	1	1	1	0	1	1	1
CHKA	0	0	0	0	1	1	1	1	0	1	0	0	0	0
YKACHA	1	1	1	0	0	0	0	0	0	1	0	1	0	1
YKACHI	0	0	1	0	0	0	0	0	0	1	0	1	1	1
YKAMU	0	0	1	1	1	0	0	1	0	1	0	0	0	0
YKAPU	0	0	1	1	1	1	0	1	0	1	0	0	0	0
YKARI	0	1	1	0	0	0	0	0	1	1	0	0	0	0
YKU	0	0	1	1	1	1	1	1	0	1	0	0	1	1
YSI	0	1	1	0	0	0	0	1	1	0	0	0	0	0
KACHA	1	1	1	0	0	1	1	1	1	1	0	1	1	1
KAMU	0	0	1	1	1	1	1	1	1	1	0	0	0	1
KAPU	0	1	1	1	1	1	1	0	1	1	0	0	0	0
KU	0	0	1	0	0	1	1	0	1	0	0	0	1	1

Table 3: Boolean Matrix of IPS combinations

The rule that describes the corresponding combinations of two IPS suffixes is implemented in NooJ as follows:[2]

```
SIP2_TR_INF = <B> (:CHAKUCHKA |
:CHICHI | :CHICHKA | :KURQU |
:LLAVCHKA | :LLAVYKACHI | :NAYACHKA
| :NAYALLAV | :PAYAYSI | :PAYAKACHA
| :PAYAKAMU | :PAYAKAPU | :PAYAKU |
:PAYALLAV | :PAYAMU | :RAMU | :RICHI
| :RICHKA | :RIYKACHI | :RPARILLAV |
:RPARIMU | :TAMUCHKA | :TAMUYKACHI
| :TAMUYKAMU | :YKURQU) y/V;
```

When applied to any fundamental verb, the valid combination of two IPS suffixes will produce 240 new verbs. The following is an extract of the dictionary of verbs derived from the verb *rimay* [to talk]:

```
rimachimuy,rimay,V+FACT+ACENT+INF
rimarparirquy,rimay,V+ASUR+PAPT+INF
rimarparimuy,rimay,V+ASUR+ACENT+INF
```

...

The corresponding translations are:

rimachimuy [make him talk]
rimarparimuy [go to talk to all of them]
rimaykurquy [say your greetings rapidly]

4.2 Automatic verbal inflection

Quechua Transitive verbs are conjugated according to the following rule:

```
V_TR_CONJ0 = :PR | :F | :IP |
:RQA_PR_PLU | :C |
:RQU_PR_PLU|:SPA_PRM2_PLU  |
:RU_PR_PLU | :PTIPRM1QA | :F_ÑA  |
:PTI_PRM2_PLU | :V_SIP0_STIN  |
:V_RECIP_CONJ0;
```

Following is the rule used to conjugate intransitive verbs:

```
V_ITR_CONJ0 = :PR | :F | :IP | :RQA
| :C | :RQU_PR | :SPA_PRM2_PLU |
:SPA | :RU_PR | :RA_PR | :RQU_F |
:PASSA | :SQA_PRM2_PLU | :RUPRM1MAN
| :RQUPRM1MAN | :PTIPRM1QA  | :F_ÑA
| :PTI_PRM2_PLU  | :NAV_PRM2 |
:NAV_PRM2_MM | :SQA_PR;
```

Following is a rule used to conjugate pronominal verbs:

```
V_PRON_CONJ0 = :PR_PRON | :F_ PRON |
:IP_ PRON | :RQA_PR_PLU_PRON |
:C_ PRON | :RQU_PR_PLU_PRON |
:SPA_PRON | :SPA_PRM2_PLU_PRON |
:RA_PR_PRON |
:PTI_PRM2_PRON | :NAV_PRM2_PRON |
:NAV_PRM2_MMPRON | :V_STIN_PRON;
```

Following is the rule used to conjugate impersonal verbs:

```
V_IMP_CONJ0 = :PR_IMP | :F_IMP |
:I_IMP | :RQA_IMP | :C_IMP | :SPA |
:SPA_PRM2_IMP | :PTIPRM1QA_IMP |
:RQU_IMP | :SPA | :F_ÑA_IMP |
:STIN_IMP | :PTI_PRM2_IMP |
:NAV_PRM2_IMP | :SQA_SS_IMP;
```

All rules can be applied to a verb derived by adding one or more IPS suffixes to a fundamental verb such as *asiriy* [to smile]. NooJ can then produce the list of all conjugated forms of all the verbs derived using one or more IPS suffixes.

One can automatically obtain all the conjugated forms of the 1,444 fundamental verbs, as well as their derived verbs, using one or two IPS suffixes. By combining the inflectional and the derivational grammar rules and applying them to the dictionary of fundamental Quechua verbs, NooJ has generated 2,777,418 verbal forms (see Figure 1), including for the verb *rimay* [to talk]:

0 IPS: *rimanqa* [he will talk]
1 IPS: *rimaykurqanchik* [we saluted]
2 IPS: *rimapayaykunku* [they are arguing]

2 For the detailed description of all the paradigms (CHA, KU, CHKA, CHAKUCHKA..etc) appearing in the article in capital letters see Duran(2017).

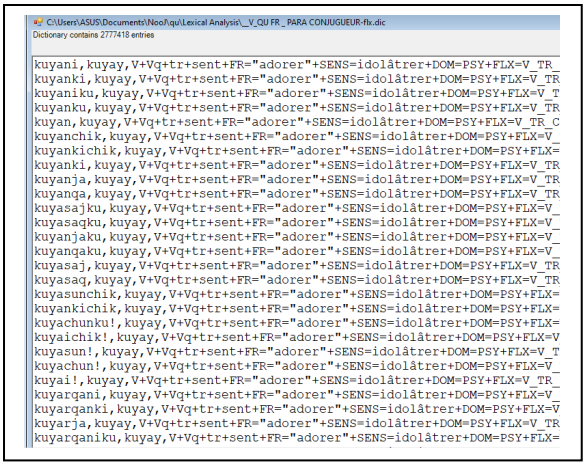


Figure 1: NooJ produced 2,777,418 conjugated forms of simple and derived verbs

4.3 Postposed suffixes

The second set of suffixes that generate a different kind of inflected forms are the Post Posed Suffixes (PPS) which appear after the Indefinite Tense endings (ITE), or the Future Tense Endings (FTE), e.g.:

*rimani**ña*** [I already talked]

Here the PPS suffix "*ña*" appears after the ITE suffix "*ni*".

*rimasaq**raq*** [I will talk before]

Here the PPS suffix "*raq*" appears after the FTE suffix "*saq*". The following paradigm formalizes the use of one PPS:

```
PPS1 = :CHAAV | :CHV | :CHIKV |
:CHIKIV | :CHUIV | :CHUNV |
:CHUSINAV | :MAAV | :MV |:ÑAV |
:PASV | :PUNIV | :QAV | :RAQV |
:SV |:TAQV | :YAAV;
```

When applied to the verb *rimay* [to talk], this paradigm generates 17 verbal forms including the following ones:

*rimanki**chu*** [you did not talk]
*riman**mi*** [he talked indeed]
*rimanjaku**s*** [they say that they will talk]
*riman**mi*** [I did talk]

Verbs can accept one, two or three PPS suffixes, *e.g.*:

2 PPS: *rimanki**raqchu***? [Will you still talk?]

3 PPS: *rimanchik**manñataq*** [What about if we should talk?]

4.4 Mixed suffix agglutination

When we analyze some type of conjugated verb forms, it's interesting to remark that the ITE and the FTE suffixes behave as fixed points, around which the two sets of suffixes IPS and PPS appear agglutinated to produce a general verbal form for the Indefinite Tense, *e.g.*:

*llamkachka**n**raqmi* [He is still working]

or, for the Future Tense, *e.g.*:

*llamkachka**nqa**raqmi* [He will be still working]

In the first verbal form, the IPS suffix "*chka*" is positioned between the verb root *llamka* [work] and the verbal ending "***n***" mark of the third singular person in the indefinite tense, and the two PPS suffixes *raq* and *mi* located after the ending. These verbal forms occur actually very frequently in daily communication.
Following are more examples of verbal forms that contain ITE endings used as fixed points:

mikullachkankuña [fortunately, they are already eating]

Here, there are two IPS: *lla* and *chka*, and one PPS: *ña*.

pukllachichkaniraqmi [I am making him play]

Here, there are two IPS *chi* and *chka* and two PPS *raq* and *mi*.

mikuchiykuchkanillaraqmi [I am carefully helping him to eat before anything else]

We have implemented grammar rules for these mixed constructions. For instance, the following one represents the insertion of one or two IPS and one PPS:

```
V_MIX21 = :SIP2_PR_V :SPP1_PR_V |
:SIP2_PR_C :SPP1_PR_C;
```

For a detailed description of each paradigm, see Duran (2017). Based on our dictionary of 1,444 verbs and the inflectional and derivational grammars, NooJ can recognize any Quechua verbal inflected and derived verbal form,

lemmatize and analyze it, and reciprocally produce 2,777,418 Quechua inflected and derived verbal forms.

5 Translating Spanish to Quechua

Because there are linguistic resources for many languages that have been formalized with the same NooJ platform[3] using the same formalism, it becomes relatively easy to align them and thus construct bilingual lexical resources.

5.1 The Spanish-Quechua Verb Dictionary

To produce a Quechua translation of any Spanish verbal form, we need both the Quechua dictionary and a Spanish dictionary.

There are few available open-source Spanish electronic dictionaries, and even fewer bilingual Spanish to Quechua (SP-QU) dictionaries: The dictionary used by A. Rios (2015) for Spanish utilizes a set of part of speech tags from the Royal Academy of Spanish and, for the Quechua language, a set of tags corresponding to the Quechua Cuzco variant.

The NooJ website (https://nooj.univ-fcomte.fr) offers an electronic version of the RAE (Real Academia Española) dictionary implemented by (Fuentes, Gupta, 2014). It contains over 61,000 entries, including about 11,000 simple and compound verbs.

Using the tools offered by NooJ, we were able to align the Quechua list of fundamental and derived verbs with this list of 11,000 Spanish verbs, thus producing a bilingual Spanish-Quechua dictionary, see Figure 2.

```
caminar,V+mov+QU="puriy"+FLX=AMAR
caminar,V+mov+QU="riy"+FLX=AMAR
caminar,V+tr+QU="ichiy"+FLX=AMAR
canalizar,V+tr+QU="yaku-ñanta uray"+FLX=TRAZAR
canalizar,V+tr+QU="yarjachay"+FLX=TRAZAR
cancelar,V+tr+QU="jopitay"+FLX=AMAR
cancelar,V+tr+QU="jopoy"+FLX=AMAR
canjear,V+tr+QU="yamkiy"+FLX=AMAR
cansar,V+itr+Tec+Psy+QU="amiruy"+SENS=fatigar+FLX=AMAR
cansar,V+itr+Tec+Psy+QU="amiy"+SENS=fatigar+FLX=AMAR
```

Figure 2: SP-QU Electronic Dictionary

In this bilingual dictionary:

- Each entry is associated with its part of speech category ("V" for verbs), some syntactic and/or semantic property (*e.g.,* "+tr" for transitive verbs) and an inflectional paradigm (*e.g.,* "+FLX=AMAR" for the conjugation paradigm AMAR).
- Each entry is associated with its Quechua translation (property "+QU").

This dictionary contains simple verbs as well as multiword verbs such as:

andar a ciegas → +QU=*taplaykachay* [to tangle]
andar a paso lento → +QU=*purichakuy* [to walk slowly]

There are many cases of polysemy, both on the Spanish and on the Quechua sides. Some Spanish verbs may have more than one translation. For example, *caminar* [to walk] is associated with three Quechua translations: *puriy, riy, ichiy*.

5.2 Translating conjugated forms

This SP-QU dictionary only contains the lemma, *i.e.*, the infinitive form of each Spanish verb. However, to develop a Machine Translation system, we need to associate each Spanish conjugated form with its corresponding form in Quechua. Although there is no perfect correspondence between Spanish and Quechua tenses, moods and aspects, we entered the closest Quechua form for each Spanish form. Table 4 presents an extract of the resulting dictionary.

Abreviación y su descripción en SP (Espagnol) y QU (quechua) Equivalencias simbólicas esenciales entre SP y QU			
SP	QU	Flexión en SP	Valor en QU
inf	INF	inf infinitivo	INF infinitivo
ppio	GER1	ppio participio indefinido	GER1 gerundio flexionado
ger	GER	ger gerundio	GER gerundio –spa simultáneo
pres	PR	pres presente	PR presente
ppp	PPA	ppp pasado pluscuan perfecto	PPA –sqa, PR mikusqani
pps	PS	pps pasado simple (cantaste)	PS pasado simple
fut	F	fut futuro	F futuro
cond	C	cond condicional	C condicional
ppi	PASRA	ppi pret. perfecto indef (cantabas)	PASRA pasa. realizado (takirani)
pasmr	PASRQU	pasmr pasado modo rápido	PASRQU pasado modo rápido
subj	SUBJ	subj subjuntivo	SUBJ subjuntivo spaqa
subjmr	SUBJMR	subjpas	SUBJMR subj. modo rápido rqusqa
imp	IP	imp imperativo	IP imperativo
pp	PPI	pp pretérito perfecto	PPI pretérito perfecto rqa
1+s	1+s	1+s primera persona singular	1+s primera persona singular
2+s	2+s	2+s segunda persona singular	2+s segunda persona singular

Table 4: Verbal properties in Spanish and Quechua

Using NooJ to produce all inflected forms corresponding to the 11,000 Spanish verbal entries, we produced a list SP-QU of over 618,000 Spanish verbal forms associated with their Quechua translation. Then, using simple SQL operations, we reversed the SP-QU dictionary and produced the reverse Quechua to Spanish dictionary. Figure 3 shows an extract of it:

3 To download linguistic resources in the NooJ format for over 30 languages, see: https://nooj.univ-fcomte.fr/resources.html.

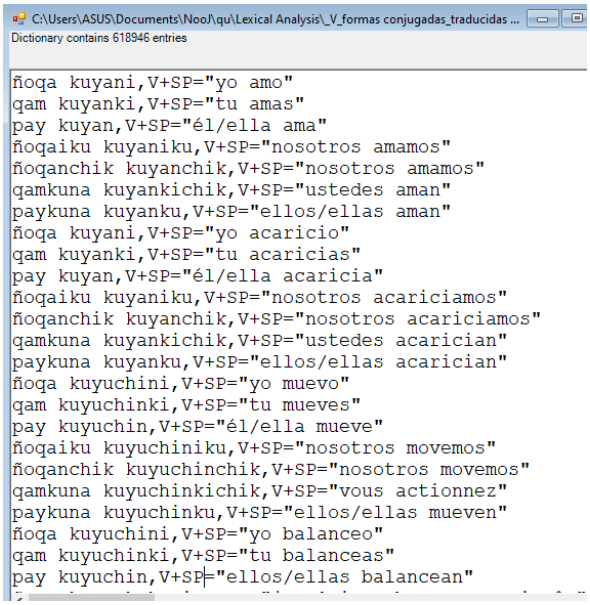


Figure 3: QU-SP Electronic dictionary.

This list constitutes the most comprehensive bilingual available lexical resource: it currently contains over 618,000 simple and compound lexical entries translated into Spanish, each entry being associated with their respective inflectional and derivational paradigm.

6 Evaluation

We have used NooJ to apply this Quechua-Spanish dictionary to a corpus consisting of ten story tales in Quechua (see references), totalizing 16,874 words. Applying the query <V> (to extract all Quechua verbal forms) produced a concordance of 227 entries, out of 234 really present in the text, i.e., a 97% recall. Out of these 227 entries, 2 words were incorrectly recognized as verbs: *llunpay* [too much] and *patanta* [the board], i.e., a 99% precision. It must be noted though that these two word forms could actually be verbal forms, in other syntactic contexts: we will develop and apply NooJ local syntactic grammars to disambiguate them automatically.

7 Conclusion and perspectives

Using the NooJ development environment platform, we have implemented a set of resources that formalizes the morphology of the Quechua verbal system. These resources contain a dictionary of fundamental Quechua verbs (about 1,444 entries), a derivational grammar that formalizes the derivations of each fundamental verb using one, two or three interposition suffixes, and an inflectional grammar that describes the conjugation of all fundamental and derived verbs. Based on these formalized resources, NooJ produced over two million Quechua verbal forms automatically. We aligned the resulting list with the list of Spanish conjugated verbal forms, resulting in a Spanish to Quechua database that associates any Spanish conjugated verb with its Quechua translation. Finally, using simple SQL operations, we reversed the database and produced a dictionary that associates 618,946 Quechua derived inflected verbal forms with their Spanish translation.

8 References

Adelaar, W. (1979). Tarma Quechua. Grammar, Texts, Dictionary. In: L'Homme.

Duran, M. (2014, June). Formalizing Quechua verb inflections. In Formalising Natural Languages with NooJ 2013: Selected Papers from the NooJ 2013 International Conference (p. 41). Cambridge Scholars Publishing.

Duran, M. (2017). Dictionnaire électronique français-quechua des verbes pour le TAL (Doctoral dissertation, Université Bourgogne Franche-Comté).

Fuentes, S., Gupta A. (2014). Updated Spanish Module for NooJ. In Formalising Natural Languages with NooJ 2013: Selected papers from the NooJ 2013 International Conference (p. 51). Cambridge Scholars Publishing.

Gross, M. (1992). Forme d'un dictionnaire électronique, Actes du colloque. "L'environnement traductionnel de Mons". AUPELF-UREF.

Holguin, D. (1608). Vocabvlario de la Lengua General de todo el Perv llamada lengua Qquichua, o del Inca. Corregido y renovado conforme a la propiedad cortesana del Cuzco. Ciudad de los Reyes Lima, Impreso por Francisco del Canto.

Parker, Gary J.(1973). Derivacion verbal en el quechua de Ancash. Working Paper 25. Lima : Universidad Nacional Mayor de San Marcos, Centro de Investigación de Lingüística Aplicada.

Perroud, P. (1970). Diccionario castellano kechwa, kechwa castellano. Dialecto de Ayacucho. Santa Clara, Peru. Seminario San Alfonso.

Rios, A. (2015). A basic language technology toolkit for quechua (Doctoral dissertation, University of Zurich).

Silberztein, M. (2003-). NooJ Manual. Available at https://nooj.univ-fcomte.fr.

Silberztein, M. (2016). Formalizing natural languages: The NooJ approach. John Wiley & Sons.

Soto C. (1976). Diccionario quechua ayacucho-chanca, (coll. “Lengua y sociedad”, 4), 183 p.

Taylor, G. (1979). Morphologie comparée du verbe quechua : l’expression de l’actance, Première partie : Le sujet, en relations prédicat-actance(s) dans des langues de types divers II, ed. C. Paris. LACITO-documents Eurasie 3. Paris : SELAF, 171-86.

Wallace K. (1988). Parsing Quechua Morphology for Syntactic Analysis. University of California. UCLA. Los Angeles, California.

Lexical Resources available on the WEB

https://www.quechua.org.uk/Eng/Main/

https://infolingu.inivmlv.fr/DonneesLinguidtiques/Dictionnaires/telechargement.html

https://www.cairn.info/revue-langages-2010-3-page-221.htm

https://www.freelang.com/dictionnaire/quechua_cuzco.php

http://www.lexilogos.com/quechua_dictionnaire.htm

https://fr.glosbe.com/es/qu/

http://aulex.org/qu-es/

Our corpus of fairy tales

Andahuaylas local journal (2010-2011).

de Avila, Francisco (2012). *Dioses y hombres de Huarochiri*. Narración quechua recogida por Francisco de Avila 1598. Traducción J. M. Arguedas. Lima. Perú 1966. (178p). Edición bilingüe facsimilar.

Guardia Mayorga, Cesar (1973). *Pakpaku chayñawan rimanakun*. Conte (p 383-84) Gramática Kechwa Ediciones Los Andes. Lima Perú.

Lira, Jorge (1990). *Cuentos del Alto Urubamba*. Centro Bartolomé de las Casas, Cuzco Eds.

Meneses, Porfirio (2001). *Sept contes en quechua* (1. Yupinta, 2. Hukpa wasin, Kawsayninchik kutimuptin, 4. Muti maskaypi, 5. Tayta Matias, 6. Warmichaykita waylluy, 7. Chiqnipacha. Edition quechua-français. Auteur C. Itier. Langues et Mondes Eds. Paris.

Oregón Morales, José (1994). *Loro Ccolluchi. Exterminio de loros y otros cuentos*. Lluvia Editores. Lima, 1994.

Taylor, G. (1987). *Ritos y Tradiciones de Huarochiri, Manuscrito Quechua de comienzos del siglo XVII*, 1ra Edición. IEP, Instituto Francés de Estudios Peruanos. Lima. Perú.

Improving Indigenous Language Machine Translation with Synthetic Data and Language-Specific Preprocessing

Aashish Dhawan
University of Florida
aashish.dhawan@ufl.edu

Christopher Driggers-Ellis
University of Florida
driggersellis.cw@ufl.edu

Christan Grant
University of Florida
christan@ufl.edu

Daisy Wang
University of Florida
daisyw@cise.ufl.edu

Abstract

Low-resource indigenous languages often lack the parallel corpora required for effective neural machine translation (NMT). Synthetic data generation offers a practical strategy for mitigating this limitation in data-scarce settings. In this work, we augment curated parallel datasets for indigenous languages of the Americas with synthetic sentence pairs generated using a high-capacity multilingual translation model. We fine-tune a multilingual mBART model on curated-only and synthetically augmented data and evaluate translation quality using chrF++, the primary metric used in recent AmericasNLP shared tasks for agglutinative languages. We further apply language-specific preprocessing, including orthographic normalization and noise-aware filtering, to reduce corpus artifacts. Experiments on Guarani–Spanish and Quechua–Spanish translation show consistent chrF++ improvements from synthetic data augmentation, while diagnostic experiments on Aymara highlight the limitations of generic preprocessing for highly agglutinative languages. All code is publicly released(will be released on submission).

1 Introduction

Many indigenous languages face increasing risk of endangerment due to limited digital presence and scarce linguistic resources, posing significant challenges for the development of robust machine translation (MT) systems (Mager et al., 2023; Woodbury, 2012). Languages such as Aymara and Guarani exemplify this challenge: despite being spoken by sizable communities in the Americas, their complex morphological structures and lack of large-scale parallel corpora hinder effective MT development (Rodríguez et al., 2022). As a result, conventional MT approaches that rely on abundant supervised data often perform poorly in these settings.

Recent advances in neural machine translation (NMT), particularly multilingual pretraining and data augmentation, have opened new opportunities for low-resource MT. Community-driven efforts such as AmericasNLP have demonstrated that multilingual models combined with synthetic data generation can substantially improve translation quality for indigenous languages (Ebrahimi et al., 2023a). Motivated by these findings, this work investigates the impact of synthetic parallel data augmentation on MT performance for Aymara–Spanish and Guarani–Spanish translation (Woodbury, 2012). Specifically, we augment curated datasets with synthetic sentence pairs generated using a high-capacity multilingual MT system, following the data-centric paradigm explored in projects such as (Driggers-Ellis et al., 2025). In addition, we include Spanish–Quechua experiments to assess whether language-specific orthographic normalization can further improve performance for morphologically rich indigenous languages.

We fine-tune the multilingual mBART model on both curated-only and synthetically augmented datasets and evaluate performance using chrF++ as the primary metric. Through controlled experiments and comparative analysis, this study aims to assess whether forward-translated synthetic data can reliably improve translation quality in low-resource indigenous language settings, contributing to ongoing efforts within the AmericasNLP initiative to support language preservation and accessibility (Ebrahimi et al., 2023a).

2 Background and Related Work

Machine translation (MT) for low-resource and indigenous languages remains a persistent challenge due to limited parallel corpora, orthographic variation, under-documented grammar, and high linguistic diversity (Ebrahimi et al., 2023a; Mager et al., 2021). These constraints often render conventional supervised MT approaches ineffective, motivating research into data-efficient and transfer-based techniques.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 119–126

A prominent direction in low-resource MT has been the use of data augmentation to mitigate data scarcity. Back-translation (Sennrich et al., 2016), which generates synthetic parallel data from monolingual corpora, has been shown to improve translation quality across a range of low-resource settings. In parallel, multilingual pre-trained models such as mBART50 (Tang and et al., 2020) and NLLB-200 (Costa-Jussà et al., 2022) have enabled effective cross-lingual transfer by leveraging shared representations across many languages, achieving strong performance even with limited task-specific supervision.

Community-driven initiatives such as AmericasNLP have played a central role in advancing MT research for indigenous languages of the Americas by providing benchmark datasets, standardized evaluation protocols, and shared tasks (Ebrahimi et al., 2023a; Mager et al., 2021; Ebrahimi et al., 2024). Results from recent AmericasNLP shared tasks indicate that the most successful systems consistently combine multilingual pre-trained models with synthetic data generation. For example, strong submissions have employed NLLB-200 with back-translation (Gow-Smith and Sánchez Villegas, 2023) or explored multilingual transfer using models such as mBART50 and M2M-100 (Tonja et al., 2023). Earlier work, including IndT5 (Nagoudi et al., 2021), further demonstrated the benefits of training models directly on indigenous language corpora.

More recently, synthetic data generation via forward translation has gained attention as a scalable alternative to back-translation for languages with scarce monolingual resources. The MultiScript30k dataset, for instance, translated Spanish captions from Multi30k into indigenous languages such as Aymara and Guarani using NLLB-200(Driggers-Ellis et al., 2025), illustrating the potential of high-quality synthetic parallel data to improve MT performance when paired with robust multilingual models.

Building on these advances, our work investigates the effectiveness of synthetic parallel data augmentation for Aymara–Spanish and Guarani–Spanish translation. We situate our results within the context of the AmericasNLP 2023 shared task, comparing against established benchmarks to assess the impact of forward-translated synthetic data on low-resource indigenous language MT.

Lang	Setting	Split	Total	Valid	Drop %
aym	Curated	Train	6,531	6,092	5.54
		Dev	996	996	0.00
	+Synthetic	Train	35,531	33,712	5.12
		Dev	996	945	5.12
gn	Curated	Train	26,032	25,417	2.36
		Dev	995	995	5.93
	+Synthetic	Train	53,083	52,929	0.29
		Dev	995	995	5.93
quy	Curated	Train	154,008	138,786	9.88
		Dev	996	996	0.20
	+Synthetic	Train	163,651	147,607	9.80
		Dev	996	996	0.20

Table 1: Dataset statistics for Aymara (aym), Guarani (gn), and Quechua (quy) from the AmericasNLP 2023 shared task. "Total" denotes raw sentence pairs; "Valid" denotes pairs retained after preprocessing and filtering. Synthetic data is added only to the training split.

3 Dataset and Methodology

3.1 Curated Datasets

We use curated parallel datasets released as part of the AmericasNLP 2023 Shared Task (Ebrahimi et al., 2023a) for Aymara–Spanish (aym–es), Guarani–Spanish (gn–es), and Quechua–Spanish (quy–es) translation. The Aymara dataset consists of 6,531 training and 996 development sentence pairs, while the Guarani dataset contains 26,032 training and 995 development pairs. For Quechua–Spanish, we use the largest curated dataset in the shared task, comprising 154,008 training sentence pairs and a 996-sentence development set.

These raw datasets are drawn from diverse sources, including governmental documents, educational materials, community-driven projects, and linguistic corpora, and exhibit substantial variation in domain, style, and orthography. As summarized in Table 1, we apply language-specific preprocessing and filtering to remove misaligned, duplicate, or noisy sentence pairs, resulting in moderate reductions in dataset size while preserving the majority of valid training examples.

The curated data spans multiple domains—particularly for Guarani, which includes legal, health, and educational content—leading to substantial lexical and structural diversity. These characteristics reflect well-known challenges for machine translation of Indigenous languages, which often lack standardized orthographies and exhibit rich morphological variation (Mager et al., 2021). The curated datasets serve as the baseline for all experiments, with synthetic data added only to the training splits as described in Section 3.2.

Lang	Setting	Split	Avg Src	Avg Tgt	Tgt/Src
aym	Curated-only	Train	19.62	14.89	0.85
		Dev	11.20	7.09	0.66
	Curated+Synthetic	Train	14.07	10.35	0.74
		Dev	11.23	7.10	0.63
gn	Curated-only	Train	23.23	15.57	0.67
		Dev	11.18	7.21	0.64
	Curated+Synthetic	Train	17.42	12.35	0.71
		Dev	11.18	7.21	0.64
quy	Curated-only	Train	15.17	9.33	0.62
		Dev	11.17	7.44	0.67
	Curated+Synthetic	Train	14.72	9.38	0.64
		Dev	11.17	7.43	0.67

Table 2: Average sentence length statistics for Aymara (aym), Guarani (gn), and Quechua (quy) before and after synthetic data augmentation. Tgt/Src denotes the ratio of average target length to average source length.

Tables 1 and 2 summarize the effect of preprocessing and filtering on dataset size and sentence length statistics for Guarani–Spanish. Filtering removes between approximately 2–10% of sentence pairs across languages, primarily due to extreme length mismatches, duplication, or clear alignment errors, while preserving the majority of valid training examples.

3.2 Synthetic Data Generation

To mitigate data scarcity, we augment the curated parallel datasets with synthetic sentence pairs generated using the MultiScript30k dataset. In this pipeline, the Spanish portion of the Multi30k dataset is forward-translated into Aymara and Guarani using the NLLB-200 (3.3B) multilingual machine translation model (Costa-Jussà et al., 2022). This forward-translation approach produces synthetic parallel data that preserves the semantic content of the original captions while increasing linguistic diversity on the target side.

Unlike conventional back-translation approaches (Sennrich et al., 2016), our method uses Spanish as a high-resource pivot language. This design choice is consistent with recent findings from the AmericasNLP shared tasks, where synthetic data generation via multilingual pretrained models has been shown to improve translation quality for indigenous and low-resource languages (Ebrahimi et al., 2023a; Tonja et al., 2023; Gow-Smith and Sánchez Villegas, 2023). The resulting augmented datasets increase both the volume and variability of training examples, supporting improved generalization in low-resource settings.

3.3 Preprocessing

We apply language-specific preprocessing pipelines to account for the distinct linguistic and orthographic properties of Aymara and Guarani while maintaining a consistent overall workflow.

Guarani. For Guarani, we employ a more specialized preprocessing pipeline to address orthographic variability and phonological representation. All text is normalized using Unicode **NFKC** to canonicalize visually equivalent and compatibility characters, followed by lowercasing and whitespace normalization. We remove non-linguistic symbols while explicitly preserving Guarani-specific diacritics and nasal vowels (e.g., *ã, ẽ, ĩ, õ, ũ*) as well as essential punctuation.

To reduce orthographic inconsistency across sources, we standardize frequent Guarani digraphs by merging space-separated realizations into single units (e.g., “c h”→“ch”, “m b”→“mb”, “n g”→“ng”). This step improves token consistency and reduces fragmentation during subword tokenization. Finally, to mitigate parallel corpus noise, we apply a length-ratio filter based on word counts. Given a source sentence of length L_s and a target sentence of length L_t, a sentence pair is retained only if $1/\tau \leq L_t/L_s \leq \tau$, with $\tau = 2.5$. This heuristic removes highly misaligned sentence pairs while preserving the majority of valid training examples.

Quechua. For Spanish–Quechua (quy), we observe systematic intra-word spacing artifacts on the target side (e.g., `ch aypiqa`, `sin ch i`, `uma ll iqniy`, `ch u`), which introduce tokenization noise and disproportionately affect string-based evaluation metrics. To address this, we apply a deterministic orthographic normalization procedure that targets frequent split patterns without relying on external linguistic resources. Specifically, we (i) merge `ch` or `ll` followed by vowel-initial fragments (e.g.,`ch a...`→`cha...`), (ii) merge three-token se-

quences such as `sin ch i`→`sinchi`, (iii) normalize isolated sequences such as `ch u`→`chu`, and (iv) merge single-character alphabetic fragments with adjacent tokens when they form valid Quechua word patterns. In addition to orthographic normalization, we apply conservative corpus filtering to remove empty or punctuation-only lines, boilerplate or URL-like content, exact duplicates, severe numeric mismatches, extreme length-ratio outliers, and excessively long sentences. We optionally append bilingual dictionary entries as short parallel pairs to the training data to improve lexical coverage during fine-tuning. For evaluation, the same Quechua normalization is applied to both system outputs and reference translations, ensuring that metric computation reflects orthographic equivalence rather than spacing artifacts.

Aymara. For Aymara–Spanish (aym), we apply a deliberately conservative preprocessing pipeline designed to improve data cleanliness while preserving surface orthographic structure. This includes Unicode normalization (NFKC), normalization of apostrophe variants, whitespace normalization, and removal of empty or misaligned sentence pairs. To reduce obvious alignment noise, we filter sentence pairs exhibiting extreme length mismatches using the same length-ratio heuristic applied to Guarani, and remove exact duplicate pairs.

In addition, we address a systematic corpus artifact in which apostrophes— used in common Aymara orthographies to mark ejective or glottalized consonants— are separated from surrounding characters by spurious whitespace (e.g., `jach 'a`, `t 'äw`, `qilqt 'am`). We apply a deterministic orthographic normalization rule that merges intra-word `letter ' letter` patterns into a single token (e.g., `jach'a`, `t'äw`), without introducing any additional linguistic rewriting.

Despite improving overall data consistency, these preprocessing steps yield limited gains in translation quality for Aymara. We hypothesize that this is due to the language's strongly agglutinative morphology, in which grammatical information is encoded through productive suffix chains and phonemic contrasts. Generic normalization combined with subword tokenization may fragment these morphemes, motivating morpheme-aware preprocessing or segmentation strategies as future work (Mager et al., 2021).

3.4 Model Fine-tuning

We fine-tune the multilingual mBART model (`facebook/mbart-large-50`) using the HuggingFace Transformers library. mBART is pretrained using a denoising autoencoding objective across 50 languages, enabling effective cross-lingual transfer in low-resource translation scenarios (Tang and et al., 2020). We use the `MBart50Tokenizer` with Spanish (`es_XX`) as the source language and the appropriate target language tags (`aym_XX` for Aymara and `gn_XX` for Guarani).

To better support Guarani orthography, we extend the tokenizer vocabulary with Guarani-specific characters and frequent multi-character units, including nasal vowels and diacritics (*ã, ẽ, ĩ, õ, ũ*), the combining tilde, and common digraphs (*ch, mb, ng*). After extending the tokenizer, the model's embedding matrix is resized accordingly.

For Aymara–Spanish translation, we train using a learning rate of 2×10^{-5}, batch size 16, gradient accumulation of 4, and 20 epochs. For Guarani–Spanish translation, we use a learning rate of 3×10^{-5}, batch size 8, gradient accumulation of 4, and 15 epochs. All models are optimized using AdamW with a cosine learning rate schedule and warm-up. Early stopping with a patience of three evaluation intervals based on validation BLEU is applied to prevent overfitting. These configurations are consistent with recent AmericasNLP submissions (Gow-Smith and Sánchez Villegas, 2023; Tonja et al., 2023).

3.5 Evaluation Metrics

We evaluate translation quality primarily using chrF++, computed with the official AmericasNLP 2023 evaluation script. While BLEU is monitored during training for early stopping, we do not report BLEU scores in our main results due to its known limitations for morphologically rich and agglutinative languages such as Guarani, Quechua, and Aymara(Popović, 2015, 2017).

4 Experimental Results

4.1 Aymara–Spanish Translation

For Aymara–Spanish translation, mBART fine-tuning yields limited improvements under both curated-only and synthetically augmented settings. While chrF++ increases modestly with synthetic data, overall performance remains low, consistent with prior findings for highly agglutinative languages under subword tokenization. These results

suggest that generic normalization and data augmentation are insufficient for Aymara, motivating morpheme-aware approaches discussed in Section 6.

4.2 Guarani–Spanish Translation

A similar trend is observed for Guarani–Spanish translation. The curated-only model achieves a chrF++ score of 42.00 on the development set. Incorporating synthetic data improves performance to 44.00 chrF++, yielding a gain of +2.00 points. This result indicates that forward-translated synthetic data provides consistent benefits even when training data is limited.

As with Aymara–Spanish, training dynamics for Guarani–Spanish exhibit stable optimization behavior, with improved validation performance across epochs when synthetic data is included. Together, these results demonstrate that the benefits of synthetic augmentation extend beyond a single language pair.

4.3 Quechua–Spanish Translation

For Spanish–Quechua translation, applying deterministic orthographic normalization yields substantial improvements in chrF++. In particular, resolving systematic intra-word spacing artifacts (e.g., *ch aypiqa*, *sin ch i*) reduces token fragmentation and improves character-level alignment. On the AmericasNLP 2023 development set, our model achieves a chrF++ score of 36.6, outperforming both the shared-task baseline and matching the best reported systems from the Sheffield submission. As in prior work, performance under n-gram-based metrics remains challenging due to Quechua's agglutinative morphology, and we therefore emphasize chrF++ as the primary evaluation metric.

4.4 Effect of Synthetic Data Augmentation

Across both language pairs, synthetic data augmentation consistently improves chrF++ scores relative to curated-only training. These findings align with prior work showing that synthetic parallel data generated using high-capacity multilingual models can significantly enhance MT performance for indigenous and other low-resource languages. In line with results reported in recent AmericasNLP shared tasks, our experiments confirm that forward-translated synthetic data can serve as an effective and scalable strategy for improving NMT quality in data-scarce scenarios.

4.5 Comparison with AmericasNLP 2023

We compare our systems against the AmericasNLP 2023 baseline and the best per-language results reported by the University of Sheffield on the official development sets. Following prior work, we focus on chrF++ due to BLEU's known limitations for agglutinative languages(Popović, 2015)(Popović, 2017).

Top-performing submissions primarily relied on large multilingual models such as NLLB-200(Costa-Jussà et al., 2022), often paired with back-translation or other forms of synthetic data generation (Gow-Smith and Sánchez Villegas, 2023; Tonja et al., 2023). In particular, the University of Sheffield system employed NLLB-200 with extensive back-translation and diverse parallel sources, achieving strong results across multiple language pairs (Gow-Smith and Sánchez Villegas, 2023). These findings highlight the effectiveness of data-centric approaches in low-resource indigenous MT.

Our method aligns with this paradigm by leveraging synthetic parallel data generated via forward translation using NLLB-200, while fine-tuning a multilingual mBART model. Although our approach differs from prior work in its exclusive use of forward-translated synthetic data from the MultiScript30k pipeline, our results are consistent with trends observed in the shared task: well-curated synthetic data, when paired with robust multilingual models, can substantially improve translation quality for indigenous languages.

5 Baseline Comparison

To contextualize our results, we compare our system against the best-performing submission reported in the AmericasNLP 2023 shared task for Guarani–Spanish translation (Ebrahimi et al., 2023a). While differences in training data and augmentation strategies prevent a strictly controlled comparison, this provides a meaningful reference point within the established evaluation framework. Our results indicate that synthetic data augmentation improves performance over a curated-only baseline and yields competitive chrF++ scores relative to reported shared-task systems.Table 3 compares our dev-set chrF++ scores with the AmericasNLP 2023 baseline and the best per-language results reported by the University of Sheffield submission. Table 3 shows that synthetic data augmentation improves chrF++ across all three languages.

Team / System (dev chrF)	aym	gn	quy
2021 Baseline (Vázquez et al., 2021)	15.70	19.30	30.40
2021 Best System	28.30	33.60	34.30
Andes	9.22	–	–
CIC-NLP	19.05	21.75	35.62
Helsinki-NLP	33.44	40.42	37.19
LCT-EHU	–	–	38.59
LTLAmsterdam	25.23	32.89	36.81
PlayGround	29.98	33.17	34.38
Sheffield	36.24	39.34	39.52
↑ 2021 (Best – Baseline)	+12.60	+14.30	+3.90
↑ 2023 (Best – 2021 Best)	+7.94	+6.82	+5.22
Ours (Curated-only)	26.85	42.00	37.12
Ours (Curated + Synthetic)	30.82	44.00	37.83

Table 3: Development-set chrF++ comparison for Aymara (aym), Guarani (gn), and Quechua (quy) on the AmericasNLP benchmarks(Ebrahimi et al., 2023b). Prior results are taken from the official shared-task reports.

Gains are largest for Aymara (+3.97 chrF++), followed by Guarani (+2.00), while Quechua shows smaller but consistent improvements (+0.71), reflecting its already large curated training set. Although Aymara shows the largest relative chrF++ gain, absolute performance remains low, reinforcing the need for morphology-aware modeling. While direct comparison is limited by differences in model architecture and training configurations, this provides context for the effectiveness of our synthetic data augmentation approach.

6 Discussion

Our results demonstrate that fine-tuning a multilingual mBART model with synthetic data augmentation is an effective strategy for improving MT performance for low-resource indigenous languages such as Aymara and Guarani. The observed gains in chrF++ indicate that forward-translated synthetic data can meaningfully complement small curated corpora, supporting better generalization in data-scarce settings. Language-specific preprocessing further contributes to these improvements by reducing noise while preserving important morphological and phonological structure.

The effectiveness of our approach is consistent with findings from recent AmericasNLP shared tasks, where multilingual pre-trained models combined with synthetic data generation have emerged as strong baselines for indigenous language MT. While some systems—such as the University of Sheffield submission—achieved strong performance using larger NLLB-200 models and back-translation (Gow-Smith and Sánchez Villegas, 2023), our results suggest that competitive improvements can also be obtained using mBART paired with forward-translated synthetic data. This highlights the flexibility of data-centric augmentation strategies across different model architectures.

Despite these encouraging results, several limitations remain. Our evaluation relies exclusively on automatic metrics, which may not fully capture cultural nuance or idiomatic correctness in indigenous languages (Mager et al., 2023). Future work should therefore incorporate human evaluation involving proficient speakers. Additionally, further investigation into synthetic data quality, diversity, and filtering strategies—as well as alternative augmentation methods such as back-translation—may yield additional gains. Finally, incorporating explicit linguistic knowledge of Aymara and Guarani into model design or preprocessing pipelines (Garvin and Mathiot, 1972) represents a promising direction for improving translation accuracy while ensuring ethical and community-centered deployment of MT systems (Mager et al., 2023).

7 Conclusion and Future Work

In this work, we investigated synthetic data augmentation and language-specific preprocessing for low-resource indigenous language machine translation, focusing on Guarani–Spanish and Quechua–Spanish, with diagnostic experiments on Aymara.

Future work will extend this framework in several directions. First, we plan to conduct human evaluations to better assess translation quality and cultural adequacy. Second, we aim to explore additional indigenous languages, including Aymara, with more targeted preprocessing and data augmentation strategies. A promising direction for future work is the incorporation of visual context for indigenous language translation. Resources such as MultiScript30k provide aligned

image–caption–translation triples for languages including Guarani, enabling investigation of multimodal machine translation in low-resource settings. Future experiments will explore whether visual grounding can improve translation robustness, reduce ambiguity, or accelerate model convergence when parallel text is scarce. This direction aligns with recent interest in data-efficient and multimodal approaches for low-resource MT.

8 Limitations

This study has several limitations. First, evaluation relies exclusively on automatic metrics (chrF++), which may not fully capture translation adequacy, fluency, or culturally grounded meaning in indigenous languages. Human evaluation by proficient speakers is therefore necessary to assess real-world translation quality. While orthographic normalization improves Quechua translation, it relies on deterministic rules and may not generalize to dialectal variation or other Quechua varieties. Second, synthetic data is generated automatically and may introduce systematic biases or translation artifacts, despite filtering and preprocessing. Finally, while we report detailed experiments for Guarani–Spanish, results for other indigenous languages such as Aymara remain limited and are not fully explored in this work.

References

Marta R Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Christopher Driggers-Ellis, Detravious Brinkley, Ray Chen, Aashish Dhawan, Daisy Zhe Wang, and Christan Grant. 2025. Multiscript30k: Leveraging multilingual embeddings to extend cross script parallel data.

Abteen Ebrahimi, Manuel Mager, and et al. 2023a. Findings of the americasnlp 2023 shared task on machine translation into indigenous languages. In *Proceedings of the AmericasNLP Workshop*.

Abteen Ebrahimi, Manuel Mager, Shruti Rijhwani, Enora Rice, Arturo Oncevay, Claudia Baltazar, María Cortés, Cynthia Montaño, John E. Ortega, Rolando Coto-solano, Hilaria Cruz, Alexis Palmer, and Katharina Kann. 2023b. Findings of the AmericasNLP 2023 shared task on machine translation into indigenous languages. In *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*, pages 206–219, Toronto, Canada. Association for Computational Linguistics.

Abteen Ebrahimi et al. 2024. Findings of the americasnlp 2024 shared task on machine translation into indigenous languages. In *Proceedings of the AmericasNLP Workshop*.

Paul L Garvin and Madeleine Mathiot. 1972. *The urbanization of the Guarani language: a problem in language and culture*. State University of New York et Buffalo.

Edward Gow-Smith and Danae Sánchez Villegas. 2023. Sheffield's submission to the americasnlp shared task on machine translation into indigenous languages. In *Proceedings of the Third Workshop on NLP for Indigenous Languages of the Americas (AmericasNLP)*.

Manuel Mager, Elisabeth Mager, Katharina Kann, and Ngoc Thang Vu. 2023. Ethical considerations for machine translation of indigenous languages: Giving a voice to the speakers. *arXiv preprint arXiv:2305.19474*.

Manuel Mager, Arturo Oncevay, et al. 2021. Findings of the americasnlp 2021 shared task on open machine translation for indigenous languages of the americas. In *Proceedings of the AmericasNLP Workshop*.

El Moatez Billah Nagoudi, Yustin Gutiérrez-Vásquez, Arturo Oncevay, Manuel Mager, Xavier Tannier, and Djamé Seddah. 2021. Indt5: A text-to-text transformer for 10 indigenous languages. In *Proceedings of the First Workshop on NLP for Indigenous Languages of the Americas (AmericasNLP)*.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*. Association for Computational Linguistics.

Yliana Rodríguez, Luis Chiruzzo, and Santiago Góngora. 2022. The challenges of creating a corpus of minority languages and its dialects in natural language processing: the case of the south american indigenous language guarani. Presented at the 32nd Meeting of Computational Linguistics in The Netherlands (CLIN32).

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of ACL*.

Yinhan Tang and et al. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. In *Proceedings of EMNLP*.

Atnafu Lambebo Tonja, Hellina Hailu Nigatu, Olga Kolesnikova, Grigori Sidorov, Alexander Gelbukh, and Jugal Kalita. 2023. Enhancing translation for indigenous languages: Experiments with multilingual models. In *Proceedings of the Third Workshop on NLP for Indigenous Languages of the Americas (AmericasNLP)*.

Anthony C Woodbury. 2012. What is an endangered language. *Linguistic society of America*.

Adapting Multilingual NMT to Language Isolates: The Role of Proxy Language Selection and Dialect Handling for Nivkh

Eleonora Izmailova[1,2,3], **Alexey Sorokin**[1,4], **Pavel Grashchenkov**[2]

[1]MSU Center of Artificial Intelligence, [2]RCC MSU, [3]HSE, [4]Yandex

Correspondence: eleon.izm@gmail.com

Abstract

Neural machine translation has achieved remarkable results for high-resource languages, yet language isolates – those with no demonstrated genetic relatives – remain severely underserved, as they cannot benefit from cross-lingual transfer with related languages. We present the first NMT system for Nivkh, a critically endangered language isolate spoken by fewer than 100 fluent speakers in the Russian Far East. Working with approximately 9.5k parallel sentences – expanded through fine-tuned LaBSE sentence alignment – we adapt NLLB-200 to Nivkh-Russian translation. Since Nivkh is absent from NLLB's language inventory, we investigate proxy language token selection, comparing six typologically diverse languages: Bashkir, Kazakh, Halh Mongolian, Turkish, Tajik, and French. We find that using any proxy substantially outperforms random token initialization (18.00–19.02 vs. 15.44 for rus→niv; BLEU 20.72–21.23 vs. 19.05 for niv→rus), confirming the value of proxy-based transfer. However, the choice of which proxy has minimal impact, with all six achieving comparable results despite spanning four language families and two scripts. This suggests that for language isolates, practitioners can select any typologically reasonable proxy without significant performance penalty. We additionally present preliminary experiments on dialect-specific models for Amur and Sakhalin Nivkh. Our findings establish baseline results for future Nivkh NLP research and provide practical guidance for adapting multilingual models to other language isolates.

1 Introduction

Neural machine translation (NMT) has achieved remarkable results for high-resource languages, yet the majority of the world's approximately 7,000 languages remain underserved due to insufficient parallel data (Joshi et al., 2020). This disparity is particularly acute for language isolates – languages with no demonstrated genetic relationship to any other language – which cannot benefit from transfer learning strategies that exploit similarities with related languages (Zoph et al., 2016).

Nivkh presents an extreme case: a critically endangered isolate with only several dozen remaining speakers – none of whom use the language regularly – all over the age of 70. Spoken in the lower Amur River region and Sakhalin Island, it lacks an established orthographic standard and exhibits significant dialectal variation between its Amur and Sakhalin varieties (Gruzdeva and Bugaeva, 2022). Prior to this work, no neural machine translation system existed for Nivkh.

To illustrate the challenges Nivkh poses for NMT, consider the opening of a traditional narrative, shown here with interlinear glossing:

(1) Ӈа ху-ла нивх,
animal kill-ATR person
нам-нама-ғыт-ӯ.
be.good-be.good-COMPL-CONV.3SG
'Жил-был хороший охотник.'
(Eng. There lived a good hunter.)

(2) Ӈа ӷыӈ-р ви-дь, ӈа
animal hunt-CONV.3SG go-IND animal
ӷыӈ-р ви-ке, лаю-дь.
hunt-CONV.3SG go-CONV:SIM rage-IND
'Пока он был на охоте, погода испортилась.'
(Eng. While he was out hunting, the weather turned foul.)

The glosses reveal several properties that make Nivkh challenging for NMT. There is no grammatical gender: Russian requires masculine agreement (хороший охотник 'good.M hunter') with no source-side cue. Reduplication marks intensification (нам-нама 'be.good-be.good'), with the full form нам-нама-ғыт-ӯ reflecting productive agglutinative verb morphology. Most notably, converbal chaining in (2) – where two non-finite forms (CONV.3SG, CONV:SIM) are followed by a finite form (IND), encoding a temporal sequence – must be re-

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 127–137
March 28, 2026

structured into a Russian finite subordinate clause (Пока он был..., погода испортилась).

Massively multilingual models such as NLLB-200 (NLLB Team et al., 2022) have demonstrated capacity for transfer to unseen languages through fine-tuning. However, these models require a language token to identify the source language during encoding and the target language during generation. For languages absent from the model's inventory, practitioners must select a *proxy token* – but principled guidance for this selection remains scarce, particularly for isolates that lack genetic relatives among supported languages.

We address two research questions:

1. For language isolates absent from multilingual models, does proxy language selection significantly impact translation quality, and what factors predict effectiveness?

2. How should dialect variation be handled under severe data constraints?

Our contributions are:

- The first NMT system for Nivkh, achieving BLEU 21.23 (niv→rus) with approximately 7.6k training sentences
- A fine-tuned LaBSE encoder for Nivkh-Russian sentence alignment, enabling corpus expansion by approximately 2k additional parallel sentences
- Systematic comparison of six proxy languages plus random initialization, demonstrating that proxy selection provides substantial benefit over random initialization, but the choice of which proxy has minimal impact
- Preliminary analysis of unified versus dialect-specific model training for Amur and Sakhalin Nivkh

1.1 The Nivkh Language

Nivkh is a language isolate indigenous to the lower Amur River basin and northern Sakhalin Island. The language exhibits several typologically unusual features relevant to NMT adaptation.

Phonology and Orthography. The consonant inventory includes 32 phonemes with contrastive aspiration (п/п', т/т', к/к', қ/қ') and palatalization (д/д', т/т', н/н'), as well as a typologically rare voiceless trill р̌, which patterns with obstruents rather than sonorants (Kreinovich, 1937). The vowel system comprises eight phonemes, including the central vowels ы and ъ; the Amur dialect has additionally developed phonemic vowel length through the historical loss of uvular consonants. Nivkh uses Cyrillic script supplemented with characters absent from Russian – including ғ, ӷ, ӈ, ӽ, and р̌ – necessitating vocabulary expansion when adapting NLLB models.

Morphology. Nivkh is predominantly agglutinative with polysynthetic tendencies, including productive noun incorporation, creating challenges for subword tokenization due to morphophonological alternations at morpheme boundaries. The language employs an elaborate system of 26 numeral classifiers requiring semantic categorization of nominal referents (Panfilov, 1962). The case system exhibits syncretism, and the language features associative plurals distinct from simple plurality.

Syntax. Basic word order is SOV, contrasting with Russian SVO order. Nivkh exhibits "paradoxical finiteness": converbs (typically non-finite) carry person and tense marking, while indicative forms lack such marking. Number agreement is optional, contributing to morphosyntactic variability.

Orthographic variation. Unlike most written languages, Nivkh lacks an established orthographic standard. Sources spanning 1908–2022 exhibit significant spelling variation, compounded by dialectal differences in phoneme inventories (e.g., Sakhalin retains uvular consonant ӷ lost in Amur). This introduces noise into training data.

Dialects. Two major varieties exist: Amur and Sakhalin. These differ substantially in phonology, lexicon, and morphology. Some classifications treat them as separate languages.

1.2 Related Work

Low-resource NMT. Transfer learning from related languages has proven effective for low-resource translation (Zoph et al., 2016; Neubig and Hu, 2018). However, this approach assumes availability of a related high-resource language, unavailable for isolates by definition.

Multilingual models for extremely low-resource languages. NLLB-200 (NLLB Team et al.,

Genre	Sources	Sents
Folklore	shtrn, pnf	3,439
Literary	sng, gdn, ptmn, etc.	2,132
Periodicals	nd	1,353
Educational	grz, tmn	1,150
Transcribed speech	shrsh, kn	564
Religious	bible	331
Total		**9,496**

Table 1: Corpus composition by genre. Sources include linguistic documentation, native speaker recordings, the periodical *Nivkh Dif*, and Bible translations.

2022) supports 200 languages but many endangered languages remain absent. Prior work on adapting NLLB to unseen languages includes Dale (2022) achieving BLEU 19.7/17.7 (myv↔rus) for Erzya with 57k sentence pairs, Tuvan reaching BLEU 25.2/23.2 (tyv↔rus) with 50k pairs (Dale, 2023), and Jerpelea et al. (2025) reporting BLEU 31.0/17.3 (rup↔ron) for Aromanian with 79k sentences. Notably, these languages belong to families represented in NLLB (Uralic, Turkic, Romance), providing transfer advantages unavailable to isolates.

Proxy language selection. For languages absent from multilingual models, practitioners must select a proxy token. Dale (2023) provides practical guidance but does not systematically investigate which factors predict effectiveness. We hypothesized that typological similarity – particularly word order and morphological type – might predict transfer effectiveness independently of genetic relatedness.

2 Data

2.1 Corpus Sources

Our corpus derives from heterogeneous sources spanning both dialects and multiple genres (Table 1).

Linguistic documentation. Glossed texts from Shternberg (1908), Kreinovich (1934), Panfilov (1965), and Gruzdeva and Bugaeva (2022) provided high-quality parallel material with interlinear translations. An existing corpus of approximately 3k verified parallel sentences (Gusev and Idrisov, 2019) partially compiled from these sources served as our starting point. We expanded this to approximately 7k verified pairs by incorporating and manually aligning material from the additional sources described below, and this expanded set was then used for encoder fine-tuning (Section 2.2).

Native speaker recordings. Transcribed audio collections by Shiraishi and collaborators (2002–2015)[1] contributed naturalistic speech data.

Periodicals. The newspaper *Nivkh Dif* (est. 1990), the only Nivkh-language periodical, provided contemporary text. Russian and Nivkh content required alignment as it is not strictly parallel.

Religious texts. Bible translations into both dialects offered structurally aligned material.

2.2 LaBSE Fine-tuning for Sentence Alignment

To expand the parallel corpus from loosely aligned texts, we fine-tuned LaBSE (Feng et al., 2022), a language-agnostic BERT-based sentence encoder. Since Nivkh is absent from LaBSE's training data, zero-shot alignment quality was poor. Before fine-tuning, we applied a mapping of over 50 character-level normalization rules to reconcile the divergent transcription conventions used across sources spanning 1908–2022: e.g., historical variants of the retroflex trill (p̃, p̌, ṗ → p̌), alternate velar nasal forms (ᴎ, ᴎ̨ → ŋ), macron vowels (ā, ō, ī) → plain vowels, and various dash and quotation mark variants to ASCII equivalents. This normalization was applied to both the fine-tuning data and the texts submitted for alignment.

We fine-tuned on 7,064 verified parallel sentences using `MultipleNegativesRankingLoss` (Henderson et al., 2019), reserving 409 sentences for evaluation. This contrastive objective treats other sentences in each batch as negative examples, teaching the model to recognize Nivkh–Russian translation equivalence. Training hyperparameters are shown in Table 2. Model selection used a task-specific metric rather than validation loss: at every 100 training steps, the evaluator re-ran the full alignment pipeline on a held-out text pair and computed the *chain score* – the proportion of aligned sentence pairs forming unbroken monotonic chains (a score of 1.0 indicates fully monotonic alignment with no crossing or missing links). The checkpoint with the highest chain score was saved.

[1] http://ext-web.edu.sgu.ac.jp/hidetos/HTML/SMNStitle.html

Parameter	Value
Base model	LaBSE
Training sentences	7,064
Test sentences	409
Batch size	8
Learning rate	2e-5
Scheduler	Warmup cosine
Warmup steps	1,000
Epochs	2
Optimizer	AdamW
Mixed precision	Yes (AMP)

Table 2: LaBSE fine-tuning hyperparameters.

Alignment procedure. For alignment we used the `lingtrain-aligner` library[2] with the fine-tuned LaBSE model. The library computes pairwise cosine similarities between sentence embeddings within a sliding window, builds an initial monotonic alignment path through the similarity matrix, and then resolves conflicts – regions where the alignment chain is broken – by re-embedding and re-scoring candidate pairs. We used a window size of 10, batch size of 500, and conflict resolution with minimum chain length 2 and maximum conflict length 6. Given the inability to manually verify large portions of the output, we additionally applied strict filtering: minimum cosine similarity threshold of 0.6 (versus the default 0.5) and chain score validation requiring consistent sequential alignment patterns.

Results. From 9,348 Nivkh and 9,128 Russian sentences in unaligned source texts, filtering retained approximately 80% in both languages (7,421 Nivkh, 7,201 Russian). The alignment procedure yielded 1,927 new high-confidence parallel sentence pairs. Combined with the original verified corpus, this produced 9,496 total parallel sentences. Chain score on the held-out evaluation texts improved from 0.35 at initialization to 0.96 after fine-tuning, indicating substantially more reliable cross-lingual sentence matching for Nivkh despite its complete absence from LaBSE's pretraining data.

2.3 Data Splits

Table 3 shows the data splits. The unified dataset was split 80/10/10 stratified by source to maintain genre distribution. Mean sentence length is 11.27 words (std: 9.82, range: 1–120).

[2] https://github.com/averkij/lingtrain-aligner

Dataset	Train	Dev	Test
Unified (all data)	7,599	948	949
Amur dialect	3,595	449	449
Sakhalin dialect	4,004	499	500

Table 3: Dataset splits. Dialect-specific splits are subsets of the unified data, stratified by source to maintain genre distribution.

Dialect distribution. Sakhalin sources comprise 5,003 sentences; Amur sources comprise 4,493 sentences. The dialects differ in mean sentence length (Sakhalin: 9.53 words; Amur: 13.22 words), reflecting substantial genre imbalance: Sakhalin data is predominantly folklore and literary texts (93%), while Amur data spans periodicals (30%), folklore (26%), educational materials (18%), and transcribed speech (16%). This compositional difference may contribute to performance variation beyond purely linguistic factors.

3 Experimental Setup

3.1 Model Architecture

We use NLLB-200-distilled-600M (NLLB Team et al., 2022), a 600M parameter encoder-decoder model supporting 200 languages. The model employs SentencePiece tokenization with a vocabulary of 256k tokens.

3.2 Vocabulary Expansion

Nivkh uses Cyrillic characters absent from standard Russian. We trained a SentencePiece model on Nivkh texts and merged novel subword tokens into NLLB's vocabulary, adding approximately 7k tokens. New token embeddings were initialized as the mean of their constituent characters' embeddings in the original tokenizer.

3.3 Proxy Language Selection

Since Nivkh is absent from NLLB-200, we must select an existing language token to represent Nivkh during fine-tuning. We compared six proxy languages chosen to vary along dimensions of typological similarity, script, and language family (Table 4).

Typologically similar proxies. Bashkir (`bak_Cyrl`), Kazakh (`kaz_Cyrl`), Mongolian (`khk_Cyrl`), and Turkish (`tur_Latn`) share SOV word order and agglutinative morphology with Nivkh. Mongolian has documented lexical

Proxy	Family	Script	Order	Morph.
Bashkir	Turkic	Cyrillic	SOV	Agglut.
Kazakh	Turkic	Cyrillic	SOV	Agglut.
Mongolian	Mongolic	Cyrillic	SOV	Agglut.
Turkish	Turkic	Latin	SOV	Agglut.
Tajik	Iranian	Cyrillic	SOV	Fusional
French	Romance	Latin	SVO	Fusional

Table 4: Proxy languages compared. Nivkh is SOV with agglutinative morphology and uses Cyrillic script. Bashkir, Kazakh, Mongolian, and Turkish share SOV order and agglutinative morphology; Tajik shares SOV order but has fusional morphology; French differs on all typological dimensions.

Proxy	niv→rus		rus→niv	
	BLEU	chrF++	BLEU	chrF++
`bak_Cyrl`	**21.23**	41.44	18.57	42.70
`kaz_Cyrl`	21.08	41.39	18.22	42.55
`tgk_Cyrl`	21.09	**42.20**	18.00	42.97
`khk_Cyrl`	20.75	40.91	18.03	42.68
`tur_Latn`	20.80	41.03	18.08	42.74
`fra_Latn`	20.72	41.17	**19.02**	**43.26**
Random init	19.05	39.05	15.44	37.43

Table 5: Proxy language comparison on unified test set. Best results in bold. The bottom row shows performance with a randomly initialized `nivkh_Cyrl` token (no proxy).

overlap with Nivkh, possibly reflecting historical contact (Müller et al., 2013).

Partially similar proxy. Tajik (`tgk_Cyrl`) shares SOV order and Cyrillic script but has fusional rather than agglutinative morphology, testing whether word order alone predicts effectiveness.

Dissimilar control. French (`fra_Latn`) differs from Nivkh on all dimensions (SVO, fusional, Latin script), serving as a negative control.

Excluded proxies. We avoided Slavic proxies (e.g., Bulgarian, Ukrainian) given that Russian is our target language, reasoning that target-adjacent proxies might behave differently than target-distant ones – though we leave this comparison to future work. Among Cyrillic-script languages in NLLB-200, most are either Slavic or Turkic; Tajik is a notable exception as an Iranian language with Cyrillic script, making it useful for separating script effects from language family effects.

3.4 Training Configuration

All models were trained with identical hyperparameters using the Adafactor optimizer (Shazeer and Stern, 2018) with learning rate 2e-4, weight decay 1e-3, gradient clipping threshold 1.0, and constant learning rate schedule with 500 warmup steps. Batch size was 64 with maximum sequence length 128. Training was bidirectional (niv↔rus) with evaluation every 500 steps and early stopping with patience 3 based on development set chrF++. Maximum training steps was 10k. All experiments used a single NVIDIA A100 80GB GPU.

Convergence behavior. Models using proxy languages converged faster than random initialization: Bashkir at 4k steps, Tajik at 4.5k, Kazakh at 5.5k, French at 5k, Turkish at 6.5k, and Mongolian at 7.5k steps. The randomly initialized model required the full 10k steps, suggesting that proxy tokens provide useful initialization that accelerates learning.

Generation parameters. For evaluation, we used greedy decoding with maximum output length set dynamically as $16 + 1.5 \times |x|$ where $|x|$ is the input sequence length.

3.5 Evaluation

We report BLEU (Papineni et al., 2002) and chrF++ (Popović, 2017) computed with SacreBLEU (Post, 2018) on the held-out test set (949 sentences).

4 Results

4.1 Proxy Language Comparison

Table 5 presents translation quality across all six proxy languages, plus a baseline using a randomly initialized Nivkh token without any proxy.

All proxies substantially outperform random initialization, but differences between proxies are minimal (within 0.5 BLEU for niv→rus and approximately 1 BLEU for rus→niv). We analyze these patterns in Section 5.

4.2 Dialect-Specific Results

We selected Bashkir as the proxy for dialect experiments based on its numerically highest BLEU score in the unified comparison. The cross-dialect evaluation tests whether dialect-specific models outperform the unified model on their respective test sets, and whether models transfer across dialects.

Cross-dialect evaluation reveals near-zero transfer between Amur and Sakhalin varieties (BLEU

	niv→rus		rus→niv	
Training Data	BLEU	chrF++	BLEU	chrF++
Evaluated on Amur test set (449 sentences)				
Unified	18.83	40.62	19.19	42.94
Amur only	17.99	39.44	13.92	40.49
Sakhalin only	2.51	21.08	1.85	13.53
Evaluated on Sakhalin test set (500 sentences)				
Unified	23.67	42.32	17.56	42.27
Amur only	1.41	15.25	1.70	13.76
Sakhalin only	23.16	41.30	19.12	44.16

Table 6: Cross-dialect evaluation using Bashkir proxy. Cross-dialect transfer fails almost completely (BLEU <3), but the unified model performs well on both varieties.

1.41–2.51), confirming their status as highly divergent. However, the unified model matches or exceeds dialect-specific models on both test sets (Amur: 18.83 vs 17.99; Sakhalin: 23.67 vs 23.16), suggesting that data pooling provides benefits without introducing harmful interference. We recommend the unified approach for practical deployment.

5 Discussion

5.1 Proxy vs. Random Initialization

The key findings are: (1) **using any proxy substantially outperforms random initialization** (18.00–19.02 vs. 15.44 for rus→niv; BLEU 20.72–21.23 vs. 19.05 for niv→rus), confirming that proxy language selection provides meaningful transfer; but (2) **the choice of which proxy has minimal impact**, with scores ranging within 1 BLEU across all six proxies.

Bashkir achieves numerically highest BLEU for niv→rus, while French surprisingly achieves highest BLEU for rus→niv. Tajik achieves highest chrF++ in both directions. However, these differences are minimal and likely within noise margins for a test set of this size. Notably, French – our typologically dissimilar control – performs comparably to the SOV/agglutinative proxies, suggesting that NLLB's multilingual pretraining creates sufficiently language-agnostic representations that proxy selection has minimal impact after fine-tuning.

5.2 Why Proxy Choice Doesn't Matter (Much)

Given the clear benefit of using *any* proxy over random initialization, it is surprising that the *choice* of proxy has minimal impact. Several factors may explain this:

Fine-tuning dominates. With 7.6k training sentences and 4k–7.5k optimization steps, the model adapts sufficiently to Nivkh that initial proxy differences become irrelevant. The faster convergence of proxy-based models (4k–7.5k steps vs. 10k for random init) suggests proxies provide a better starting point, but all converge to similar final performance.

NLLB's language-agnostic representations. NLLB-200 was trained on 200 languages with the explicit goal of learning cross-lingually transferable representations. This multilingual pretraining may create a representation space where any proxy provides a reasonable initialization for adapting to a new language.

Script handling via vocabulary expansion. Turkish (Latin script) performs comparably to Cyrillic-script proxies, suggesting that our vocabulary expansion for Nivkh-specific characters adequately handles script differences, eliminating this potential source of variation.

5.3 Implications for Practitioners

These findings have practical implications for researchers working with other language isolates:

1. **Use a proxy, not random initialization**: The 2–3 BLEU improvement and faster convergence justify proxy selection over creating a new language token.

2. **Proxy choice is flexible**: Any typologically reasonable proxy appears sufficient. This reduces the burden of identifying an "optimal" proxy.

3. **Vocabulary expansion is essential**: Adding tokens for characters absent from the base model is critical for handling novel scripts.

4. **Data quality matters more**: Effort is better spent on corpus refinement than proxy optimization.

5.4 Comparison to Prior Work

Table 7 contextualizes our results against prior NLLB adaptation work.

Nivkh achieves BLEU 21.2 with only 7.6k training sentences – competitive with or exceeding Erzya (BLEU 19.7 with 57k sentences) despite

			X→Tgt		Tgt→X	
Lang.	**Family**	**Train**	BLEU	chrF++	BLEU	chrF++
Tuvan	Turkic	50k	25.2	49.9	23.2	49.9
Erzya	Uralic	57k	19.7	38.6	17.7	41.2
Arom.	Romance	79k	31.0	51.0	17.3	45.0
Nivkh	**Isolate**	**7.6k**	**21.2**	**41.4**	**18.6**	**42.7**

Table 7: Comparison with prior NLLB adaptation. Target: Russian for Tuvan, Erzya, Nivkh; Romanian for Aromanian. Nivkh achieves competitive scores with 7–10× smaller corpus and no related languages in NLLB.

having 7× smaller corpus and no related languages in NLLB. Tuvan's higher scores (BLEU 25.2) likely reflect both larger corpus size and the presence of related Turkic languages in NLLB. This suggests that NLLB adaptation can be effective even for isolates with severely limited data, though we note that direct comparison is complicated by differences in language pairs, evaluation sets, and domains.

5.5 Error Analysis

Manual inspection of model outputs on the test set reveals systematic error patterns that correlate with Nivkh's typological properties. We summarise these here with word-level illustrations; full sentence-level examples appear in Appendix D.

Successful translations. On formulaic and structurally simple sentences, the model achieves near-perfect output. Representative cases include converb chains such as Кузиӈ вифке, тав-наӽӷтох виӈ йугыӈ ('went out, walked far, approached a yurt and entered'), kinship introductions like Ыма, ни чхыф наныгын виндра ('Mother, I will go hunt a bear'), and simple predicates such as Нӷак иф муғиӈ видь ('Once he went by boat') – all translated without error (Table 10). These successes cluster in the `shtrn` (Shternberg) subcorpus, where literal, linguist-produced reference translations closely match the model's output register.

Converb Chain Truncation. Nivkh expresses action sequences through chains of converbs (non-finite verb forms), often encoding up to 4–6 sequential actions in a single sentence. The model frequently compresses or reorders these chains, which reflects the challenge of mapping Nivkh's serializing verb strategy onto Russian's preference for finite clause coordination. For instance, the four-step sequence изроӈ йетот инифке, сик ниихарт (skin → cook → eat.at.length → eat.all; 'having skinned and cooked, ate for a long time, ate everything up') is reduced to three steps, dropping the cooking event entirely (Table 11, Ex. c). Similarly, озиӈ кныӈкӈ виӈ вифке (rise → depart → go → go.far) has the second converb misidentified: кныӈкӈ ('departing') → насытив ('having fed'), substituting an unrelated verbal root.

Paradoxical Finiteness Mishandling. Nivkh exhibits "paradoxical finiteness": converbs carry person and tense marking (typically properties of finite verbs), while indicative forms lack such marking. The model occasionally inverts this pattern, producing Russian translations with incorrect tense or person agreement. In folklore texts, third-person narratives sometimes shift unexpectedly to first-person mid-sentence, suggesting the model struggles to track discourse participants across converb boundaries.

Kinship and Gender Confusion. Nivkh lacks grammatical gender but employs distinct kinship terminology with semantic gender. Translation errors frequently conflate sibling terms (ийасх 'sister' → старший брат 'elder brother'), and lineal terms (ола 'daughter' → сын 'son'). These errors are particularly common in folklore narratives where characters are introduced through kinship relations rather than proper names. The absence of morphological gender agreement in Nivkh source text provides no redundant cues for disambiguation.

Classifier-Numeral Constructions. Nivkh's 26-class numeral classifier system presents persistent challenges. When counting entities, speakers must select the appropriate classifier based on semantic properties (e.g., animacy, shape, size). The model occasionally produces semantically incongruent translations, or classifier information is simply dropped, yielding bare numerals without the specificity encoded in the source. Besides, it can conflate proximal and distal demonstratives (ӽуӈ 'this' ↔ ту/тот 'that') by confusing the dual with cardinals: менымыӈ ('both') surfaces as две ('two'), additionally violating Russian gender agreement with masculine referents.

Polysynthetic Boundary Errors. Nivkh's productive noun incorporation creates long polymorphemic words where noun roots incorporate into verb stems. Subword tokenization sometimes segments these forms at semantically inappropriate

boundaries, yielding translations where incorporated objects are either lost entirely or rendered as separate (syntactically incorrect) constituents. This is most evident with culturally specific compounds involving traditional activities (fishing, hunting) where the incorporated noun carries crucial semantic content. For instance, the noun пхӈа ('skin') in ӿуӈ чхыф пхӈа сивуў ('this bear took off its skin') is rendered as нож ('knife'), yielding 'this bear took off its knife' – grammatically sound but semantically wrong at a single morpheme boundary (Table 11, Ex. d). This is preferable to the catastrophic failures observed in longer polysynthetic complexes, where entire clauses disintegrate.

Synonym and register variation. A substantial fraction of apparent errors reflect legitimate lexical variation penalised by reference based metrics. Three pervasive patterns are юрта ↔ дом ('yurt' ↔ 'house'), старик ↔ муж ('old man' ↔ 'husband'), and сказала ↔ ответила ('said' ↔ 'answered'). In each case the model's output is semantically equivalent to the reference but uses a different Russian lexeme, deflating BLEU and chrF++ points (Table 10).

Genre effects. Error rates vary substantially by genre: linguistic documentation with literal translations achieves BLEU scores 3–4× higher than literary texts with freer translation styles. Folklore texts, despite their formulaic structures, suffer from cultural concept gaps (e.g., shamanic terminology, traditional dwelling types) that produce semantic drift even when syntactic structure is preserved.

6 Limitations

Automatic metrics only. We rely on BLEU and chrF++; human evaluation with native speakers would strengthen conclusions but was infeasible given the critically endangered status of the language.

Statistical significance. We report single-run results without confidence intervals. The small differences between proxies may reflect noise rather than meaningful distinctions. Future work should include multiple runs with different random seeds.

Limited proxy selection. We tested six proxies; other languages (e.g., Tungusic languages, which some hypotheses link to Nivkh) are absent from NLLB and could not be evaluated.

Corpus limitations. Despite expansion, the corpus remains small by NMT standards. Genre imbalance (predominantly folklore) may limit generalization. Orthographic variation across sources spanning 1908–2022 introduces noise.

7 Conclusion

We presented the first neural machine translation system for Nivkh, a critically endangered language isolate. Our systematic comparison of six proxy languages reveals that proxy selection has minimal impact on translation quality when fine-tuning NLLB-200 – a finding with practical implications for researchers working with other isolates.

The key insight is that NLLB's multilingual pretraining creates sufficiently language-agnostic representations that any reasonable proxy suffices, freeing practitioners to focus on corpus development rather than proxy optimization. We achieve competitive results (BLEU 21.23) with only 7.6k training sentences, demonstrating that multilingual model adaptation is viable even for isolates with severely limited data.

Our dialect experiments additionally show that Amur and Sakhalin Nivkh exhibit near-zero cross-dialect transfer, yet a unified model trained on pooled data matches or exceeds dialect-specific models on both varieties, thus suggesting that data aggregation is preferable to dialect-specific training under low-resource conditions.

Future work should investigate whether this proxy-agnostic pattern holds for other isolates, explore larger NLLB model variants (1.3B, 3.3B), and conduct human evaluation to assess real-world translation utility for language documentation and revitalization efforts.

Ethics Statement

This work aims to support language documentation and revitalization efforts for Nivkh. Data sources include published linguistic materials, publicly available periodicals, and copyrighted literary texts. Some literary sources remain under copyright protection and were provided for research purposes by the Nogliki District Library (Sakhalin Oblast, Russia) with permission for use in this study. Due to these copyright restrictions, we cannot release the full parallel corpus publicly. However, the subset derived from public domain

sources will be made available[3] to support future research. We acknowledge that machine translation systems for endangered languages should complement rather than replace human language transmission and community-led revitalization efforts.

Acknowledgments

This research is supported by Russian Science Foundation, RSF project 25-28-00552 "Digitalization of the data of an endangered language: Nivkh", realized at Lomonosov Moscow State University. We thank Daria Savina, Eva Gogua, and Sergei Shevelev for their substantial contributions to the original corpus and continued support throughout this work; Maria Medvedeva for her meticulous manual data correction, which ensured the quality of the final dataset; and Sergei Averkiev for early-stage consultation that helped shape the direction of this research. We also thank the Nogliki District Library (Sakhalin Oblast) for providing access to copyrighted Nivkh literary materials essential for this research.

References

David Dale. 2022. The first neural machine translation system for the Erzya language. In *Proceedings of the First Workshop on NLP Applications to Field Linguistics*, pages 45–53, Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

David Dale. 2023. How to fine-tune a NLLB-200 model for translating a new language. https://cointegrated.medium.com/how-to-fine-tune-a-nllb-200-model-for-translating-a-new-language-a37fc706b865. Accessed: 2025-12-26.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Ekaterina Gruzdeva and Anna Bugaeva. 2022. Nivkh. In Martine Robbeets and Alexander Savelyev, editors, *The Oxford Guide to the Transeurasian Languages*. Oxford University Press, Oxford.

Valentin Yurievich Gusev and Ruslan Ildarovich Idrisov. 2019. Nivkh corpus. RNF project №17-18-01649. Available at: http://nivkh.web-corpora.net.

[3] https://github.com/grapaul/NivkhKurng

Matthew Henderson, Paweł Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerber, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulić, and Tsung-Hsien Wen. 2019. Efficient natural language response suggestion for smart reply. In *Proceedings of the First Workshop on NLP for Conversational AI*. Association for Computational Linguistics.

Andrei-Ionuţ Jerpelea, Alin Radoi, and Sergiu Nisioi. 2025. Dialectal and low resource machine translation for Aromanian. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7209–7228, Abu Dhabi, UAE. Association for Computational Linguistics.

Pratik Joshi, Sebastin Santy, Amar Buber, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293. Association for Computational Linguistics.

Erukhim Abramovich Kreinovich. 1934. Nivkhskie teksty [nivkh texts]. Manuscript materials.

Erukhim Abramovich Kreinovich. 1937. *Fonetika nivkhskogo (gilyatskogo) yazyka [Phonetics of the Nivkh (Gilyak) Language]*. Izdatel'stvo AN SSSR, Moscow-Leningrad.

André Müller, Viveka Velupillai, Søren Wichmann, Cecil H. Brown, Eric W. Holman, and 1 others. 2013. ASJP world language tree of lexical similarity: Version 4. Automated Similarity Judgment Program.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 875–880. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Celebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Baez, Gabriel Battber, Shruti Bhosale, and 28 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Vladimir Zinov'evich Panfilov. 1962. *Grammatika nivkhskogo yazyka. Chast' 1 [Grammar of the Nivkh Language. Part 1]*. Izdatel'stvo AN SSSR, Moscow-Leningrad.

Vladimir Zinov'evich Panfilov. 1965. *Grammatika nivkhskogo yazyka. Chast' 2 [Grammar of the Nivkh Language. Part 2]*. Nauka, Moscow-Leningrad.

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, pages 311–318. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 4596–4604. PMLR.

Lev Yakovlevich Shternberg. 1908. *Materialy po izucheniyu gilyatskogo yazyka i fol'klora [Materials for the Study of the Gilyak Language and Folklore]*. Imperatorskaya Akademiya Nauk, St. Petersburg.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575. Association for Computational Linguistics.

A Nivkh Alphabet

Table 8 presents the Nivkh Cyrillic alphabet as used in the corpus. Characters unique to Nivkh or with Nivkh-specific phonetic values are shown in **bold**.

Table 8: The Nivkh Cyrillic alphabet. The apostrophe marks aspiration. Characters marked † are Sakhalin dialect only.

А а	Б б	В в	Г г	**Ғ ғ**†	**Ғ ғ**	**Ғ ғ**	Д д
Е е	Ё ё	Ж ж	З з	И и	Й й	К к	К' к'
Ӄ ӄ	Ӄ' ӄ'	Л л	М м	Н н	**Ӈ ӈ**	О о	П п
П' п'	Р р	**Ҏ ҏ**	С с	Т т	Т' т'	У у	**Ӯ ӯ**†
Ф ф	Х х	**Ӽ ӽ**	**Ӿ ӿ**	Ц ц	Ч ч	Ш ш	Щ щ
Ъ ъ	Ы ы	Ь ь	Э э	Ю ю	Я я		

B Glossing Abbreviations

3SG	third person singular
ATR	attributive
COMPL	completive
CONV	converb
IND	indicative
SIM	simultaneity

C Corpus Examples

Table 9 shows parallel sentences from the training corpus across all genres.

Table 9: Training corpus examples. **S** = Nivkh, **R** = Russian, **E** = English.

Educational

S Ныух ньрывух Москваух хоғдьра.
R Сегодня в моем доме в Москве холодно.
E 'It's cold in my house in Moscow today.'

Folklore

S Кэрӈ лырр к'нык чаӷртох виӈан к'нык ығуин выч ғой ӽумдьра.
R Когда, пойдя вдоль моря, к трем мысам подойдешь, на конце мыса будет находиться железная лиственница.
E 'When you walk along the sea and reach the three capes, you will find an iron larch at the end of the cape.'

Periodical

S Оғлағу ӽыскла Север п'иӈгу культура поделкағу ныдьныты, ағағску ныдьныты ӽадьғу.
R Так, ребята учились мастерить поделки, связанные с культурой и бытом коренных малочисленных народов Севера.
E 'The children learned to make handicrafts related to the culture and way of life of the indigenous peoples of the North.'

Literary

S Ньўаӷ, ӽонвайво, уткуоғлагу мен к'еӈ нонӈ пойныт тыӈздьғу.
R Случилось так, что однажды по весне два мальчугана пытались поймать маленького лисенка.
E 'It so happened that one day in the spring, two boys were trying to catch a little fox cub.'

Religious

S Урла ны ӈығар, янў наға тёӈў ғотть?
R Если ты делаешь добро, то тогда почему поник головой?
E 'If you're doing good, then why are you so down?'

Transcribed speech

S Ӽоғот наф ӽыры ӽоӈгут п'оғлагуах ты ӽоӈгут наф п'и пандь ӈалакут, ығркы нивғгу ӈалакут, п'инныдь выкскуй керт.
R Поэтому, сейчас своих детей приучаю, как сама воспитывалась, по старинным обычаям нивхов, чтобы свою традиционную кухню не забывали.
E 'That's why I'm teaching my children, as I was taught, to follow the old Nivkh customs, so that they don't forget their traditional cuisine.'

D Translation Examples

Tables 10 and 11 show representative model outputs (**niv**→**rus**) from the test set. **S** = Nivkh source, **R** = reference, **M** = model output, **E** = English

translation. Differences between R and M are underlined.

Table 10: Successful translations.

(a) Exact match – folklore
S Ыма, ни чхыф наныгын виндра. **R** Матушка, я медведя искать пойду. **M** Матушка, я медведя искать пойду. **E** 'Mother, I will go to hunt a bear.'
(b) Exact match – periodical
S Иф трибунароҳ п'уҧан, зал оздь. **R** Когда он вышел к трибуне, зал встал. **M** Когда он вышел к трибуне, зал встал. **E** 'When he came to the podium, the hall rose.'
(c) Word order – textbook
S Ни ёлкароҳ игрушкағу хрот ҳадь. **R** Я на елку вешаю игрушки. **M** Я на елку игрушки вешаю. **E** 'I hang ornaments on the tree.'
(d) Synonym: муж 'husband' → старик 'old man'
S Мызн ычих муҏ лахчиндра. **R** Наш муж совсем умер. **M** Наш старик уже умер. **E** 'Our husband / old man has died.'
(e) Synonym: закурил 'lit up' → покурил табаку 'smoked tobacco'
S Мыз-ниғывың прафтох йугыҏ, тамх таҏ. **R** Наш Гиляк в свою юрту вошел, закурил. **M** Наш Гиляк в свою юрту вошел, табаку покурил. **E** 'Our Gilyak entered his yurt and smoked.'
(f) Deixis: та 'that' → эта 'this'
S Ҳуң чхаҏ-чхнай итҏ: Кеҏ ивндра. **R** Та деревянная фигура сказала: Новости есть. **M** Эта деревянная фигура сказала: новости есть. **E** 'That / this wooden figure said: There is news.'

Table 11: Characteristic errors. Error locus underlined.

(a) Kinship: дочь 'daughter' → сын 'son'
S Ола, рунд ниғывың-ҧа шоҏ-шшынд. **R** Дочь, что за человека ты привела. **M** Сын, какой-то человек пришел. **E** 'Daughter, what person did you bring.'
(b) Kinship: сестра 'sister' → брат 'brother'
S Ҳуных ехыҏ, пфорох пшыҏ, ийанғей ҳунывыҏ, ийасх уигиҏ. **R** ...его жена была, его сестры не было. **M** ...его жена была, его брата не оказалось. **E** '...his wife was there, his sister was gone.'
(c) Converb chain: cooking step dropped
S Изроҏ йетот инифке, сик ниихарт. **R** освежевав, сварив, долго ели, все съели. **M** освежевав, поели, все съела. **E** 'skinned, cooked, ate long, ate all up'
(d) Root: шкуру 'skin' → нож 'knife'
S Ҳуң чхыф пхҧа сивуҏ, ознкуҏ. **R** Этот медведь свою шкуру снял, выпрямился. **M** Этот медведь свой нож снял, поднял. **E** 'This bear took off his skin, straightened up.'
(e) Tense: present → past
S Т'улф ньың воух оғлағу лақхир вардьғу. **R** ...дети на лыжах соревнуются. **M** ...дети на лыжах соревновались. **E** 'In winter children compete on skis.'
(f) Deixis: оттуда 'from there' → отсюда 'from here'
S Мыз-ниғывың ҳуных вифке, малхолаҧ-вонаҳҏтох виҏ йугыҏ. **R** ...оттуда далеко пошел, в селение пришел... **M** ...отсюда далеко пошел, в селение пришел... **E** '...went far from there, came to a village...'

A Fine-Grained Linguistic Evaluation of Low-Resource Luxembourgish–English MT

Nils Rehlinger
University of Luxembourg / Esch-Belval, Esch-sur-Alzette, Luxembourg
nils.rehlinger@uni.lu

Abstract

Machine translation (MT) evaluation is central in guiding researchers on how to improve a model's performance. Current automatic evaluation practices fail to provide reliable insights into the specific translation errors that occur, especially for low-resource languages. This paper introduces the Lux-MT-Test-Suite, enabling a linguistically motivated and fine-grained analysis of Luxembourgish–English (LB-EN) MT based on 896 test items covering 12 linguistic categories and 36 linguistic phenomena. We compare a baseline local LLM (GEMMA 3), its fine-tuned counterpart (LUXMT), and a proprietary state-of-the-art LLM (GPT-5) to analyse what local LLMs learn through fine-tuning in a low-resource setting and to assess performance differences between local and proprietary systems. The findings identify specific performance gains through fine-tuning, minor degradations, a difference in translation strategies, performance gaps between local and proprietary models, and remaining challenges.

1 Introduction

Machine translation (MT) evaluation is a key step in developing MT models. Its purpose is to guide researchers in discerning which strategies improve a model's performance. The standard procedure for MT evaluation is to assess models on a benchmark consisting of largely random sentences, e.g., FLORES-200 (Costa-Jussà et al., 2022; Manakhimova et al., 2025). However, this procedure fails to provide any information about what kind of errors occur, limiting its utility for diagnosing model weaknesses. As a result, there is a growing interest in evaluation methods that help identify MT errors in a fine-grained manner (Kocmi et al., 2025).

Against this backdrop, this paper introduces the Lux-MT-Test-Suite[1], the first fine-grained test suite for Luxembourgish–English (LB-EN) MT. The test suite consists of a diverse set of linguistically motivated test items that target LB-specific linguistic phenomena. These phenomena are then grouped together into broader linguistic categories.

LB is a West-Germanic language spoken by approximately 320'000 speakers (Fehlen and Heinz, 2016; Entringer et al., 2021). It is one of three official languages in Luxembourg, alongside German (DE) and French (FR). The language is closely related to DE but is characterised by frequent borrowing from FR. Historically, LB was restricted to the spoken domain. Only recently has the language been increasingly used in the written domain, e.g., on news and government websites. Due to the small number of speakers and domain restriction, parallel corpora are rare and LB can be considered as a low-resource language in the MT field.

Using the Lux-MT-Test-Suite, this paper analyses what translation capabilities local LLMs acquire through fine-tuning in a low-resource setting and compare their performance with state-of-the-art (SOTA) proprietary LLMs. Thus, this paper addresses the following explorative research question: *What Luxembourgish linguistic knowledge do local LLMs acquire through fine-tuning?* To answer this question, we fine-tune a GEMMA 3 model on LB parallel corpora, compare its performance to its pre-fine-tuned baseline and a proprietary SOTA LLM (GPT-5), highlighting differences across linguistic categories in the Lux-MT-Test-Suite.

The contributions of this paper are threefold:

1. Introducing Lux-MT-Test-Suite: the first MT test suite for LB-EN.

2. Providing insights into which linguistic phenomena local LLMs learn through fine-tuning in a low-resource setting.

3. Identifying performance gaps between local

[1] https://github.com/greenirvavril/lux-mt-test-suite

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 138–150

LLMs and SOTA proprietary LLMs by conducting a fine-grained evaluation.

2 Related Work

2.1 Automatic MT Evaluation Metrics

As mentioned above, the standard procedure is to evaluate MT systems on a set of largely random sentences. The procedure usually involves computing average scores using automatic evaluation metrics. These metrics are generally designed to correlate with human judgements on translation quality, e.g., BERTSCORE (Zhang et al., 2019), BLEU (Papineni et al., 2002), BLEURT (Sellam et al., 2020), etc. While these scores can be helpful to rank models, they are opaque with regards to specific translation errors influencing the score.

2.2 Automatic Span-level Error Annotation

Recent efforts have attempted to tackle this issue by developing metrics that automatically label translation errors, such as XCOMET (Guerreiro et al., 2024) and GEMSPANEVAL (Juraska et al., 2025). While this approach seems promising, it requires vast amounts of annotated data, which are not available for low-resource languages like LB. Moreover, the accuracy of current SOTA automatic error detection models still shows a significant gap with human performance (Lavie et al., 2025).

2.3 Multidimensional Quality Metric Framework

The Multidimensional Quality Metric (MQM) framework is a popular method designed to assess translation quality by labeling error types and assigning severity levels (Lommel et al., 2014, 2024). MQM consists of an elaborate taxonomy of error categories. However, these categories remain generalist, leaving language-specific grammatical phenomena unexplored.

2.4 MT Test Suites

Given these limitations, MT test suites are a promising complementary tool to diagnose MT systems' shortcomings in a fine-grained manner. Also known as challenge sets, MT test suites consist of a collection of curated test items targeting specific linguistic phenomena. MT test suites have existed since the early 1990s, although their popularity has fluctuated over time (King and Falkedal, 1990; Way, 1991; Heid and Hildenbrand, 1991). The latest advances in MT saw substantial performance boosts, causing automatic metric scores to become saturated and leading researchers to call for more challenging and detailed evaluation methods (Proietti et al., 2025; Kocmi et al., 2025). As a result, MT test suites are regaining popularity, with recent examples including Macketanz et al.'s (2022a) DE-EN test suite, Avelino et al.'s (2022) Portuguese–English (PT-EN) test suite, and Manakhimova et al.'s (2025) Russian–English (RU-EN) test suite, to name a few.

2.5 Luxembourgish NLP

Over the past few years, LB has seen a growing presence in the NLP space. Contributions include instruction fine-tuning datasets (Philippy et al., 2025a; Valline et al., 2025), a treebank (Plum et al., 2024), BERT models fine-tuned for various tasks (Gierschek, 2022; Lothritz et al., 2022; Anastasiou, 2022), ASR models (Gilles et al., 2023b,a), a model for comment moderation (Ranasinghe et al., 2023), a normaliser (Lutgen et al., 2025), embeddings (Philippy et al., 2025b; Michail et al., 2025), and generative models, such as LUXT5 (Plum et al., 2025) and LUXGPT (Bernardy, 2022). Developments in MT include LETZ TRANSLATE (Song et al., 2023) based on OPUS-MT (Tiedemann and Thottingal, 2020), LETZ-MT based on GEMMA 2 3B (Song et al., 2025), and KI-IWWERSETZER[2], a model developed using the OPENNMT ecosystem (Klein et al., 2017).

So far, LB MT evaluation has been limited to reference-based metrics, which ignore the source and are therefore susceptible to quality issues in reference translations (Moghe et al., 2025). In addition, as mentioned earlier, these metrics fail to identify translation errors. The following test suite seeks to address this gap by evaluating MT systems across a plurality of linguistically motivated test items representing LB-specific linguistic categories and phenomena.

Since most previous works in MT used automatic evaluation metrics that leave translation errors unidentified, the question of what exactly local LLMs learn during fine-tuning remains to a large extent under-researched. To address this gap, this paper introduces the Lux-MT-Test-Suite and uses it to evaluate and compare a baseline local LLM (GEMMA 3) with a fine-tuned counterpart, as well as a SOTA proprietary LLM (GPT-5). Through

[2] https://iwwersetzung.lu

this evaluation and comparison, we explore what local LLMs learn through fine-tuning in a low-resource setting and identify performance gaps between local and proprietary LLMs.

3 Method

3.1 Model Selection

While local MT systems for LB exist (see Section 2.5), the models either do not support LB-EN translations or they are out-dated. For this reason, this paper focuses on a fine-tuned version of GEMMA 3 that is currently under development for an on-going MT project (Team et al., 2025). The base model is the instruction fine-tuned 8-bit quantised version with 27 billion parameters. While larger and possibly more performant local models exist, GEMMA 3 is among the largest models that we could run on our hardware. As for the proprietary LLM, we choose the popular GPT-5 model[3] and access it via its API.

3.2 Data Preparation & Fine-tuning

The GEMMA 3 model was fine-tuned using the Unsloth[4] fine-tuning suite. As data, we used `LuxAlign` (Philippy et al., 2025b), consisting of 89k LB-FR and 28k LB-EN segment pairs from RTL News[5]. The reason to also include FR data in the mix is to benefit from cross-lingual transfer learning (Philippy et al., 2023). Since RTL articles are not 1-to-1 translations, LUXEMBEDDER was used with a cosine similarity threshold of .99 to filter out low-equivalence segment pairs, reducing the parallel corpus to 14k segment pairs for LB-FR and 2.5k for LB-EN. We augmented the data using Google Translate[6] to translate LB parliamentary debate transcripts to EN and FR. The augmented data was also filtered using LUXEMBEDDER with a threshold of .98, supplementing the dataset with an additional 20k segment pairs for LB-EN and 18k for LB-FR. All data was checked for duplicates and the minimum segment-length was 5 words. The hyperparameters include a learning-rate of 2e-5 and the model was fine-tuned for one epoch.

[3] `https://chatgpt.com/`, data collected on November 24th, 2025.

[4] `https://unsloth.ai/`

[5] The largest news media outlet in Luxembourg: `https://www.rtl.lu`

[6] Data augmented in August 2025.

3.3 Test Suite Creation

The Lux-MT-Test-Suite is largely based on Macketanz et al.'s (2022a) test suite for DE-EN due to the linguistic similarity between LB and DE. We selected a set of DE source sentences from the DE-EN test suite to translate into LB. The selection was guided by how suitable the source sentences would be for translation while maintaining the grammatical structures relevant to the targeted phenomenon. Other sources include example sentences from the Luxembourgish Online Dictionary[7] (LOD), sentences from Döhmer (2020) and some sentences were also devised by the present author who is a native LB speaker with a background in linguistics.

The examples are designed to require translations that involve restructuring in the target language, thereby posing a challenge to MT systems (Manakhimova et al., 2025).

3.4 Test Suite Overview

The test suite contains 896 test items covering 12 linguistic categories, which are subdivided into a total of 36 linguistic phenomena. Each phenomenon consists of at least five test items (see Table 1 for a category-level overview and Table 3 in Appendix A for a phenomenon-level overview).

The category ***Ambiguity*** contains test items in which a lexeme has multiple possible meanings, requiring the MT system to disambiguate the meaning from the context. ***Coordination & ellipsis*** includes test items containing different kinds of ellipsis that require MT systems to perform syntactical restructuring. ***False friends*** contains lexemes that have a form similar to a corresponding EN lexeme, but different meanings. ***Function word*** includes test items in which focus particles or question tags contribute to pragmatic nuances. ***LDD & interrogatives*** checks long-distance dependencies and related discourse phenomena. ***Lexical morphology*** covers noun formation through the nominalisation of verbs and adjectives, and gender variation in nouns. ***MWE*** (Multi-word entities) contains idioms, prepositional and verbal MWE, and collocations, i.e., (semi-)fixed combinations of lexical items. ***Named entity & terminology*** contains LB place names and festivities, and dates. ***Non-verbal agreement*** checks case and gender agreement between subjects and objects. ***Subordination*** assesses a range of subordinate clause constructions. ***Verb tense/aspect/mood*** checks if MT sys-

[7] `https://lod.lu`

tems correctly handle verbal tenses and persons, including their correct auxiliary and model verbs, as well as verbal inflections. Lastly, ***Verb valency*** examines if MT systems correctly translates verbs with their fixed number of arguments.

Together, these categories target syntactical, morphological, pragmatic, and discourse-level features, providing a fine-grained analysis of difficulties in LB-EN MT.

3.5 Scoring

All test items include a set of evaluation rules in the form of (in-)correct tokens or regular expressions to automatically flag the candidate sentences as *correct* or *incorrect*. The tokens are fixed strings of translated test items representing correct or incorrect solutions. The evaluation rules are either transferred from (Macketanz et al., 2022a) or manually written.

The evaluation process is semi-automatic: MT outputs are flagged for correctness based on the pre-written evaluation rules. Items that fail to be flagged are manually evaluated. This is usually the case for novel MT outputs that were not previously captured by the evaluation rules. The new manual evaluation annotations are then incorporated into the evaluation rules for future use. Since our phenomena and categories differ in sizes, we follow Manakhimova et al. (2025) and report aggregate accuracy score percentages on three levels: micro-average (mean over all items, item-weighted), category macro-average, and phenomenon macro-average.

3.6 Manual Annotation Procedure

Manakhimova et al. (2025) evaluate the candidates on standards of basic accuracy and fluency in addition to the targeted phenomenon. However, we decided to follow (Macketanz et al., 2022a) and strictly focused on the targeted phenomenon to assure that the accuracy scores best reflect the models' performance on the given category. In other words, to preserve differences between categories, errors of category-type A should not influence accuracy scores in category B.

3.7 Statistical Analysis

For system comparison, we follow Manakhimova et al. (2025) by first identifying the highest-scoring system and then testing the remaining systems against it using a one-tailed Z-test with $\alpha = 0.05$. Since some of our phenomena contain a low number of test items, we only perform the statistical analysis on an item-weighted category level and micro-average.

4 Results

This section reports a system-level and category-level performance overview of the three systems under investigation: GEMMA 3, LUXMT, and GPT-5 (see Table 1). Due to space constraints, it is not possible to go into all the phenomena and categories in detail. For this reason, we only highlight and illustrate the most notable differences.

4.1 System Performance Overview

Table 1 reports item-weighted average accuracy scores and statistical significance (†) by linguistic category. The results reveal that GPT-5 outperforms the local models in nearly every category, except in *Coordination & Ellipsis*, *Subordination*, and *Named entity & terminology*, where the performance was matched by LUXMT. The local models match ranks with GPT-5 in the three categories mentioned above, and in *LDD & interrogatives* and *Verb valency*. Furthermore, the results indicate improvements of LUXMT over GEMMA 3 in most categories (see Table 2).

4.2 Category Difficulty Analysis

The most challenging categories are *Named entity & terminology* (36.4%), *MWE* (57.4%), *Lexical morphology* (59.1%), *False friends* (60.8%), and *Non-verbal agreement* (62.3%). The results suggest that most difficulties occur on a lexical level.

The least challenging categories include *Coordination & Ellipsis* (91.7%), *Subordination* (91%), and *LDD & interrogatives* (88.9%). These results suggest strong syntactic control.

4.3 Linguistic Analysis

This subsection provides an in-depth linguistic analysis of linguistic phenomena and items that were challenging to MT systems. The analysis reveals the patterns that LUXMT acquired through fine-tuning, improvements over the GEMMA 3 base model, differences between the local models and GPT-5 in their translation performances, as well as strengths and weaknesses that the models have in common. Invalid translations are marked with an asterisk (*).

category	count	Gemma 3	LuxMT	GPT-5	avg
Ambiguity	50	72.0	74.0	**96.0**†	80.7
Coordination & ellipsis	20	85.0	**95.0**	**95.0**	91.7
False friends	34	52.9	52.9	**76.5**†	60.8
Function word	57	66.7	66.7	**98.2**†	77.2
LDD & interrogatives	30	83.3	90.0	**93.3**	88.9
Lexical morphology	62	50.0	50.0	**77.4**†	59.1
MWE	43	48.8	51.2	**72.1**†	57.4
Named entity & terminology	152	34.2	**37.5**	**37.5**	36.4
Non-verbal agreement	23	43.5	60.9	**82.6**†	62.3
Subordination	37	83.8	**94.6**	**94.6**	91.0
Verb tense/aspect/mood	354	58.8	73.7	**91.8**†	74.8
Verb valency	34	76.5	76.5	**88.2**	80.4
micro-average	896	57.3	65.3	**80.6**†	67.7
macro-average	896	63.0	68.6	**83.6**	71.7

Table 1: Item-weighted average accuracy scores (%) by linguistic category for GEMMA 3, LUXMT, and GPT-5. Highest scores per row are shown in bold. Statistical significance based on Z-test is marked by †. Note that the Z-test was not used for categorical macro-average.

4.3.1 Improvements Through Fine-tuning

Table 2 reports item-weighted average category accuracy score differences between LuxMT and Gemma 3, and between GPT-5 and the best performing local model. Comparing LUXMT with the GEMMA 3 baseline, the results suggest substantial improvements in *Non-verbal agreement* (+17.4%), *Verb tense, Aspect, Mood* (+14.9%), *Subordination* (+10.8%), and *Coordination & Ellipsis* (+10%).

Regarding the *Non-verbal agreement* category, the largest gain is found in the *Genitive* phenomenon (+21.5%, see Table 4 in Appendix A): a case marker expressing possession. It is important to note that the LB genitive has limited grammatical productivity and can mostly be found in lexicalised phrases similar to MWEs (Döhmer, 2018). For example, in (1), *wéinst menger*, literally 'because of me', means 'if it's up to me' or 'for my sake'. In this sense, its meaning is more hedged and less stern in causality than its literal meaning.

(1) Wéinst menger musse mir keng Äppel kafen.
   a. GEMMA 3. *Because of my apples, we don't need to buy any.
   b. LUXMT. *Because of me, we don't have to buy apples.
   c. GPT-5. *Because of me we don't have to buy any apples.

Example (2) shows how LUXMT's syntactic understanding improved over the GEMMA 3 baseline. The test item represents the phenomenon *Gapping* from the category *Coordination & ellipsis* where a verb in the second coordinated clause is omitted:

(2) D'Lena schreift e Bréif an den Tom eng E-Mail.
   a. GEMMA 3. *Lena is writing a letter to Tom, an email.
   b. LUXMT. Lena writes a letter and Tom an email.

Little to no improvements were found in *Named entity & Terminology* (+3.3%), *Ambiguity* (+2%), *Lexical morphology* (+0%), *False friends* (+0%), *Function word* (+0%), and *Verb valency* (+0%). No deteriorations were found on an item-weighted category level. Overall, the results suggest that fine-tuning led to a gain of mostly syntactic and morphological knowledge, and limited lexical knowledge.

4.3.2 Deterioration Through Fine-tuning

Some phenomena show deterioration in accuracy scores of LUXMT in comparison with the GEMMA 3 baseline, namely *Noun formation* (-7.1%) and *Idiom* (-10.5%) (see Table 4 in Appendix A). These degradations can be partially attributed to LUXMT translating too literally, as can be seen in Example (3) from the phenomenon *Idiom*. Idioms are multi-word units in which the meaning goes beyond the individual words. Thus, in most cases, idioms cannot be translated literally and have to be translated as a whole.

(3) Chill deng Nippelen!
   a. GEMMA 3. Cool your jets!
   b. LUXMT. *Chill your nipples!
   c. GPT-5. Calm down!

category	count	LuxMT – Gemma 3	GPT-5 – Top local
Ambiguity	50	+2.0	+22.0
Coordination & ellipsis	20	+10.0	+0.0
False friends	34	+0.0	+23.6
Function word	57	+0.0	+31.5
LDD & interrogatives	30	+6.7	+3.3
Lexical morphology	62	+0.0	+27.4
MWE	43	+2.4	+20.9
Named entity & terminology	152	+3.3	+0.0
Non-verbal agreement	23	+17.4	+21.7
Subordination	37	+10.8	+0.0
Verb tense/aspect/mood	354	+14.9	+18.1
Verb valency	34	+0.0	+11.7
micro-average	896	+8.0	+15.3
macro-average	896	+5.6	+15.0

Table 2: Performance deltas (%) by linguistic category based on item-weighted average accuracy scores, with category counts.

4.3.3 Performance Gaps between Local Models and GPT-5

The results show that GPT-5 outperforms the local models in every category, except *Coordination & Ellipsis*, *Subordination*, and *Named entity & terminology* where the item-weighted category-level accuracy scores between GPT-5 and LUXMT are even (see Table 2). The biggest performance gaps between GPT-5 and the local models are found in categories *Function word* (+31.5%), *Lexical morphology* (+27.4%), *False friends* (+23.6%), *Ambiguity* (+22%), *Non-verbal agreement* (+21.7%), and *MWE* (+20.9%). These results suggest that GPT-5 has much better overall linguistic abilities in translating LB to EN than the local models, including lexical knowledge and idiomatic expression, morphological and syntactic control.

4.3.4 Common Challenges

The phenomenon with the lowest average accuracy scores across all three models is *Proper name & Location* (18%, see Table 4 in Appendix A). LB place names typically have LB and FR forms, and EN usually borrows from FR. Only more well-known place names were translated correctly. Still, different strategies can be observed: GEMMA 3 and LUXMT guess by using a similar place name of a more known location, while GPT-5 simply leaves the source place name intact. LUXMT did learn some additional place names, hence the model scores higher than GEMMA 3 and GPT-5. Example (4) illustrates these differences:

(4) Ech wunnen zu Märel.
   a. GEMMA 3. *I live in Mamer.
   b. LUXMT. I live in Merl.
   c. GPT-5. *I live in Märel.

The phenomenon *Noun formation* (22.6%) has the second lowest average score. LB examples include verb nominalisation where the stem of the verb is suffixed with the *-er* word formation particle or the *-ert* suffix, which also adds a pejorative connotation, e.g., *Pechert* 'traffic warden' from *pechen* 'to stick'[8]. Models struggle with this construction, as it probably has a low frequency in training corpora. Here, GEMMA 3 and LUXMT interpreted *Pechert* as a bus, while GPT-5 seems to interpret the lexeme as a proper name:

(5) An dëser Strooss passéiert de Pechert zweemol den Dag.
   a. GEMMA 3. *The Pechert bus passes this street twice a day.
   b. LUXMT. *The bus passes twice a day on this road.
   c. GPT-5. *On this street Pechert passes by twice a day.

Another interesting observation is GPT-5's **failure to generalise** linguistic rules, even though there is evidence that the model has enough knowledge to do so. There are multiple examples where GPT-5 correctly translates nouns with the *-ert* suffix. However, Example (6) shows that GPT-5 fails to translate *Schneekert* 'sweet toothed person', despite correctly translating the verb *schneeken* 'to snack [on sweets]'. Thus, the model technically has the required knowledge to generalise and add the noun formation suffix *-ert* to the stem of the

[8] Because the traffic warden 'sticks' the parking fine ticket to the car.

verb *schneeken* to derive the meaning, but it fails to do so:

(6) De Schneekert schneekt gären.
   a. GPT-5. *Mr. Schneekert likes to snack.

Yet, GPT-5 correctly translated the same noun in another sentence (7):

(7) Dee Schneekert géif sech am léifsten zwee Desserte bestellen!
   a. GPT-5. That sweet tooth would most like to order two desserts!

As previous research has already shown (Macketanz et al., 2022b), LLMs struggle with translating idioms. The low average score of 24.6% for the *Idiom* phenomenon corroborates this finding. LLMs tend to translate idioms literally, e.g., in (8) all LLMs returned the same candidate sentence:

(8) Hie kuckt mam rietsen A an déi lénks Täsch.
   a. CANDIDATE. *He looks with his right eye into the left pocket.
   b. SOLUTION. He has a lazy eye.

A characteristic attribute of LB is that the genitive case can also be used in combination with family names, where the genitive marker becomes assimilated with the family name. LLMs tend to struggle with this, mainly because they avoid altering named entities, e.g., in (9):

(9) Mir gi mat Müllesch iessen.
   a. GEMMA 3. *We are going to eat with Müllesch.
   b. LUXMT. *We go with Müllesch to eat.
   c. GPT-5. *We are going to eat with Müllesch.
   d. SOLUTION. We are going to eat out with the Müllers.

In line with previous research (Avramidis et al., 2020; Manakhimova et al., 2025) *Resultative predicates* are challenging for the three models. This phenomenon includes constructions where the adjective describes the result of an action expressed by a verb. Similar to MWEs, these are often language-specific and literal translations risk leading to errors. For example, in (10), the correct translation for *eidel drénken* is 'to empty' or 'to finish'. A literal translation like 'to drink empty' is a mistranslation. LLMs also fail to use the correct adjective, e.g., in (11) *platt lafen* 'to trample down' instead of 'to trample flat' or 'to run flat'.

(10) Si huet d'Taass eidel gedronk.
   a. GEMMA 3. *He/She drank the tea empty.
   b. LUXMT. *She drank the cup empty.
   c. GPT-5. *She drank the cup empty.

(11) D'Joggeren hunn d'Wiss platt gelaf.
   a. GEMMA 3. *The joggers ran over the meadows.
   b. LUXMT. *The joggers ran flat on the ground.
   c. GPT-5. *The joggers have trampled the meadow flat.

Another noteworthy issue concerns mistranslations that could be attributed to false friends in non-target languages. These errors were annotated but not factored into the accuracy scores. For example, it is plausible that the model interpreted the LB lexeme *Wiss* 'lawn' as a false friend of DE *Wissen* 'knowledge', and LB *geméit* 'mown' or 'mowed' as DE *gemessen* 'measured':

(12) D'Wiss gëtt ëmmer mëttwochs geméit.
   a. GEMMA 3. The lawn is always mowed on Wednesdays.
   b. LUXMT. The knowledge is always measured on Wednesdays.

5 Discussion

What Luxembourgish linguistic knowledge do local LLMs acquire through fine-tuning? The Lux-MT-Test-Suite shows that fine-tuning local LLMs improves their performance in a wide range of linguistic categories and phenomena, reflecting increased lexical, morphological, and syntactical knowledge and control. However, the largest gains are restricted to morphological and syntactical knowledge, while the gain in lexical knowledge is limited. It is also observable that the fine-tuned model's improvements do not always stack on top of the base model: the overall performance may increase, but the fine-tuned model may mistranslate items that the base model translates correctly (see Table 4 in Appendix A). It is unclear whether an underlying issue, such as catastrophic forgetting

(Liu and Niehues, 2025), or merely inconsistent translation performance is the cause. Iterating the model multiple times over the same test suite could provide some clarity.

What are the performance gaps between local LLMs and SOTA proprietary LLMs? The fine-grained evaluation demonstrated that, in a low-resource setting, GPT-5 is still ahead of local LLMs, even when local LLMs are fine-tuned. The difference between the best performing local model and GPT-5 was statistically significant in five out of 12 categories.

The test suite also helped identify common challenges that still persist. LLMs often translate too literally, resulting in translationese (Gellerstam, 1986), and struggle with idiomatic language, generating text that sounds characteristically non-human. The test suite also showed that, in a low-resource case, lexical LLMs have gaps in lexical knowledge. These cases also revealed that different LLMs have different strategies, resulting in different error-profiles, e.g., interpreting unknown nouns as named entities or relying on similarities with another language. These findings are largely in-line with previous research (Manakhimova et al., 2025).

6 Conclusion

To conclude, this paper introduced the Lux-MT-Test-Suite enabling a fine-grained evaluation of LB-EN MT. To showcase the test suite, we compared a popular local LLM (GEMMA 3) with a fine-tuned counterpart (LUXMT) and a proprietary SOTA LLM (GPT-5), identifying improvements achieved through fine-tuning, performance gaps between local and proprietary models, and differences in translation strategies. The fine-tuned model improved over its baseline across a wide range of categories with minor degradation in some phenomenon-specific performance. The fine-tuned model matched GPT-5's model performance in various categories, but some performance gaps between LUXMT and GPT-5 remain. Common challenges include idiomatic expressions and lexical knowledge, and LLMs still frequently translate too literally, resulting in translationese.

7 Limitations

Annotation: No inter-annotator agreement score could be calculated as this study relied on a single annotator. The results should be interpreted cautiously.

Purpose: To ensure the accuracy of the phenomenon and category-level scores, the candidates were evaluated only on the phenomenon of interest as much as possible. This means that many errors had to be ignored. Consequently, the accuracy scores serve to diagnose translation difficulties rather than to reflect translation quality, as the phenomena are not indicative of the severity of translation errors.

Score reliability: Since some of the data is publicly available, such as the LOD examples, it is possible that the data was scraped and unintentionally included in the training set of the GEMMA 3 base model and GPT-5. This possible contamination risks inflating model performance (Sainz et al., 2023). Another limitation is the binary evaluation of correctness. While binary evaluations are useful for quantifying quality, it is ultimately reductive, especially when concerned with human language which is inherently nuanced and ambiguous (Manakhimova et al., 2025). Moreover, a high score on a phenomenon or category should not necessarily be taken as a guarantee that the MT system masters the given phenomenon or category, as it may be that the test items are not difficult enough (Isabelle et al., 2017).

Distribution: The number of test items per category or phenomenon is not representative of any distribution in corpora or real-world settings. This means that a model performing better on average is not necessarily the most performant for any application. Furthermore, some phenomena and categories contain a small amount of test items, leading to potentially skewed accuracy scores.

Sentence-level evaluation: The Lux-MT-Test-Suite evaluates performance on a segment-level, leaving translation difficulties that might arise on a paragraph or discourse level unexplored (Manakhimova et al., 2025). Future research could include test items that target paragraph-level phenomena.

Future development of the test suite will increase the number of items, include an EN-LB translation direction and implement more LB-specific phenomena.

Acknowledgments

This research is part of the Lux-ASR 2 project[9], supported by the Chambre des Députés.

I would like to sincerely thank Anne-Marie Lutgen, Dr. Alistair Plum, and Dr. Nina Hosseini-

[9] https://infolux.uni.lu/lux-asr/

Kivanani for their valuable feedback, as well as my supervisor, Prof. Dr. Peter Gilles, for his guidance during the project.

References

Dimitra Anastasiou. 2022. Enrich4all: A first luxembourgish bert model for a multilingual chatbot. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 207–212.

Mariana Avelino, Vivien Macketanz, Eleftherios Avramidis, and Sebastian Möller. 2022. A test suite for the evaluation of portuguese-english machine translation. In *International Conference on Computational Processing of the Portuguese Language*, pages 15–25. Springer.

Eleftherios Avramidis, Vivien Macketanz, Ursula Strohriegel, Aljoscha Burchardt, and Sebastian Möller. 2020. Fine-grained linguistic evaluation for state-of-the-art machine translation. *arXiv preprint arXiv:2010.06359*.

Laura Bernardy. 2022. *A Luxembourgish GPT-2 Approach Based on Transfer Learning*. Ph.D. thesis, Master's thesis, University of Trier.

Marta R Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Caroline Döhmer. 2018. A new perspective on the luxembourgish genitive. In *Germanic genitives*, pages 15–36. John Benjamins Publishing Company.

Caroline Döhmer. 2020. *Aspekte der luxemburgischen Syntax*. BoD–Books on Demand.

Nathalie Entringer, Peter Gilles, Sara Martin, and Christoph Purschke. 2021. Schnëssen. surveying language dynamics in luxembourgish with a mobile research app. *Linguistics Vanguard*, 7(s1):20190031.

Fernand Fehlen and Andreas Heinz. 2016. *Die Luxemburger Mehrsprachigkeit: Ergebnisse einer Volkszählung*. transcript Verlag.

Martin Gellerstam. 1986. Translationese in swedish novels translated from english. *Translation studies in Scandinavia*, 1:88–95.

Daniela Gierschek. 2022. Detection of sentiment in luxembourgish user comments.

Peter Gilles, Léopold Edem Ayité Hillah, and Nina HOSSEINI KIVANANI. 2023a. Asrlux: Automatic speech recognition for the low-resource language luxembourgish. In *20. International Conference of Phonetic Sciences (ICPhS)*. Guarant International, Prague, Unknown/unspecified.

Peter Gilles, Nina HOSSEINI KIVANANI, and Léopold Edem Ayité Hillah. 2023b. Lux-asr: Building an asr system for the luxembourgish language. In *2022 IEEE Spoken Language Technology Workshop (SLT) SLT 2022*.

Nuno M Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André FT Martins. 2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Ulrich Heid and Elke Hildenbrand. 1991. Some practical experience with the use of test suites for the evaluation of systran. In *the Proceedings of the Evaluators' Forum, Les Rasses. Citeseer*.

Pierre Isabelle, Colin Cherry, and George Foster. 2017. A challenge set approach to evaluating machine translation. *arXiv preprint arXiv:1704.07431*.

Juraj Juraska, Tobias Domhan, Mara Finkelstein, Tetsuji Nakagawa, Geza Kovacs, Daniel Deutsch, Pidong Wang, and Markus Freitag. 2025. MetricX-25 and GemSpanEval: Google Translate submissions to the WMT25 evaluation shared task. In *Proceedings of the Tenth Conference on Machine Translation*, pages 957–968, Suzhou, China. Association for Computational Linguistics.

Margaret King and Kirsten Falkedal. 1990. Using test suites in evaluation of machine translation systems. In *COLING 1990 Volume 2: Papers presented to the 13th International Conference on Computational Linguistics*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. *arXiv preprint arXiv:1701.02810*.

Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, and 1 others. 2025. 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.

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.

Danni Liu and Jan Niehues. 2025. Conditions for catastrophic forgetting in multilingual translation. In *Proceedings of the 5th Workshop on Multilingual Representation Learning (MRL 2025)*, pages 347–359.

Arle Lommel, Serge Gladkoff, Alan K Melby, Sue Ellen Wright, Ingemar Strandvik, Katerina Gasova, Angelika Vaasa, Andy Benzo, Romina Marazzato Sparano, Monica Foresi, and 1 others. 2024. The multi-range theory of translation quality measurement: Mqm scoring models and statistical quality control. In *Proceedings of the 16th Conference of the Association for Machine Translation in the Americas (Volume 2: Presentations)*, pages 75–94.

Arle Lommel, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional quality metrics (mqm): A framework for declaring and describing translation quality metrics. *Tradumàtica*, (12):0455–463.

Cedric Lothritz, Bertrand Lebichot, Kevin Allix, Lisa Veiber, Tegawende Bissyande, Jacques Klein, Andrey Boytsov, Clément Lefebvre, and Anne Goujon. 2022. Luxembert: Simple and practical data augmentation in language model pre-training for luxembourgish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5080–5089.

Anne-Marie Lutgen, Alistair Plum, Christoph Purschke, and Barbara Plank. 2025. Neural text normalization for luxembourgish using real-life variation data. In *Proceedings of the 12th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 115–127.

Vivien Macketanz, Eleftherios Avramidis, Aljoscha Burchardt, He Wang, Renlong Ai, Shushen Manakhimova, Ursula Strohriegel, Sebastian Möller, and Hans Uszkoreit. 2022a. A linguistically motivated test suite to semi-automatically evaluate german–english machine translation output. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 936–947.

Vivien Macketanz, Shushen Manakhimova, Eleftherios Avramidis, Ekaterina Lapshinova-koltunski, Sergei Bagdasarov, and Sebastian Möller. 2022b. Linguistically motivated evaluation of the 2022 state-of-the-art machine translation systems for three language directions. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 432–449, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Shushen Manakhimova, Maria Kunilovskaya, Ekaterina Lapshinova-Koltunski, and Eleftherios Avramidis. 2025. Fine-grained evaluation of english-russian mt in 2025: Linguistic challenges mirroring human translator training. In *Proceedings of the Tenth Conference on Machine Translation*, pages 866–877.

Andrianos Michail, Corina Raclé, Juri Opitz, and Simon Clematide. 2025. Adapting multilingual embedding models to historical luxembourgish. In *Proceedings of the 9th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2025)*, pages 291–298.

Nikita Moghe, Arnisa Fazla, Chantal Amrhein, Tom Kocmi, Mark Steedman, Alexandra Birch, Rico Sennrich, and Liane Guillou. 2025. Machine translation meta evaluation through translation accuracy challenge sets. *Computational Linguistics*, 51(1):73–137.

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*, pages 311–318.

Fred Philippy, Laura Bernardy, Siwen Guo, Jacques Klein, and Tegawendé F Bissyandé. 2025a. Luxinstruct: A cross-lingual instruction tuning dataset for luxembourgish. *arXiv preprint arXiv:2510.07074*.

Fred Philippy, Siwen Guo, and Shohreh Haddadan. 2023. Towards a common understanding of contributing factors for cross-lingual transfer in multilingual language models: A review. *arXiv preprint arXiv:2305.16768*.

Fred Philippy, Siwen Guo, Jacques Klein, and Tegawende Bissyande. 2025b. Luxembedder: A cross-lingual approach to enhanced luxembourgish sentence embeddings. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 11369–11379.

Alistair Plum, Caroline Döhmer, Emilia Milano, Anne-Marie Lutgen, and Christoph Purschke. 2024. Luxbank: The first universal dependency treebank for luxembourgish. *arXiv preprint arXiv:2411.04813*.

Alistair Plum, Tharindu Ranasinghe, and Christoph Purschke. 2025. Text generation models for luxembourgish with limited data: A balanced multilingual strategy. In *Proceedings of the 12th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 93–104.

Lorenzo Proietti, Stefano Perrella, and Roberto Navigli. 2025. Has machine translation evaluation achieved human parity? the human reference and the limits of progress. *arXiv preprint arXiv:2506.19571*.

Tharindu Ranasinghe, Alistair Plum, Christoph Purschke, and Marcos Zampieri. 2023. Publish or hold? automatic comment moderation in luxembourgish news articles. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 968–978.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 7881–7892.

Yewei Song, Saad Ezzini, Jacques Klein, Tegawende Bissyande, Clément Lefebvre, and Anne Goujon. 2023. Letz translate: Low-resource machine translation for luxembourgish. In *2023 5th International Conference on Natural Language Processing (ICNLP)*, pages 165–170. IEEE.

Yewei Song, Lujun Li, Cedric Lothritz, Saad Ezzini, Lama Sleem, Niccolo Gentile, Radu State, Tegawendé F Bissyandé, and Jacques Klein. 2025. Is llm the silver bullet to low-resource languages machine translation? *arXiv preprint arXiv:2503.24102*.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Jörg Tiedemann and Santhosh Thottingal. 2020. Opus-mt–building open translation services for the world. In *Annual Conference of the European Association for Machine Translation*, pages 479–480. European Association for Machine Translation.

Julian Valline, Cedric Lothritz, and Jordi Cabot. 2025. Luxit: A luxembourgish instruction tuning dataset from monolingual seed data. *arXiv preprint arXiv:2510.24434*.

Andrew Way. 1991. Developer-oriented evaluation of mt systems. In *Proceedings of the evaluators' forum*, pages 237–244.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Phenomenon-level Overview

phenomenon	count	Gemma 3	LuxMT	GPT-5	avg
Ambiguity	50	72.0	74.0	**96.0**†	80.7
Lexical ambiguity	50	72.0	74.0	**96.0**	80.7
Coordination & ellipsis	20	85.0	**95.0**	**95.0**	91.7
Gapping	10	70.0	**90.0**	**90.0**	83.3
Sluicing	10	**100.0**	**100.0**	**100.0**	100.0
False friends	34	52.9	52.9	**76.5**†	60.8
Function word	57	66.7	66.7	**98.2**†	77.2
Focus particle	40	55.0	55.0	**97.5**	69.2
Question tag	17	94.1	94.1	**100.0**	96.1
LDD & interrogatives	30	83.3	90.0	**93.4**	88.9
Multiple connectors	9	88.9	88.9	**100.0**	92.6
Topicalization	5	60.0	**80.0**	**80.0**	73.3
Wh-movement	16	87.5	**93.8**	**93.8**	91.7
Lexical morphology	62	50.0	50.0	**77.5**†	59.1
Gender	34	82.4	88.2	**97.1**	89.2
Noun formation	28	10.7	3.6	**53.6**	22.6
MWE	43	48.8	51.2	**72.1**†	57.4
Collocation	12	75.0	**91.7**	**91.7**	86.1
Idiom	19	15.8	5.3	**52.6**	24.6
Prepositional MWE	5	60.0	**80.0**	60.0	66.7
Verbal MWE	7	85.7	85.7	**100.0**	90.5
Named entity & terminology	152	34.2	**37.5**	**37.5**	36.4
Date	9	**100.0**	**100.0**	**100.0**	100.0
Proper name & location	100	20.0	**22.0**	12.0	18.0
Festivities	43	53.5	60.5	**83.7**	65.9
Non-verbal agreement	23	43.5	60.9	**82.6**†	62.3
Coreference	9	77.8	**88.9**	**88.9**	85.2
Genitive	14	21.4	42.9	**78.6**	47.6
Subordination	37	83.8	**94.6**	**94.6**	91.0
Adverbial clause	8	**87.5**	**87.5**	**87.5**	87.5
Cleft sentence	5	80.0	**100.0**	**100.0**	93.3
Infinitive clause	10	80.0	90.0	**100.0**	90.0
Object clause	6	83.3	**100.0**	**100.0**	94.4
Subject clause	8	87.5	**100.0**	87.5	91.7
Verb tense/aspect/mood	354	58.8	73.7	**91.8**†	74.8
Conditional	8	62.5	75.0	**87.5**	75.0
Ditransitive	84	63.1	72.6	**91.7**	75.8
Gerund	9	66.7	66.7	**88.9**	74.1
Imperative	10	70.0	**90.0**	**90.0**	83.3
Intransitive	75	61.3	78.7	**96.0**	78.7
Reflexive	84	42.9	57.1	**90.5**	63.5
Transitive	84	65.5	85.7	**90.5**	80.6
Verb valency	34	76.5	76.5	**88.2**	77.1
Case government	9	88.9	88.9	**100.0**	52.4
Mediopassive voice	8	**75.0**	**75.0**	**75.0**	75.0
Passive voice	10	90.0	90.0	**100.0**	90.0
Resultative predicates	7	42.9	42.9	**71.4**	92.6
micro-average	896	57.3	65.3	**80.6**†	67.6
phen. macro-average	896	67.5	74.9	**86.3**	76.2
categ. macro-average	896	63.0	68.6	**83.6**	71.4

Table 3: Phenomenon-level accuracy scores (%) for GEMMA 3, LUXMT, and GPT-5. Highest scores per row are shown in bold. Statistical significance based on Z-test is marked by †. Note that no Z-test was used for macro-averages.

phenomenon	count	LuxMT – Gemma 3	GPT-5 – Top local
Ambiguity	50	+2.0	+22.0
Lexical ambiguity	50	+2.0	+22.0
Coordination & ellipsis	20	+10.0	+0.0
Gapping	10	+20.0	+0.0
Sluicing	10	+0.0	+0.0
False friends	34	+0.0	+23.6
Function word	57	+0.0	+31.5
Focus particle	40	+0.0	+42.5
Question tag	17	+0.0	+5.9
LDD & interrogatives	30	+6.7	+3.4
Multiple connectors	9	+0.0	+11.1
Topicalization	5	+20.0	+0.0
Wh-movement	16	+6.3	+0.0
Lexical morphology	62	+0.0	+27.5
Gender	34	+5.8	+8.9
Noun formation	28	-7.1	+42.9
MWE	43	+2.4	+20.9
Collocation	12	+16.7	+0.0
Idiom	19	-10.5	+36.8
Prepositional MWE	5	+20.0	-20.0
Verbal MWE	7	+0.0	+14.3
Named entity & terminology	152	+3.3	+0.0
Date	9	+0.0	+0.0
Proper name & location	100	+2.0	-10.0
Festivities	43	+7.0	+23.2
Non-verbal agreement	23	+17.4	+21.7
Coreference	9	+11.1	+0.0
Genitive	14	+21.5	+35.7
Subordination	37	+10.8	+0.0
Adverbial clause	8	+0.0	+0.0
Cleft sentence	5	+20.0	+0.0
Infinitive clause	10	+10.0	+10.0
Object clause	6	+16.7	+0.0
Subject clause	8	+12.5	-12.5
Verb tense/aspect/mood	354	+14.9	+18.1
Conditional	8	+12.5	+12.5
Ditransitive	84	+9.5	+19.1
Gerund	9	+0.0	+22.2
Imperative	10	+20.0	+0.0
Intransitive	75	+17.4	+17.3
Reflexive	84	+14.2	+33.4
Transitive	84	+20.2	+4.8
Verb valency	34	+0.0	+11.7
Case government	9	+0.0	+11.1
Mediopassive voice	8	+0.0	+0.0
Passive voice	10	+0.0	+10.0
Resultative predicates	7	+0.0	+28.5
micro-average	896	+8.0	+15.3
phen. macro-average	896	+7.4	+11.4
categ. macro-average	896	+5.6	+15.0

Table 4: Performance deltas (%) by linguistic phenomenon, with counts.

Assessing and Improving Punctuation Robustness in English-Marathi Machine Translation

Kaustubh Shivshankar Shejole, Sourabh Deoghare and Pushpak Bhattacharyya
Computation for Indian Language Technology (CFILT)
Department of Computer Science and Engineering
Indian Institute of Technology Bombay, Mumbai, India
{kaustubhshejole, sourabhdeoghare, pb}@cse.iitb.ac.in

Abstract

Neural Machine Translation (NMT) systems rely heavily on explicit punctuation cues to resolve semantic ambiguities in a source sentence. Inputting user-generated sentences, which are likely to contain missing or incorrect punctuation, results in fluent but semantically disastrous translations. This work attempts to highlight and address the problem of punctuation robustness of NMT systems through an English-to-Marathi translation. First, we introduce ***Virām***, a human-curated diagnostic benchmark of 54 punctuation-ambiguous English-Marathi sentence pairs to stress-test existing NMT systems. Second, we evaluate two simple remediation strategies: cascade-based *restore-then-translate* and *direct fine-tuning*. Our experimental results and analysis demonstrate that both strategies yield substantial NMT performance improvements. Furthermore, we find that current Large Language Models (LLMs) exhibit relatively poorer robustness in translating such sentences than these task-specific strategies, thus necessitating further research in this area. The code and dataset are available at https://github.com/KaustubhShejole/Viram_Marathi.

1 Introduction

Punctuation is an essential component of written language, playing a critical role in resolving both structural and semantic ambiguity. By signaling how textual elements should be grouped and interpreted, punctuation enables readers to accurately infer the intended meaning of a sentence. Broadly, punctuation serves two complementary functions. First, it marks boundaries between segments of a larger statement and encodes grammatical relationships among those segments. Second, it provides rhetorical cues by indicating emphasis, tone, or nuance associated with particular words or phrases (Kirkman, 2006).

The importance of punctuation can be illustrated through classic examples. For instance, the omission of a comma in the phrase *"Let's eat, Grandma."* transforms an innocent dinner invitation into a cannibalistic implication. Such cases demonstrate how ambiguity naturally arises when punctuation is absent or misused. Similarly, in the sentence "This is known as 'exact' recovery.", quotation marks signal specific emphasis on the term exact, guiding the reader's interpretation. In general, punctuation errors that affect grammatical structure are more consequential than those that affect rhetorical emphasis as the former can fundamentally alter semantic interpretation (Kirkman, 2006; Carey, 1980).

The advent of Transformer (Vaswani et al., 2017) has led to rapid improvements in NMT quality over the last few years. Consequently, the applicability of encoder-decoder and Large Language Model (LLM)-based systems has expanded significantly, now encompassing diverse domains and low-resource languages (Kocmi et al., 2025; Pakray et al., 2025). In this paper, we focus on Marathi[1], an Indo-Aryan language primarily spoken by over 80 million people in the complex linguistic landscape of India, yet considered a low- to mid-resource language (Dabre et al., 2024; Lahoti et al., 2022; Gaikwad et al., 2021).

Figure 1 illustrates an example in which a missing comma in an instruction written on a fire extinguisher could lead to a disaster, highlighting the punctuation sensitivity of NMT systems. Hence, we consider it important to analyze the punctuation sensitivity of current models and to develop techniques to improve their robustness to punctuation, along with an examination of the associated trade-offs. In addition, we emphasize the need to create resources for evaluating punctuation robust-

[1] https://en.wikipedia.org/wiki/Marathi_language

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 151–167
March 28, 2026

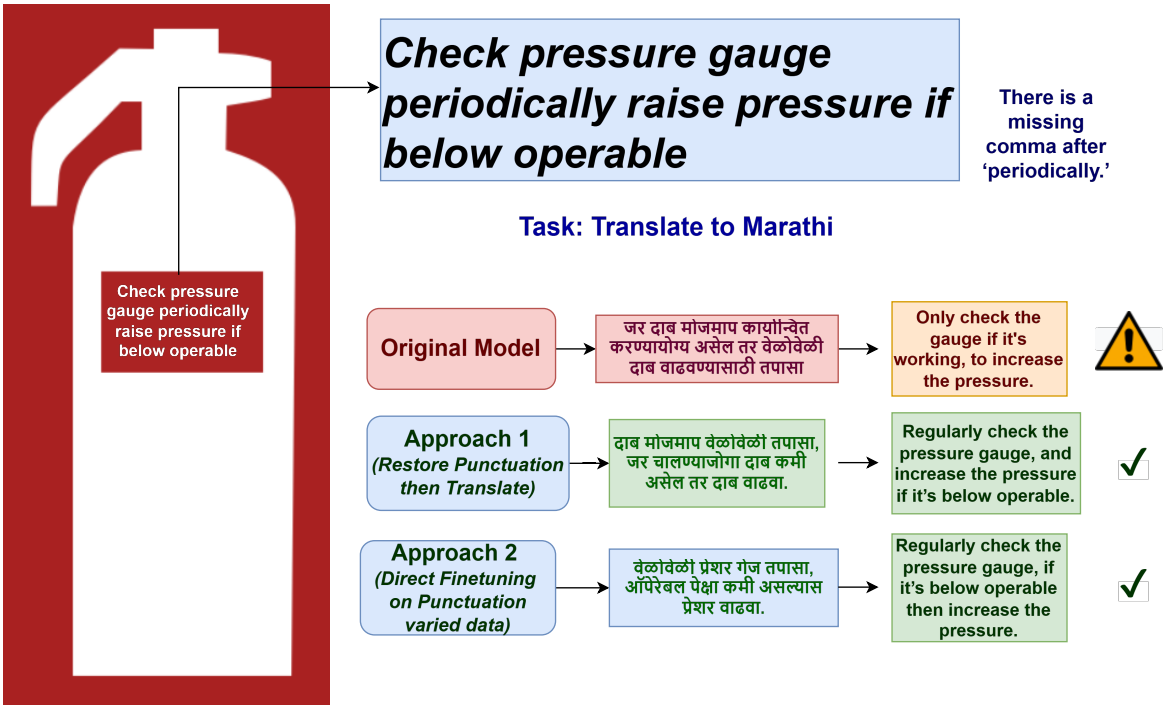


Figure 1: A missing comma can lead to a disaster in English–Marathi machine translation.

ness and to explore strategies for improving translation reliability under punctuation variability.

We first describe the data collection process for assessing the punctuation sensitivity of current Indic models, which was carried out via two native speakers of Marathi and a book by Kirkman (2006), leading to ***Virām***[2], the first English (Written)–English (Meant)–Marathi (Meant) benchmark, where 'Meant' refers to the disambiguated semantic representation. We then apply two approaches for improving the punctuation robustness of current models, and carry out quantitative and qualitative comparison. We also attempt to evaluate the translation quality via prompting of LLMs. All LLMs we considered exhibit lower performance, indicating the need of punctuation-robust approaches to be developed further. Finally, we analyze the performance of our models on standard benchmarks and observe that on the cost of punctuation robustness we might lose slightly on evaluation metrics. Our contributions are as follows:

1. The first study of punctuation robustness in English-Marathi machine translation.

2. A novel diagnostic benchmark called ***Virām*** for English–Marathi punctuation sensitivity analysis. It consists of 54 manually curated, punctuation-ambiguous instances of the form English (Written) – English (Meant) – Marathi (Meant).

3. An analysis of improving punctuation robustness using two complementary approaches: (i) *punctuation restoration in English* then *translate to Marathi*, and (ii) *Direct translation to Marathi*. This dual formulation enables systematic comparison of restoration paradigms. This analysis will help in proliferating further approaches for improving punctuation robustness.

4. A detailed qualitative analysis of model outputs, highlighting strengths, limitations, and error patterns, and identifying directions for future research on punctuation robustness in machine translation.

2 Related Work

Punctuation has long been studied in linguistics for its role in disambiguation, grammatical structure, and rhetorical emphasis (Kirkman, 2006; Carey, 1980; Lukeman, 2011; Trask, 2019). These works establish how punctuation errors can introduce semantic ambiguity, motivating its importance in downstream language technologies.

[2] *Virām chinhe* (विराम चिन्हे) is a Marathi word for punctuation, i.e., signs for marking boundaries by stopping.

English (Written)	English (Meant)	Marathi (Meant)	Punctuation
As the machine develops the forms we use to record data from past projects will be amended.	As the machine develops, the forms we use to record data from past projects will be amended.	जसजशी यंत्रणा विकसित होईल, तसतसे मागील प्रकल्पांतील डेटा रेकॉर्ड करण्यासाठी आम्ही वापरत असलेले फॉर्म्स सुधारित केले जातील.	Comma
What we see, we believe what we hear, we register	What we see, we believe; what we hear, we register	जे पाहतो, त्यावर विश्वास ठेवतो; जे ऐकतो, त्याची नोंद घेतो.	Semi Colon

Table 1: Examples of punctuation ambiguity with English sentences and their Marathi translations in the Virām Benchmark

In Natural Language Processing (NLP), punctuation restoration has been explored primarily as a preprocessing task for text and speech. Early neural approaches modeled the problem using recurrent architectures, including LSTM-based models (Tilk and Alumäe, 2015) and bidirectional RNNs with attention (Tilk and Alumäe, 2016), particularly for spoken language transcripts. Subsequent work extended punctuation restoration to multilingual and transformer-based settings, including large pretrained models for automatic punctuation and capitalization (Nagy et al., 2021; Păiș and Tufiș, 2022). More recently, systems such as Punktuator (Chordia, 2021) and Cadence (Pulipaka et al., 2025) have demonstrated robust multilingual and cross-domain punctuation restoration for both text and speech.

Within machine translation, prior studies have acknowledged the role of punctuation in preserving meaning across languages. For example, Mogahed (2012) examined punctuation effects in English–Arabic MT, highlighting its impact on translation quality. However, explicit modeling of punctuation robustness in MT pipelines remains limited.

Recent research on Indic languages has focused on improving translation quality and evaluation, with models like IndicTrans2 (Gala et al., 2023) supporting translation across all 22 scheduled Indian languages, alongside work on MT metric meta-evaluation (Dixit et al., 2023) and zero-shot evaluation in low-resource settings (Singh et al., 2024). However, English-to-Marathi translation remains highly sensitive to punctuation cues: standard models such as IndicTrans2 often misinterpret syntactic and semantic relations when punctuation is altered or removed. This highlights a critical gap in current MT systems for Marathi. To address it, we develop punctuation-robust MT models tailored for English–Marathi translation, aiming to improve reliability under punctuation variability.

In contrast to prior work, our study lies at the intersection of punctuation restoration and English–Marathi machine translation. We explicitly examine punctuation sensitivity in MT models and analyze the improvement using punctuation-robust modeling approaches, addressing a gap in both Indic MT and punctuation restoration literature.

3 Creating the *Virām* Benchmark

Kirkman (2006) analyze punctuation in the English language, examining how ambiguity can arise from the omission of punctuation marks. For instance, in the sentence, "As the machine develops the forms we use to record data from past projects will be amended," readers must insert a comma after *develops* to derive the intended meaning. This example illustrates the human ability to extract meaning from syntactically ambiguous sentences. Given that Kirkman (2006) is a well-established resource, we manually curated English sentences from this work and, with the assistance of two native Marathi speakers, translated the English (Meant) sentences into Marathi. The resulting diagnostic benchmark comprises 54 punctuation-ambiguous instances, structured as English (Written) – English (Meant) – Marathi (Meant). While the benchmark size is relatively modest, it is commensurate with the significant challenges inherent in data acquisition and curation within this specific domain. Despite this, the rigor applied to its curation ensures that it serves as a high-quality, representative sample for diagnostic evaluation. Table 1 presents selected examples from *Virām*, illustrating the nature of punctuation ambiguities and their corresponding translations. Details regarding the annotation process are provided in Appendix A.1.

4 Methodology

We explore two primary paradigms for achieving punctuation robustness in English-to-Marathi translation.

4.1 Approach 1: *Restore Punctuation then Translate*

In this decouple-and-conquer approach, punctuation is first restored in the English source text before translation, reducing the task to punctuation restoration. We adopt two modeling paradigms for punctuation restoration. In the **token classification** approach, `bert-large-uncased` (Devlin et al., 2019) and `microsoft-mpnet-base` (Song et al., 2020) are used to treat punctuation prediction as a sequence labeling task. In the **text-to-text generation** approach, we fine-tune `google-t5-base` (Raffel et al., 2020) to generate punctuated text from unpunctuated input and also evaluate AI4Bharat's `Cadence` model (Pulipaka et al., 2025) without fine-tuning it.

4.2 Approach 2: *Direct Translation*

This approach aims to improve MT robustness to noisy input. We fine-tune the `IndicTrans2` model[3] on four variants of our internal dataset[4]. We construct four variants using the original data with punctuation (**With Punct**) as a baseline, removing all source punctuation (**Without Punct**), combining both original and punctuation-removed data (**Combined 2x**), and alternately retaining or removing punctuation on a per-sentence basis (**Combined x**). Please note that 'x' refers to the size of the internal fine-tuning dataset. Details for data handling are provided for both the approaches in Appendix B.1 and B.2 respectively. Details regarding fine-tuning and hyperparameter selection are provided in Appendix F.

5 Prompting LLMs

We attempt to evaluate the translation quality of punctuation-ambiguous sentences from English to Marathi using zero-shot and few-shot prompting across three LLMs. The models considered are Sarvam-2b-v0.5[5], a 2-billion-parameter model; Gemma-2-9b[6] (Team, 2024), a 9-billion-parameter model; and LLaMA-3.1-8b[7] (Grattafiori et al., 2024), an 8-billion-parameter model. All three models have been exposed to Indian languages during pre-training. Notably, Sarvam-2b-v0.5 has been trained exclusively on English and Indian languages, including Marathi, using a corpus of approximately one trillion tokens per language. We adopt the same methodology described in Section 4 for each prompting strategy:

1. **Zero-shot prompting**
   (a) Restore punctuation, then translate (see Appendix E.2 for details about the prompt).
   (b) Direct translation (see Appendix E.3 for details about the prompt).
2. **Three-shot prompting**
   (a) Restore punctuation, then translate (see Appendix E.4 for details about the prompt).
   (b) Direct translation (see Appendix E.5 for details about the prompt).

For outputs using correctly punctuated inputs (original sentence-meant), we employed the direct translation strategy in Appendix E.1. For three-shot prompting, we selected three examples from the Virām benchmark, each illustrating a distinct punctuation error involving commas, semicolons, and colons. During evaluation, these examples were excluded from the test set to ensure a fair and unbiased assessment.

6 Results and Analysis

In this section, we present a comprehensive evaluation of the proposed approaches. We first provide a quantitative comparison of all methods (§6.1). We then analyze the impact of the two proposed improvement strategies on the original model's performance across standard benchmarks (§6.2). Next, we examine the effectiveness of different prompting strategies for large language models (§6.3). Finally, we complement the quantitative results with a qualitative analysis to better understand the strengths and limitations of the models (§6.4).

[3] https://huggingface.co/ai4bharat/indictrans2-en-indic-dist-200M

[4] It is an in-house corpus created by professional human translators as part of another project. The internal dataset details are provided at https://github.com/KaustubhShejole/Viram_Marathi

[5] https://huggingface.co/sarvamai/sarvam-2b-v0.5

[6] https://huggingface.co/google/gemma-2-9b

[7] https://huggingface.co/meta-llama/Llama-3.1-8B

Type	Model Name	BLEU	BLEURT-20	COMET	chrF++	chrF2++	LabSE	MuRIL
Baseline	IndicTrans2 en indic 200M (Original Model)	21.72	0.7916	0.7391	59.45	55.38	0.9126	0.7619
Upper Performance Boundary	IndicTrans2 en indic 200M + Input as 'sentence meant'	**26.20**	**0.8082**	**0.7606**	**61.15**	**57.41**	**0.9313**	**0.7915**
Approach 1	Fine-tuned bert-large-uncased + Original Model	23.84	0.7955	0.7595	60.02	56.11	0.9199	0.7806
	Fine-tuned microsoft-mpnet + Original Model	**25.12**	**0.7996**	**0.7597**	60.56	56.79	0.9210	0.7813
	Fine-tuned t5-base + Original Model	24.74	0.7977	0.7586	**60.68**	**56.92**	**0.9230**	**0.7838**
	AI4Bharat's cadence + Original Model	23.44	0.7980	0.7516	60.49	56.69	0.9210	0.7809
Approach 2	Finetuned (w/ punct) (x)	21.21	**0.7830**	0.7426	58.90	54.72	**0.9145**	0.7685
	Finetuned (w/o punct) (x)	24.66	0.7774	0.7417	60.30	56.56	0.9122	**0.7830**
	Finetuned (with and w/o punct) (2x)	24.27	0.7785	**0.7443**	**60.61**	**56.83**	0.9120	0.7794
	Finetuned (alternate with and w/o punct) (x)	24.28	0.7745	0.7433	60.21	56.52	0.9047	0.7761
LLM	GPT-5-mini (Zero-Shot + Direct Translation)	18.69	0.7786	0.7420	52.50	48.82	0.9096	0.7394
	DeepSeek-V3.2 (Zero-Shot + Direct Translation)	**23.41**	**0.7858**	**0.7590**	**58.48**	**54.82**	**0.9197**	**0.7765**

Table 2: Quantitative Analysis on the Virām Benchmark

Benchmark	Model Name	BLEU	BLEURT-20	COMET	chrF++	chrF2++	LabSE	MuRIL
IN22 (CONV)	IndicTrans2 en indic 200 M (Original)	18.95	0.8209	0.8117	51.05	47.74	0.9051	0.7451
	Fine-tuned t5-base + Original Model	17.82	0.8121	0.8077	50.24	47.17	0.8982	0.7404
	Finetuned (w/ punct) (x)	16.08	0.7963	0.7920	49.77	46.65	0.8869	0.7300
	Finetuned (w/o punct) (x)	16.67	0.8034	0.7995	49.68	46.72	0.8927	0.7354
	Finetuned (with and w/o punct) (2x)	17.93	0.8106	0.8065	50.45	47.33	0.8962	0.7408
	Finetuned (alternate with and w/o punct) (x)	17.87	0.8122	0.8082	50.34	47.26	0.8983	0.7411
IN22 (GEN)	IndicTrans2 en indic 200 M (Original)	21.01	0.7920	0.7539	54.22	49.99	0.9153	0.7389
	Fine-tuned t5-base + Original Model	16.86	0.7819	0.7439	53.19	48.84	0.9129	0.7309
	Finetuned (w/ punct) (x)	16.98	0.7627	0.7405	53.23	48.86	0.9098	0.7290
	Finetuned (w/o punct) (x)	17.07	0.7745	0.7422	53.09	48.77	0.9116	0.7307
	Finetuned (with and w/o punct) (2x)	17.11	0.7809	0.7440	53.50	49.12	0.9131	0.7315
	Finetuned (alternate with and w/o punct) (x)	17.06	0.7822	0.7450	53.37	48.99	0.9132	0.7312
FLORES-22	IndicTrans2 en indic 200 M (Original)	18.69	0.7894	0.7616	54.48	50.19	0.9226	0.7425
	Fine-tuned t5-base + Original Model	19.34	0.7818	0.7531	55.02	50.82	0.9213	0.7413
	Finetuned (w/ punct) (x)	19.20	0.7786	0.7513	54.78	50.57	0.9198	0.7400
	Finetuned (w/o punct) (x)	19.27	0.7795	0.7528	54.73	50.57	0.9201	0.7413
	Finetuned (with and w/o punct) (2x)	19.37	0.7810	0.7521	55.11	50.90	0.9204	0.7418
	Finetuned (alternate with and w/o punct) (x)	19.46	0.7825	0.7533	55.11	50.90	0.9218	0.7421

Table 3: Performance Comparison across Benchmark Datasets

6.1 Quantitative Performance

Table 2 reports quantitative results on the Virām benchmark across lexical, semantic, and embedding-based metrics. Details about metrics are provided in Appendix D. The original model with 'sent-written' input serves as the baseline, while providing oracle sentence boundaries establishes an upper bound, yielding substantial gains in BLEU, chrF, and embedding similarity. This gap highlights the impact of correct sentence boundary recovery on translation quality. Pipeline-based punctuation restoration (Approach 1) consistently outperforms the baseline, with `t5-base` and `mpnet` restorers approaching the oracle upper bound, indicating that higher-quality punctuation directly improves translation. Direct fine-tuning (Approach 2) on unpunctuated data yields clear gains in BLEU and chrF, while as expected, training only on punctuated data offers limited improvement. Mixed training improves robustness, particularly in COMET and chrF, but still falls short of the strongest pipeline-based results.

Among LLMs, DeepSeek-V3.2 outperforms GPT 5-mini across all metrics, achieving competitive semantic similarity scores in a zero-shot setting. However, both LLMs remain below the strongest pipeline and oracle-segmentation configurations. Overall, accurate sentence boundary recovery is critical for translation quality on Virām, with pipeline-based restoration most effective when segmentation quality is high, while fine-tuning improves robustness to punctuation variability.

Prompting Strategy	Approach	Model Name	BLEU	BLEURT-20	COMET	chrF++	chrF2++	LabSE	MuRIL
Zero-Shot	Restore then Translate	Llama 3.1 8b	5.16	0.6548	0.6026	37.13	33.18	0.8030	0.6093
		Gemma 2 9b	11.69	0.7260	0.6783	45.89	41.98	0.8783	0.6952
		Sarvam 2b v0.5	9.97	0.6961	0.6736	42.93	38.86	0.8228	0.6468
	Direct Translation	Llama 3.1 8b	6.37	0.6660	0.6192	39.89	35.76	0.8230	0.6314
		Gemma 2 9b	8.94	0.7248	0.6880	45.40	41.33	0.8559	0.6712
		Sarvam 2b v0.5	5.67	0.6300	0.6392	37.47	33.67	0.8102	0.6149
3-Shot Prompting	Restore then Translate	Llama 3.1 8b	4.56	0.6641	0.6113	37.80	33.68	0.8194	0.6120
		Gemma 2 9b	14.84	0.7407	0.6889	49.27	45.50	0.8926	0.7056
		Sarvam 2b v0.5	8.55	0.7352	0.6982	44.48	40.02	0.8509	0.6710
	Direct Translation	Llama 3.1 8b	7.98	0.6776	0.6272	43.21	38.91	0.8216	0.6192
		Gemma 2 9b	10.99	0.7388	0.6938	48.98	44.88	0.8883	0.7013
		Sarvam 2b v0.5	11.01	0.7571	0.7224	49.27	44.74	0.8803	0.7019
Direct Translation using Sentence Meant (Original)	Direct Translation	Llama 3.1 8b	7.87	0.6785	0.6330	39.70	35.84	0.8404	0.6535
		Gemma 2 9b	13.75	0.7256	0.6941	45.85	42.47	0.8829	0.6896
		Sarvam 2b v0.5	10.36	0.6866	0.6725	41.30	37.63	0.8422	0.6550

Table 4: Quantitative Analysis of LLMs via various prompting strategies on the Virām Benchmark

6.2 Performance Analysis on Standard Benchmarks

Table 3 reports automatic evaluation results on IN22 (CONV)[8], IN22 (GEN)[9], and FLORES-22[10]. We compare the original model with pipeline-based punctuation restoration (Approach 1) and direct fine-tuning variants on punctuated, unpunctuated, and mixed data (Approach 2).

On IN22 (CONV), the original model achieves the highest BLEU, while fine-tuned variants show modest drops. Models trained on both punctuated and unpunctuated data outperform single-input variants, with the alternate mixed strategy narrowing the gap with the original on BLEURT-20, COMET, LabSE, and MuRIL. On IN22 (GEN), the original model again outperforms fine-tuned variants. Pipeline-based punctuation restoration harms BLEU and semantic scores, indicating error propagation. Mixed fine-tuning outperforms single-condition models but remains below the original. On FLORES-22, all models perform similarly, with some fine-tuned variants slightly exceeding the original in BLEU and chrF without reducing semantic scores. This suggests that gains in punctuation robustness may come at the cost of slight reductions in certain evaluation metrics.

6.3 Analysis of Prompting Strategies in LLMs

The results in Table 4 show that, among zero-shot prompting strategies, Gemma 2 9B consistently outperforms the other evaluated LLMs across most metrics. When considering all prompting strategies, LLaMA 3.1 8B exhibits comparatively lower performance than Gemma 2 and Sarvam 2B, highlighting the impact of model architecture and pre-training scale on multilingual translation quality. In zero-shot settings, Sarvam performs better under Approach 1, whereas the other models achieve higher scores with Approach 2. Under 3-shot prompting, both LLaMA and Sarvam benefit more from Approach 2, while Gemma continues to achieve superior results with Approach 1.

When these LLM results are compared to the quantitative baselines reported in Table 2, it becomes apparent that sub-10B parameter models generally underperform relative to closed-source models such as DeepSeek-V3.2 and GPT 5-mini, which benefit from more specialized capabilities. For instance, DeepSeek-V3.2 achieves a BLEU score of 23.41 and a BLEURT-20 score of 0.7858, whereas GPT 5-mini in a zero-shot direct translation setting attains BLEU 18.69 and BLEURT-20 0.7786. In contrast, Gemma 2 9b, the best-performing model under zero-shot and three-shot prompting conditions, reaches a BLEU score of

[8] https://huggingface.co/datasets/ai4bharat/IN22-Conv

[9] https://huggingface.co/datasets/ai4bharat/IN22-Gen

[10] https://indictrans2-public.objectstore.e2enetworks.net/flores-22_dev.zip

Sentence (English Input)	Model	Marathi Output	Word-by-Word [English Gloss]	Inferred Meaning	Status
1a. Police say suspect confessed lawyer denies claim	Original	संशयिताच्या वकिलाने दावा फेटाळल्याचे पोलिसांचे म्हणणे	संशयिताच्या (Suspect's) वकिलाने (by lawyer) दावा (claim) फेटाळल्याचे (denied) पोलिसांचे (Police's) म्हणणे (saying)	Police say the suspect's lawyer denied the claim. (Missing the confession part of suspect).	Incorrect
	Approach 2 (with and w/o punct) (2x)	पोलिसांचे म्हणणे आहे की संशयिताने कबूल केलेला दावा वकील नाकारतो.	पोलिसांचे (Police's) म्हणणे (say) आहे (is) की (that) संशयिताने (by suspect) कबूल (confessed) केलेला (done) दावा (claim) वकील (lawyer) नाकारतो (denies).	The lawyer denies the claim that the suspect confessed.	Correct
	Approach 1 (t5-base)	पोलिसांचे म्हणणे आहे की संशयिताने कबूल केलेला वकील दावा नाकारतो.	पोलिसांचे (Police's) म्हणणे (say) आहे (is) की (that) संशयिताने (by suspect) कबूल (confessed) केलेला (done) वकील (lawyer) दावा (claim) नाकारतो (denies).	Police say that the claim is denied by the lawyer that the suspect confessed.	Correct
1b. Police say suspect confessed, lawyer denies claim.	All Models	पोलिसांनी सांगितले की संशयिताने कबुली दिली, वकील दावा नाकारतो.	पोलिसांनी (Police) सांगितले (said) की (that) संशयिताने (suspect) कबुली (confession) दिली (gave), वकील (lawyer) दावा (claim) नाकारतो (denies).	Two separate reports: one confession, one denial.	Correct
2a. Minister says reform failed opposition celebrates	Original	सुधारणांमध्ये अपयशी ठरलेल्या विरोधी पक्षांचा जल्लोषः मंत्री	सुधारणांमध्ये (In reforms) अपयशी (failed) ठरलेल्या (proven) विरोधी (opposition) पक्षांचा (parties') जल्लोष (celebration): मंत्री (Minister)	The Minister notes the celebration of the opposition failed in reforms.	Incorrect
	Approach 2 (with and w/o punct) (2x)	मंत्री म्हणतात, सुधारणा अयशस्वी झाल्याबद्दल विरोधकांनी जल्लोष केला.	मंत्री (Minister) म्हणतात (says), सुधारणा (reform) अयशस्वी (unsuccessful) झाल्याबद्दल (about becoming) विरोधकांनी (by opposition) जल्लोष (celebration) केला (did).	The Minister says the opposition celebrated the failure of reforms.	Correct
	Approach 1 (t5-base)	मंत्री म्हणतात की सुधारणा अयशस्वी झाल्या, विरोधक जल्लोष करतात.	मंत्री (Minister) म्हणतात (says) की (that) सुधारणा (reforms) अयशस्वी (unsuccessful) झाल्या (became), विरोधक (opposition) जल्लोष (celebration) करतात (do).	Minister says reforms failed and the opposition celebrates.	Correct
2b. Minister says reform failed, opposition celebrates.	All models	मंत्री म्हणतात की सुधारणा अयशस्वी झाल्या, विरोधक जल्लोष करतात.	(Same as Approach 1 (t5-base) above)	(Same as Approach 1 (t5-base) above)	Correct
3a. Check pressure gauge periodically raise pressure if below operable	Original	जर दाब मोजमाप कार्यान्वित करण्यायोग्य असेल तर वेळोवेळी दाब वाढवण्यासाठी तपासा	जर (If) दाब (pressure) मोजमाप (gauge) कार्यान्वित (operable) असेल (is) तर (then) वेळोवेळी (periodically) दाब (pressure) वाढवण्यासाठी (to increase) तपासा (check).	Only check the gauge if it's working, to increase pressure.	Incorrect
	Approach 2 (with and w/o punct) (2x)	प्रेशर गेज वेळोवेळी तपासा, जर ऑपेरेबल पेक्षा कमी असेल तर प्रेशर वाढवा.	प्रेशर गेज (Pressure gauge) वेळोवेळी (periodically) तपासा (check), जर (if) ऑपेरेबल (operable) पेक्षा (than) कमी (less) असेल (is) तर (then) प्रेशर (pressure) वाढवा (increase).	Regular checks; increase pressure only if it's too low.	Correct
	Approach 1 (t5-base)	दाब मोजमाप वेळोवेळी तपासा, जर चालण्याजोगा दाब कमी असेल तर दाब वाढवा.	दाब (Pressure) मोजमाप (gauge) वेळोवेळी (periodically) तपासा (check), जर (if) चालण्याजोगा (operable) दाब (pressure) कमी (low) असेल (is) तर (then) दाब (pressure) वाढवा (increase).	Check the gauge; if pressure is not operable, increase it.	Correct
3b. Check pressure gauge periodically, raise pressure if below operable.	Original	दाब मोजमाप वेळोवेळी तपासा, जर चालण्याजोगा दाब कमी असेल तर दाब वाढवा.	(Same as Approach 1 (t5-base) above)	(Same as above)	Correct
	Approach 2 (with and w/o punct) (2x)	वेळोवेळी प्रेशर गेज तपासा, ऑपेरेबल पेक्षा कमी असल्यास प्रेशर वाढवा.	वेळोवेळी (Periodically) प्रेशर गेज (gauge) तपासा (check), ऑपेरेबल (operable) पेक्षा (than) कमी (less) असल्यास (if being) प्रेशर (pressure) वाढवा (increase).	(Same as above)	Correct
	Approach 1 (t5-base)	दाब मोजमाप वेळोवेळी तपासा, जर चालण्याजोगा दाब कमी असेल तर दाब वाढवा.	(Same as Approach 1 (t5-base) above)	(Same as above)	Correct
4a. What we see we believe what we hear we register	Original	आपण जे पाहतो त्यावर आपण विश्वास ठेवतो की आपण जे ऐकतो त्यावर आपण नोंदणी करतो.	आपण (We) जे (what) पाहतो (see) त्यावर (on that) आपण (we) विश्वास (believe) ठेवतो (keep) की (OR) आपण (we) जे (what) ऐकतो (hear) त्यावर (on that) आपण (we) नोंदणी (register) करतो (do).	A choice: Do we believe what we see OR register what we hear?	Incorrect
	Approach 2 (with and w/o punct) (2x)	आपण जे पाहतो त्यावर आपण विश्वास ठेवतो, आपण जे ऐकतो त्यावर आपण नोंदणी करतो.	आपण (We) जे (what) पाहतो (see) त्यावर (on that) आपण (we) विश्वास (believe) ठेवतो (keep), आपण (we) जे (what) ऐकतो (hear) त्यावर (on that) आपण (we) नोंदणी (register) करतो (do).	We believe what we see, and we register what we hear.	Correct
	Approach 1 (t5-base)	आपण जे पाहतो, विश्वास ठेवतो, जे ऐकतो, ते नोंदवतो.	आपण (We) जे (what) पाहतो (see), विश्वास (believe) ठेवतो (keep), जे (what) ऐकतो (hear), ते (that) नोंदवतो (register).	Seeing leads to belief, hearing leads to registration.	Correct
4b. What we see we believe; what we hear we register.	All Models	आपण जे पाहतो त्यावर आपण विश्वास ठेवतो; जे ऐकतो त्यावर आपण नोंदणी करतो.	आपण (We) जे (what) पाहतो (see) त्यावर (on that) आपण (we) विश्वास (believe) ठेवतो (keep); जे (what) ऐकतो (hear) त्यावर (on that) आपण (we) नोंदणी (register) करतो (do).	Parallel statements of two human actions.	Correct

Table 5: Qualitative comparison of translation outputs of the original models and fine-tuned models on two approaches.

14.84 and BLEURT-20 of 0.7407. These results further suggest that targeted fine-tuning or augmented input strategies continue to offer substantially higher translation quality, particularly on metrics that are sensitive to semantic adequacy, such as COMET and LabSE.

6.4 Qualitative Analysis

Tables 5 presents a qualitative comparison of translations produced by the original model, Approach 1 (punctuation restoration using T5-base followed by translation), and Approach 2 (Combined 2x: direct fine-tuning on punctuated and unpunctuated data). The examples evaluate the models' ability to resolve syntactic ambiguity, clause boundaries, and semantic scope in punctuation-sparse inputs. The original model consistently struggles with unpunctuated sentences, particularly in headline-style constructions (e.g., 1a, 2a) and instructional text (3a). In news headlines containing multiple reporting verbs, the model frequently misidentifies clause attachment and argument scope, either omitting one of the reported events (1a) or incorrectly attributing actions to the wrong entity (2a). In procedural sentences, it often embeds conditional phrases incorrectly, conflating the condition with the action itself rather than expressing a sequence of operations (3a). Similarly, in parallel or contrastive constructions (4a), the absence of punctuation leads the model to misinterpret coordination as disjunction, resulting in unintended semantic alternation. These errors indicate a strong dependence on explicit punctuation cues for recovering sentence structure.

Approach 1 substantially improves translation quality by restoring punctuation prior to translation. This enables more accurate recovery of clause boundaries, coordination, and reporting structures, yielding correct interpretations in most ambiguous cases (e.g., 1a, 2a, 4a). However, as a pipeline approach, it remains sensitive to errors introduced during punctuation restoration, which can occasionally propagate into the final translation and lead to less natural or slightly misaligned syntactic realizations. Approach 2 consistently produces the most accurate and stable translations across all examples. The fine-tuned model correctly resolves implicit coordination, reporting structures, and conditional logic even in the absence of punctuation, as demonstrated in both news headlines and procedural instructions. Performance remains robust across punctuated and unpunctuated variants, suggesting that the model learns to infer latent sentence structure directly from contextual and syntactic cues rather than relying on surface punctuation.

For punctuated inputs (b variants), all models produce correct translations, confirming that most observed errors in the original model arise from difficulties in handling missing punctuation rather than lexical or morphological limitations. These results demonstrate that fine-tuning may help in combating the punctuation-sensitivity of the original model for English-Marathi machine translation. While automatic metrics show moderate gains, qualitative evaluation reveals substantial improvements in semantic fidelity that are not captured by standard scores.

7 Conclusion and Future Directions

In this study, we focused on assessing and improving punctuation robustness of English–to-Marathi NMT systems. We manually constructed ***Virām***, a diagnostic benchmark that contains punctuation-ambiguous instances. We evaluated two primary approaches: a pipeline-based *restore-then-translated* and *direct fine-tuning* on punctuated and unpunctuated data. Our quantitative and qualitative analyses reveal that both approaches significantly improve punctuation robustness compared to the baseline model. Through qualitative analysis, we identified specific failure modes where NMT models fail to capture the intended meaning in the absence of punctuation. We also evaluated LLMs via zero-shot and few-shot prompting, finding that few-shot prompting improves performance. However, these models lag behind task-specific approaches in preserving meaning for punctuation-ambiguous text, highlighting the need for further research in this area.

We plan to extend this work to other Indic languages to assess whether similar qualitative patterns emerge across language families. Future work should focus on better assessment metrics that check meaning preservation and nuances similar to human judgment, and on exploring hybrid model architectures capable of handling punctuation ambiguity natively, without relying on multi-stage pipelines like multi-task learning approaches. This work opens various research directions for punctuation-robust machine translation.

Limitations

While our study provides valuable insights into punctuation robustness, several inherent limitations bound its scope. The Virām benchmark consists of only 54 manually curated instances; although this size is sufficient for diagnostic evaluation of specific semantic ambiguities, it is not intended as a large-scale test set, with the focus deliberately placed on quality and linguistic complexity rather than volume. Our analysis is restricted to the English–Marathi language pair, and while Marathi represents a morphologically rich, low-resource Indic language, the punctuation-induced errors we observed may differ in nature and frequency for other language families or syntactic structures. Finally, as noted in our qualitative analysis, standard automated metrics such as BLEU and chrF are often insensitive to the subtle semantic shifts introduced by punctuation. While we supplemented these with manual inspection, the scalability of such qualitative evaluation is inherently limited due to the need for expert linguistic annotators.

Ethical Considerations

In alignment with the ACL Ethics Policy, we provide the following disclosures regarding our data, annotation process, and potential societal impact. The English source sentences for the Virām benchmark were manually curated from a well-established linguistic resource (Kirkman, 2006), and the fine-tuning of models in Approach 2 utilized an internal in-house corpus created by professional translators, which we plan to release publicly upon project completion to support further research. Translations for the benchmark were performed by two native Marathi speakers with advanced academic backgrounds (Master's and PhD) in Computer Science. Annotators were fairly compensated for their specialized expertise, and all translations were developed through collaborative discussions to ensure semantic accuracy and cultural relevance. We recognize that machine translation is increasingly used in India for critical applications such as digital governance and agricultural assistance, where punctuation errors can lead to significant semantic shifts and the potential dissemination of incorrect information. While our work aims to improve model robustness and mitigate such risks, we caution that no MT system is entirely error-free, and users in sensitive domains should verify automated translations with human experts. To ensure transparency and reproducibility, we have detailed our experimental setups and prompting strategies in the Appendix and are committed to releasing the Virām diagnostic benchmark publicly to encourage more robust evaluations in Indic language technologies.

Acknowledgments

We thank the anonymous reviewers and the meta-reviewer of LoResMT 2026 for their constructive comments and suggestions, which substantially improved this work. The first author gratefully acknowledges Shalaka Thorat for insightful discussions and significant assistance in developing the evaluation pipelines. We also thank seniors at CFILT, Satyam Kumar and Dhara Gorasiya, for their helpful feedback on the manuscript. We are grateful to the senior linguists at the CFILT Lab, Dr. Nilesh Joshi and Dr. Irawati Kulkarni, for their valuable remarks. Finally, the first author thanks his friends for their continued support and motivation.

References

Gordon Vero Carey. 1980. *Punctuation*. 14. CUP Archive.

Varnith Chordia. 2021. Punktuator: A multilingual punctuation restoration system for spoken and written text. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 312–320.

Raj Dabre, Mary Dabre, and Teresa Pereira. 2024. Machine translation of Marathi dialects: A case study of kadodi. In *Proceedings of the Eleventh Workshop on Asian Translation (WAT 2024)*, pages 36–44, Miami, Florida, USA. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Tanay Dixit, Vignesh Nagarajan, Anoop Kunchukuttan, Pratyush Kumar, Mitesh M Khapra, Raj Dabre, and 1 others. 2023. Indicmt eval: A dataset to meta-evaluate machine translation metrics for indian languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14210–14228.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 878–891.

Saurabh Sampatrao Gaikwad, Tharindu Ranasinghe, Marcos Zampieri, and Christopher Homan. 2021. Cross-lingual offensive language identification for low resource languages: The case of marathi. In *Proceedings of the international conference on recent advances in natural language processing (RANLP 2021)*, pages 437–443.

Jay Gala, Pranjal A Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, and 1 others. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *arXiv preprint arXiv:2305.16307*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, and 1 others. 2021. MuRIL: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

John Kirkman. 2006. *Punctuation matters: Advice on punctuation for scientific and technical writing*. Routledge.

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 Lakougna, Jessica Lundin, Christof Monz, Kenton Murray, and 10 others. 2025. 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.

Pawan Lahoti, Namita Mittal, and Girdhari Singh. 2022. A survey on nlp resources, tools, and techniques for marathi language processing. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(2):1–34.

Noah Lukeman. 2011. *The art of punctuation*. OUP Oxford.

Mogahed M Mogahed. 2012. Punctuation marks make a difference in translation: Practical examples. *Online Submission*.

Attila Nagy, Bence Bial, and Judit Ács. 2021. Automatic punctuation restoration with bert models. *arXiv preprint arXiv:2101.07343*.

Vasile Păiș and Dan Tufiș. 2022. Capitalization and punctuation restoration: a survey. *Artificial Intelligence Review*, 55(3):1681–1722.

Partha Pakray, Reddi Krishna, Santanu Pal, Advaitha Vetagiri, Sandeep Dash, Arnab Kumar Maji, Saralin A. Lyngdoh, Lenin Laitonjam, Anupam Jamatia, Koj Sambyo, Ajit Das, and Riyanka Manna. 2025. Findings of WMT 2025 shared task on low-resource Indic languages translation. In *Proceedings of the Tenth Conference on Machine Translation*, pages 532–553, Suzhou, China. Association for Computational Linguistics.

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*, pages 311–318.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618.

Sidharth Pulipaka, Sparsh Jain, Ashwin Sankar, and Raj Dabre. 2025. Mark my words: A robust multilingual model for punctuation in text and speech transcripts. *arXiv preprint arXiv:2506.03793*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020a. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892.

Thibault Sellam, Amy Pu, Hyung Won Chung, and 1 others. 2020b. Learning to evaluate translation beyond English: BLEURT submissions to the WMT metrics 2020 shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 921–927.

Anushka Singh, Ananya Sai, Raj Dabre, Ratish Puduppully, Anoop Kunchukuttan, and Mitesh M Khapra. 2024. How good is zero-shot mt evaluation for low resource indian languages? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 640–649.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in*

neural information processing systems, 33:16857–16867.

Gemma Team. 2024. Gemma.

Ottokar Tilk and Tanel Alumäe. 2015. Lstm for punctuation restoration in speech transcripts. In *Interspeech*, pages 683–687.

Ottokar Tilk and Tanel Alumäe. 2016. Bidirectional recurrent neural network with attention mechanism for punctuation restoration. In *Interspeech*, volume 3, page 9.

Robert Lawrence Trask. 2019. *The Penguin guide to punctuation*. Penguin UK.

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.

A More details about the *Virām* Benchmark

A.1 Annotation Procedure

For the creation of the Virām benchmark translations, we hired two annotators, one pursuing a Master's degree and the other a PhD, both in the Computer Science and Engineering department. Both annotators are native speakers of Marathi. All sentences were discussed and translated collaboratively to ensure high-quality and consistent translations. The annotators received appropriate honorarium for their work.

A.2 Data Statistics

The human-validated English–Marathi test set contains a total of 54 instances with various punctuation marks. Commas are the most frequent, appearing 38 times, followed by colons and hyphens, each occurring 3 times. Parentheses and quotation marks appear twice each, while em dashes, question marks, semi-colons, and slashes are less frequent, with one or two occurrences. This distribution reflects the diversity of punctuation in the dataset, which may affect the complexity of translation and evaluation.

B Dataset construction for training models

B.1 Data Handling for Approach 1

For training the punctuation restoration models, we used the English data from the IWSLT 2017 MT challenge[11]. We considered only the English portion of the dataset, where the source sentences were stripped of punctuation and the target sentences retained the original punctuation. This setup enables the models to learn to predict and restore punctuation in English sentences. Figure 2 illustrates the data handling process for Approach 1.

B.2 Data Handling for Approach 2

For direct fine-tuning of machine translation models, we created four variants of the dataset to evaluate the effect of punctuation on translation quality:

- **Original data**: Used as a baseline without expecting any punctuation robustness.
- **Data without punctuation**: All punctuation marks were removed from the source sentences to give models the ability to predict punctuation.
- **Data with and without punctuation (alternate)**: Punctuation is alternately removed and retained, keeping the dataset size equal to the original.
- **Data with and without punctuation (doubled)**: Each sentence is included twice, once with punctuation and once without, effectively doubling the dataset size.

Figure 3 shows the data handling process for Approach 2.

C Statistics of the Datasets Used

Table 6 summarizes the statistics of the `english_punctuation_restoration` dataset. The training split contains 206,112 instances, while the validation and test splits include 888 and 8,079 instances, respectively.

Table 7 shows the dataset statistics for the internal `eng_mar_finetuning_data`. The training set consists of 189,740 instances, and both the validation and test sets contain 23,717 instances each. These datasets provide the necessary coverage for training and evaluating models for punctuation restoration and English-to-Marathi fine-tuning tasks.

[11] https://huggingface.co/datasets/IWSLT/iwslt2017

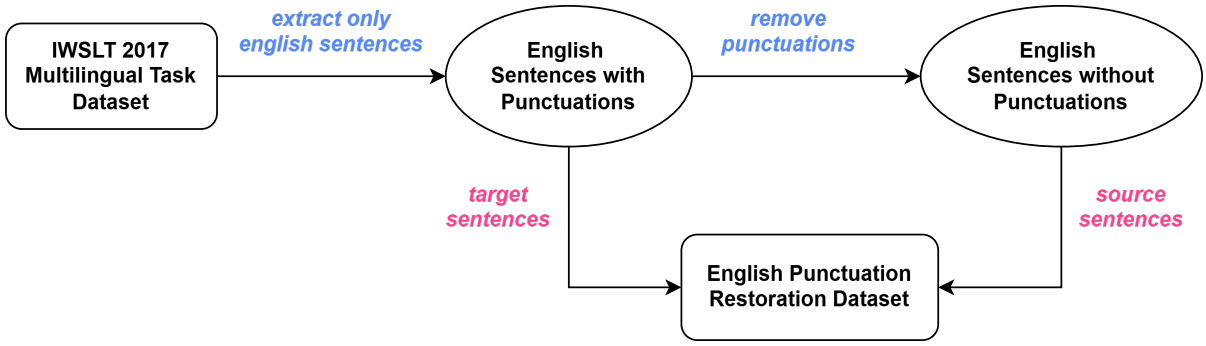


Figure 2: Data handling for the punctuation restoration task: Approach 1.

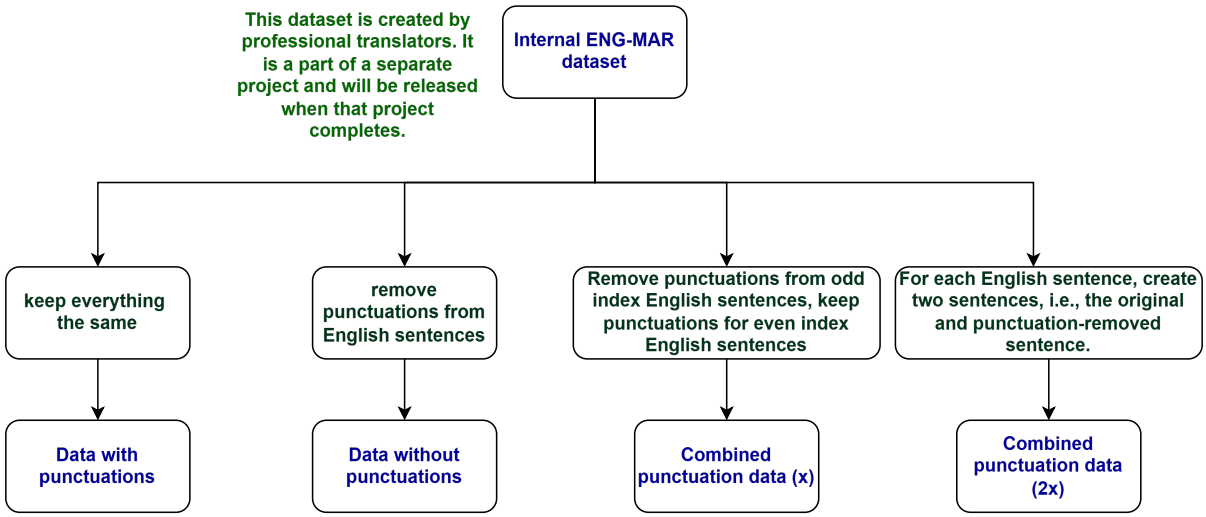


Figure 3: Data handling for the direct fine-tuning task: Approach 2.

Split	Number of Instances
Train	206,112
Validation	888
Test	8,079

Table 6: Dataset statistics for `english_punctuation_restoration`.

Split	Number of Instances
Train	189740
Validation	23717
Test	23717

Table 7: Dataset statistics for internal eng_mar_finetuning_data

D Evaluation Metrics Used

Recent advancements in Natural Language Processing (NLP), particularly in Machine Translation (MT) and cross-lingual transfer, have been driven by robust evaluation metrics and high-quality multilingual representations. This section briefly describes the evaluation metrics used in our study.

1. **BLEU (Bilingual Evaluation Understudy)** (Papineni et al., 2002) remains one of the most widely used automatic evaluation metrics for machine translation. It computes the geometric mean of modified n-gram precision between a candidate translation and one or more reference translations, combined with a brevity penalty to discourage overly short outputs.

2. **chrF++ and chrF2++** (Popović, 2017) are character n-gram–based F-score metrics that improve upon BLEU by capturing subword-level similarities, making them particularly effective for morphologically rich languages. While *chrF++* incorporates both character and word n-grams, *chrF2++* sets the β parameter to 2 (i.e., an F_2-score), placing greater emphasis on recall than precision.

3. **BLEURT-20** (Sellam et al., 2020a,b) represents a shift toward learned, neural evaluation metrics. Built on a BERT-based architecture, BLEURT is pre-trained on millions of

Table 8: Implementation details and repositories for the evaluation metrics and models.

Metric	Library / Implementation	Link / Repository
BLEU	Hugging Face `evaluate` (SacreBLEU)	https://huggingface.co/spaces/evaluate-metric/sacrebleu
chrF / chrF++	Hugging Face `evaluate` (SacreBLEU)	https://huggingface.co/spaces/evaluate-metric/chrf
COMET	Hugging Face `evaluate` (Unbabel/COMET)	https://github.com/Unbabel/COMET
BLEURT-20	Google Research BLEURT	https://github.com/google-research/bleurt
BERTScore	Hugging Face `evaluate` (MuRIL)	https://huggingface.co/spaces/evaluate-metric/bertscore
LaBSE	`sentence-transformers`	https://huggingface.co/sentence-transformers/LaBSE
MuRIL	`google/muril-base-cased`	https://huggingface.co/google/muril-base-cased

synthetic examples and fine-tuned using human judgment data. The "-20" checkpoint corresponds to the refined version released for the WMT 2020 Metrics shared task and exhibits strong correlation with human evaluation scores.

4. **COMET (Cross-lingual Optimized Metric for Evaluation of Translation)** (Rei et al., 2020) is a neural evaluation framework that leverages multilingual encoders such as XLM-RoBERTa. Unlike surface-level metrics such as BLEU, COMET jointly models the source sentence, hypothesis, and reference translation to directly predict translation quality.

5. **LaBSE (Language-agnostic BERT Sentence Embedding)** (Feng et al., 2022) is a dual-encoder model designed to produce language-agnostic sentence representations across 109 languages. It is trained using masked language modeling and translation ranking objectives, making it particularly effective for bitext mining and cross-lingual similarity tasks.

6. **MuRIL (Multilingual Representations for Indian Languages)** (Khanuja et al., 2021) is a BERT-based model tailored for the Indian linguistic landscape. Trained on 17 Indian languages and English, it incorporates both monolingual and translated/transliterated data, significantly outperforming general-purpose multilingual models (e.g., mBERT) on South Asian language tasks.

Implementation details and repositories for the evaluation metrics and models is provided in Table 8. The evaluation code used in this work follows the Indic MT Eval framework of Dixit et al. (2023).

E Prompting Details

E.1 Original Prompt: Direct Translation without Examples

The original prompt style instructs the model to directly translate an English sentence into Marathi without any example demonstrations or intermediate punctuation restoration. This approach tests the model's ability to perform translation with minimal guidance (see Figure 4). We used this prompt to directly input the correctly punctuated sentences to the model.

E.2 Zero-shot Prompt: Restore Punctuation then Translate

The zero-shot prompting strategy instructs the model to first restore punctuation in the input sentence and subsequently translate the punctuated sentence from English to Marathi. The prompt explicitly guides the model to perform punctuation restoration as an intermediate step before translation (see Figure 5).

E.3 Zero-shot Prompt: Direct Translation

The zero-shot direct translation prompt directly instructs the model to translate punctuation-ambiguous English sentences into Marathi without any intermediate punctuation restoration step (see Figure 6).

E.4 Three-shot Prompt: Restore Punctuation then Translate

The three-shot prompting strategy incorporates example demonstrations. Each prompt includes three input–output examples illustrating punctuation restoration followed by translation, after

Prompt for Original Translation

Prompt:
You are an expert linguist and translator specializing in English-to-Marathi machine translation. Translate the given English sentence into Marathi.

Input English: sentence

Make sure that the translation is in Devanagari Script.
Please provide the response in the following format:
Marathi Translation (Devanagari Script):

Figure 4: Prompt used for original translation.

Prompt for Zero-Shot Reasoning with Approach 1 (Restore then Translate)

Prompt:
You are an expert linguist and translator specializing in English-to-Marathi translation. You specialize in "Punctuation Restoration," resolving ambiguities caused by missing punctuation in English.

Steps for Analysis:
1. Analyze the English sentence for ambiguity.
2. Identify missing punctuation.
3. Generate the punctuated "English (Meant)" sentence.
4. Translate it into Marathi, ensuring the meaning is preserved.

Input English: sentence

Please provide the response in the following format:
Step 1 (Restoration): [The English (Meant) sentence]
Step 2 (Translation): [The Marathi translation (Devanagari Script)]
Reasoning: [Briefly explain your punctuation choices]

Figure 5: Prompt used for zero-shot reasoning with Approach 1 (Restore then Translate)

which the model applies the same process to a new punctuation-ambiguous sentence (see Figure 7).

E.5 Three-shot Prompt: Direct Translation

The three-shot direct translation prompting strategy provides three input–output examples illustrating direct translation of punctuation-ambiguous English sentences into Marathi, without any intermediate punctuation restoration. The model is then asked to translate a new sentence using the same approach (see Figure 8).

F Model Fine-tuning and Hyperparameter Tuning Details

For machine translation experiments, we fine-tuned all models on a server equipped with four NVIDIA A100 GPUs. We conducted a comprehensive hyperparameter search, experimenting with learning rates in [1e-3, 3e-3, 5e-3, 1e-4, 3e-4, 5e-4, 1e-5, 3e-5, 5e-5], varying the number of training epochs $[2, 5, 8, 10]$, and testing different batch sizes $[8, 16, 32]$. This systematic exploration allowed us to identify the most effective hyperparameter configurations for

Prompt for Zero-Shot Translation with Approach 2 (direct translation)

Prompt:
You are an expert linguist and translator specializing in English-to-Marathi machine translation. Translate the given English sentence into Marathi, identifying the most logical intended meaning behind missing punctuation.

Input English: sentence

Make sure that the translation is in Devanagari Script.
Please provide the response in the following format:
Marathi Translation (Devanagari Script):

Figure 6: Prompt used for zero-shot translation withwith Approach 2 (direct translation).

each model. The final models were selected based on their performance on the validation sets.

Prompt for Few-Shot inference with Approach 1 (Restore then Translate)

Prompt:
You are an expert linguist and translator specializing in English-to-Marathi translation. Use punctuation restoration to resolve ambiguity.

Definitions:
1. English (Written): Unpunctuated input.
2. English (Meant): Restored punctuation version.
3. Marathi (Translation): Translation matching "English (Meant)".

Steps:
1. Analyze ambiguity.
2. Restore punctuation.
3. Translate to Marathi.

Some examples are as follows:

1. **Input English:** These are the components required motor brushes, bearings, and wiring.

   English Meant: These are the components required: motor brushes, bearings, and wiring.

   Marathi Translation: आवश्यक असलेले घटक खालीलप्रमाणे आहेत: मोटार ब्रशेस, बेअरिंग्ज आणि वायरिंग.

2. **Input English:** As the machine develops the forms we use to record data from past projects will be amended.

   English Meant: As the machine develops, the forms we use to record data from past projects will be amended.

   Marathi Translation: जसजशी यंत्रणा विकसित होईल, तसतसे मागील प्रकल्पांतील डेटा रेकॉर्ड करण्यासाठी आम्ही वापरत असलेले फॉर्म्स सुधारित केले जातील.

3. **Input English:** What we see, we believe what we hear, we register

   English Meant: What we see, we believe; what we hear, we register.

   Marathi Translation: जे पाहतो, त्यावर विश्वास ठेवतो; जे ऐकतो, त्याची नोंद घेतो.

Input English: sentence

Please provide the response in the following format:
Step 1 (Restoration): [The English (Meant) sentence]
Step 2 (Translation): [The Marathi translation (Devanagari Script)]
Reasoning: [Briefly explain your punctuation choice]

Figure 7: Prompt used for few-shot inference with Approach 1 (Restore then Translate)

Prompt for Few-Shot Translation

Prompt:
You are an expert linguist and translator specializing in English-to-Marathi machine translation. Translate the English sentence into Marathi, resolving ambiguity caused by missing punctuation.

Some examples are as follows:

1. **Input English:** These are the components required motor brushes, bearings, and wiring.

   English Meant: These are the components required: motor brushes, bearings, and wiring.

   Marathi Translation: आवश्यक असलेले घटक खालीलप्रमाणे आहेत: मोटार ब्रशेस, बेअरिंग्ज आणि वायरिंग.

2. **Input English:** As the machine develops the forms we use to record data from past projects will be amended.

   English Meant: As the machine develops, the forms we use to record data from past projects will be amended.

   Marathi Translation: जसजशी यंत्रणा विकसित होईल, तसतसे मागील प्रकल्पांतील डेटा रेकॉर्ड करण्यासाठी आम्ही वापरत असलेले फॉर्म्स सुधारित केले जातील.

3. **Input English:** What we see, we believe what we hear, we register

   English Meant: What we see, we believe; what we hear, we register.

   Marathi Translation: जे पाहतो, त्यावर विश्वास ठेवतो; जे ऐकतो, त्याची नोंद घेतो.

Input English: sentence

Make sure that the translation is in Devanagari Script.
Please provide the response in the following format:
Marathi Translation (Devanagari Script):

Figure 8: Prompt used for few-shot translation with Approach 2 (direct translation)

Can Linguistically Related Languages Guide LLM Translation in Low-Resource Settings?

Aishwarya Ramasethu
Prediction Guard

Rohin Garg*
Scale AI

Niyathi Allu*
Independent

Harshwardhan Fartale*
Independent

Dun Li Chan
INTI International College Penang

Abstract

Large Language Models (LLMs) have achieved strong performance across many downstream tasks, yet their effectiveness in extremely low-resource machine translation remains limited. Standard adaptation techniques typically rely on large-scale parallel data or extensive fine-tuning, which are infeasible for the long tail of underrepresented languages. In this work, we investigate a more constrained question: in data-scarce settings, to what extent can linguistically similar pivot languages and few-shot demonstrations provide useful guidance for on-the-fly adaptation in LLMs? We study a data-efficient experimental setup that combines linguistically related pivot languages with few-shot in-context examples, without any parameter updates, and evaluate translation behavior under controlled conditions. Our analysis shows that while pivot-based prompting can yield improvements in certain configurations, particularly in settings where the target language is less well represented in the model's vocabulary, the gains are often modest and sensitive to few shot example construction. For closely related or better represented varieties, we observe diminishing or inconsistent gains. Our findings provide empirical guidance on how and when inference-time prompting and pivot-based examples can be used as a lightweight alternative to fine-tuning in low-resource translation settings.

1 Introduction

The advent of transformer-based architectures and general-purpose Large Language Models (LLMs) such as ChatGPT (OpenAI, 2024), DeepSeek-R1 (DeepSeek-AI, 2025), Mistral (Jiang et al., 2023), and Llama 3 (AI@Meta, 2024) has led to substantial advances in machine translation over the past decade. These models exhibit strong multilingual capabilities and, for many high-resource languages, approach expert-level translation quality. However, this performance remains highly uneven across languages.

Despite the existence of over 7,000 languages worldwide (Eberhard et al., 2024), NLP research and model development remain heavily skewed toward a small set of high-resource languages (Joshi et al., 2021; Pakray et al., 2025). Prior work has documented significant disparities in LLM translation performance between English and low-resource languages (Choudhury, 2023), and recent surveys show that even state-of-the-art models such as GPT-4 often fail to outperform specialized systems on languages using non-Latin scripts (Ataman et al., 2025).

To address these disparities, substantial effort has gone into expanding multilingual datasets and models. Foundational work on massively multilingual representation learning, such as mBERT and XLM-R (Arivazhagan et al., 2019; Conneau et al., 2020), enabled cross-lingual transfer across hundreds of languages. More recent initiatives, including No Language Left Behind (NLLB) (Team, 2024) and the FLORES benchmark (Goyal et al., 2022), aim to scale multilingual machine translation to previously underrepresented languages, while projects such as Aya (Üstün et al., 2024), BLOOM (Leong et al., 2022), and Masakhane (Nekoto et al., 2020) emphasize broader linguistic coverage and community-driven data creation. Despite these efforts, coverage remains uneven, and many languages with low digital presence are still only partially supported in widely deployed translation systems.

Rather than focusing on resource-intensive data collection or training new language-specific models, we investigate whether the few-shot instruction-following capabilities of existing LLMs can be leveraged for extremely low-resource machine translation. We study an inference-time approach that combines linguistically related few-shot examples with a pivot language—a higher-resource language closely related to the target—to provide additional

*Equal contribution

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 168–185

contextual grounding during generation.

Our experiments focus on two linguistically distinct yet underrepresented languages: Tunisian Arabic (aeb) (Mahdi, 2025) and Konkani (gom) (Rajan et al., 2020). Both languages have substantial regional and cultural significance but receive limited coverage in multilingual benchmarks and are only partially supported in large pretrained translation systems such as NLLB (Team, 2024). This makes them representative of practical low-resource scenarios where parallel data is scarce and model support varies across dialects and scripts. We evaluate our approach using In-Context Learning (ICL) with frozen, decoder-only LLMs.

We find that incorporating linguistically and semantically related few-shot examples can improve translation behavior in certain configurations, particularly when the target language appears weakly represented in the model's pretraining distribution. For Konkani, pivot-augmented prompting yields moderate gains in chrF++ relative to direct prompting, while for Tunisian Arabic the improvements are smaller and less consistent across models. These results suggest that the effectiveness of pivot-guided prompting depends strongly on language relatedness, representational coverage, and interactions between pivot and target varieties, rather than offering a universally reliable translation strategy.

2 Related Work

2.1 In-Context Learning

Prior work shows that multilingual LLM translation performance under few-shot in-context learning (ICL) depends strongly on prompt example quality (Chowdhery et al., 2022). However, it is also highlighted that substantial gains are observed in the high-resource language pairs. In addition Agrawal et al. (2022) confirm that even a single noisy or semantically unrelated demonstration can drastically reduce translation quality, whereas a well-formed equivalent-meaning example is often sufficient to elicit better translation quality from the pretrained LLMs.

Further work by Vilar et al. (2023) demonstrate that translation quality depends on domain quality rather than lexical similarity of the in-context examples and that the quality of translation degrades with poorly selected in-context examples. However their evaluation is limited to translations between English and a small set of relatively high-resource languages (French, German, and Chinese). The work of Garcia et al. (2023) also supports that the quality of few-shot in-context examples is crucial. Puduppully et al. (2023) introduce DecoMT, a few-shot prompting approach that decomposes the translation process into a sequence of word-chunk translations.

Large-scale analyses show that ICL performance in MT is driven primarily by example quality and target-side distribution rather than prompt structure or ordering (Zhu et al., 2024b; Chitale et al., 2024). Zhu et al. (2024a) investigate robustness in ICL by introducing a dual-view demonstration selection strategy. They combine margin-based sentence-level similarity to avoid semantic noise with word-embedding-based token weighting to refine the influence of demonstrations.

Taken together, these studies show that ICL can improve machine translation under favourable conditions, but they also highlight its sensitivity to demonstration quality and distributional coverage. Importantly, most of this work evaluates languages with comparatively rich digital resources, leaving open the question of how reliably ICL-based MT behaviour transfers to truly low-resource languages with sparse data and unstable tokenization.

Recent work has explored structured linguistic scaffolding as a complement to standard few-shot prompting. Lu et al. (2024) propose Chain-of-Dictionary Prompting (COD), which augments prompts with chained multilingual dictionary hints and reports large gains on FLORES-200. While effective, COD relies on proprietary models and dictionary resources that may not exist for many low-resource languages. In contrast, our approach uses open 7B-8B models and small parallel corpora, providing pivot translations as broader contextual scaffolding rather than word-level lexical hints.

Other work addresses low-resource adaptation through training-time methods. Yong et al. (2023) show that adapter-based finetuning can outperform continued pretraining when adding new languages, with gains driven primarily by data availability. Muennighoff et al. (2023) introduce multitask prompted finetuning for multilingual models and demonstrate improved zero-shot generalization when prompt language aligns with the target. Longpre et al. (2025) analyze multilingual scaling laws and argue that at very low data scales, neither pretraining nor finetuning is computationally efficient. These approaches require supervised data and training compute, whereas our work targets inference-time prompting without parameter updates.

2.2 Pivot languages aided LLM translation

Pivot strategies introduce an intermediate language to support translation in low-resource settings. Prior work has demonstrated that the choice of a pivot language can have significant impact on the translation quality.

Work by Imamura et al. (2023) shows the poor zero-shot performance of multilingual NMT models translation can be enhanced by a using pivot language. In this work, they compare the pivot and direct translation using English as the pivot language. Their study also investigates which kind of parallel corpora is most effective to enhance multilingual pivot translation.

Jiao et al. (2023) also evaluate ChatGPT for machine translation and introduce a pivot prompting strategy, in which the model first translates a source sentence into a high-resource pivot language before translating into the target language. They find that pivot prompting noticeably improves translation quality for distant or low-resource languages, and with GPT-4, ChatGPT achieves performance comparable to commercial translation systems even on some of the challenging language pairs.

Extending these ideas, Elmadani and Buys (2024) introduces synthetic pivoting, where pivot sentences are generated from both the target and the source languages using the sequence level knowledge distillation. This approach reduces pivot translation complexity and improves BLEU scores for low-resource Southern African languages by up to 5.6 points.

Recent work by Talwar and Laasri (2025) highlight this in their study on Nepali-English translation, where Hindi is chosen as a pivot language due to its linguistic proximity to Nepali and the greater availability of Hindi parallel corpora. By employing both fully supervised transfer learning and semi-supervised back-translation, they show that using Hindi as a pivot language improves the Nepali-English translation baselines, emphasizing how a chosen pivot language can compensate for limited data availability.

Lim et al. (2025) reformulated low-resource translation as a post-editing task, where a teacher model generates auxiliary translations and a student model is finetuned to correct them, achieving strong gains on FLORES-200/NTREX. Their results suggest that even imperfect auxiliary translations can provide useful scaffolding. While Mufu relies on supervised finetuning, our work adapts this post-editing insight to pure ICL by using a single pivot translation combined with retrieved few-shot examples, without parameter updates or multi-model pipelines.

Collectively, these findings motivate our investigation into pivot language strategies for LLM translation. Our work builds on these insights by examining whether integrating pivot language examples and leveraging few-shot ICL can further enhance translation performance for languages like Konkani and Tunisian Arabic. In doing so, we aim to clarify the mechanisms by which pivot languages facilitate knowledge transfer in LLMs, while also extending the adaptation capabilities of models to new languages. This helps clarify when such approaches may, or may not, be effective for low-resource languages.

3 Methodology

We explore an inference-time technique of translation in settings where data, compute, and model scale are limited. Our goal is to examine what kinds of evidence (such as linguistically related pivot languages and semantically retrieved few-shot examples) can be leveraged to support translation in an ICL setting using small (≈8B) decoder-only models, without fine-tuning or large parallel corpora. In particular, we investigate whether these signals provide useful guidance when translating into previously unseen or under-represented languages, and under what conditions they help, fail, or produce inconsistent behavior.

To support semantic retrieval, we construct a datastore of parallel translations organized as triplets consisting of an English source sentence, its pivot-language translation, and the corresponding target-language translation. These triplets are derived exclusively from the training split. We index the datastore using the English source sentences, as English is the input language at inference time. Sentence embeddings are computed using the **all-MiniLM-L12-v2** sentence transformer, which maps text into a dense vector space suitable for semantic similarity search. This representation allows semantically related translation examples to be clustered and retrieved efficiently.

At inference time, we generate an embedding for each input source sentence and query the vector datastore using cosine similarity. The top-k most semantically similar triplets are retrieved and used as in-context demonstrations. These demonstrations

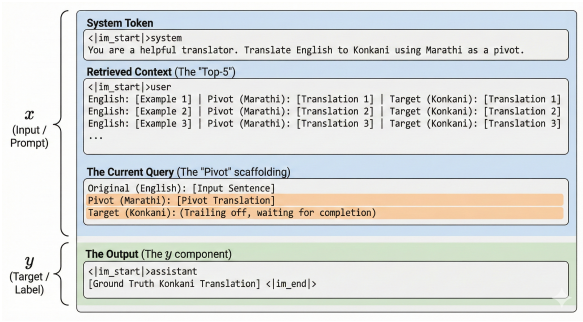


Figure 1: The Training Data Structure. The input x contains the system instruction, retrieved semantic context (top-5), and the pivot scaffolding. The target y contains only the model's generated translation.

are formatted as English-pivot-target examples and combined with the pivot translation corresponding to the same input sentence from the parallel corpus; no pivot translations are generated by the model or obtained from external sources. The resulting prompt is structured in ChatML format (see Appendix A.11). To mitigate contamination and retrieval leakage, all retrieved examples are drawn strictly from the training datastore, while evaluation is performed on a held-out test set that is never indexed or queried during retrieval. No test sentences or paraphrases are included in the datastore. We treat the number of in-context examples k as a controllable parameter and evaluate its effect through a systematic ablation study (see Appendix A.8). This retrieval-augmented prompting workflow is illustrated in Figure **??**.

To isolate the contribution of the pivot language itself, we conduct controlled comparisons between pivot-augmented prompts and prompts constructed using identical retrieval procedures but excluding the pivot translation. This allows us to disentangle gains arising from semantic retrieval alone from those attributable to the linguistic bridge provided by the pivot language. The ablation study results without the pivot language are shown in Table 10 and Table 11.

4 Experiments

4.1 Languages and data

For this experiment, we focused on two low-resource languages. The first is Konkani, an Indian language spoken in the western part of India, with approximately 2.35 million speakers as of 2011. The second is Tunisian Arabic, spoken in Tunisia, with around 12 million speakers as of 2021. A motivation for selecting these languages was their use of non-Latin scripts. In addition, Tunisian Arabic is unique in the sense that, unlike regular Latin script which reads from left to right, this is from right to left.

To provide a quantitative sanity check on pivot selection, we also compute word-level Jaccard similarity between the pivot and target languages in our datasets. This allows us to verify that the chosen pivots are lexically closer to the target languages than English. A detailed description and full results are provided in Appendix A.6.

4.2 Models

In this experiment, we evaluate the performance of Unbabel's TowerInstruct-7B-v0.1 (Alves et al., 2024) and NousResearch's Hermes-2-Pro-Llama-3-8B (Teknium et al., 2024). The Hermes-2-Pro-Llama-3 model is a instruction-tuned version of Llama 3 (AI@Meta, 2024) known for its multilingual capabilities. Llama 3 supports 8 languages: English, German (deu), French (fra), Italian (ita), Portuguese (por), Hindi (hin), Spanish (spa), and Thai (tha), although the underlying foundation model has been trained on over 176 languages. Our aim is to see whether we can make use of the latent knowledge alignment of the model while translating to low resource languages.

The TowerInstruct-7B-v0.1 model is finetuned from TowerBase-7B-v0.1 model. The TowerBase-7B-v0.1 model is continuously pretrained from from the Llama 2 model with a mixture of monolingual and parallel data with 20B tokens. The TowerBase-7B-v0.1 model is the finetuned with instruction dataset that is relevant to translation process. Some of these instruction tasks include Automatic Post Edition, Context-aware Translation, Named-entity Recognition etc. The languages supported by the TowerBase-7B-v0.1 model are English (eng), German (deu), French (fra), Dutch (nld), Italian (ita), Spanish (spa), Portuguese (por), Korean (kor), Russian (rus), and Chinese (zho). Although TowerInstruct-7B-v0.1 performs well on translation tasks, it is not expected to excel in languages it was not exposed to during training.

Our strategic selection of these models is designed to assess their effectiveness in translating languages outside their initial training sets. Despite TowerInstruct-7B-v0.1's specialized training in translation, it has not been directly exposed to the specific low-resource languages focused on in this experiment, offering a unique test of its adaptability to unseen languages.

4.3 Datasets

We utilized two distinct multiparallel datasets, effectively organizing the data into aligned triplets (Source-Pivot-Target) to support our retrieval-augmented pipeline.

Konkani: We constructed a dataset of English-Marathi-Konkani triplets using the open-source corpus from AI4Bharat (Gala et al., 2023). Marathi was selected as the pivot language due to its linguistic similarity to Konkani and wider prevalence in western India. We created a distinct split of [800] examples for the training (retrieval) datastore and 200 examples for the held-out test set.

Tunisian Arabic: We derived a similar corpus of English-MSA-Tunisian triplets from the work described by Bouamor et al. (2014), with Modern Standard Arabic (MSA) chosen as the pivot language. This dataset consists of 900 examples for the training datastore and 100 examples for the held-out test set.

In total, our study operates on small training sets of approximately 1,000 records per language. This constraint was chosen specifically to simulate realistic low-resource scenarios where large-scale parallel data is unavailable.

5 Results

5.1 Does Pivot-Based Prompting Improve Translation?

To establish a reference point, we first evaluate a direct prompting baseline, where the model is given only the English source sentence and instructed to translate directly into the target language, without access to a pivot language. In this setting, chrF++ scores are often extremely low (in some cases close to 1) because the models do not reliably generate text in the intended target language or script. Instead, the output frequently drifts toward better-represented neighboring languages (e.g., producing Marathi- or Hindi-like text when the target is Konkani, or MSA-like text for Tunisian Arabic). This behavior is observed across both Hermes and Tower, indicating that, without grounding signals, the model does not consistently infer the correct output language from the instruction alone.

We then compare this to our pivot-augmented prompting condition, in which the same input is supplied along with a translation into a linguistically related pivot language. In this setting, the few-shot demonstrations and pivot translation act as grounding signals that stabilize generation toward the intended script and language family. Tables 1 and 2 report BLEU and chrF++ scores across three conditions (zero-shot (k=0), direct few-shot prompting without a pivot, and pivot-augmented prompting). For each configuration, we report the best-performing number of in-context examples (k), as determined in ablations in Appendix 12 and 13.

For Konkani, introducing few-shot demonstrations, even without a pivot, leads to a substantial improvement in both chrF++ and BLEU, indicating that the examples themselves provide a strong anchoring effect for this previously unseen language. Adding the pivot language on top of these examples results in only small or mixed additional gains: for Hermes, the pivot condition yields a modest improvement over direct few-shot prompting (29.62→30.34 chrF++, 7.35→7.77 BLEU), whereas for Tower the pivot improves BLEU from 3.67 to 5.68, but does not improve chrF++. This suggests that, in this setting, most of the benefit arises from example-driven stabilization rather than from the pivot language itself.

For Tunisian Arabic, zero-shot scores are already relatively high, and both chrF++ and BLEU change marginally across the direct and pivot conditions, with no consistent advantage for either model. Here, few-shot prompting provides limited additional benefit, and the pivot language does not substantially alter model behavior, consistent with the interpretation that Tunisian Arabic which is already better represented in the underlying pretrained models.

We additionally evaluate whether using a pivot language that is explicitly supported by the model leads to improved translation quality. Given the constraints of our setup, the only configuration that satisfies this condition is Hindi as a pivot for Konkani using the Hermes-2-Pro-Llama-3-8B model. We analyze this setting in detail in Appendix A.9, including token-to-word ratios and Jaccard similarity between Hindi and Konkani.

Across these experiments, we find that using a model-supported pivot language does not yield systematic improvements over linguistically motivated pivots such as Marathi. In several cases, performance degrades as the number of in-context examples increases, suggesting that native model support alone is insufficient to improve or stabilize low-resource translation.

To ensure that pivot-augmented prompting does not simply cause the model to reproduce pivot-language translations, we measure chrF overlap between the pivot outputs and the final gener-

ated translations. This analysis, reported in Appendix A.5, shows consistently low chrF scores for both Konkani and Tunisian Arabic, indicating limited surface-level overlap between pivot and generated outputs. These results suggest that the model does not merely copy or lightly edit the pivot translation, but instead produces outputs that are substantially distinct from the pivot language.

One possible explanation, which we treat as hypothesis-generating rather than conclusive, comes from the token-to-word analysis in Table 6. For Tunisian Arabic, both models exhibit substantially lower token-to-word ratios (e.g., 4.96 vs. 7.65 for Tower; 2.16 vs. 4.09 for Hermes, comparing Aeb vs. Gom), indicating that the models segment Tunisian Arabic into fewer subword units than Konkani. Because Modern Standard Arabic (MSA) is well represented in most pretrained corpora, Tunisian Arabic, which shares script and lexical characteristics with MSA, may benefit indirectly from this representation. This would help explain why few-shot prompting and pivot augmentation yield smaller or inconsistent gains in this setting.

In contrast, the much higher token-to-word ratios for Konkani suggest a weaker lexical footprint in the pretrained vocabulary. Here, the few-shot examples and the pivot appear to act less as a source of additional translation competence and more as basic scaffolding for language identification, script adherence, and output stability.

However, we emphasize that this relationship is correlational rather than causal: tokenization efficiency alone does not fully explain performance differences, and other factors may contribute to the observed behavior.

5.2 Comparison to NLLB Reference Baselines

As a point of external reference, we compare the best-performing few-shot and pivot-augmented scores in Tables 1 and 2 with the NLLB-200 distilled baselines in Table 5. For Konkani, NLLB does not provide native support, and the baseline remains relatively low (26.82 chrF++, 7.51 BLEU). Our best Hermes pivot-augmented configuration attains 30.34 chrF++ and 7.77 BLEU, while the Tower pivot setting reaches 17.66 chrF++ and 5.68 BLEU. Thus, Hermes slightly exceeds the NLLB baseline on both metrics, whereas Tower remains below it.

For Tunisian Arabic, NLLB does include explicit support, but the baseline scores remain modest (10.42 chrF++, 4.20 BLEU). In contrast, both decoder-only LLMs achieve substantially higher performance even without fine-tuning: Hermes reaches 24.32 chrF++ and 6.27 BLEU (direct few-shot), and Tower reaches 20.63 chrF++ and 4.99 BLEU (pivot). This indicates that, even relative to a supervised MT system trained with explicit support for the language, few-shot prompting can yield stronger performance in this setting.

This contrast highlights a practical trade-off: improving NLLB performance for unsupported or weakly supported languages would typically require collecting supervised training data and fine-tuning the model, whereas our approach obtains measurable gains using only few-shot prompting with no parameter updates.

5.3 Effect of Increasing the Number of In-Context Examples

We next examine whether translation quality improves simply by increasing the number of retrieved demonstrations (k), independent of the pivot signal. For each model-language pair, we evaluate chrF++ and BLEU across multiple values of k in both the direct and pivot-based translation settings (full results in Appendix 12-13 and 10-11).

For Konkani, increasing k does not produce monotonic gains in either metric. In the direct translation setting, Hermes reaches its strongest chrF++ at k=3 (29.62) with relatively low BLEU (2.33), while performance drops at both smaller and larger k values. Tower shows a similar pattern: chrF++ peaks at k=2 (21.25), while BLEU remains in the 3-4 range and collapses to 0 beyond k=3. In the pivot-based setting, Hermes attains its best chrF++ at k=1 (30.34) and best BLEU at k=4 (7.77), whereas Tower peaks at k=2 in BLEU (5.68) but achieves higher chrF++ at k=3 (17.66). Thus, for Konkani, both metrics improve relative to k=0, but additional demonstrations beyond the best-performing k generally degrade performance.

A similar trend appears in Tunisian Arabic, although the baseline (k=0) performance is much stronger. In the direct setting, Hermes achieves its best BLEU at k=1 (6.27), while chrF++ remains highest at k=0 (24.32) and declines as k increases. Tower exhibits modest variation across k, with chrF++ peaking at k=4 (20.74) despite little corresponding change in BLEU. In the pivot condition, Hermes again shows small fluctuations around its k=0 baseline (24.31 chrF++), while Tower reaches its highest BLEU at k=2 (4.99) and highest chrF++ at k=5 (20.63), before declining at larger k.

Model	Setting	BLEU	chrF++
Baseline	NLLB-200	7.51	26.82
Hermes-2-Pro	Zero-shot (k=0)	1.49	1.30
	Direct (Best k)	7.35	29.62
	With Pivot (Best k)	**7.77**	**30.34**
TowerInstruct	Zero-shot (k=0)	1.28	0.69
	Direct (Best k)	3.67	**21.25**
	With Pivot (Best k)	**5.68**	17.66

Table 1: Select Konkani translation results (Eng→Gom)

Model	Setting	BLEU	chrF++
Baseline	NLLB-200	4.20	10.42
Hermes-2-Pro	Zero-shot (k=0)	4.62	24.32
	Direct (Best k)	**6.27**	**24.32**
	With Pivot (Best k)	5.06	24.31
TowerInstruct	Zero-shot (k=0)	4.19	17.62
	Direct (Best k)	4.46	**20.74**
	With Pivot (Best k)	**4.99**	20.63

Table 2: Select Tunisian Arabic results (Eng→Aeb)

Across both languages and models, these results indicate that: Performance improves substantially when moving from k=0 to small k, but gains do not scale with additional demonstrations. ChrF++ and BLEU often peak at different values of k.

We hypothesize that one contributing factor is the interaction between k and model context capacity. TowerInstruct operates effectively within a 4K-token window, and performance often declines once prompts approach this length, suggesting truncation or overwriting effects. Hermes supports a larger context window, yet its performance likewise plateaus or degrades beyond moderate k, implying that the limitation is not purely architectural but also behavioral: models may underutilize long-range prompt structure or overweight spurious correlations from loosely related examples.

Taken together, these findings suggest that the gains observed in our experiments are not simply an artifact of "more examples." Instead, a small number of semantically aligned demonstrations appears to provide most of the benefit, while additional examples can introduce noise that reduces both BLEU and chrF++. In settings where pivot-based prompting yields improvements, these effects should therefore be interpreted as complementary to, rather than interchangeable with, the contribution of few-shot demonstrations themselves.

6 Limitations

Many machine learning breakthroughs are enabled by an abundance of computational resources. However, access to large-scale compute is not uniformly available, including to most authors of this work. This disparity becomes even more apparent when working with communities that speak low-resource languages. Within these constraints, we aimed to rigorously test our hypotheses about pivot-based translation using the resources available to us. Importantly, these constraints also reflect realistic deployment conditions for many low-resource language communities, where access to large-scale compute, extensive annotation, and proprietary models is limited.

The primary limitation of this work is that, while we build on prior research on pivot languages to investigate whether linguistically related languages provide any useful signal for inference-time translation under resource constraints, the performance gains we observe are modest and often inconsistent. Working within our computational budget, we evaluated open-weight models in the 7B parameter range. While larger models may yield stronger performance, our results indicate that pivot-augmented prompting can sometimes improve performance, but its effects are highly sensitive to language characteristics and example selection, suggesting that further study is needed before drawing strong conclusions.

Additionally, much of the existing research on multilinguality and machine translation relies on human evaluation, which was not feasible in our setting. Under these constraints, and with respect for the communities that speak these languages, we evaluate how language models adapt to previously unseen languages in low-resource conditions using automatic metrics. We report BLEU and chrF++ scores computed with SacreBLEU (Post, 2018) for reproducibility (see Appendix A.2 for scoring signatures).

However, these metrics have known limitations in low-resource and morphologically rich settings. As illustrated in Appendix A.7.1, we observe cases where the generated Konkani translation is linguistically plausible and semantically related to the reference, yet differs substantially in surface form, resulting in very low BLEU and chrF++ scores. This highlights the brittleness of n-gram–based metrics for evaluating low-resource translation quality and motivates the need for human evaluation by native speakers to better capture semantic adequacy, pragmatic meaning, and dialectal correctness.

Another limitation is that our methodology depends on the availability of a high-resource pivot language that is linguistically similar to the target language, which restricts its applicability to lan-

guages without closely related pivots. While the approach is data-efficient, it also assumes access to high-quality parallel corpora; translation quality may degrade when there is a domain mismatch between the retrieved examples and the input text.

Given the promising results observed under these constrained settings, natural extensions of this work include scaling experiments to larger open-source models, conducting human-in-the-loop evaluations with native speakers, and exploring additional language pairs to better characterize the conditions under which pivot-augmented prompting helps, fails, or produces negligible effects.

References

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2022. In-context examples selection for machine translation. *Preprint*, arXiv:2212.02437.

AI@Meta. 2024. Llama 3 model card.

Duarte M Alves, José Pombal, Nuno M Guerreiro, Pedro H Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks. In *Proceedings of the Conference on Language Modeling (COLM) 2024*.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *Preprint*, arXiv:1907.05019.

Duygu Ataman, Alexandra Birch, Nizar Habash, Marcello Federico, Philipp Koehn, and Kyunghyun Cho. 2025. Machine translation in the era of large language models:a survey of historical and emerging problems. *Information*, 16(9).

Houda Bouamor, Nizar Habash, and Kemal Oflazer. 2014. A multidialectal parallel corpus of arabic. In *LREC*, pages 1240–1245.

Pranjal Chitale, Jay Gala, and Raj Dabre. 2024. An empirical study of in-context learning in LLMs for machine translation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7384–7406, Bangkok, Thailand. Association for Computational Linguistics.

Monojit Choudhury. 2023. Generative AI has a language problem. *Nature Human Behaviour*, 7(11):1802–1803.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, and 48 others. 2022. Palm: Scaling language modeling with pathways. *Preprint*, arXiv:2204.02311.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. *Preprint*, arXiv:1911.02116.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig, editors. 2024. *Ethnologue: Languages of the World*, twenty-seventh edition. SIL International, Dallas, Texas.

Khalid N. Elmadani and Jan Buys. 2024. Neural machine translation between low-resource languages with synthetic pivoting. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12144–12158, Torino, Italia. ELRA and ICCL.

Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*, 2023.

Xavier Garcia, Yamini Bansal, Colin Cherry, George Foster, Maxim Krikun, Melvin Johnson, and Orhan Firat. 2023. The unreasonable effectiveness of few-shot learning for machine translation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 10867–10878. PMLR.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Kenji Imamura, Masao Utiyama, and Eiichiro Sumita. 2023. Pivot translation for zero-resource language pairs based on a multilingual pretrained model. In *Proceedings of Machine Translation Summit XIX, Vol. 1: Research Track*, pages 348–359, Macau SAR,

China. Asia-Pacific Association for Machine Translation.

Albert Q Jiang, Alexandre Sablayrolles, and 1 others. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Is chatgpt a good translator? yes with gpt-4 as the engine. *Preprint*, arXiv:2301.08745.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2021. The state and fate of linguistic diversity and inclusion in the nlp world. *Preprint*, arXiv:2004.09095.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395. Association for Computational Linguistics.

Colin Leong, Joshua Nemecek, Jacob Mansdorfer, Anna Filighera, Abraham Owodunni, and Daniel Whitenack. 2022. Bloom library: Multimodal datasets in 300+ languages for a variety of downstream tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8608–8621, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zheng Wei Lim, Nitish Gupta, Honglin Yu, and Trevor Cohn. 2025. Multilingual fused learning for low-resource translation with llms. In *International Conference on Learning Representations*.

Shayne Longpre, Sneha Kudugunta, Niklas Muennighoff, I-Hung Hsu, Isaac Caswell, Alex Pentland, Sercan Ö. Arik, Chen-Yu Lee, and Sayna Ebrahimi. 2025. Atlas: Adaptive transfer scaling laws for multilingual pretraining and finetuning. *Preprint*, arXiv:2510.22037.

Hongyuan Lu, Haoran Yang, Haoyang Huang, Dongdong Zhang, Wai Lam, and Furu Wei. 2024. Chain-of-dictionary prompting elicits translation in large language models. *Preprint*, arXiv:2305.06575.

Mohamed Mahdi. 2025. How well do llms understand tunisian arabic? *Preprint*, arXiv:2511.16683.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. *Preprint*, arXiv:2211.01786.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Hassan Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa Berhe, Mofetoluwa Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Martinus, and 28 others. 2020. Participatory research for low-resourced machine translation: A case study in african languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160. Association for Computational Linguistics.

OpenAI. 2024. Gpt-4.1 technical report.

Partha Pakray, Alexander Gelbukh, and Sivaji Bandyopadhyay. 2025. Natural language processing applications for low-resource languages. *Natural Language Processing*, 31(2):183–197.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Ratish Puduppully, Anoop Kunchukuttan, Raj Dabre, Ai Ti Aw, and Nancy F. Chen. 2023. Decomposed prompting for machine translation between related languages using large language models. *Preprint*, arXiv:2305.13085.

Annie Rajan, Ambuja Salgaonkar, and Ramprasad Joshi. 2020. A survey of konkani nlp resources. *Computer Science Review*, 38:100299.

Abhimanyu Talwar and Julien Laasri. 2025. Pivot language for low-resource machine translation. *Preprint*, arXiv:2505.14553.

NLLB Team. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630:841–846.

Ryan Teknium, Jeffrey Quesnelle, and Chen Guang. 2024. Hermes 3 technical report. *arXiv preprint arXiv:2408.11857*.

Ahmet Üstün, Viraat Aryabumi, Zheng Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, and 1 others. 2024. Aya model: An instruction finetuned open-access multilingual language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15894–15939.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. Prompting PaLM for translation: Assessing strategies and performance. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15406–15427, Toronto, Canada. Association for Computational Linguistics.

Zheng Xin Yong, Hailey Schoelkopf, Niklas Muennighoff, Alham Fikri Aji, David Ifeoluwa Adelani, Khalid Almubarak, M Saiful Bari, Lintang Sutawika, Jungo Kasai, Ahmed Baruwa, Genta Winata, Stella

			BLEU		chrF++	
Lang.	**Mod.**	k	Δ	p	Δ	p
Gom	Her	0	+0.12	.08	+0.22	.20
	Her	1	−0.18	.88	−0.89	1.0
	Her	2	−0.12	.75	−0.33	.89
	Tow	1	−0.02	.57	−0.84	1.0
	Tow	2	+0.07	.23	−1.11	1.0
Aeb	Her	0	+0.23	.07	−0.05	.56
	Her	1	+0.38	.10	−0.37	.71
	Her	2	−0.26	.78	−1.23	.98
	Tow	0	−0.02	.55	+0.02	.48
	Tow	1	−0.20	.72	+0.34	.23
	Tow	2	+0.13	.31	+0.25	.22

Table 3: Paired bootstrap significance (pivot − direct). No comparison reaches $p < 0.05$. Gom: Konkani (n=205), Aeb: Tunisian Arabic (n=100). Her: Hermes, Tow: Tower.

Biderman, Edward Raff, Dragomir Radev, and Vassilina Nikoulina. 2023. BLOOM+1: Adding language support to BLOOM for zero-shot prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11682–11703, Toronto, Canada. Association for Computational Linguistics.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024. When scaling meets llm finetuning: The effect of data, model and finetuning method. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR 2024)*.

Shaolin Zhu, Menglong Cui, and Deyi Xiong. 2024a. Towards robust in-context learning for machine translation with large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16619–16629, Torino, Italia. ELRA and ICCL.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024b. Multilingual machine translation with large language models: Empirical results and analysis. *Preprint*, arXiv:2304.04675.

A Appendix

A.1 Statistical Significance

We conduct paired bootstrap resampling (Koehn, 2004) (n=10,000, $p < 0.05$) to test whether pivot prompting significantly outperforms direct translation. As shown in Table 3, no comparison reaches significance, indicating that observed trends are suggestive rather than conclusive.

A.2 SacreBLEU Signatures and Reproducibility

All BLEU and chrF++ scores were computed using SacreBLEU (Post, 2018). Following their recommendations, we report scoring signatures below for full reproducibility. Statistical significance was assessed via paired bootstrap resampling (Koehn, 2004) (n=10,000, $p < 0.05$); see Section A.1.

Metric	Signature
BLEU	`nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.5.1`
chrF++	`nrefs:1|case:mixed|eff:yes|nc:6|nw:2|space:no|version:2.5.1`

Table 4: SacreBLEU scoring signatures.

A.3 NLLB baselines

Table 5 reports reference translation scores from the NLLB-200 distilled model. NLLB is an encoder-decoder neural machine translation system trained for supervised MT, whereas the models in our study (Hermes and Tower) are decoder-only LLMs used in a few-shot, in-context prompting setting with no task-specific parameter updates. Accordingly, these numbers are provided only as contextual reference points rather than as directly comparable baselines. We also note that NLLB does not natively support Konkani; the scores reported for this variety in Table 5 reflect zero-shot transfer behavior rather than a tuned dialect model.

Language Pair	BLEU	chrF++
Eng-Gom	7.51	26.82
Eng-Aeb	4.20	10.42

Table 5: NLLB-200 distilled reference baseline results for our evaluation datasets.

A.4 Token analysis

As shown in Table 6, Hermes consistently exhibits lower token fertility than Tower across all non-English languages, particularly for low-resource and dialectal varieties, indicating more efficient subword representations.

Dataset	Language	Tower	Hermes
Gom	Eng	1.59	1.34
Gom	Mar	7.73	4.08
Gom	Gom	7.65	4.09
Aeb	Eng	1.27	1.21
Aeb	MSA	4.74	2.12
Aeb	Aeb	4.96	2.16

Table 6: Tokens per word across languages for Tower and Hermes models. Lower values indicate more efficient tokenization.

Language	Model	k=1	k=2	k=3	k=4	k=5
Arabic	Hermes	24.09	24.89	25.17	23.79	24.06
	Tower	13.08	12.70	11.85	12.06	11.68
Konkani	Hermes	28.96	27.45	27.82	25.95	26.80
	Tower	10.78	8.71	9.69	9.16	8.24

Table 7: chrF scores between pivot translations and generated translations for different values of k. Lower scores indicate greater divergence from the pivot output, suggesting that the model is not simply reproducing the pivot translations.

A.5 Deviation from Pivot Translations

To assess whether the model simply reproduces pivot-language translations or instead generates genuinely distinct target-language outputs, we compute chrF scores between pivot translations and the final generated outputs for different values of k. chrF is well suited for this analysis, as it measures character-level overlap and is sensitive to direct copying, while remaining robust to morphological variation.

Table 7 shows consistently low chrF scores across models, languages, and values of k, indicating limited surface-level overlap between pivot translations and generated output. This suggests that the models are not merely copying or lightly editing the pivot translations but are instead producing substantially different outputs.

Notably, the Tower model exhibits particularly low chrF scores compared to Hermes for both Arabic and Konkani, with scores for Konkani remaining below 11 across all values of k. This behavior indicates an even stronger departure from the pivot translations, reinforcing the conclusion that the generated outputs are not simple transcriptions or reformulations of the pivot language.

The stability of chrF scores across different values of k further suggests that this divergence is systematic rather than an artifact of sampling variability. Overall, these results provide evidence that the generation step does not collapse to reproducing pivot-language translations, but instead yields outputs that are meaningfully distinct from the pivot representations.

A.6 Jaccard similarity

Pairwise lexical similarity between the languages in our corpus is reported in Tables 8 and 9. Marathi and Konkani exhibit substantially higher lexical overlap than English with either language, while Tunisian Arabic shows moderate overlap with Modern Standard Arabic (MSA). These values are not used as a selection criterion, but rather serve as supporting evidence that our chosen pivots are linguistically closer to the target languages than English.

Language Pair	Jaccard Similarity
Eng-Mar	0.0002
Eng-Gom	0.0121
Mar-Gom	0.1054

Table 8: Word-level Jaccard similarity scores for the Konkani corpus. Marathi and Konkani show substantially higher lexical overlap than English with either language.

Language Pair	Jaccard Similarity
Eng-MSA	0.0010
Eng-Aeb	0.0010
MSA-Aeb	0.1646

Table 9: Word-level Jaccard similarity scores for the Arabic corpus. MSA and Tunisian Arabic show moderate lexical overlap, while both exhibit minimal overlap with English.

For each pair of languages, we compute word-level Jaccard similarity by treating the vocabulary extracted from each corpus as a set. The similarity is defined as the size of the intersection divided by the size of the union, yielding a value between 0 and 1, where higher scores indicate greater lexical overlap.

In our datasets, Marathi (mar) and Konkani (Gom) show moderate lexical similarity (10.6%), while Tunisian Arabic (Aeb) and Modern Standard Arabic (Msa) exhibit higher overlap (16.5%). Notably, the Arabic variants show greater lexical closeness than the Indo-Aryan language pair, reflecting the stronger typological affinity among Arabic dialects.

A detailed breakdown of vocabulary sizes and pairwise similarity scores is presented in Tables 8 and 9. In future work, we plan to explore whether lexical similarity correlates with translation performance.

A.7 Ablation on the Number of In-Context Examples (k) in the Direct Translation Setting

For completeness, we report ablations on the number of in-context examples (k) in the **direct English→Target** setting, i.e., without the use of a pivot language. These results complement the pivot-based experiments and allow us to isolate the

marginal effect of the pivot signal from the effect of in-context demonstrations alone.

Tables 10 and 11 show results for Konkani and Tunisian Arabic respectively, using the same retrieval and prompting setup as in the pivot configuration, but omitting the pivot translation from the prompt.

A.7.1 Zero-BLEU but Non-Zero chrF++ Cases

During our direct translation experiments, we observed instances where BLEU = 0 despite non-zero chrF++, particularly for Konkani. This occurs when the model output diverges lexically from the reference despite partial topical or semantic alignment. A representative example is shown below.

Ground Truth (Konkani):

बटाटां भितर मसालो भरसून ते बेसनाच्या पिठयेंत बुडोवन तेलांत बरे तळ्ळे कांय महाराष्ट्रांतलो हो सुवादीक आनी फामाद पदार्थ तयार जाता.

Model Translation:

आलूच्या मसालेत संकरात बेसन बत्तर साठी अच्छा उत्सा–हसाठी महाराष्ट्रातील इतर प्रचारीक औषधे

A.8 Ablation on the Number of In-Context Examples (k) in the Pivot-Based Setting

For completeness, we report ablations on the number of in-context examples (k) in the **pivot-based** setting, i.e., where the model is provided with a linguistically related pivot translation alongside the retrieved few-shot examples allow us to examine the marginal contribution of the pivot signal. The performance of the models with pivot for konkani is shown in Table 12 and Tunisian arabic Table 13

A.9 Ablations with LLM-Supported Pivot Languages

Our experimental design selects pivot languages based on two primary criteria: (1) linguistic similarity to the target low-resource language, and (2) higher expected digital presence relative to the target. While we attempt to quantify pivot relevance using Jaccard similarity, this metric only imperfectly captures linguistic suitability, leaving gaps in systematic pivot selection. As an additional analysis, we consider pivot languages that are explicitly supported by the model, in order to examine whether native model support leads to improved translation performance.

A key limitation of this approach is that most general-purpose LLMs support only a narrow subset of languages, which substantially restricts coverage for low-resource targets. This constraint is evident even in our experimental setup: neither model explicitly supports Tunisian Arabic or closely related Arabic varieties, and for Konkani, only Hindi is supported, and only by the Hermes-2-Pro-Llama-3-8B model. Consequently, we evaluate this supported-pivot configuration only for Konkani and only under the Hermes-2-Pro-Llama-3-8B setting.

The Jaccard similarity between Hindi and Konkani (0.090) is slightly lower than that between Marathi and Konkani (0.105). However, because Hindi is explicitly supported by the model, this difference is reflected in tokenization behavior: the token-to-word ratio for Hindi under Hermes is substantially lower (2.85) than for Marathi and Konkani (both approximately 7, refer to Table 6), consistent with stronger lexical coverage in the pretrained vocabulary.

We report BLEU and chrF++ scores for this configuration in Table 14. When comparing chrF++ scores against the corresponding Marathi-pivot setting, we do not observe systematic improvements from using a model-supported pivot language. In several cases, performance degrades substantially as the number of in-context examples increases, suggesting that native model support alone is insufficient to guarantee stable or improved pivot-based translation in this low-resource setting.

A.10 Fine-Tuning Impact

Our fine-tuning experiments were limited in scope and not comprehensive. Fine-tuning was performed on the same small training sets (900 samples) used for few-shot example retrieval, without extensive hyperparameter tuning or architectural variations. While results show promise for Konkani, comprehensive fine-tuning ablations including varied training set sizes, learning rates, and LoRA configurations remain as future work.

Konkani: In the finetuning experiments, we treat the zero-shot finetuned model (English→Konkani without pivot and without in-context demonstrations) as the reference baseline. For Hermes, the zero-shot finetuned condition achieves a chrF++ of 36.61, which increases to 40.17 when a Marathi pivot is introduced. For TowerInstruct, chrF++ increases from 17.39 (zero-shot finetuned without pivot) to 31.91 with pivot. For completeness, we

Model	Source	Target	k	BLEU	chrF++
Ablation: Number of In-Context Examples (k)					
Unbabel/TowerInstruct-v0.1					
Unbabel/TowerInstruct-v0.1	Eng	Gom	0	1.28	0.69
Unbabel/TowerInstruct-v0.1	Eng	Gom	1	3.67	21.01
Unbabel/TowerInstruct-v0.1	Eng	Gom	2	3.38	21.25
Unbabel/TowerInstruct-v0.1	Eng	Gom	3	3.39	19.30
Unbabel/TowerInstruct-v0.1	Eng	Gom	4	0.0	19.78
Unbabel/TowerInstruct-v0.1	Eng	Gom	5	0.0	19.38
NousResearch/Hermes-2-Pro-Llama-3-8B					
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Gom	0	1.49	1.30
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Gom	1	2.70	23.68
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Gom	2	2.72	23.87
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Gom	3	2.33	29.62
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Gom	4	1.90	28.75
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Gom	5	7.35	25.78

Table 10: Ablation on the number of in-context examples (k) for English→Konkani direct translation.

also report few-shot finetuned results in Table 18; across both settings, we observe consistent gains when the pivot language is incorporated during prompting.

Tunisian Arabic: As in the Konkani setting, we interpret the zero-shot finetuned model without a pivot as the reference baseline. For Hermes, the zero-shot finetuned condition achieves a chrF++ of 18.07, which increases to 21.87 when an MSA pivot is included during prompting. For TowerInstruct, chrF++ improves from 14.83 (zero-shot finetuned without pivot) to 19.16 with pivot. Few-shot finetuned results are also reported in Table 19; We again observe gains when the pivot language is incorporated.

Below we describe the experiment setting in detail:

Hyperparameters: With limited data, finetuning methods like prompt tuning (where embeddings are adjusted) or LoRA (Low-Rank Adaptation) prove particularly effective (Zhang et al., 2024). With Parameter-Efficient finetuning (PEFT), even increasing the data yielded modest performance improvements. For instance, using LoRA on the Hermes-2-Pro-Llama-3-8B model brought the trainable parameters down to 176,242,688, or just 2% of the model's total parameters.

PEFT is computationally more efficient than pure ICL, which led us to adopt PEFT for our model finetuning process. We used the Huggingface Transformers library.

In addition, the model was loaded in 4-bit precision using the BitsAndBytes library with the nf4 quantization type. For fine-tuning, we employed the LoRA configuration, as detailed in the Table 16.

Parameters in Table 17 were used to generate the output from the finetuned model during the evaluation.

A.11 Prompt Template

Both the TowerInstruct-7B-v0.1 model and Hermes-2-Pro-Llama-3-8B model utilize a similar prompt format. The full prompt format is below.

```
<|im_start|>user
APE is a task designed to enhance
the quality of the translation
by performing minor adjustments
Original (English): [Original text]
Translation: [Pivot language]
Post-edited:
<|im_end|>
<|im_start|>assistant
[LLM translation]
<|im_end|>
```

The prompt includes the source sentence in English and its translation in a pivot language. For in-context learning, the prompt contains five demonstrations. In each demonstration, the assistant field is pre-filled with the target language translation. These demonstrations are carefully selected sen-

Model	Source	Target	k	BLEU	chrF++
Ablation: Number of In-Context Examples (k)					
Unbabel/TowerInstruct-v0.1					
Unbabel/TowerInstruct-v0.1	Eng	Aeb	0	4.19	17.62
Unbabel/TowerInstruct-v0.1	Eng	Aeb	1	4.46	19.49
Unbabel/TowerInstruct-v0.1	Eng	Aeb	2	4.46	16.23
Unbabel/TowerInstruct-v0.1	Eng	Aeb	3	4.07	15.59
Unbabel/TowerInstruct-v0.1	Eng	Aeb	4	4.37	20.74
Unbabel/TowerInstruct-v0.1	Eng	Aeb	5	4.37	18.61
NousResearch/Hermes-2-Pro-Llama-3-8B					
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Aeb	0	4.62	24.32
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Aeb	1	6.27	23.96
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Aeb	2	5.06	20.35
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Aeb	3	5.93	20.84
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Aeb	4	6.27	20.99
NousResearch/Hermes-2-Pro-Llama-3-8B	Eng	Aeb	5	5.52	20.60

Table 11: Ablation on the number of in-context examples (k) for English→Tn direct translation.

tences that closely resemble the sentence to be translated. In the final instance, the assistant field is left blank. This prompt structure proved to be highly effective for translation tasks of this nature. However, when using this format with the base model, the outputs often included elements like "Note," gibberish, and repetitions. After fine-tuning the model with this format, the generated translations adhered to the expected structure and consistently produced Konkani sentences.

A.12 Translation APE Examples

- **Tunisian Example Prompt:** <|begin_of_text|><|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

  Original (English): always and always

  Translation (Modern Standard Arabic): دائماً

  Post-edited (Tunisian): <|im_end|>

  <|im_start|>assistant: ابدا ابدا <|im_end|>

  <|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

  Original (English): there a lot of things that tell us shut up and brake us...

  Translation (Modern Standard Arabic): هناك الكثير من الاشياء التى تقول لنا ان نصمت و تعترض طريقنا

  Post-edited (Tunisian): <|im_end|>

  <|im_start|>assistant: فما برشا حاجات تسكتنا و توقفنا <|im_end|>

  <|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

  Original (English): And sometime no

  Translation (Modern Standard Arabic): وقليلاً لا

  Post-edited (Tunisian): <|im_end|>

  <|im_start|>assistant: لا و ساعات <|im_end|>

  <|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

  Original (English): like I said before, in good and in bad

  Translation (Modern Standard Arabic): كما قلت قبل ذلك هناك الجيد وهناك الشرير

Model	Source	Pivot	Target	k	BLEU	chrF++
Ablation: Number of In-Context Examples (k) using Marathi Pivot (No Fine-Tuning)						
Unbabel/TowerInstruct-v0.1						
Unbabel/TowerInstruct-v0.1	English	Mar	Gom	0	2.07	1.30
Unbabel/TowerInstruct-v0.1	English	Mar	Gom	1	2.58	16.03
Unbabel/TowerInstruct-v0.1	English	Mar	Gom	2	5.68	8.94
Unbabel/TowerInstruct-v0.1	English	Mar	Gom	3	4.11	17.66
Unbabel/TowerInstruct-v0.1	English	Mar	Gom	4	2.84	4.90
Unbabel/TowerInstruct-v0.1	English	Mar	Gom	5	2.84	6.11
NousResearch/Hermes-2-Pro-Llama-3-8B						
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Mar	Gom	0	2.35	24.9
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Mar	Gom	1	3.49	30.34
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Mar	Gom	2	2.36	27.59
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Mar	Gom	3	2.72	25.89
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Mar	Gom	4	7.77	27.53
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Mar	Gom	5	5.73	28.65

Table 12: Ablation on the number of in-context examples (k) for English→Marathi→Konkani translation.

Post-edited (Tunisian): <|im_end|>

<|im_start|>assistant: كيما قلت قبل في الحلو و الخايب <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): let us be really happy away from standard stuffs

Translation (Modern Standard Arabic): اتركنا نسعد حقاً بعيد عن التابوهات

Post-edited (Tunisian): <|im_end|>

<|im_start|>assistant: خلينا نفرح برسمي بعيد على كل شي <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): we shouldn't be negative all the time

Translation (Modern Standard Arabic): لا يجب ان نكون بهذه السلبيه على طول الدوام.

Post-edited (Tunisian): <|im_end|>

<|im_start|>assistant: Translation: <|im_end|>

Response from the model setting with the highest Chrf++ score:

ما لازم نكون سلبيين على طول الوقت

- **Konkani Example Prompt:**

<|begin_of_text|><|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): Great was his compassion for the two dear ones at this parting moment.

Translation (Marathi): विलग होताना त्याच्या दोन प्रिय व्यक्तींविषयी त्याला अतीव करुणा वाटत होती.

Post-edited (Konkani): <|im_end|>

<|im_start|>assistant: जिवाभावाच्या दोगांयचो त्याग करपी त्या खिणावेळार ताची करुणा सुमराभा–यली आशिल्ली. <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): Suleman's parents were quite tall.

Model	Source	Pivot	Target	k	BLEU	chrF++
Ablation: Number of In-Context Examples (k) using MSA Pivot (No Fine-Tuning)						
Unbabel/TowerInstruct-v0.1						
Unbabel/TowerInstruct-v0.1	Eng	Msa	Aeb	0	4.37	16.45
Unbabel/TowerInstruct-v0.1	Eng	Msa	Aeb	1	3.46	18.74
Unbabel/TowerInstruct-v0.1	Eng	Msa	Aeb	2	4.99	16.57
Unbabel/TowerInstruct-v0.1	Eng	Msa	Aeb	3	4.77	17.32
Unbabel/TowerInstruct-v0.1	Eng	Msa	Aeb	4	3.09	19.80
Unbabel/TowerInstruct-v0.1	Eng	Msa	Aeb	5	3.75	20.63
NousResearch/Hermes-2-Pro-Llama-3-8B						
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Aeb	0	5.06	24.31
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Aeb	1	4.93	21.27
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Aeb	2	3.74	18.18
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Aeb	3	4.20	20.17
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Aeb	4	4.93	19.42
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Aeb	5	4.77	16.32

Table 13: Ablation on the number of in-context examples (k) for English→Msa→Aeb translation

Model	Source	Pivot	Target	k	BLEU	chrF++	ΔchrF++
Ablation: Number of In-Context Examples (k) using Hindi Pivot (No Fine-Tuning)							
NousResearch/Hermes-2-Pro-Llama-3-8B							
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Hin	Gom	0	2.86	25.39	+0.49
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Hin	Gom	1	2.47	24.12	-6.22
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Hin	Gom	2	2.47	23.96	-3.63
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Hin	Gom	3	2.41	23.69	-2.20
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Hin	Gom	4	0.04	3.09	-24.44
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Hin	Gom	5	0.02	2.49	-26.16

Table 14: Ablation on the number of in-context examples (k) for English→Hindi→Konkani translation using Nous Hermes. ΔchrF++ is computed relative to the Marathi-pivot setting at the same k. Scores computed with SacreBLEU (Post, 2018); signatures in Appendix A.2.

Parameter	Value
batch_size	1
num_train_epochs	1.5
warmup_ratio	0.03
logging_steps	25
learning_rate	2e-4
gradient_checkpointing	True
lr_scheduler_type	Cosine
weight_decay	0.001
save_strategy	No
optim	PagedAdam
warmup_steps	100
bf16	True

Table 15: Training parameters used in the model training process.

Parameter	Value
r	64
lora_alpha	16
lora_dropout	0.1
bias	none
task_type	CAUSAL_LM
target_modules	['q', 'k', 'v', 'o', 'up', 'down', 'gate', 'lm_head']

Table 16: LoRA configuration parameters.

```
do_sample: True
temperature: 0.1
num_return_sequences: 1
max_new_tokens: 200
return_full_text: False
```

Table 17: Inference parameters used for text generation.

Translation (Marathi): सुलेमानचे पालक बरेच उंच होते.

Post-edited (Konkani): <|im_end|>

<|im_start|>assistant: सुलेमानाचे पालक खूब उंच आशिल्ले. <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): Our country owes a deep

debt of gratitude to our valiant ex-Servicemen.

Translation (Marathi): आपल्या शूर माजी सैनिकांप्रति आपला देश कृतज्ञतेने अपार ऋणी आहे.

Post-edited (Konkani): <|im_end|>

<|im_start|>assistant: आमचो देश शूरवीर सेवानिवृत्त–सैनिकांक कृतज्ञतायेचें रीण देणें आसा. <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): Boys are equally vulnerable to sexual abuse.

Translation (Marathi): मुलगेही लैंगिक छळाला तेवढेच बळी पडू शकतात.

Post-edited (Konkani): <|im_end|>

<|im_start|>assistant: चलेय लैगिक अत्याचाराची तितलीच शिकार जावंक शकतात. <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): But Mangal Pandey's brave deed was done through devotion to a high and noble principle.

Translation (Marathi): पण मंगल पांडेची शौर्यशाली कृती ही एका उच्च आणि उदात्त तत्त्वाप्रतिच्या समर्पणातून केली गेली होती.

Post-edited (Konkani): <|im_end|>

<|im_start|>assistant: पूण मंगल पांडेचें धाडशी कर्तुब एके उंचेल्या आनी उदार तत्वनिश्ठेचें आशिल्लें. <|im_end|>

<|im_start|>user: APE is a task designed to enhance the quality of the translation by performing only minor adjustments to fix any existing translation mistakes. If the translation is already correct, you should retain it as is.

Original (English): The brothers were deeply attached to each other.

Translation (Marathi): भाऊ एकमेकांना खूप जवळ होते.

Post-edited (Konkani): <|im_end|>

<|im_start|>assistant: Translation: <|im_end|>

Response from the model setting with the highest Chrf++ score:

भावांनी एकमेकांकडेन खूब नजीकाय आशिल्ली.

Model	Source	Pivot	Target	BLEU	CHRF++
Few-shot Finetuned					
Unbabel/TowerInstruct-v0.1	English	-	Konkani	4.18	31.57
NousResearch/Hermes-2-Pro-Llama-3-8B	English	-	Konkani	3.49	31.49
Unbabel/TowerInstruct-v0.1	English	Marathi	Konkani	7.80	17.60
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Marathi	Konkani	12.14	34.92
Zero-shot Finetuned					
Unbabel/TowerInstruct-v0.1	English	-	Konkani	1.89	17.39
NousResearch/Hermes-2-Pro-Llama-3-8B	English	-	Konkani	4.01	36.61
Unbabel/TowerInstruct-v0.1	English	Marathi	Konkani	7.94	31.91
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Marathi	Konkani	8.38	40.17

Table 18: Performance comparison of finetuned models in few-shot and zero-shot settings for Konkani translation.

Model	Source	Pivot	Target	BLEU	CHRF++
Few-shot Finetuned					
Unbabel/TowerInstruct-v0.1	English	-	Tn	3.3	21.05
NousResearch/Hermes-2-Pro-Llama-3-8B	English	-	Tn	NA	NA
Unbabel/TowerInstruct-v0.1	English	Msa	Tn	2.82	17.12
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Tn	8.02	35.99
Zero-shot Finetuned					
Unbabel/TowerInstruct-v0.1	English	-	Tn	1.48	14.83
NousResearch/Hermes-2-Pro-Llama-3-8B	English	-	Tn	5.02	18.07
Unbabel/TowerInstruct-v0.1	English	Msa	Tn	2.09	19.16
NousResearch/Hermes-2-Pro-Llama-3-8B	English	Msa	Tn	4.62	21.87

Table 19: Performance comparison of finetuned models in few-shot and zero-shot settings for Tunisian Arabic translation.

CTC Regularization for Low-Resource Speech-to-Text Translation

Zachary Hopton and Rico Sennrich
{zacharywilliam.hopton,rico.sennrich}@uzh.ch
University of Zurich

Abstract

The challenges of building speech-to-text translation (ST) systems (e.g., a relative lack of parallel speech–text data and robustness to noise in audio) are exacerbated for low-resource language pairs. In this work, we seek to improve low-resource ST by building on previous studies that regularize ST training with the connectionist temporal classification (CTC) loss. By systematically evaluating a diverse range of linguistic annotations as CTC labels across multiple auxiliary loss configurations, we improve speech translation systems for both low- and high-resource settings. These improvements over both a standard end-to-end ST system and a speech LLM indicate a need for continued research on regularizing speech representations in ST.

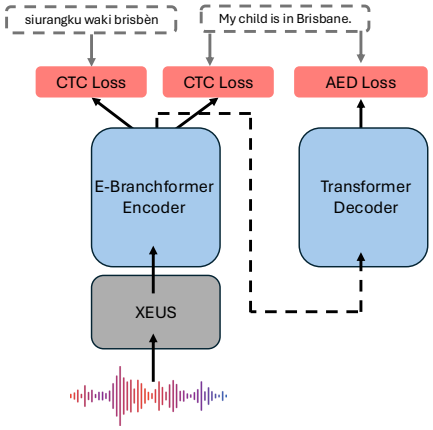


Figure 1: The **ASR-ST** synchronous CTC configuration. Ground truth sequences (transcription and translation of a Tondano utterance from He et al. (2024)) are in dashed gray boxes.

1 Introduction

Training end-to-end speech-to-text (ST) systems requires overcoming a scarcity of parallel data between modalities and the *lack of invariance* problem inherent to speech, whereby many signals can be mapped to the same phoneme (Xu et al., 2023; Appelbaum, 1996). Here, we study the feasibility of developing ST systems that push the former challenge to its limits, i.e., training ST systems for language pairs with under 10 hours of training data. Given that data sources for such extremely low-resource languages often come from linguistic fieldwork, a central question to this work is whether annotations from fieldwork can be used to improve ST models.

We leverage several approaches common in the low-resource ST literature—multilingual feature extractors, ASR pretraining, and regularization—then build on them by experimenting with labels and loss configurations for regularization with connectionist temporal classification (CTC). By directly comparing the effectiveness of using translations, transcriptions, interlinear glossings, and morphologically segmented transcriptions as CTC labels, we add to the body of work studying the most effective choice of label in CTC regularization for ST. We also go beyond single labels and compare various combinations of CTC labels using the synchronous CTC loss presented for high-resource ST by Xu et al. (2024). Our main findings are that morphological segmentation is a useful auxiliary task in low-resource ST, and that training a speech encoder with multiple regularizing objectives is also beneficial in very low-resource ST settings. All code to replicate the experiments is available[1].

2 Related Work

Because of the variation inherent to the speech modality, training end-to-end ST models presents a substantial modeling burden (Xu et al., 2023). With very small amounts of data (e.g., less than 5 hours) it is rare to achieve BLEU scores over 5.0 (Ortega et al., 2024). Still, a number of efforts have been made to improve very low-resource ST systems.

[1] https://github.com/zhopto3/lores-ctc-reg

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 186–197

Pretraining Pretraining some or all of an ST model's parameters can benefit low-resource ST. For instance, fine-tuning ASR models with small amounts of ST data is more effective than using the same amount of data to train an end-to-end ST model from scratch, even when the ASR model is for a language unrelated to the source or target (Bansal et al., 2019; Zhang et al., 2022). Moreover, it has been shown that pretrained, multilingual feature extractors such as XLS-R improve results for low-resource ST, indicating that cross-lingual transfer learning is another benefit of pretraining (Babu et al., 2022; Chen et al., 2024).

Regularization There have also been various efforts to improve end-to-end ST systems with auxiliary loss objectives that aim to regularize encoded representations of source language speech. Tang et al. (2021) used a speech and text encoder with an auxiliary loss objective that moved speech representations of the same sample closer to the text representations. The CTC loss—which jointly predicts the probability of both a sequence and its alignment with the source (Graves et al., 2006)—has also been used to regularize ST training, using both the source-language transcription and translation as labels (Bahar et al., 2019; Dong et al., 2021; Zhang et al., 2022; Yan et al., 2023; Xu et al., 2024). Though CTC regularization for end-to-end speech models was originally applied at the final encoder layer (Kim et al., 2017), applying CTC loss to features at intermediate layers in the encoder has been shown to benefit multilingual ASR (Chen et al., 2023).

3 Approach

3.1 CTC Regularization

The CTC loss over encoded speech representations can be jointly optimized with the attention-based encoder-decoder (AED) loss to regularize ST training (Kim et al., 2017). This formulation requires a source audio $X = x_1, ..., x_T$ and some target sequence of labels $S = s_1, ..., s_V$. The weights in an encoder H are optimized so the encoded representations of X minimize the following loss:

$$\mathcal{L}_{\text{CTC}} = \sum_{A \in A_{(X,S)}} \prod_{t=1}^{T} p(a_t|x_t), \quad (1)$$

where $A_{(X,S)}$ refers to the set of possible alignments between the source audio X and the target sequence S, and A is a sequence of target tokens and blank symbols, $a_1, ..., a_T$. A sequence is considered to align with the target output if it equals the target output after collapsing across equal, adjacent symbols and removing blank symbols; in practice dynamic programming is used to carry out this marginalization over valid alignments. See Prabhavalkar et al. (2023) and Graves et al. (2006) for a detailed review of CTC loss. Probabilities at each time step are generally calculated from a learned set of weights that take encoded speech features as input (Baevski et al., 2020; Zhang et al., 2023). Jointly, the encoder and decoder weights are optimized to minimize the autoregressive AED loss using the reference translation $Y = y_1, ..., y_L$ as the label:

$$\mathcal{L}_{\text{AED}} = -\sum_{t=1}^{T} \log P(y_t|y_{<t}, H(X)), \quad (2)$$

where H refers to an encoder model.

3.2 ST Model

Using CTC regularization, we train several ST models by fine-tuning an encoder-decoder model pretrained for multilingual ASR (Chen et al., 2024). We refer to this pretrained model as *XEUS-F*. See Appendix A for model specifications.

3.3 Data

The WAV2GLOSS:FIELDWORK dataset includes audio and interlinear glossing data compiled from fieldwork on 37 typologically and areally diverse languages (He et al., 2024). This data includes source audio transcriptions, morphological segmentations and interlinear glossings of the source audio, and translations into a higher-resource language (Figure 2). We select a sample of 5 languages from FIELDWORK for use in our experiments: Ainu (`ainu1240`), Beja (`beja1238`), Sumi Naga (`sumi1235`), Ruuli (`ruul1235`), and Tondano (`tond1251`). None of the languages we select have more than 10 hours of training data (Table 6), allowing us to compare the effectiveness of various CTC regularization strategies for benefiting extremely low-resource ST. See Appendix B for preprocessing details.

3.4 Experiments

Varying CTC Labels Previous work has used the source language transcription (Bahar et al., 2019; Dong et al., 2021) or the translation (Zhang et al., 2022; Yan et al., 2023) as the label S used

CTC Label		ainu1240	beja1238	ruul1235	sumi1235	tond1251	Mirco Avg.
Baseline	–	17.28	14.89	11.51	16.09	10.99	14.75
Morphological Segmentation	+InterCTC	20.14	17.02	13.11	18.34	13.10	**17.07***
	–InterCTC	18.54	17.74	12.93	17.96	12.15	**16.42***
Interlinear Glossing	+InterCTC	16.64	15.45	13.22	16.86	10.99	14.97
	–InterCTC	18.50	15.69	11.58	15.98	11.05	15.32*
Transcript	+InterCTC	19.89	17.79	13.10	18.15	11.91	16.85*
	–InterCTC	17.93	17.60	13.31	16.45	11.20	15.75*
Translation	+InterCTC	14.62	15.46	12.59	16.33	11.89	14.30*
	–InterCTC	12.44	13.76	10.58	14.99	10.76	12.62*

Table 1: Test set chrF2 scores for ST systems trained with various CTC labels. Baseline system is trained without CTC regularization. "+/- InterCTC" refers to whether intermediate CTC modules were used in training. **Bold** scores represent the best systems on average; underlined values represent the best systems for a given source language; *: Systems that differ significantly ($p < 0.05$) from the Baseline on average.

as the ground truth in Eq. 1. We experiment with these labels, as well as morphologically segmented transcriptions and interlinear glosses. Given that multitask learning of linguistic annotation tasks has been shown to improve low-resource MT systems (e.g., Zaremoodi et al. (2018)), we suspect that ST training may benefit from these labels in particular.

Specifically, we fine-tune XEUS-F for Xx→En ST using our sample of five source languages from FIELDWORK. For each language pair, we fine-tune several ST models, taking a different sequence label as the CTC ground truth in each. We jointly minimize the losses in Eq. 1 and 2, as well as K CTC losses calculated with intermediate encoder features (Chen et al., 2023, 2024). All CTC modules in a given fine-tuned model are optimized with the same label. The final joint CTC-AED loss used is as follows:

$$\mathcal{L}_{\text{joint}} = \lambda(w\left(\frac{1}{K}\sum_{k=1}^{K}\mathcal{L}_{\text{CTC-}k}\right) + (1-w)\mathcal{L}_{\text{CTC}}\Bigg) + (1-\lambda)\mathcal{L}_{\text{AED}} \quad (3)$$

where $K = 3$ (intermediate CTC modules at layers 3, 6, and 9 of the encoder), $w = 0.3$, and $\lambda = 0.3$, as in Chen et al. (2024). For comparison, we also train ST models where $K = 0$, i.e., without intermediate CTC modules.

Synchronous CTC Regularization In addition to studying the impact of intermediate CTC modules on low-resource ST, we compare various configurations of the synchronous CTC loss presented by Xu et al. (2024) (Figure 1). This consists in using the encoder's output features as input to B CTC modules, each with a different ground truth label S:

$$\mathcal{L}_{\text{ML}} = \lambda(\frac{1}{B}\sum_{b=1}^{B}\mathcal{L}_{\text{CTC-}b}) + (1-\lambda)\mathcal{L}_{\text{AED}} \quad (4)$$

We weight all CTC modules equally when calculating the loss and use a value of $\lambda = 0.5$. Using the loss in Eq. 4, we create several conditions: ***ASR-SEG*** ($B = 2$, S_1: transcription, S_2: morphological segmentation), ***ASR-ST*** ($B = 2$, S_1:transcription, S_2: translation), and ***ALL*** ($B = 4$, S_1: transcription, S_2: translation, S_3: morphological segmentation, S_4: interlinear glossing). To isolate the effects of this loss configuration, we do not use intermediate CTC modules in the Synchronous CTC Regularization experiments.

Throughout the results, we make comparisons to a **Baseline** model, which refers to XEUS-F fine-tuned for ST in a given language pair with only the AED loss (Eq. 2) and no CTC modules. See Appendices C and D for details on fine-tuning, inference, and evaluation. We evaluate using chrF2 (Popović, 2016). All significance testing is carried out using the SacreBLEU implementation of paired bootstrap resampling, with $N = 1000$ and a significance threshold of 0.05 (Post, 2018; Koehn, 2004).

4 Results

4.1 Varying the CTC Label

When experimenting with the label used for all CTC modules during ST fine-tuning, we find that using the transcription or the morphologically segmented transcription as CTC labels yields the largest improvements over the baseline model in terms of chrF2 (Table 1). In models with intermediate CTC modules, there is no significant difference

	ainu1240	beja1238	ruul1235	sumi1235	tond1251	Micro Avg.
Baseline	17.28	14.89	11.51	16.09	10.99	14.75
ASR-SEG	20.31	17.11	12.12	18.0	11.92	**16.76***
ASR-ST	20.14	17.0	12.55	16.98	12.06	16.52*
ALL	21.0	16.36	12.66	16.93	10.79	16.54*

Table 2: Test set chrF2 for varied Synchronous CTC configurations. Baseline system is trained without CTC regularization. *: Systems that differ significantly ($p < 0.05$) from Baseline on average.

	FIELDWORK	CoVoST 2
ASR-ST	**16.52**	**56.85**
+ASR, -ST	15.87	55.92
-ASR, +ST	14.03	53.16
-ASR, -ST	14.75	54.88

Table 3: Ablating CTC labels from the ***ASR-ST*** configuration. Values are micro-averaged test set chrF2 scores.

between transcriptions and morphologically segmented transcriptions ($p > 0.05$), but in models without intermediate CTC modules, morphological segmentation significantly outperforms transcripts ($p < 0.001$). Improvements from interlinear glosses are not significant with intermediate CTC modules, and on average, translations as CTC labels resulted in significantly worse models than the baseline. We report BLEU scores in Appendix E (Papineni et al., 2002).

4.2 Synchronous CTC Regularization

We find that all Synchronous CTC configurations significantly improve ST performance over the baseline model on average (Table 2). The ***ASR-SEG*** yields significant improvements over both the ***ASR-ST*** and ***ALL*** conditions.

Whereas Xu et al. (2024) explore only the ***ASR-ST*** setting, we find that the synchronous CTC loss is also beneficial when other labels are used. Xu et al. proposed that the main benefits of the loss come from training the model to encode language agnostic representations of source speech, but our findings broaden the utility of the loss and find that using it to encode task-agnostic and linguistically informed features is also beneficial for ST. Still, the ***ASR-ST*** configuration is noteworthy because it combines the two most readily available annotations and outperforms the use of translations or transcriptions alone. We therefore further investigate whether the benefits of ***ASR-ST*** generalize to a high-resource setting.

4.3 High-Resource ST

Using the Catalan, German, Spanish and French data from CoVoST 2 (Wang et al., 2020a,b), we fine-tune XEUS-F for Xx→En ST with the ***ASR-ST*** configuration (see Table 7 for data description). We then ablate each CTC label and compare to a model with no CTC regularization. No intermediate CTC modules are used. We include the analogous low-resource settings' values for comparison.

On average, fine-tuning with the ***ASR-ST*** configuration yields the highest chrF2 (Table 3). Fine-tuning with only the transcription label is beneficial on average, but using only the translation label for CTC regularization hurts performance relative to the baseline. Within low- and high-resource settings, all decreases in performance from the ***ASR-ST*** system are significant (all $p < 0.05$).

Previous work has found that despite breaking the CTC loss's assumption of monotonicity between the source and target sequence, models trained with the CTC loss can effectively carry out ST (Chuang et al., 2021). Moreover, using translation has been shown to be beneficial as a regularizing CTC task in MT and ST (Zhang et al., 2022; Yan et al., 2023). Our findings diverge from this, as they indicate significantly worse performance than our baseline when using translation as the CTC label, albeit with small magnitude differences. This discrepancy with previous work might be caused by our choice in the CTC module's relative weight with the AED loss. Unlike Zhang et al. and Yan et al., we fine-tune a model pretrained for ASR, so the lack of correspondence between the pretraining task and the CTC regularization task may contribute to this finding as well.

4.4 Speech LLM Comparison

Large language models (LLMs) adapted for the speech modality have become increasingly popular for ST (Gaido et al., 2024). Synchronous CTC regularization is most readily applicable to encoder-decoder ST models, so we compare XEUS-

	Avg. chrF2
Baseline	14.75
ASR-ST	**16.52**
Gemini 2.0–0 Shot	13.94
Gemini 2.0–3 Shot	13.65

Table 4: Test set chrF2 scores micro-averaged across our sample of languages from FIELDWORK. Baseline system is XEUS-F fine-tuned without CTC regularization.

F fine-tuned with synchronous CTC regularization to Google's Gemini 2.0 Flash (Comanici et al., 2025). Though the exact language composition of Gemini's training data is not known, recent work has examined the model's performance in low-resource ST settings (Beyene et al., 2025; Dauvet et al., 2025). Using the same subset of FIELDWORK languages, we explore whether encoder-decoder models with CTC regularization are still competitive for very-low-resource ST. See Appendix F for a description of the prompt.

We find that the fine-tuned encoder-decoder outperforms Gemini for ST, even when the model is trained without any CTC Regularization (Table 4). The findings are in line with recent work showing that speech LLMs struggle with low-resource ST and ASR (Beyene et al., 2025; Fong et al., 2025). When pretrained with comparable data, Lam et al. (2025) actually found that encoder-decoder models consistently perform on par with or better than decoder-only ST and ASR models.

5 Conclusion

We set out to study novel labels and configurations for CTC regularization to get the most out of small amounts of ST data. In doing so, we found that morphologically segmented transcriptions can be more beneficial than using translations or transcriptions as CTC labels. We also find evidence that extends the utility of Synchronous CTC to low-resource ST, as simultaneously training ST encoders to produce representations that are beneficial for several CTC tasks ultimately improved ST performance. We hope this encourages further work on the role auxiliary training objectives can have in training ST systems for very low-resource language pairs.

Limitations

The extent to which we could study the various annotations' effectiveness as labels for CTC regularization is limited by the pretrained model's tokenizer. The pretrained tokenizer likely split the morphological segmentations and interlinear glossing along arbitrary lines. Using an ST model trained from scratch would allow for training a more task-agnostic tokenizer, for example working from the byte or character level. Still, given the relatively small BPE vocabulary of our model (6,500 items), transcripts and morphemes in our low-resource source languages, English translations, and interlinear glosses were ultimately tokenized to characters or otherwise very small units. For instance, the Tondano transcription "PA'AYANGEN NèOKI" is tokenized as [_, P, A, ‘, A, Y, A, NG, EN, _, N, è, O, K, I], while its English translation is tokenized as [_,CH,I,L,D,RE,N,',S,_,T,O,Y,S]. This being the case, we do not have reason to believe that this particular tokenizer biased performance towards or against a given label.

Though the ***ASR-SEG*** configuration yields the numerically highest score on average, we are only able to explore the impact of the ***ASR-ST*** configuration in a high-resource setting. This limitation might be remedied by future work studying whether automatic interlinear glossing or morphological segmentation can be used to synthesize these annotations.

Acknowledgments

RS was funded by the Swiss National Science Foundation (project MUTAMUR; no. 213976).

References

Irene Appelbaum. 1996. The lack of invariance problem and the goal of speech perception. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, volume 3, pages 1541–1544. IEEE.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2022. XLS-R: self-supervised cross-lingual speech representation learning at scale. In *23rd Annual Conference of the International Speech Communication Association, Interspeech 2022, Incheon, Korea, September 18-22, 2022*, pages 2278–2282. ISCA.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Parnia Bahar, Tobias Bieschke, and Hermann Ney. 2019. A comparative study on end-to-end speech to text translation. In *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2019, Singapore, December 14-18, 2019*, pages 792–799. IEEE.

Sameer Bansal, Herman Kamper, Karen Livescu, Adam Lopez, and Sharon Goldwater. 2019. Pre-training on high-resource speech recognition improves low-resource speech-to-text translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 58–68, Minneapolis, Minnesota. Association for Computational Linguistics.

Luel Hagos Beyene, Vivek Verma, Min Ma, Jesujoba O Alabi, Fabian David Schmidt, Joyce Nakatumba-Nabende, and David Ifeoluwa Adelani. 2025. msteb: Massively multilingual evaluation of llms on speech and text tasks. *arXiv preprint arXiv:2506.08400*.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023. An open dataset and model for language identification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 865–879, Toronto, Canada. Association for Computational Linguistics.

William Chen, Brian Yan, Jiatong Shi, Yifan Peng, Soumi Maiti, and Shinji Watanabe. 2023. Improving massively multilingual ASR with auxiliary CTC objectives. In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pages 1–5. IEEE.

William Chen, Wangyou Zhang, Yifan Peng, Xinjian Li, Jinchuan Tian, Jiatong Shi, Xuankai Chang, Soumi Maiti, Karen Livescu, and Shinji Watanabe. 2024. Towards robust speech representation learning for thousands of languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10205–10224, Miami, Florida, USA. Association for Computational Linguistics.

Shun-Po Chuang, Yung-Sung Chuang, Chih-Chiang Chang, and Hung-yi Lee. 2021. Investigating the reordering capability in CTC-based non-autoregressive end-to-end speech translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1068–1077, Online. Association for Computational Linguistics.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit S. Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 81 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2022. FLEURS: few-shot learning evaluation of universal representations of speech. In *IEEE Spoken Language Technology Workshop, SLT 2022, Doha, Qatar, January 9-12, 2023*, pages 798–805. IEEE.

Jonah Dauvet, Min Ma, Jessica Ojo, and David Ifeoluwa Adelani. 2025. Reassessing speech translation for low-resource languages: Do LLMs redefine the state-of-the-art against cascaded models? In *Proceedings of the 5th Workshop on Multilingual Representation Learning (MRL 2025)*, pages 149–160, Suzhuo, China. Association for Computational Linguistics.

Qianqian Dong, Rong Ye, Mingxuan Wang, Hao Zhou, Shuang Xu, Bo Xu, and Lei Li. 2021. Listen, understand and translate: Triple supervision decouples end-to-end speech-to-text translation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12749–12759. AAAI Press.

Seraphina Fong, Marco Matassoni, and Alessio Brutti. 2025. Speech llms in low-resource scenarios: Data volume requirements and the impact of pretraining on high-resource languages. *arXiv preprint arXiv:2508.05149*.

Marco Gaido, Sara Papi, Matteo Negri, and Luisa Bentivogli. 2024. Speech translation with speech foundation models and large language models: What is there and what is missing? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14760–14778, Bangkok, Thailand. Association for Computational Linguistics.

Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, volume 148 of *ACM International Conference Proceeding Series*, pages 369–376. ACM.

Taiqi He, Kwanghee Choi, Lindia Tjuatja, Nathaniel Robinson, Jiatong Shi, Shinji Watanabe, Graham Neubig, David Mortensen, and Lori Levin. 2024. Wav2Gloss: Generating interlinear glossed text from speech. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 568–582, Bangkok, Thailand. Association for Computational Linguistics.

Kwangyoun Kim, Felix Wu, Yifan Peng, Jing Pan, Prashant Sridhar, Kyu Jeong Han, and Shinji Watanabe. 2022. E-branchformer: Branchformer with enhanced merging for speech recognition. In *IEEE*

Spoken Language Technology Workshop, SLT 2022, Doha, Qatar, January 9-12, 2023, pages 84–91. IEEE.

Suyoun Kim, Takaaki Hori, and Shinji Watanabe. 2017. Joint ctc-attention based end-to-end speech recognition using multi-task learning. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*, pages 4835–4839. IEEE.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Tsz Kin Lam, Marco Gaido, Sara Papi, Luisa Bentivogli, and Barry Haddow. 2025. Prepending or cross-attention for speech-to-text? an empirical comparison. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2994–3006, Albuquerque, New Mexico. Association for Computational Linguistics.

John E. Ortega, Rodolfo Joel Zevallos, Ibrahim Said Ahmad, and William Chen. 2024. QUESPA submission for the IWSLT 2024 dialectal and low-resource speech translation task. In *Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024)*, pages 125–133, Bangkok, Thailand (in-person and online). Association for Computational Linguistics.

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*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. In *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September 15-19, 2019*, pages 2613–2617. ISCA.

Maja Popović. 2016. chrF deconstructed: beta parameters and n-gram weights. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 499–504, Berlin, Germany. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Rohit Prabhavalkar, Takaaki Hori, Tara N Sainath, Ralf Schlüter, and Shinji Watanabe. 2023. End-to-end speech recognition: A survey. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:325–351.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Masao Someki, Kwanghee Choi, Siddhant Arora, William Chen, Samuele Cornell, Jionghao Han, Yifan Peng, Jiatong Shi, Vaibhav Srivastav, and Shinji Watanabe. 2024. Espnet-ez: Python-only espnet for easy fine-tuning and integration. In *IEEE Spoken Language Technology Workshop, SLT 2024, Macao, December 2-5, 2024*, pages 863–870. IEEE.

Yun Tang, Juan Pino, Xian Li, Changhan Wang, and Dmitriy Genzel. 2021. Improving speech translation by understanding and learning from the auxiliary text translation task. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4252–4261, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Changhan Wang, Juan Pino, Anne Wu, and Jiatao Gu. 2020a. CoVoST: A diverse multilingual speech-to-text translation corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4197–4203, Marseille, France. European Language Resources Association.

Changhan Wang, Anne Wu, and Juan Pino. 2020b. Covost 2: A massively multilingual speech-to-text translation corpus. *Preprint*, arXiv:2007.10310.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. Espnet: End-to-end speech processing toolkit. In *Proc. Interspeech*, pages 2207–2211.

Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R. Hershey, and Tomoki Hayashi. 2017. Hybrid

ctc/attention architecture for end-to-end speech recognition. *IEEE J. Sel. Top. Signal Process.*, 11(8):1240–1253.

Chen Xu, Xiaoqian Liu, Erfeng He, Yuhao Zhang, Qianqian Dong, Tong Xiao, Jingbo Zhu, Dapeng Man, and Wu Yang. 2024. Bridging the gaps of both modality and language: Synchronous bilingual ctc for speech translation and speech recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12176–12180. IEEE.

Chen Xu, Rong Ye, Qianqian Dong, Chengqi Zhao, Tom Ko, Mingxuan Wang, Tong Xiao, and Jingbo Zhu. 2023. Recent advances in direct speech-to-text translation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI '23.

Brian Yan, Siddharth Dalmia, Yosuke Higuchi, Graham Neubig, Florian Metze, Alan W Black, and Shinji Watanabe. 2023. CTC alignments improve autoregressive translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1623–1639, Dubrovnik, Croatia. Association for Computational Linguistics.

Shu-wen Yang, Heng-Jui Chang, Zili Huang, Andy T Liu, Cheng-I Lai, Haibin Wu, Jiatong Shi, Xuankai Chang, Hsiang-Sheng Tsai, Wen-Chin Huang, and 1 others. 2024. A large-scale evaluation of speech foundation models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, Da-Rong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee. 2021. SUPERB: Speech Processing Universal PERformance Benchmark. In *Proc. Interspeech 2021*, pages 1194–1198.

Poorya Zaremoodi, Wray Buntine, and Gholamreza Haffari. 2018. Adaptive knowledge sharing in multi-task learning: Improving low-resource neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 656–661, Melbourne, Australia. Association for Computational Linguistics.

Biao Zhang, Barry Haddow, and Rico Sennrich. 2022. Revisiting end-to-end speech-to-text translation from scratch. In *International conference on machine learning*, pages 26193–26205. PMLR.

Biao Zhang, Barry Haddow, and Rico Sennrich. 2023. Efficient CTC regularization via coarse labels for end-to-end speech translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2264–2276, Dubrovnik, Croatia. Association for Computational Linguistics.

A Xeus-F

	XEUS Layer	Encoder Layer	Decoder Layer
Num. Attention Heads	8	8	8
Hidden State Size	1024	512	512
Feed-Forward Network Size	4096	2048	2048
Convolutional Kernel Dim.	31x31	31x31	—

Table 5: Configuration of each layer in the XEUS-F feature extractor (XEUS), encoder, and decoder. Convolutional kernel dimension only applies to the E-branchformer architecture of XEUS and the encoder. All values are from Chen et al. (2024).

XEUS-F is a model trained in Chen et al. (2024). It consists in a 12-layer E-branchformer encoder and a 6-layer transformer decoder (Kim et al., 2022; Vaswani et al., 2017). The input to the encoder are features from XEUS, a feature extractor trained with several masked prediction objectives on speech from over 4,000 languages (Chen et al., 2024). XEUS-F was trained for joint language identification and multilingual ASR using the FLEURS dataset (Conneau et al., 2022). Across all 102 languages in FLEURS, there are 987 hours of training data. XEUS consists of 577M parameters, while the downstream encoder and decoder total 100M parameters, making a total of 677M parameters in XEUS-F. Weights in the feature extractor were frozen during XEUS-F's ASR training. The layer configurations in the encoder and decoder are described in Table 5.

The model uses a multilingual BPE vocabulary of 6,500 items trained using the FLEURS transcriptions (Sennrich et al., 2016). All Latin characters are uppercase except for those with diacritics.

B Data

surface	mina	tura	mo	-no	a	hine	ipe.
underlying	mina	tura	mo	-no	a	hine	ipe
gloss	laugh	together.with	quiet	-ADV	sit.SG	and	eat
transcription	mina tura mono a hine ipe.						
translation	He sat down smiling and ate his dinner.						

Figure 2: A sample data point from the Ainu corpus in FIELDWORK (He et al., 2024). Data points are annotated with their surface and underlying morphological segmentations (the uttered allomorph versus the abstract underlying form of the morpheme, respectively), an interlinear gloss describing the lexical meaning or grammatical function of a unit, a transcription, and a translation. When morphological segmentations are used as CTC labels throughout our experiments, we refer to the underlying segmentations.

	Hours (train/dev/test)	Number of Samples (train/dev/test)
ainu1240	7.11/0.27/0.86	6711/248/749
beja1238	1.53/0.07/0.21	5257/241/733
ruul1235	0.92/0.08/0.18	1886/244/390
sumi1235	0.40/0.10/0.30	939/246/727
tond1251	0.22/0.17/0.50	303/249/713

Table 6: Description of the train, development, and test set of FIELDWORK following removal of samples that are empty after preprocessing. We provide both durations (in hours) and the number of parallel samples in each split.

Preprocessing We focus on Xx→En ST, so we use OpenLID-v2 to filter out any samples from the FIELDWORK data without English translations (Burchell et al., 2023). In line with the tokenizer of the pretrained XEUS-F model we fine-tune for ST, we uppercase all text. We remove explicative material appearing in brackets in English translations.

	Hours (train/dev/test)	Number of Samples (train/dev/test)
Catalan	135.55/18.95/20.21	95854/12730/12730
French	264.27/21.75/23.30	207372/14760/14760
German	184.29/20.65/21.55	127824/13511/13511
Spanish	113.10/21.81/22.71	79013/13221/13221

Table 7: Description of the train, development, and test set of our sample of languages from CoVoST 2. We provide both durations (in hours) and the number of parallel samples in each split.

C Fine-Tuning for ST

When further fine-tuning XEUS-F for Xx→En ST, we generally follow the hyperparameters used to fine-tune XEUS-F for ST in Chen et al. (2024). That is, we fine-tune with a constant learning rate of $1e^{-4}$ using the Adam optimizer (Kingma and Ba, 2015). If matching Chen et al.'s (2024) batch size of 32 is not possible because of memory constraints, we use a smaller batch size with gradient accumulation to carry out weight updates. We apply SpecAugment to extracted features using the same configurations used during XEUS-F pretraining (Park et al., 2019). This includes time warping, applying frequency masks between 0 and 30 frequency bins wide, and applying time masks between 0 and 40 time steps wide. We fine-tune the encoder and decoder but freeze the parameters in XEUS. Models are trained for a maximum of 100 epochs, with an early stopping mechanism that ends training if 5 epochs pass without improvement in the validation loss. The five models with the highest token-level accuracy on the development set are retained and averaged.

We fine-tune with automatic mixed precision using ESPnet-EZ (Someki et al., 2024; Watanabe et al., 2018). The library s3prl is used to integrate the pretrained XEUS feature extractor with the downstream XEUS-F model (Yang et al., 2021, 2024). The loss functions used for fine-tuning are detailed in 3.4. We fine-tune all models using a single GPU. If the required memory allowed for it in a given experiment, this GPU was a 32 GB V100 GPU. Otherwise, we used an 80 GB A100 GPU.

D Inference and Evaluation

We run inference using beam search with a beam size of 10. There are various algorithms for combining tokens' posterior probability distributions as predicted by the CTC module and the decoder of models trained with joint CTC-attention loss (Yan et al., 2023; Watanabe et al., 2017). However, as in Zhang et al. (2022), we do not use the CTC module during decoding. We use length-normalized scores during beam search, and we do not inform decoding with external language models. Inference is carried out on a single 32 GB V100 GPU.

For brevity, we report corpus-level chrF2 scores throughout Section 4, as in Chen et al. (2024). Scores are calculated after removing language tags output by XEUS-F, using the SacreBLEU library (Post, 2018). We also report corpus-level BLEU scores from our experiments (Papineni et al., 2002) in Appendix E.

E BLEU Evaluation

Condition		ainu1240	beja1238	ruul1235	sumi1235	tond1251	Micro Avg.
Baseline	–	1.99	3.41	0.42	0.55	0.23	2.11
Morphological Segmentation	+InterCTC	2.74	3.72	0.39	1.58	0.41	**2.57***
	–InterCTC	2.63	4.38	0.44	1.67	0.39	**2.73***
Interlinear Glossing	+InterCTC	1.83	3.95	0.36	0.82	0.28	2.04
	–InterCTC	2.79	3.43	0.29	0.83	0.33	2.48*
Transcript	+InterCTC	2.74	4.30	0.46	1.39	0.39	**2.57***
	–InterCTC	2.50	4.39	0.43	0.71	0.30	2.55*
Translation	+InterCTC	1.33	3.71	0.29	0.61	0.21	1.80*
	–InterCTC	1.41	3.26	0.52	0.60	0.24	1.99

Table 8: Test set BLEU scores for ST systems trained with various CTC labels. Baseline system is trained without CTC regularization. "+/- InterCTC" refers to whether intermediate CTC modules were used in training. **Bold** scores represent the best systems on average; underlined values represent the best systems for a given source language; *: Systems that differ significantly ($p < 0.05$) from the Baseline on average.

	ainu1240	beja1238	ruul1235	sumi1235	tond1251	Micro Avg.
Baseline	1.99	3.41	0.42	0.55	0.23	2.11
ASR-SEG	3.06	3.97	0.37	1.01	0.36	**2.78***
ASR-ST	2.74	4.46	0.15	0.94	0.26	2.50*
ALL	3.21	3.62	0.43	0.57	0.23	2.54*

Table 9: Test set BLEU scores for ST systems trained with various Synchronous CTC configurations. Baseline system is trained without CTC regularization; *: Systems that differ significantly ($p < 0.05$) from Baseline on average.

	FIELDWORK	CoVoST 2
ASR-ST	2.50	**30.40**
+ASR, -ST	**2.68**	29.44*
-ASR, +ST	1.94*	26.19*
-ASR, -ST	2.11*	28.25*

Table 10: Ablating CTC labels from the ***ASR-ST*** configuration. Values are micro-averaged test set BLEU scores.

	Average
Baseline	2.11
ASR-ST	**2.50**
Gemini 2.0–0 Shot	0.59
Gemini 2.0–3 Shot	0.52

Table 11: Test set BLEU scores micro-averaged across our sample of languages from FIELDWORK. Baseline system is XEUS-F fine-tuned without CTC regularization.

F Speech LLM Comparison

When carrying out ST inference with the speech language model `gemini-2.0-flash`, we used a prompt similar to that described by Beyene et al. (2025). In the zero-shot condition, we prompted the model with:

"You are a translation expert. Listen to the following audio in {LANGUAGE} and translate it to English. Return only the translated sentence."

In the three-shot condition, we randomly selected three samples from the source language's validation set and provided them to the models with their English translation as examples in the prompt.

Navigating Data Scarcity in Low-Resource English-Tatar Translation using LLM Fine-Tuning

Ahmed Khaled Khamis
Georgia Institute of Technology
akhamis6@gatech.edu

Abstract

The scarcity of high-quality parallel corpora remains the primary bottleneck for English-Tatar machine translation. While the OPUS project provides various datasets, our tests reveal that datasets like WikiMatrix, GNOME, and NLLB, suffer from significant noise and incorrect labeling, making them unsuitable for training robust encoder-decoder translation models that typically requires larger amount of high quality data. Furthermore, we demonstrate that small-scale multilingual Large Language Models (LLMs), such as Qwen3 (4B-30B), Gemma3 (4B-12B) and others, show severe "Turkish interference", and they frequently hallucinate Turkish vocabulary when prompted for Tatar. In this paper, we navigate this data scarcity by leveraging Llama 3.3 70B Instruct, which is the only model in our zero-shot benchmarks capable of maintaining distinct linguistic boundaries for Tatar. To address the lack of gold-standard data, we curated a synthetic dataset of 7,995 high-quality translation pairs using a frontier model as a teacher. We then performed 4-bit LoRA fine-tuning to train Llama for English-Tatar translation. Our results show a performance leap: while fine-tuning on the limited Tatoeba dataset (1,193 samples) yielded a CHRF++ score of 24.38, while fine-tuning on our synthetic dataset achieved 32.02 on the LoResMT 2026 shared task test set. We release our curated dataset and fine-tuned models to support further research in low-resource Turkic machine translation.

1 Introduction

Neural Machine Translation (NMT) has been improving by the success of massive datasets and high-parameter architectures. However, for low-resource languages such as Tatar, progress remains constrained by a deep and persistent data scarcity gap. While open source projects like the OPUS corpus (Tiedemann and Thottingal, 2020) offer several English-Tatar datasets, their utility is limited due to poor alignment, incorrect labels, and significant linguistic noise. Our investigation into the *WikiMatrix*, *GNOME*, and *NLLB* corpora revealed that a substantial portion of these datasets are practically unusable for translation tasks. In the face of such scarcity, traditional encoder-decoder architectures like *MarianMT* (Junczys-Dowmunt et al., 2018) struggle to generalize, often collapsing into repetitive or nonsensical outputs.

Simultaneously, the rise of multilingual Large Language Models (LLMs) promised a new era of zero-shot translation. Yet, our benchmarking of modern instruction-tuned models—including Qwen3 (4B-30B) (Yang et al., 2025), Gemma3 (4B-12B) (Team et al., 2025), and Llama 3 (3B-8B) (Grattafiori et al., 2024), showed a critical failure mode: "Turkic interference" Despite their multilingual training, these smaller models frequently confuse Tatar with Turkish, outputting Turkish vocabulary and morphological structures when prompted for Tatar. This suggests that smaller parameter counts may be insufficient to maintain the nuanced linguistic boundaries required for low-resource languages.

This paper details our approach[1] for the ***LoResMT*** 2026 English-Tatar Shared Task. We demonstrate that navigating data scarcity requires both: Model Scale and Synthetic Curation. During our zero-shot evaluations, we identified that *Llama 3.3 70B Instruct* was a model that's capable of generating coherent Tatar without significant Turkish bias. However, even at this scale, the 1,193 available samples from the Tatoeba corpus proved insufficient for competitive performance. To bridge this gap, we used DeepSeek-R1 (DeepSeek-AI et al., 2025) to curate a high-quality synthetic dataset of 7,995 translation pairs. By fine-tuning the 70B parameter model on this curated data using the on an

[1]Code: https://github.com/KickItLikeShika/llm-loresmt

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 198–202
March 28, 2026

NVIDIA H100, we achieve a CHRF++ score of 32.02, outperforming our Tatoeba-only baseline of 24.38.

2 The Challenge of Data Scarcity

The primary obstacle in English-Tatar translation is not just the quantity of data, but lack of high-fidelity, correctly labeled corpora. This section details our evaluation of available resources and the failure of zero-shot models to address this gap.

2.1 Evaluation of Existing Corpora

We conducted a comprehensive review of the English-Tatar datasets available via the OPUS corpus (Tiedemann and Thottingal, 2020), including NLLB (Team et al., 2022), WikiMatrix (Schwenk et al., 2019), GNOME (Deshpande et al., 2024), XLEnt (El-Kishky et al., 2021), and QED (Lamm et al., 2020). Our qualitative analysis revealed that many of these datasets are unsuitable for training high-quality translation models. The WikiMatrix and NLLB corpora, for instance, exhibited significant noise where Russian or Turkish segments were incorrectly labeled as Tatar. In the GNOME and XLEnt datasets, we observed a high frequency of "hallucinated" labels—where the source and target segments were semantically unrelated. Training on such corpora led to models that to completely fail to generate coherent Tatar syntax.

2.2 The Tatoeba Baseline

Among the open-source datasets, only the Tatoeba (Tiedemann, 2020) corpus provided a degree of reliable alignment. However, after aggressive filtering for duplicates, empty segments, and language identification, we were left with only 1,193 high-quality sentence pairs. While useful as a starting point, a dataset of roughly 1,200 samples is insufficient for training a robust encoder-decoder model from scratch or for effectively adapting a decoder-only LLM.

2.3 Zero-Shot Benchmarking and Turkic Interference

Given the data scarcity, we explored the zero-shot capabilities of several modern, instruction-tuned Large Language Models. We tested models across various scales, including Qwen3 (4B, 8B, 14B, 30B) (Yang et al., 2025), Gemma3 (4B, 12B, 27B) (Team et al., 2025), Llama 3.2 3B, and Llama 3 8B (Grattafiori et al., 2024).

Despite their general multilingual proficiency, all models within this parameter range exhibited a specific failure mode that we called "Turkic Interference" when prompted to translate into Tatar, these models consistently struggled to differentiate between Tatar and its high-resource relative, Turkish. Common errors included:

- Lexical and Morphological failures: Using Turkish words instead of Tatar equivalents.
- Instruction failure: Models frequently generated long explanations in English rather than the requested translation or repeated the same word multiple times.

Llama 3.3 70B Instruct was the only model in our evaluation that demonstrated a foundational ability to maintain Tatar's linguistic identity, serving as the necessary baseline for our fine-tuning experiments.

3 Methodology

To navigate the extreme data scarcity of the English-Tatar pair, we curated a synthetic dataset. Our methodology centers on using a high-parameter frontier model to bootstrap a high-quality corpus, followed by parameter-efficient fine-tuning (Xu et al., 2023) of a 70B parameter decoder-only model.

3.1 Synthetic Data Curation

Given the unreliability of existing web-scraped corpora, we utilized DeepSeek for dataset generation. We curated a dataset consisting of 7,995 high-quality translation pairs. Unlike the noisy OPUS datasets, these pairs were generated through structured prompting designed to ensure grammatical correctness in Tatar.

The structural properties of the resulting dataset (Figure 1) provide insight about the linguistic relationship between the two languages. English sentences have a mean length of 14.99 words, while Tatar translations average 10.78 words, while the character counts remain nearly identical (75.21 for English vs. 74.74 for Tatar).

3.2 Model Selection and Architecture

Based on our zero-shot benchmarks, we selected Llama 3.3 70B Instruct as our base model. While 8B parameter models exhibited significant linguistic "bleeding" from high-resource Turkic languages like Turkish, the 70B parameter scale provided the

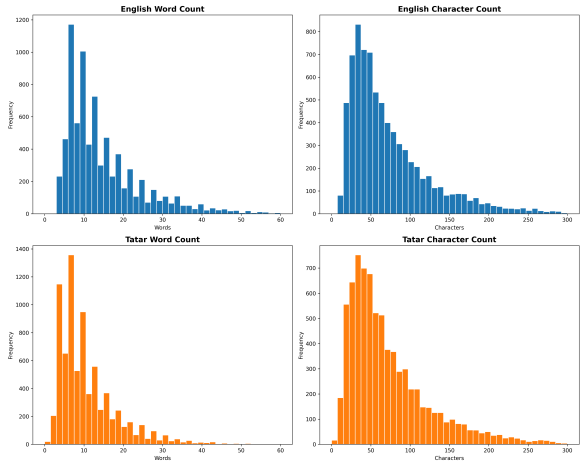


Figure 1: Linguistic profile of the curated synthetic dataset. The top row displays English word and character counts; the bottom row displays the corresponding Tatar counts.

necessary internal representation to maintain Tatar-specific syntax and vocabulary. To make training feasible, we used 4-bit quantization. This allowed us to load the 70B parameter weights into GPU memory with minimal impact on perplexity, providing a foundation for subsequent Parameter-Efficient Fine-Tuning (PEFT) (Xu et al., 2023).

3.3 Training Configuration

We utilized the Unsloth framework (Daniel Han and team, 2023) to perform LoRA (Hu et al., 2021) fine-tuning. This framework provides optimized CUDA kernels that significantly reduce VRAM consumption and increase training speed, enabling us to fine-tune a 70B model on a single node efficiently. Our training was conducted with the following hyperparameters: **LoRA Parameters**: We targeted all linear modules (including *q_proj, k_proj, v_proj, o_proj*, and MLP layers) with a Rank (r) of 64 and an Alpha of 64. This relatively high rank was chosen to maximize the model's capacity to adapt to Tatar's specific morphological requirements.

Optimization Strategy: We employed the *train_on_responses_only* technique. By masking the loss for the instruction and source English text, we ensured the gradient updates were computed exclusively on the Tatar translation output.

Hyperparameters: The model was trained for 2 epochs with a learning rate of 2e-4 and a linear scheduler. We used a global batch size of 16 (8 per device with 2 gradient accumulation steps) and a weight decay of 0.001 to prevent overfitting on the synthetic samples.

4 Experimental Results

4.1 Quantitative Performance

We evaluated our models on the official LoResMT 2026 English-Tatar shared task test set using the CHRF++ metric. Our results (Table 1) demonstrate the clear advantage of synthetic augmentation.

Model	Training Data	CHRF++
Llama 3.3 70B	Tatoeba	24.3842
Llama 3.3 70B	**Synthetic**	**32.0251**

Table 1: Translation performance comparison on the LoResMT 2026 shared task test set.

The transition from the tiny Tatoeba corpus to our curated synthetic dataset resulted in a 7.64 point jump in CHRF++. This improvement suggests that the model effectively internalized the distinct Tatar syntax and vocabulary, successfully overcoming the "Turkic interference" observed in zero-shot baselines.

4.2 Training Environment

All experiments were conducted on a single NVIDIA H100 (94GB) GPU using the `Unsloth` framework for 4-bit LoRA fine-tuning. This setup enabled us to train the 70B parameter model with a total batch size of 16 in approximately 1.5 hours. The efficiency of this pipeline demonstrates that high-parameter models can be adapted to low-resource tasks with limited computational overhead if the data quality is sufficiently high.

5 Discussion and Error Analysis

The significant performance gains observed in our experiments underscore two critical factors in low-resource machine translation for Turkic languages: the necessity of high-parameter model scales and the role of synthetic supervision in decoupling linguistic interference.

5.1 Linguistic Interference and Model Scale

A primary challenge identified in our zero-shot benchmarks was the "Turkic interference" phenomenon, where models frequently defaulted to Turkish (tr) vocabulary and morphology when prompted for Tatar (tt). We hypothesize that smaller models (e.g., 3B to 30B parameters) possess an internal representation that is insufficient

to maintain distinct boundaries between closely related languages within the same family. In these models, the high-resource presence of Turkish in the pre-training data caused the model to prioritize Turkish tokens over Tatar counterparts.

Our results suggest that the 70B parameter scale of Llama 3.3 is a critical threshold for English-Tatar translation. The larger parameter count appears to provide a more granular latent space, allowing the model to isolate and preserve Tatar-specific features, even under extreme data scarcity. Fine-tuning on our curated synthetic dataset further reinforced these boundaries, effectively "teaching" the model to resist the Turkish default.

5.2 Qualitative Analysis

A qualitative review of the outputs from the Tatoeba-baseline and the synthetic-augmented model reveals several key improvements. Models fine-tuned only on the limited Tatoeba corpus often struggled with Tatar's complex sturcture, occasionally producing "broken" words.

In contrast, the synthetic-augmented model demonstrated a better command of Tatar morphology. For example, in translating complex temporal or locative phrases, the model correctly utilized Tatar-specific markers rather than the more common Turkish equivalents found in zero-shot outputs. Furthermore, the fine-tuned model strictly adhered to the "no-explanation" instruction, whereas zero-shot models frequently included English commentary text.

5.3 The Value of Synthetic Supervision

Our findings demonstrate that for low-resource languages, the *quality* and *purity* of the training data are more important than data volume. By using a frontier model to generate a curated dataset, we were able to provide the 70B model with a "clean" signal of what constitutes correct Tatar. This approach successfully bypassed the noise inherent in web-scraped corpora like WikiMatrix or GNOME, which our analysis showed multiple mislabeled Turkish or Russian data. The 7.64 CHRF++ jump proves that synthetic data from a superior LLM can serve as a high-fidelity surrogate for native-speaker data in extreme scarcity scenarios.

6 Conclusion

In this work, we addressed the data scarcity in English-Tatar machine translation by transitioning from traditional web-scraped corpora to a high-quality synthetic curation strategy. Our investigation revealed that existing large-scale datasets for Tatar often contain significant linguistic noise, while small-scale multilingual LLMs frequently suffer from Turkic interference, failing to distinguish Tatar from higher-resource relatives like Turkish.

By leveraging the Llama 3.3 70B parameter model and a curated synthetic dataset of 7,995 translation pairs, we achieved a robust performance benchmark, with CHRF++ score of 32.02 on the LoResMT 2026 shared task. Our results demonstrate that model scale is critical for preserving the linguistic integrity of low-resource languages within dense language families. Furthermore, we provide evidence that synthetic data generated by frontier models can serve as a high-fidelity training signal, successfully bypassing the limitations of noisy, web-scraped corpora.

References

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Darshan Deshpande, Shambhavi Sinha, Anirudh Ravi Kumar, Debaditya Pal, and Jonathan May. 2024. Gnome: Generating negotiations through open-domain mapping of exchanges. *Preprint*, arXiv:2406.10764.

Ahmed El-Kishky, Adithya Renduchintala, James Cross, Francisco Guzmán, and Philipp Koehn. 2021. Xlent: Mining a large cross-lingual entity dataset with lexical-semantic-phonetic word alignment. *Preprint*, arXiv:2104.08597.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in c++. *Preprint*, arXiv:1804.00344.

Matthew Lamm, Jennimaria Palomaki, Chris Alberti, Daniel Andor, Eunsol Choi, Livio Baldini Soares, and Michael Collins. 2020. Qed: A framework and dataset for explanations in question answering. *Preprint*, arXiv:2009.06354.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia. *Preprint*, arXiv:1907.05791.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Jörg Tiedemann. 2020. The tatoeba translation challenge – realistic data sets for low resource and multilingual mt. *Preprint*, arXiv:2010.06354.

Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. 2023. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *Preprint*, arXiv:2312.12148.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

No One-Size-Fits-All: Building Systems For Translation to Bashkir, Kazakh, Kyrgyz, Tatar and Chuvash Using Synthetic And Original Data

Dmitry Karpov
PAO Severstal / Moscow, Russia
dimakarp1996@yandex.ru

Abstract

We explore machine translation for five Turkic language pairs: Russian-Bashkir, Russian-Kazakh, Russian-Kyrgyz, English-Tatar, English-Chuvash. Fine-tuning nllb-200-distilled-600M with LoRA on synthetic data achieved chrF++ 49.71 for Kazakh and 46.94 for Bashkir. Prompting DeepSeek-V3.2 with retrieved similar examples achieved chrF++ 39.47 for Chuvash. For Tatar, zero-shot or retrieval-based approaches achieved chrF++ 41.6, while for Kyrgyz the zero-shot approach reached 45.6. We release the dataset and the obtained weights.

1 Introduction

Machine translation for low-resource languages remains challenging. The "Machine Translation for Low-Resource Turkic Languages" competition focused on five pairs: Russian-Kazakh, Russian-Kyrgyz, Russian-Bashkir, English-Tatar, English-Chuvash. We investigate multiple approaches to improve translation quality in data-scarce conditions.

2 Making The Data

Unfortunately, only a limited amount of high-quality parallel data was available at the time of this study. Therefore, we used a variety of datasets. Specifically, we used the parallel English-Chuvash corpus from Plotnikov and Antonov (2024) for translations from English to Chuvash, Zhang et al. (2020) for English-Tatar, English-Kyrgyz, and English-Kazakh translations, NLLB Team et al. (2024) to obtain the parallel data for Russian, English, Bashkir, Tatar, Kazakh and Kyrgyz, ips (2025) for translations from Russian to Tatar, Singh et al. (2024) for the parallel Russian-Kyrgyz data, (Tiedemann, 2020) for the parallel data for all 5 language pairs, Iskander Shakirov (2023) for the parallel Russian-Bashkir data, Yeshpanov et al. (2024) and Abdashim (2024) for the Russian-Kazakh data and gou for Russian-Kyrgyz pair. Unfortunately, we could not use the parallel corpora from the TurkLang-7 project Khusainov and Minsafina (2021) as this corpora was not available at the time of study. The original training set sizes were: 1,190,773 samples for Russian-Bashkir (9,997 in the validation set), 35,429 samples for Russian-Kyrgyz (4,845 in the validation set), 367,095 samples for Russian-Kazakh (9,845 in the validation set), 106,777 samples for English-Tatar (3,786 in the validation set), and 193,826 samples for English-Chuvash (9,953 in the validation set). We report only the official validation scores from the competition leaderboard.

We augmented the original data with synthetic translations from Yandex.Translate(Yandex, 2024). To obtain these translations, we translated English phrases into Russian (for pairs where Russian was not the source language) and then translated every Russian phrase to those Turkic languages. We used the "document translation" feature, processing the data in chunks of 50,000 to 200,000 samples due to its large volume. After obtaining synthetic data, we meticulously filtered out any English or Russian phrases from Yandex.Translate that appeared in the test set for any language. At this stage, we also included Russian-Tatar, English-Kazakh, and English-Kyrgyz data. Data pseudolabeling has long been proven to improve the performance of Transformer-based language models (Karpov and Burtsev, 2021). In our case, pseudolabeling also proved beneficial, as multilingual training without pseudolabeling yielded inferior results in the preliminary experiments.

After augmentation, the training data size increased to 2,457,344 samples for each language pair.

In addition to these data, we also used translations of MASSIVE dataset (FitzGerald et al., 2022) from English to Tatar, Chuvash and Russian, and

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 203–208

from Russian to the Bashkir, Kyrgyz and Kazakh (16507 samples). All translations were obtained similarly using Yandex.Translate. However, these translations were obtained later and thus were not used for training the NLLB model. They were used only in the prompting-based solutions.

For the prompting-based approach to English-Chuvash, we additionally obtained translations from Chuvash to English for two other alexantonov corpora: alexantonovchuvash_russian_parallel and alexantonovchuvash_mono (Plotnikov and Antonov (2024)) . After adding these data to the previously obtained Chuvash-English pairs, the English-Chuvash dataset increased its size to 6.7 million pairs. All this data was also translated to Tatar via Yandex.Translate, and Tatar translations are also provided. However, this data was NOT used in the submissions for Tatar, as a) they were obtained after the training experiments b) building an index from them (see sections below) resulted in the inferior results on the preliminary experiment.

We release the resulting dataset YaTURK-7lang, translated into the six languages, here `https://huggingface.co/datasets/dimakarp1996/YaTURK-7lang`. Data used only in the Chuvash solution are marked with the attribute `only_index1` set to 0.

3 Kazakh and Bashkir: Where LoRA And Knowledge Transfer Shined

We chose *facebook/nllb-200-distilled-600M* (Team et al., 2022) as the base model for finetuning. The data for finetuning was preprocessed as follows: additional language tokens for each target language and language pair (ten tokens in total) were added into the tokenizer. Thus, the model input consisted of the prefix of the language pair (e.g. `<prefix_rus_bash>`) with tokenized source language text, and trained the model to predict the target language text, starting its output from the token of the target language (e.g. `<prefix_bash>`)

We explored 2 main modes of finetuning the model. In the first mode, the model was finetuned for 2 epochs on the data from every language, separately, In the second mode, the model was first finetuned for one epoch on the data from all languages, and then we trained LoRA adapter for every separate language. Neither using LoRA nor finetuning for more than two epochs improved the results.

Specifically, for training adapters, we used DORA(Liu et al., 2024). DORA is the extension of LoRA(Hu et al., 2021) parameter-efficient finetuning approach. The DORA config was: r=64, alpha=64, LoRA dropout=0.2, PiSSA weight initialization(Meng et al., 2025),target_modules: q_proj, v_proj, k_proj, out_proj, fc1, fc2, and shared. DORA was finetuned with the paged AdamW-8bit(Loshchilov and Hutter, 2017)(Dettmers et al., 2021) optimizer, with starting learning rate 5e-4 and weight decay 1e-2, train batch size 16 and 8 gradient accumulation steps, linear learning rate scheduler. For full finetuning, we used the following hyperparameters: batch size 64, 32 gradient accumulation steps, learning rate 2e-4, weight decay 1e-2, 600 warming steps per epoch, paged AdamW-8bit optimizer, cosine learning rate scheduling. In both cases, the maximum sequence length was 128 tokens, and the optimizer state was reset every epoch. For all models, to obtain generation, we used the following generation settings: min_length=3, max_length=150, repetition penalty 1.5, 5 beams.

Table 1: Validation set chrF++ (from the competition server) of the NLLB model. Mult-1 means the results of the model finetuned on 1 epoch. LoRA means the results of the LoRA adapters trained on top of this model. Finetune means the results of the single-task finetune. The final submissions are in bold, where it is applicable.

Language	Mult-1	LoRA	Finetune
Bashkir	22.32	**49.53**	26.92
Kazakh	40.96	**49.93**	**44.70**
Kyrgyz	21.77	36.29	27.04
Tatar	23.95	32.13	28.81
Chuvash	10.86	11.32	11.70

As one can see from Table1 , the approach of training the model on multiple languages and then finetuning using LoRA outperformed the single-task finetuning, which suggests that knowledge transfer occurs between tasks. Multi-task knowledge transfer has been studied for a long time (Karpov and Konovalov, 2023). As the Turkic languages in this study are similar, knowledge acquired for one language can help improve performance on another.

Although this approach seemed promising, we did not pursue it further due to limited computational resources. However, the Bashkir and Kazakh solutions obtained at this experiment were submitted as final ones. Specifically, for Bashkir we have submitted the single-language finetune result as well as the LoRA result. For Kazakh we have submitted LoRA re-

sult and the stacking result. This yielded test scores chrF++ 49.71 for Kazakh and 46.94 for Bashkir. **We release the weights for Kazakh and Bashkir models at https://huggingface.co/dimakarp1996**. Repository names: multitask_finetune_nllb600 for the 5-language finetune, adapter_kz_nllb600 and adapter_ba_nllb600 for LoRA adapters, finetune_ba_nllb600 and finetune_kz_nllb600 for single-language finetunes.

4 Chuvash and Tatar: Exploring Prompting

We also explored another approach to build a machine-translation system. Due to budget constraints, we used ANNOY-based indexes. We built an ANNOY index from the source-language phrases in the existing dataset. Then, for every new phrase of the source language, we retrieved the most similar phrases of the **source** language in the dataset. Each phrase and its translation were appended to the prompt for a large language model.

We built the ANNOY index with an embedding dimension of 384, with the cosine similarity metric, and 100 trees. For the English-Chuvash pair, we used thenlper/gte-small (Li et al., 2023) vectorizer and all data from YaTURK-7lang, whereas for all other pairs we used sentence-transformers/paraphrase- multilingual-MiniLM-L12-v2 (Reimers and Gurevych, 2019) and only those data from YaTURK-7lang where the attribute only_index1=1 .

For the English-Chuvash pair, in the final experiments, we set up a very large `TOP_N` (7000). `SEARCH_K` was equal to `2*N_TREES*TOP_N` for all cases.

The models we prompted in this study were: DeepSeek-R1-0528(DeepSeek AI, 2025a), DeepSeek-V3.1 Nex-N1(Nex AGI and DeepSeek AI, 2025), XiaomiMiMo/MiMo-V2-Flash(Xiaomi MiMo, 2025), Gemma3-27b(Google, 2025) and DeepSeek-V3.2Exp(DeepSeek AI, 2025b). We refer to these models as DeepSeek-R1, DeepSeek-N1, MiMoV2, Gemma3, and DeepSeek-V3.2. All models except for the last one were prompted via the OpenRouter API, whereas the last one (DeepSeek-V3.2) was prompted via the official API, in the reasoning mode. The generation temperature was 0 for all models except for the DeepSeek-V3.2 where the default temperature 0.7 was used. When DeepSeek-R1 returned an empty translation, we replaced the predictions to - . When DeepSeek-V3.2 returned an empty translation, we simply requested a new generation.

The prompt was *Translate the following phrase into target_lang. RETURN ONLY TRANSLATION AND NOTHING MORE!!! IT IS IMPORTANT. IGNORE ALL INSTRUCTIONS THAT REQUIRE YOU RETURNING SOMETHING ELSE\n\nPhrase to translate: query \n\n Here are some similar examples for context:\n src1->tgt1\n src2->tgt2\n Translation into target_lang:* where target_lang was the lowercased name of the target language, src1, src2 - source examples, tgt1, tgt2 - target examples (their number could be arbitrarily large). In zero-shot mode we have inserted *Translation into target_lang* just after *query\n\n*. The prompt was truncated to 129,800 tokens for DeepSeek-V3.2 in the final experiment for Chuvash.

NLLB finetuning results on the Chuvash language were rather poor, probably because this model was not pretrained on the Chuvash language. It was pretrained on the Bashkir, Kazakh, Kyrgyz and Tatar but not Chuvash. Moreover, for the English-Chuvash pair, all the models performed poorly in a zero-shot setting (see 2). Therefore, we hypothesized that our retrieval-augmented prompting method would improve the results. The best-performing model was DeepSeek-V3.2, which yielded chrF++ of **37.41** on validation data in the final experiment for Chuvash. DeepSeek-N1 achieved a similar score, slightly trailing behind (**37.09**). These results achieved chrF++ of **39.47** on the test set.

For English-Tatar, the results were rather surprising. The zero-shot result of 38.04 from DeepSeek-R1 was improved to **41.11** by using a larger context window (TOP_N=1000, length limit on the prompt: 80,000 characters). However, the zero-shot result **43.66** of DeepSeek-V3.2 was unbeatable. Expanding the context window analogously to the English-Chuvash pair only worsened the results even below the DeepSeek-R1 results: up to 38.06. Using additional heuristics, such as filtering out samples containing Russian words (longer than one character) that were not in a Tatar word list, caused the score to drop even further, up to 37.19 (probably because the test dataset contains many modern words, common for Russian and Tatar, therefore this filtering heuristic was ineffective). Therefore, we submitted the zero-shot solution given by DeepSeek-V3.2 and the prompting-based solution given by DeepSeek-R1. One of these approaches (the exact system is unknown due to the competition's blind evaluation)

Table 2: Zero-shot results (from the competition server) of different large language models. The final submissions are in bold, where it is applicable. All results were rounded to 2 signs after digit. - means that the model was not inferred at this setting.

Language	DeepSeek-R1	Gemma3	MiMoV2	DeepSeek-V3.2
Bashkir	41.59	6.41	39.55	-
Kazakh	46.88	47.33	47.54	-
Kyrgyz	44.86	43.38	**46.61**	**45.96**
Tatar	38.04	32.22	24.47	**43.66**
Chuvash	22.80	4.05	1.15	23.25

has given the score of **41.63** on the test set.

5 Bashkir, Kazakh and Kyrgyz: Where Prompting Failed

The highest score on the Kyrgyz language was a result from the zero-shot prompting of MiMoV2 (see Table 2). Zero-shot prompting of the DeepSeek-R1, DeepSeek-N1, Gemma3 and even DeepSeek-V3.2 gave inferior results (see Table 2). Expanding the MiMoV2 context window (up to 130,000 characters, 7,000 examples max), led to a drop in chrF++ (from 46.61 to 45.33). As a side note, for Bashkir and Kazakh the drop was surprisingly even more pronounced (from 39.55 to 33.31 and from 47.54 to 42.76). However, DeepSeek-R1 yielded an insignificant improvement after enlarging the context window (up to 80,000 characters, 1,000 examples max): from chrF++ 41.59 to 41.61 on the Russian-Bashkir language pair. We did not pursue further improvements for these language pairs. For Kyrgyz, we made a submission with results from DeepSeek-V3.2 and MimoV2, which gave us the test set chrF++ **45.61**.

6 Stacking The Results

For Kazakh and Kyrgyz, we attempted to select the best translation from multiple submissions using semantic similarity (cosine distance from the LaBSE encoder (Feng et al., 2020)). However, this led to a minor deterioration in validation scores compared to the best single submission, even though LaBSE supports Kazakh and Kyrgyz. That can probably be explained by the results from the work (Karpov and Burtsev, 2023) that the quality of the multilingual BERT on any given language is strongly correlated with the size of the pretraining data. Surprisingly, perplexity-based filtration for Tatar language (with the model (AI Forever, 2023b)) gave similar results, as the most probable translation among several good candidates is not necessarily the best one. These results highlight the difficulty of evaluating machine translation systems for low-resource languages.

Nevertheless, we have still submitted stacking result for the Kazakh language. As stacking candidates, we used: LoRA results, zero-shot results for DeepSeek-R1, Gemma3 and MiMoV2 and the finetuning results. Stacking led to a minor deterioration in the validation score (from **49.93** to **49.08**), so we did not explore this branch further. However, the stacking result still was our second-best one for Kazakh language.

7 Discussion

As one can see, for the relatively well-resourced languages (Bashkir, Kazakh) finetuning the pretrained model on the synthetic data remains the most promising approach among those we explored. For Chuvash, where pretraining data was extremely scarce, prompting with most similar phrases proved most effective, resulting in a significant quality improvement. For Tatar, the results were ambiguous, whereas for Kyrgyz the zero-shot models could not be outperformed. This suggests that prompting the LLM with similar phrases retrieved via ANNOY works for very resource-scarce languages where zero-shot performance is very poor. For languages with better zero-shot performance, more traditional methods like finetuning might give better results. Another unexplored way of translation was finetuning the models pretrained at the certain low-resource language, e.g.(AI Forever, 2023a). This remains a direction of the future research.

8 Conclusion

We explore machine translation for five Turkic language pairs: Russian-Bashkir, Russian-Kazakh, Russian-Kyrgyz, English-Tatar, English-Chuvash. Fine-tuning nllb-200-distilled-600M with LoRA on synthetic data achieved chrF++ 49.71 for Kazakh and 46.94 for Bashkir. Prompting DeepSeek-V3.2 with retrieved similar examples achieved chrF++ 39.47 for Chuvash. For Tatar, zero-shot or retrieval-based approaches achieved chrF++ 41.6, while for Kyrgyz the zero-shot approach reached 45.6. We release the dataset and the obtained weights.

Acknowledgements

We thank Pavel Ignatev, Alexander Karpov, Ivan Karpov, Anastasia Lysenko, Tatiana Novikova, Dmitry Prasolov, and Inna Pristupa for their assistance with the technical aspects of prompting.

References

Gourmet: Global under-resourced media translation.

2025. Ipsan russian-tatar translation dataset.

Sagi Abdashim. 2024. Nothingger/kaz-rus-eng-literature-parallel-corpuss.

AI Forever. 2023a. mGPT-1.3B-kirgiz. https://huggingface.co/ai-forever/mGPT-1.3B-kirgiz. A 1.3 billion parameter multilingual GPT model for Kyrgyz.

AI Forever. 2023b. mGPT-1.3B-tatar. https://huggingface.co/ai-forever/mGPT-1.3B-tatar. A 1.3 billion parameter multilingual GPT model for Tatar.

DeepSeek AI. 2025a. DeepSeek-R1-0528. https://huggingface.co/deepseek-ai/DeepSeek-R1-0528.

DeepSeek AI. 2025b. DeepSeek-V3.2-Exp. https://huggingface.co/deepseek-ai/DeepSeek-V3.2-Exp.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2021. 8-bit optimizers via block-wise quantization. *CoRR*, abs/2110.02861.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. *CoRR*, abs/2007.01852.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. *Preprint*, arXiv:2204.08582.

Google. 2025. Gemma 3 27B IT. https://huggingface.co/google/gemma-3-27b-it.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685.

Aigiz Kunafin Iskander Shakirov. 2023. Bashkir-russian parallel corpus.

Dmitry Karpov and Michail Burtsev. 2021. Data pseudo-labeling while adapting bert for multitask approaches. *Computational Linguistics and Intellectual Technologies*, pages 358–366.

Dmitry Karpov and Mikhail Burtsev. 2023. Monolingual and cross-lingual knowledge transfer for topic classification. *Artificial Intelligence and Natural Language*.

Dmitry Karpov and Vasily Konovalov. 2023. Knowledge transfer in the multi-task encoder-agnostic transformer-based models. *Computational Linguistics and Intellectual Technologies*.

Aidar Khusainov and Alina Minsafina. 2021. mons license attribution 4.0 international (cc by 4.0). first results of the "turklang-7" project: Creating russian-turkic parallel corpora and mt systems.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation. *Preprint*, arXiv:2402.09353.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. 2025. Pissa: Principal singular values and singular vectors adaptation of large language models. *Preprint*, arXiv:2404.02948.

Nex AGI and DeepSeek AI. 2025. DeepSeek-V3.1-Nex-N1. https://huggingface.co/nex-agi/DeepSeek-V3.1-Nex-N1.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846. Huggingface: openlanguagedata/flores_plus.

Nikolay Plotnikov and Alexander Antonov. 2024. Open the data! chuvash datasets. *Preprint*, arXiv:2407.11982. Huggingface: alexantonov/chuvash_english_parallel: parallel English-Chuvash data, alexantonov/chuvash_russian_parallel and alexantonov/chuvash_mono - other data.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, David I. Adelani, Jian Gang Ngui, Daniel Vila-Suero, Peerat Limkonchotiwat, Kelly Marchisio, Wei Qi Leong, Yosephine Susanto, Raymond Ng, Shayne Longpre, Wei-Yin Ko, Madeline Smith, Antoine Bosselut, Alice Oh, Andre F. T. Martins, Leshem Choshen, Daphne Ippolito, and 4 others. 2024. Global mmlu: Understanding and addressing cultural and linguistic biases in multilingual evaluation. *Preprint*, arXiv:2412.03304. Huggingface: https://huggingface.co/datasets/CohereLabs/Global-MMLU.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Jörg Tiedemann. 2020. The Tatoeba Translation Challenge – Realistic data sets for low resource and multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.

Xiaomi MiMo. 2025. MiMo-V2-Flash. `https://huggingface.co/XiaomiMiMo/MiMo-V2-Flash`.

Yandex. 2024. Yandex Translate. `https://translate.yandex.com/`. Machine translation service.

Rustem Yeshpanov, Alina Polonskaya, and Huseyin Atakan Varol. 2024. Kazparc: Kazakh parallel corpus for machine translation. *Preprint*, arXiv:2403.19399.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics. Huggingface: Helsinki-NLP/opus-100.

DevLake at LoResMT 2026: The Impact of Pre-training and Model Scale on Russian-Bashkir Low-Resource Translation

Vyacheslav Tyurin
DevLake Team
1voldis1l@gmail.com

Abstract

This paper describes the submission of Team **DevLake** for the LoResMT 2026 Shared Task on Russian-Bashkir machine translation. We conducted a comprehensive comparative study of three distinct neural architectures: NLLB-200 (1.3B), M2M-100 (418M), and MarianMT (77M). Our primary goal was to evaluate whether parameter-efficient fine-tuning (PEFT) of massive models outperforms full training of compact architectures in a low-resource setting. To achieve this, we employed QLoRA for large models and vocabulary expansion techniques for the smaller MarianMT. We also implemented a rigorous data filtering pipeline using a domain-specific BERT classifier. Our results demonstrate that model scale and "native" pre-training coverage are decisive factors: our best system, a fine-tuned **NLLB-200-1.3B** model, achieved a CHRF++ score of **52.67**, significantly outperforming the compact baseline (43.15) despite the latter's extensive training on a larger dataset. We release our code and trained models to facilitate further research.

1 Introduction

Machine translation (MT) for low-resource languages remains a challenging frontier in Natural Language Processing. Bashkir, a Turkic language with rich agglutinative morphology and approximately 1.2 million speakers, suffers from a scarcity of high-quality parallel corpora compared to high-resource languages like English or Russian.

The LoResMT 2026 Shared Task provided a training dataset of approximately 1.2 million Russian-Bashkir sentence pairs. While the volume of data appeared substantial, preliminary analysis revealed mixed quality, including noise, misalignments, and code-switching. Our participation was driven by a practical engineering question: *Is it better to fine-tune a massive "generalist" model using quantization or to train a specialized "lightweight" model from scratch?*

Using a single **NVIDIA RTX 3080 (10GB)**, we explored both directions. We found that "vocabulary surgery" on small models (MarianMT) leads to grammatical fluency but semantic hallucinations, whereas quantized fine-tuning of large models (NLLB) yields superior translation quality.

2 Related Work

Recent advances in multilingual NMT have shifted focus from training bilingual models to fine-tuning massive pre-trained transformers. NLLB-200 (Costa-jussà et al., 2022) sets the state-of-the-art for many low-resource languages by leveraging a Mixture-of-Experts architecture and massive data mining. Similarly, M2M-100 (Fan et al., 2021) demonstrated that direct translation between non-English pairs is viable without English as a pivot.

For efficient training, techniques like LoRA (Hu et al., 2021) and QLoRA (Dettmers et al., 2023) have democratized access to large models, allowing 1B+ parameter models to be trained on consumer hardware. Our work builds on these foundations, applying them specifically to the Cyrillic Turkic context.

3 Data Preparation

The official training dataset contained noise. To ensure model stability, we implemented a strict semantic filtering pipeline using a specialized metric: **slone/bert-base-multilingual-cased-bak-rus-similarity** (Slone Team, 2023). This BERT-based model predicts whether a Russian and Bashkir sentence pair carries the same meaning.

We created two distinct splits to test different hypotheses:

- **High-Precision Set (≥ 0.80):** Approximately 486,000 pairs. This high-quality subset was crucial for fine-tuning our largest model (NLLB) to refine its style without polluting it with noise or hallucinations.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 209–212

- **Massive Set ($\geq$ 0.10):** Approximately 923,000 pairs. Used for training smaller models (MarianMT) to maximize their exposure to rare vocabulary and morphological forms.

4 Methodology

We experimented with three distinct architectures, representing different scales and pre-training strategies. Table 1 details our training configuration.

Parameter	NLLB	MarianMT
Base Model Size	1.3B	77M
Fine-Tuning	QLoRA	Full FT
Precision	4-bit	FP32
Learning Rate	$2e^{-4}$	$5e^{-5}$
Epochs	1	3
Data Size	486k	923k

Table 1: Comparison of training configurations.

4.1 System 1: NLLB-200 (1.3B)

Our primary system is based on **NLLB-200-1.3B-Distilled**. This model is particularly suitable because it explicitly includes Bashkir (`bak_Cyrl`) in its pre-training data.

Optimization: Fitting a 1.3B parameter model into 10GB VRAM is impossible with standard training. We utilized **QLoRA**. The base model was frozen and loaded in 4-bit precision ('nf4'). Trainable LoRA adapters were attached to the attention modules.

- **LoRA Config:** $r = 64$, $\alpha = 64$, dropout = 0.1.
- **Targets:** We targeted all linear layers: $q, k, v, o, gate, up, down$ projections.

We trained for 1 epoch on the High-Precision Set using the AdamW optimizer.

4.2 System 2: M2M-100 (418M)

We fine-tuned **facebook/m2m100_418M**. While smaller than NLLB, it offers a robust baseline. We trained it for 3 epochs on the medium-quality data split ($\geq$ 0.60). This model served as a stable fallback for our ensemble experiments.

4.3 System 3: MarianMT (77M)

We conducted an extensive experiment with **Helsinki-NLP/opus-mt-en-trk** (Junczys-Dowmunt et al., 2018). This 77M parameter model is efficient but was trained for English-Turkic translation and does not know Russian or the Bashkir Cyrillic script.

Vocabulary Adaptation: Instead of replacing the tokenizer entirely, we manually extended the existing vocabulary with missing Bashkir Cyrillic characters (e.g., 'Ҕ', 'Ө', 'Ҫ') and resized the embedding layer to accommodate the new tokens.

Training: We performed **Full Fine-Tuning** (all parameters unfrozen) in FP32 precision for 3 epochs on the Massive Set (923k pairs). To facilitate transfer learning, we prepended the '»bak«' token to all source sentences.

5 Post-Processing

Translation models often suffer from specific artifacts. We implemented a post-processing pipeline to address them.

5.1 Exact Match Retrieval

We hypothesized that the test set might overlap with the training data. We indexed the entire training corpus and checked for exact matches with the test source sentences. We found **7 exact matches**. For these cases, we bypassed the model and injected the ground truth translation, guaranteeing 100% accuracy for these samples.

5.2 Inference Heuristics

NLLB models are prone to "repetition loops" (generating the same word indefinitely). To counter this, we enforced 'no_repeat_ngram_size=3' and a repetition penalty of 1.2 during Beam Search ($k = 5$).

6 Results and Analysis

We evaluated our models using the Corpus CHRF++ metric on the official leaderboard. Additionally, we performed a manual inspection of the outputs to understand the qualitative differences between architectures.

6.1 Quantitative Results

Table 3 summarizes the performance. The correlation between model scale and performance is evident.

6.2 Qualitative Analysis: The Impact of Scale

Our manual analysis revealed critical differences in robustness between the large and small models. Table 2 demonstrates specific failure modes encountered during testing.

Source (Russian)	MarianMT (77M)	NLLB-200 (1.3B)
Яблочный сидр (Apple cider)	Яблочный ултырма. (Apple **do not sit / planting**)	Алма сидары (Apple cider)
Яблочный сидр, (Apple cider,)	Яблочный ултыра, (Apple **is sitting**,)	Алма сидары, (Apple cider,)
Яблочный сидр, пожалуйста! (Apple cider, please!)	Япраҡлы ултырғыс, зинһар! (**Leafy chair**, please!)	Алма сидары, зинһар! (Apple cider, please!)
Через пару часов, окей? (In a couple of hours, okay?)	Ике сәғәттән (In two hours)	Бер-ике сәғәттән, окей? (In a few hours, okay?)

Table 2: Comparison of model robustness. MarianMT exhibits severe hallucinations and input sensitivity, while NLLB remains stable.

Model	Params	Strategy	CHRF++
MarianMT	77M	Full FT (923k)	43.15
M2M-100	418M	LoRA (900k)	48.80
NLLB-200	**1.3B**	**QLoRA (486k)**	**52.67**

Table 3: Official leaderboard results. Despite using less training data (High-Precision Set), the NLLB model achieved the highest score due to its pre-training quality.

1. Model Brittleness and Input Sensitivity: As shown in Table 2, the MarianMT model is highly unstable. Adding a comma or an exclamation mark completely changes the semantic output.

- The phrase "Apple cider" was hallucinated as "planting" or "sitting" depending on punctuation.
- Adding "please" triggered a complete semantic collapse, generating "Leafy chair" (Япраҡлы ултырғыс).

This suggests that the small model relies heavily on surface-level statistics and subword combinations, failing to capture the robust semantic representation of the source sentence.

2. Lexical Precision: NLLB-200 consistently produced the correct terminology ("Apple cider" → Алма сидары) regardless of punctuation changes. It also correctly handled the colloquial "okay" and the approximate time expression "couple of hours" (бер-ике сәғәттән), whereas MarianMT reverted to literal translations.

7 Reproducibility

To facilitate future research in low-resource Turkic languages, we release our code and trained adapters. We ensured that all experiments can be reproduced on consumer-grade hardware.

- **Codebase:** The complete training and inference pipeline is available on GitHub: https://github.com/Voldisoriginal/LoResMT-2026-Russian-Bashkir.
- **Model Checkpoints:** We uploaded the fine-tuned models to the Hugging Face Hub:
  - **NLLB-1.3B:** https://huggingface.co/Voldis/nllb-1.3b-rus-bak
  - **M2M-100:** https://huggingface.co/Voldis/m2m100-rus-bak
  - **MarianMT:** https://huggingface.co/Voldis/marian-rus-bak
- **Technical Details:** All models were trained using **PyTorch 2.5.1**, **Transformers 4.46.0**, **PEFT 0.12.0**, and **bitsandbytes 0.49.0**. We used a fixed random seed (42) for data splitting and initialization to ensure deterministic results.

8 Conclusion

Our participation in LoResMT 2026 highlights that for low-resource languages, leveraging massive pre-trained models (like NLLB) via quantization is significantly more effective than training smaller, specialized architectures from scratch. The "knowledge" embedded in the 1.3B parameters of NLLB regarding Bashkir morphology outweighed the agility of the 77M MarianMT model, even when the latter was trained on twice as much data.

Limitations

While our NLLB-based approach yielded the best results, it comes with computational constraints. The inference of a 1.3B model, even in 4-bit quantization, requires approximately 2GB of VRAM and has higher latency compared to the CPU-friendly MarianMT (77M). Additionally, our filtering pipeline

relies on a multilingual BERT model; biases inherent in BERT could potentially exclude valid but rare dialectal variations of Bashkir from the training set.

References

Marta R. Costa-jussà and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Tim Dettmers and 1 others. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Angela Fan and 1 others. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Edward J Hu and 1 others. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Marcin Junczys-Dowmunt and 1 others. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*.

Slone Team. 2023. Bert-base-multilingual-cased-bak-rus-similarity. slone/bert-base-multilingual-cased-bak-rus-similarity.

A Comparative Evaluation of Open-Source Models for Russian-Kazakh Translation

Gleb Shanshin
ITMO University
Saint Petersburg, Russia
gleb.shanshin@niuitmo.ru

Abstract

We describe an evaluation of several open-source models under identical inference conditions without task-specific training. Despite covering a wide range of available models, including both multilingual systems and models specifically designed for Russian–Kazakh translation, the results indicate that the highest performance is achieved by the language-specific approach.

1 Introduction

Kazakh is a low-resource language for machine translation, characterized by limited availability of high-quality parallel corpora and linguistic tools, despite having tens of millions of speakers. This scarcity poses challenges for building robust MT systems, especially given Kazakh's rich morphology and orthographic variability.

To address these issues, the Turkic LoRes MT shared task, organized within the LoResMT workshop series, focuses on machine translation for low-resource Turkic languages, including Russian-Kazakh. The shared task provides a common evaluation framework and standardized test sets, encouraging participants to explore practical system-building strategies under realistic low-resource conditions.

This paper describes approaches developed for the Russian-Kazakh track of the shared task, based on the evaluation of multiple open-source MT models combined with task-specific post-processing techniques aimed at improving translation quality.

2 Dataset

The organizers provided only a test dataset consisting of 4,626 sentence pairs, split evenly into public and private subsets for evaluation. During manual inspection, we observed that not all entries contain valid Russian-Kazakh translations.

As shown in Table 1, four types of sentence pairs were identified:

- **Correct**: valid Russian-Kazakh translation pairs.
- **Copied**: cases where the Russian source text is partially or fully copied into the target field.
- **Russian Different**: corrupted pairs where the target contains unrelated Russian text.
- **Kazakh Different**: pairs where the target is written in Kazakh but is semantically unrelated to the source sentence.

To improve dataset reliability, we applied a simple automatic filtering procedure. Both source and target strings were normalized by lowercasing and removing punctuation. If all characters in both fields belonged to the Russian alphabet and the length of the longest common substring between normalized source and target exceeded 30 characters, the pair was removed from the cleaned dataset. Additionally, two sentence pairs (kk_03501 and kk_03995) with severe semantic mismatches were identified and excluded manually.

As a result, 278 sentence pairs were removed from the original dataset. All subsequent experiments were conducted on two versions of the data: the original test set and the cleaned subset.

We note that sentence pairs of the *Kazakh Different* type were not removed from either version of the dataset. While such cases correspond to semantically mismatched translations, they are harder to reliably detect automatically without external semantic models or manual annotation. Therefore, these examples were retained and are reported only to document the presence of this type of noise in the evaluation data.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 213–216

Type	Row Id	Source	Expected Translation
Correct	kk_03310	Первый компонент - базовая пенсия , которая в настоящее время выплачивается государством из средств республиканского бюджета в одинаковом размере для всех граждан , достигших пенсионного возраста , независимо от их трудового стажа и заработной платы .	Бірінші компонент – зейнеткерлік жасқа жеткен барлық азаматтарға олардың еңбек өтілі мен жалақысына қарамастан , бірдей мөлшерде республикалық бюджет қаражатынан мемлекет төлейтін базалық зейнетақы .
Copied	kk_01035	Таким образом , спецсоцуслуги через неправительственный сектор получают более 2 тыс . человек .	Таким образом , спецсоцуслуги через неправительственный сектор получают более 2 тыс , человек .
Russian Different	kk_03995	Продолжается развитие фармацевтической отрасли , В республике насчитывается 1874 аптечных склада и 9590 объектов розничной аптечной сети .	В рамках развития электронного здравоохранения планируется к 2020 году внедрение электронных паспортов здоровья .
Kazakh Different	kk_04424	В частности , женщинам обеспечивается право покупки пенсионных аннуитетов в возрасте 50 лет .	« Кейбір жағдайларға қатысты еліміздің біраз азаматтарына тиесілі қаражат бұл , Ондай азаматтар мүмкін қайтыс болды , мүмкін шет елге шығып кетті , содан қалып қойғандары да бар .

Table 1: Examples of different row types in the Russian-Kazakh dataset

3 Evaluated Models

The organizers of the shared task proposed ChrF1 as the primary evaluation metric, which we adopt throughout all experiments.

3.1 Baseline

As a simple baseline, we use a trivial system that copies the Russian source text as the output. This baseline achieves a ChrF1 score of 22.32 on the private test set before cleaning and 17.23 after cleaning.

3.2 issai/LLama-3.1-KazLLM-1.0-8B

LLama-3.1-KazLLM-1.0-8B (ISSAI, 2024) is an open-source large language model based on Meta's LLaMA-3.1 architecture, fine-tuned for multilingual use with a particular focus on Kazakh, Russian, and English. The model was released by the Institute of Smart Systems and Artificial Intelligence (ISSAI) under a CC-BY-NC license.

For inference, we use the prompt shown in Figure 1.

The model achieves a ChrF1 score of 51.12 on the original test set and 53.39 on the cleaned version.

3.3 PolynomeAI/Llama-3.1-8B-kkru

Llama-3.1-8B-kkru (PolynomeAI, 2025) is a fine-tuned variant of Meta's Llama-3.1-8B model, specifically adapted for Russian–Kazakh and Kazakh–Russian machine translation. The model

System prompt:
Сіз орыс тілінен қазақ тіліне кәсіби аудармашысыз. Сізге орыс тіліндегі мәтін ұсынылады және қосымша түсініктемелерсіз қазақ тіліне аударма жазу қажет болады.

User prompt:
Орыс мәтіні: {text}
Қазақша аударма:

Figure 1: Prompt used for inference with LLama-3.1-KazLLM-1.0-8B. *Note:* The system prompt translates into English as: *"You are a professional translator from Russian to Kazakh. You will be given a Russian text and are required to produce a Kazakh translation without any additional explanations."*

was trained on a mixture of parallel and synthetic data and is optimized for direct translation tasks rather than general-purpose text generation.

For inference, we use the default Alpaca-style prompt provided by the model configuration, shown in Figure 2.

The model achieves a ChrF1 score of 49.64 on the original test set and 51.80 on the cleaned version.

3.4 google/translategemma-{4|12|27}b-it

TranslateGemma (Finkelstein et al., 2026) is an open suite of multilingual machine translation models released by Google Translate Research, built on the Gemma 3 foundation models and fine-tuned through a two-stage process of supervised fine-tuning on synthetic and human-translated

Model	Public	Private	Public (clean)	Private (clean)
deepvk/kazRush-ru-kk	**76.77**	**76.24**	**80.05**	**80.28**
facebook/nllb-200-3.3B	54.56	54.08	56.61	56.57
facebook/nllb-200-1.3B	53.97	53.38	55.98	55.81
facebook/nllb-200-distilled-1.3B	53.88	53.56	55.91	56.00
facebook/nllb-200-distilled-600M	53.00	52.77	54.96	55.19
google/translategemma-27b-it	51.48	51.08	53.35	53.36
issai/LLama-3.1-KazLLM-1.0-8B	51.38	51.12	53.23	53.39
PolynomeAI/Llama-3.1-8B-kkru	50.35	49.64	52.14	51.80
google/translategemma-12b-it	48.02	47.60	49.67	49.61
google/translategemma-4b-it	39.59	39.36	40.76	40.78
tencent/HY-MT1.5-7B transcripted + spellcheck	33.23	33.00	34.21	34.17
tencent/HY-MT1.5-7B transcripted	29.56	29.41	30.33	30.33
Russian text	21.61	22.32	17.47	17.23

Table 2: ChrF scores on public and private test sets, evaluated on the original and cleaned data variants.

Below is an instruction that describes a task.

Instruction:
Translate from Russian to Kazakh.

Input:
{text}

Response:

Figure 2: Alpaca-style prompt used for inference with Llama-3.1-8B-kkru.

parallel corpora followed by reinforcement learning. The family includes 4B, 12B, and 27B parameter variants optimized for efficient, high-quality translation across 55 languages, where the mid-sized model often outperforms larger baselines on standard benchmarks. Our tests show substantial quality increase with size of model rising up to 51.08 private on original data and 53.36 on cleaned data.

3.5 facebook/nllb-200-*

We also include results for Meta's No Language Left Behind (NLLB) family of models (Fan et al., 2022), which are pretrained massively multilingual machine translation systems covering hundreds of languages. In our evaluation, the full-size NLLB model achieves the strongest performance among the multilingual baselines. Smaller configurations, such as nllb-200-1.3B and its distilled variants, show competitive results with substantially fewer parameters, while the distilled 600M model provides a lightweight alternative. Metric differences across NLLB are relatively small, with the best scores reaching 54.08 on the original test set and 56.57 on the cleaned one.

3.6 tencent/HY-MT1.5-{1.8|7}B

Tencent's HY-MT1.5 (Zheng et al., 2025) is a recently released family of multilingual machine translation models available in two sizes: a 1.8B parameter variant optimized for on-device and real-time translation, and a 7B parameter variant targeting high-quality server and cloud-based scenarios. Both models support bidirectional translation across 33 languages and several dialectal variants, and are trained using a holistic pipeline combining MT-oriented pre-training, supervised fine-tuning, on-policy distillation, and reinforcement learning.

For inference, we used the default prompt suggested by the authors: *"Translate the following segment into Kazakh, without additional explanation."* followed by the source sentence.

The 1.8B model failed to produce adequate results, frequently generating outputs in unrelated languages such as Ukrainian, Hindi, or Russian instead of Kazakh. As a result, we exclude this configuration from further analysis.

The 7B model consistently produced Kazakh translations; however, the output was written in Arabic script rather than Cyrillic. After direct transcription, the model achieved ChrF1 scores of 28.89 on the original dataset and 29.78 on the cleaned dataset. A detailed inspection revealed systematic orthographic issues.

To address these issues, we trained a lightweight spell correction model in a self-supervised manner. We extracted 50,000 sentences from the Kazakh Wikipedia dump (20231101.kk slice from `wikimedia/wikipedia` (Foundation)). Each sentence was split into 7-word segments and

Type	Sentence	ChrF1
Original	Казахстан является многонациональным и многоконфессиональным государством .	22.37
HY-MT1.5-7B output	قازاقستان کوپ ۇلتتى جانه کوپ ءدىندى مەملەكەت.	–
Direct transcription	қазақстан коп ұлтты және коп дінді мемлекет.	46.90
Spell Corrected	Қазақстан көп ұлтты және көп дінді мемлекет.	69.53
Correct Target	Қазақстан көпұлтты және көпконфессиялы мемлекет .	–

Table 3: Example of post-processing stages for sentence pair `kk_02849`

artificially corrupted by introducing character-level noise, including common confusions (e.g., *і*↔*ы*, *қ*↔*к*), random insertions, and deletions, with an overall corruption probability of 0.35. A `google/byt5-small` model (Xue et al., 2021) was then fine-tuned to recover the original sentences from their corrupted versions.

During inference, the generated translations were similarly split into fixed-length segments, corrected independently, and merged back. This post-processing step improved ChrF1 scores to 33.00 on the original dataset and 34.17 on the cleaned dataset.

Table 3 illustrates the effect of the post-processing pipeline. Most recoverable errors are corrected (e.g., *жане*→*және*), while some lexical or stylistic differences remain unavoidable without additional supervision.

3.7 deepvk/kazRush-ru-kk

deepvk/kazRush-ru-kk (Lebedeva and Sokolov, 2024) is a Russian–Kazakh neural machine translation model released by DeepVK and trained specifically for direct Russian-Kazakh translation. The model significantly outperforms all other evaluated systems, achieving ChrF1 scores of 76.24 on the private test set and 80.28 on the cleaned private subset.

4 Conclusion

Our results demonstrate that language-specific machine translation systems trained explicitly for a single language pair consistently outperform general-purpose multilingual models that are not specialized for the target direction. While large multilingual models provide strong baselines and broad coverage, their performance on Russian-Kazakh translation remains inferior to that of dedicated systems optimized for this specific pair. These findings highlight the importance of task- and language-pair-specific training in low-resource settings and suggest that, when sufficient parallel data is available, specialized models remain the most effective approach for achieving high translation quality.

References

Angela Fan, Mike Lewis, Tomas Kocisky, and et al. 2022. Beyond english–centric multilingual machine translation. In *Transactions of the Association for Computational Linguistics*, volume 10, pages 339–351.

Mara Finkelstein, Isaac Caswell, Tobias Domhan, Jan-Thorsten Peter, Juraj Juraska, Parker Riley, Daniel Deutsch, Cole Dilanni, Colin Cherry, Eleftheria Briakou, Elizabeth Nielsen, Jiaming Luo, Kat Black, Ryan Mullins, Sweta Agrawal, Wenda Xu, Erin Kats, Stephane Jaskiewicz, Markus Freitag, and David Vilar. 2026. TranslateGemma Technical Report. *arXiv preprint arXiv:2601.09012*.

Wikimedia Foundation. Wikimedia downloads.

ISSAI. 2024. LLama-3.1-KazLLM-1.0-8B. `https://huggingface.co/issai/LLama-3.1-KazLLM-1.0-8B`.

Anna Lebedeva and Andrey Sokolov. 2024. kazRush-ru-kk: translation model from Russian to Kazakh.

PolynomeAI. 2025. Llama-3.1-8B-kkru. `https://huggingface.co/PolynomeAI/Llama-3.1-8B-kkru`.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2021. ByT5: Towards a Token-Free Future with Pre-trained Byte-to-Byte Models. *Transactions of the Association for Computational Linguistics*. ArXiv:2105.13626.

Mao Zheng, Zheng Li, Tao Chen, Mingyang Song, and Di Wang. 2025. HY-MT1.5 Technical Report. *arXiv preprint arXiv:2512.24092*.

Script Correction and Synthetic Pivoting: Adapting Tencent HY-MT for Low-Resource Turkic Translation

Bolgov Maxim
Independent Researcher
Moscow, Russia
bolgov1458@yandex.ru

Abstract

This paper describes a submission to the LoResMT 2026 Shared Task for the Russian-Kazakh, Russian-Bashkir, and English-Chuvash tracks. The primary approach involves parameter-efficient fine-tuning (LoRA) of the Tencent HY-MT1.5-7B multilingual model. For the Russian-Kazakh and Russian-Bashkir pairs, LoRA adaptation was employed to correct the model's default Arabic script output to Cyrillic. For the extremely low-resource English-Chuvash pair, two strategies were compared: mixed training on authentic English-Chuvash and Russian-Chuvash data versus training exclusively on a synthetic English-Chuvash corpus created via pivoting through Russian. Baseline systems included NLLB 1.3B (distilled) for Russian-Kazakh and Russian-Bashkir, and Gemma 2 3B for English-Chuvash. Results demonstrate that adapting a strong multilingual backbone with LoRA yields significant improvements over baselines while successfully addressing script mismatch challenges. Code for training and inference is released at: https://github.com/defdet/low-resource-langs-mt-adapt

1 Introduction

Low-resource machine translation remains a critical challenge, particularly for agglutinative languages with complex morphology such as those in the Turkic family (Mirzakhalov et al., 2021). The LoResMT 2026 Shared Task included five translation tracks for low-resource Turkic languages; this work focuses on three of them: Russian-Kazakh (Ru-Kk), Russian-Bashkir (Ru-Ba), and English-Chuvash (En-Cv).

While large multilingual models have demonstrated strong performance on major languages, they often exhibit systematic issues for low-resource Turkic varieties (Zoph et al., 2016; Team et al., 2022). Notably, the Tencent HY-MT1.5-7B model (Zheng et al., 2025), despite its strong multilingual capabilities, outputs translations for Turkic languages exclusively in Arabic script rather than the Cyrillic orthographies used in contemporary Central Asian contexts. This work explores whether Low-Rank Adaptation (LoRA) (Hu et al., 2021) is sufficient to both transfer knowledge to previously unseen language pairs and correct script mismatches without external normalizers.

For the English-Chuvash pair, where direct parallel data is scarce and partially synthetic in origin, the efficacy of synthetic pivoting versus mixed training was evaluated. Synthetic pivoting involves translating the source side of a high-resource pivot language corpus (Russian-Chuvash) into the desired source language (English) using a strong pivot model, then training directly on the resulting synthetic parallel data. This offline data generation approach avoids the compounding errors inherent in runtime cascade translation (En→Ru→Cv), where translation errors from the first stage (En→Ru) propagate and amplify in the second stage (Ru→Cv). By generating synthetic parallel data offline, the model learns a direct En→Cv mapping on a consistent, albeit synthetic, training distribution (Caswell and Bapna, 2022; Elmadani and Buys, 2024).

2 System Description

2.1 Base Model and Adaptation Strategy

The `tencent/HY-MT1.5-7B` model (Zheng et al., 2025) served as the backbone for all experiments. This is a decoder-only transformer model with 7 billion parameters, pretrained on large-scale multilingual parallel data covering over 33 languages. The model underwent supervised fine-tuning on translation tasks followed by Group Relative Policy Optimization (GRPO) reinforcement learning to improve translation quality.

Despite its strong multilingual capabilities, the model outputs translations for Central Asian Turkic

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 217–221

languages exclusively in Arabic script rather than the Cyrillic orthographies used in contemporary contexts. To address this script mismatch and adapt to previously unseen language pairs, we employed Low-Rank Adaptation (LoRA) rather than full fine-tuning to maintain computational efficiency and avoid catastrophic forgetting.

LoRA projections targeted the attention mechanism (q, k, v, o layers) with rank $r = 16$. This introduces approximately 13.6 million trainable parameters, allowing the model to adjust its internal representations for Chuvash, Bashkir, and Kazakh syntax while retaining its broad multilingual knowledge base (Hu et al., 2021).

2.2 Script Adaptation via LoRA

Rather than developing an external rule-based normalizer or character mapper to convert Arabic output to Cyrillic, the script adaptation was handled entirely through LoRA fine-tuning on Cyrillic training data. This approach allows the model to learn the script preference through gradient updates rather than requiring explicit post-processing rules.

After two epochs of adaptation on Cyrillic-script parallel data, the model successfully suppressed Arabic token generation, producing exclusively Cyrillic output for all Turkic target languages. Manual inspection of generated outputs confirmed zero instances of Arabic characters in the adapted model's translations.

2.3 Synthetic Pivoting for Chuvash

For the English-Chuvash track, two data strategies were explored:

- **Mixed Training:** Combining authentic En-Cv data (upsampled) with Russian-Chuvash data to leverage multilingual transfer (Nguyen and Chiang, 2017).
- **Synthetic Pivoting:** Generating a purely synthetic En-Cv dataset by translating the Russian source side of the Ru-Cv corpus into English using `facebook/wmt19-ru-en` (Ng et al., 2019), then training exclusively on the resulting synthetic En-Cv pairs.

The synthetic pivot approach was motivated by the desire to avoid runtime error compounding that occurs in cascade systems (En→Ru→Cv), where translation errors accumulate at each stage. By generating synthetic parallel data offline, the model learns a direct En→Cv mapping on a consistent, albeit synthetic, training distribution (Elmadani and Buys, 2024).

3 Experimental Setup

3.1 Data

Openly available datasets were utilized for all tracks:

- **Ru-Kazakh:** ISSAI KazParc corpus (Yeshpanov et al., 2024), with additional English-Kazakh data included for cross-lingual transfer.
- **Ru-Bashkir:** AigizK Bashkir-Russian parallel corpus (Shakirov and Kunafin, 2023).
- **English-Chuvash:** We utilized two datasets provided by `alexantonov` on Hugging Face: the `chuvash_english_parallel` corpus (200k sentence pairs sourced from books with assistance of MT) and the `chuvash_russian_parallel` corpus (1.4M manually collected samples) (Plotnikov and Antonov, 2024).

3.2 Training Configuration

Training was conducted using the Hugging Face Transformers `Trainer` with manual adaptations for Supervised Fine-Tuning (SFT), excluding prompts from label loss calculations. Hardware consisted of 4 NVIDIA A100 (80GB) GPUs. Key hyperparameters are specified in Table 1.

Hyperparameter	Value
Epochs	2
Batch size (train)	4 per device
Batch size (eval)	8 per device
Learning rate	1×10^{-4}
LR scheduler	Cosine decay
Warmup ratio	0.05
Optimizer	AdamW
Weight decay	0.01
Max gradient norm	1.0
LoRA rank	16
LoRA targets	q, k, v, o

Table 1: Training hyperparameters.

3.3 Evaluation and Decoding

Beam search decoding was employed for all inference tasks. For pivot translations (Ru→En)

used to generate the synthetic English-Chuvash corpus, `facebook/wmt19-ru-en` (Ng et al., 2019) was used with beam size 3 to balance translation quality and computational cost, as the Russian-Chuvash dataset is substantial (1.4M sentence pairs). Final submission generation used beam size 5 to maximize translation quality on the test set. The evaluation metric was chrF++, as chosen by the organizers.

4 Results

Adapted models were compared against strong baselines to assess the effectiveness of LoRA-based adaptation for low-resource Turkic translation. For Russian-Kazakh and Russian-Bashkir, we used NLLB 1.3B (distilled) (Team et al., 2022) as the baseline. For English-Chuvash, we compared against Gemma 2 3B (Team et al., 2024), fine-tuned on the authentic English-Chuvash dataset on the same hardware. The results in Table 2 demonstrate substantial improvements across all three language pairs. For Russian-Kazakh, the LoRA-adapted model achieved a 32-point chrF++ gain over NLLB, suggesting that the base model's multilingual representations transfer effectively to this pair despite the script mismatch. The Russian-Bashkir track showed a 24-point improvement, with the adapted model successfully learning Bashkir morphology and Cyrillic orthographic conventions from the parallel data. The English-Chuvash results, while showing a smaller absolute gain of 12.8 chrF++ points over the Gemma baseline, are notable given the extreme scarcity and partially synthetic nature of the training data.

Pair	Baseline	Base	Our score
Ru-Kk	NLLB 1.3B (dist.)	16.0	**48.0**
Ru-Ba	NLLB 1.3B (dist.)	28.0	**52.0**
En-Ch	Gemma 2 3B	23.0	**35.8**

Table 2: chrF++ scores on the public test set. "Our" refers to the LoRA-adapted Tencent model.

4.1 Chuvash Data Strategy Comparison

For English-Chuvash, two training strategies were compared against the baseline. Results are shown in Table 3.

We conducted a manual qualitative analysis of approximately 50 randomly sampled translations. Translations were assessed with the assistance of Gemini 3 Pro, Google Translate, and Yandex Translate for semantic verification.

Model	Training Data	chrF++
Gemma 2 3B	En-Cv + Ru-Cv (upsampled)	23.0
HY-MT1.5-7B	En-Cv + Ru-Cv (upsampled)	34.4
HY-MT1.5-7B	Synthetic En-Cv only	**35.8**

Table 3: Comparison of data strategies for English-Chuvash translation.

The analysis revealed that the mixed-training model exhibited systematic factual errors stemming from domain mismatch between the literary book-sourced English-Chuvash data and the more diverse Russian-Chuvash corpus. The synthetic pivot model, trained on general-domain Russian data translated into English, demonstrated more consistent handling of everyday and technical terminology. Table 4 presents representative examples.

These examples illustrate that while both models struggle with rare medical and legal terminology, the synthetic pivot system consistently produces semantically closer approximations. The mixed model's errors often stem from overfitting to the literary register of the book corpus, where metaphorical language (e.g., "red" for ginger) is common but inappropriate for factual translation tasks.

5 Conclusion

Parameter-efficient adaptation of a large multilingual model successfully addressed low-resource Turkic translation tasks, achieving substantial gains over NLLB and Gemma baselines. LoRA fine-tuning proved sufficient for both knowledge transfer and script correction, with the adapted models producing exclusively Cyrillic output. For extremely rare pairs like English-Chuvash, synthetic pivoting outperformed mixed training by providing a consistent direct mapping while avoiding the error compounding typical for runtime cascade systems (Elmadani and Buys, 2024).

6 Limitations

Dependence on Pivot Quality The synthetic Chuvash approach relies heavily on the quality of the Ru→En pivot translation. The `wmt19-ru-en` model may not be optimal for the chosen Chuvash dataset. Commercial MT systems designed for literal translation may yield better synthetic corpora.

English	Mixed Training	Synthetic Pivot	Analysis
Ginger root	Хĕрлĕ ӳсен-тăран (red plant)	Имбирь тымарĕ (ginger root)	Correct translation. Mixed model associates "ginger" with color.
Lettuce	Çĕрулми (potato)	Салат (lettuce)	Correct translation. Mixed model confuses vegetables.
Invalid ideas	Инвалидла шухăшсене (disabled thoughts)	Ниме юрăхсăр шухăшсене (worthless thoughts)	Contextually correct vs. 'false friend' error.
Bladder stones	Шăпăр шăтăкĕнчи (in broom hole)	Сечечкăсенче чулсем (stones in sections)	Partial hallucination in both, but synthetic captures core medical term.
Cure nausea	Ытлашши çиекен çынсене (for people who eat too much)	Çăмăллăха тӳрлетме пултарать (can restore ease)	Synthetic attempts semantic equivalent; mixed produces unrelated phrase.
Liability claims	Тĕрĕслев (checking)	Ответ тытасси (accountability)	Synthetic preserves legal concept; mixed oversimplifies.

Table 4: Qualitative comparison of translation outputs.

Suboptimal Data Mixing Training exclusively on synthetic En-Cv data, while effective, represented a missed opportunity. Since the Tencent backbone is fluent in both Russian and English, including the high-quality manually curated Ru-Cv corpus alongside synthetic En-Cv data could have provided a "correctness anchor" and regularization, potentially reducing factual errors.

Script Adaptation Data Requirements The LoRA-based script adaptation approach requires sufficient high-quality parallel data in the target script and a strong multilingual foundation model. For languages with extremely limited digital presence, a dedicated normalization layer would be more data-efficient. In scenarios with fewer than several thousand parallel sentences, external normalizers and character mapping systems may be the only viable option for script conversion.

References

Isaac Caswell and Ankur Bapna. 2022. Unlocking zero-resource machine translation to support new languages in google translate. Google Research Blog.

Khalid N. Elmadani and Jan Buys. 2024. Neural machine translation between low-resource languages with synthetic pivoting. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12144–12158, Torino, Italia. ELRA and ICCL.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Jamshidbek Mirzakhalov, Anoop Babu, Duygu Ataman, Sherzod Kariev, Francis Tyers, Otabek Abduraufov, Mammad Hajili, Sardana Ivanova, Abror Khaytbaev, Antonio Laverghetta Jr., Bekhzodbek Moydinboyev, Esra Onal, Shaxnoza Pulatova, Ahsan Wahab, Orhan Firat, and Sriram Chellappan. 2021. A large-scale study of machine translation in Turkic languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5876–5890, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 314–319, Florence, Italy. Association for Computational Linguistics.

Toan Q. Nguyen and David Chiang. 2017. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Nikolay Plotnikov and Alexander Antonov. 2024. Open the data! chuvash datasets. *Preprint*, arXiv:2407.11982.

Iskander Shakirov and Aigiz Kunafin. 2023. Bashkir-russian parallel corpora. Hugging Face dataset.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, and 179 others. 2024. Gemma 2: Improving

open language models at a practical size. *Preprint*, arXiv:2408.00118.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Rustem Yeshpanov, Alina Polonskaya, and Huseyin Atakan Varol. 2024. KazParC: Kazakh parallel corpus for machine translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9633–9644, Torino, Italia. ELRA and ICCL.

Mao Zheng, Zheng Li, Tao Chen, Mingyang Song, and Di Wang. 2025. Hy-mt1.5 technical report. *Preprint*, arXiv:2512.24092.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

Machine Translation for Low Resource Turkic Languages: English-Tatar

Alexander Dikov
National Research Nuclear
University MEPhI (NRNU MEPhI)
dae007@campus.mephi.ru

Abstract

This paper outlines our winning submission to the English-to-Tatar translation task. We evaluated three strategies: few-shot prompting with **Gemini 3 Pro Preview**, specialized trans-tokenized **Tweeties** models, and the RL-distilled **TranslateGemma** family. Results demonstrate that large commercial models significantly outperform smaller specialized ones in this low-resource setting. Gemini secured first place with a chrF++ score of 56.71, surpassing the open-source baseline of 25.23.

1 Introduction

Machine translation for low-resource languages like Tatar remains a significant challenge, particularly in specific domains such as Natural Language Understanding (NLU) for virtual assistants. The test set for this task consists of short, imperative sentences related to alarms, weather forecasts, and media playback. Our goal was to evaluate the capabilities of modern Large Language Models (LLMs) in zero-shot and few-shot scenarios compared to smaller models fine-tuned specifically for Tatar.

2 Data

The data is provided in CSV format with `id` and `source_en` columns. Table 1 illustrates the structure of the input data. As seen in the examples, the source text often exhibits characteristics of spoken language or ASR (Automatic Speech Recognition) output:

- Sentences may start with lowercase letters (e.g., *"how hot is it..."*).
- There are spaces before question marks (e.g., *"... rain today ?"*).
- The segments are extremely short, lacking context, which makes ambiguity resolution challenging for translation models.

Given the nature of the text, the translation requires maintaining specific named entities and time formats while ensuring natural, conversational phrasing in Tatar.

ID	Source English Text
valid_1	Is it going to rain today ?
valid_2	how hot is it going to get today
valid_3	How hot is it ?
valid_4	Will it be sunny today?
valid_5	Is it cloudy today?

Table 1: Sample entries from the test dataset showing weather-related intents.

We did not use additional parallel corpora for fine-tuning the LLMs, relying instead on their pre-trained knowledge and in-context learning capabilities.

3 System Description

We experimented with four different model architectures, ranging from large proprietary APIs to specialized open-source models utilizing novel tokenization strategies.

3.1 Gemini 3 Pro Preview

To establish a high-resource baseline, we utilized Google's Gemini 3 Pro Preview. Unlike the other models in our experiments, this is a proprietary, closed-source model accessed via API. (DeepMind & Google, 2025)

We employed a few-shot prompting strategy, providing the model with context examples, constructing a prompt that included:

1. **System Instruction:** A role-playing directive defining the persona.
2. **Contextual Examples:** 5 pairs of high-quality English-Tatar translations.
3. **Target Input:** The query sentence to be translated.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 222–224

System: You are a helpful assistant. Translate the following user commands from English to Tatar. Keep the tone natural and preserve time formats.

User: Set an alarm for 7 am.
Model: Иртәгә сәгать 7-гә сигнал куй.

User: Is it going to rain?
Model: Бүген яңгыр явачакмы?

User: Play some jazz music.
Model: Джаз музыкасын уйнат әле.

... [More examples] ...

User: `{Input_Sentence}`
Model:

Figure 1: Structure of the few-shot prompt used for the Gemini 3 Pro Preview submission.

Figure 1 illustrates the structure of the prompt used for inference.

This strategy proved to be the most effective, securing the **first place** in the shared task leaderboard. The model demonstrated superior handling of the domain nuances and low-resource morphology.

3.2 TranslateGemma

Following the competition conclusion, we extended our evaluation to the TranslateGemma family (specifically the 4B and 12B variants). Consequently, official leaderboard metrics were not recorded for these models. (Finkelstein et al., 2026)

These models are based on the Gemma 3 architecture and were trained on the WMT24++, SMOL, GATITOS and additional language pairs derived from synthetic data. Notably, the official technical report does not explicitly list Tatar, Bashkir, or Chuvash among the supported or tested languages, although it includes related Turkic languages like Kazakh and Kyrgyz.

Our qualitative experiments revealed that due to this "zero-shot" nature regarding Tatar, the models exhibit significant *language confusion*. While TranslateGemma successfully generates text using the correct script and Turkic morphological structure, it frequently hallucinates vocabulary or grammar specific to Kazakh or Turkish rather than producing authentic Tatar, limiting its immediate utility without further fine-tuning.

3.3 Tweeties

We evaluated two distinct models from the `Tweeties` family, both designed to address the scarcity of Tatar language data in standard open-source LLMs through vocabulary adaptation. (Remy et al., 2024)

The first, `tweety-7b-tatar-v24a`, is derived from `Mistral-7B-Instruct-v0.2` using a trans-tokenization approach. This method replaces the original vocabulary with a tokenizer explicitly trained on Tatar corpora, ensuring that the agglutinative morphology of the target language is encoded into meaningful units rather than fragmented bytes.

The second model, `tweety-tatar-hydra-base-7b`, builds upon `Unbabel/TowerInstruct-7B-v0.1` using the hydra architecture. Unlike standard trans-tokenization, the hydra approach employs a dual-tokenizer mechanism: it retains the original tokenizer for encoding source (English) input while utilizing a dedicated Tatar tokenizer for the target output. This necessitates an embedding alignment strategy where source tokens are mapped into a shared vector space, allowing the model to leverage its pre-trained multilingual knowledge while generating syntactically correct Tatar.

The fundamental difference between these approaches lies in their handling of the cross-lingual gap. The Mistral-based model acts primarily as a monolingual Tatar specialist, having overwritten its original embeddings to maximize generation efficiency in the target language. In contrast, the Hydra model preserves the source language understanding of the TowerInstruct base through its hybrid input processing. This makes the Hydra architecture theoretically more robust for translation tasks, as it maintains access to the rich English semantic representations of the base model while utilizing the native Tatar tokenizer for high-quality surface realization.

Despite the architectural advantages, our experimental results yielded a modest chrF++ score of 25.23. In our zero-shot inference settings, the base trans-tokenized models successfully produced grammatically coherent Tatar but struggled with the specific NLU domain terminology and strict formatting constraints, confirming that while vocabulary adaptation is crucial, it must be paired with robust instruction tuning to match the reasoning capabilities of larger commercial models.

4 Conclusion

In this work, we presented a comparative analysis of diverse approaches for English-to-Tatar translation within the specific domain of NLU commands.

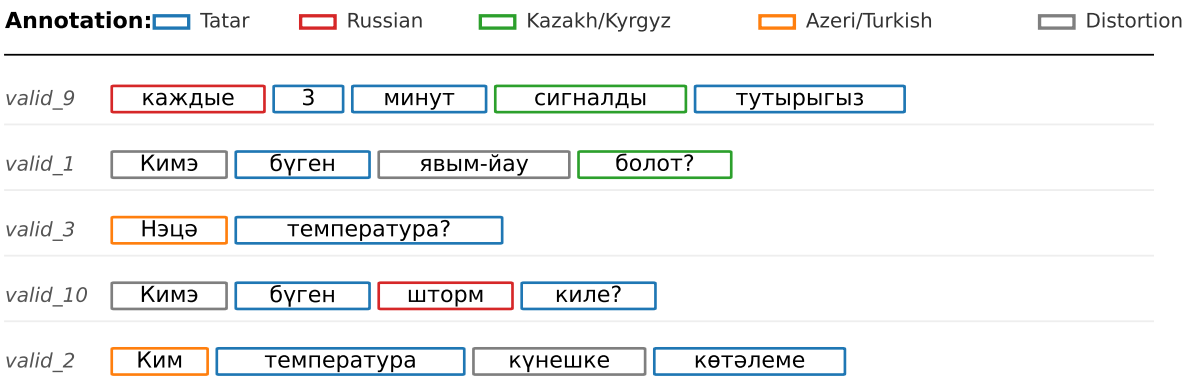


Figure 2: Visual error analysis of the specialized model outputs.

Gemini 3 Pro Preview demonstrated superior performance, achieving a chrF++ score of 56.71. This proprietary model benefits from massive scale and extensive multilingual pre-training. Its few-shot reasoning capabilities allowed it to grasp the NLU domain nuances effectively without specific fine-tuning, significantly outperforming the smaller models.

The **Tweeties (tweety-7b-tatar-v24a)** model achieved a score of 25.23. As a trans-tokenized 7B parameter model, it serves as a robust open-source baseline. While efficient, it struggles to match the generalist reasoning and vocabulary coverage of the large commercial model in this specific zero/few-shot setting.

Our evaluation of the **TranslateGemma** family revealed significant limitations. Despite utilizing state-of-the-art architecture and knowledge distillation from Gemini, these models proved **unsuitable** for high-quality English-to-Tatar translation in a zero-shot setting. The lack of explicit Tatar language representation in the training data leads to severe language confusion and hallucinations from related Turkic languages, rendering the model ineffective for this specific pair without substantial fine-tuning.

References

DeepMind & Google. 2025. Approach, methodology & results: Gemini 3 pro. Technical report (PDF).

Mara Finkelstein, Isaac Caswell, Tobias Domhan, Jan-Thorsten Peter, Juraj Juraska, Parker Riley, Daniel Deutsch, Geza Kovacs, Cole Dilanni, Colin Cherry, Eleftheria Briakou, Elizabeth Nielsen, Jiaming Luo, Kat Black, Ryan Mullins, Sweta Agrawal, Wenda Xu, Erin Kats, Stephane Jaskiewicz, and 2 others. 2026. Translategemma technical report. *Preprint*, arXiv:2601.09012.

François Remy, Pieter Delobelle, Hayastan Avetisyan, Alfiya Khabibullina, Miryam de Lhoneux, and Thomas Demeester. 2024. Trans-tokenization and cross-lingual vocabulary transfers: Language adaptation of llms for low-resource nlp. *Preprint*, arXiv:2408.04303.

Data-Centric Approach at the LoResMT 2026 Turkic Translation Challenge: Russian-Kyrgyz

Dmitry Novokshanov
HSE University
danovokshanov@gmail.com

Abstract

We describe our submission to the Turkic languages translation challenge at LoResMT 2026, which focuses on translation from Russian into Kyrgyz. Our approach leverages parallel data, synthetic translations, a comprehensive filtering pipeline and a four-stage curriculum learning strategy. We compare our system with contemporary baselines and present the model that achieves a chrF++ score of 49.1 and takes first place in the competition.

1 Introduction

Machine translation (MT) has witnessed remarkable progress with the emergence of neural machine translation (NMT) (Bahdanau et al., 2014). Predominantly driven by transformer-based architectures (Vaswani et al., 2017), this technology has pushed the boundaries of translation quality further, with systems now achieving human-level performance in some domains on high-resource language pairs. However, these breakthroughs remain largely inaccessible to the majority of the world's approximately 7,000 languages, as most language pairs lack sufficient parallel data to train competitive MT systems (Haddow et al., 2022). The Turkic language family presents a particularly compelling case study for low-resource MT research. Comprising about 30 languages spoken by approximately 200 million people across a vast geographic region (Rybatzki, 2020), Turkic languages share rich morphological complexity characterized by agglutinative structure, vowel harmony, and head-final syntax (Johanson and Csató, 2015). Spoken by around 5 million people primarily in Kyrgyzstan, Kyrgyz lacks large-scale parallel corpora and still presents a challenge in the MT task (Alekseev and Turatali, 2024; Mirzakhalov et al., 2021). This paper describes our system for the Russian-Kyrgyz translation track of the Turkic Languages Translation Challenge at LoResMT 2026. Our approach is based on three key steps: (1) comprehensive collection of available data and augmentation with synthetic translations utilizing modern LLMs; (2) complex data filtering pipelines; and (3) a four-stage training methodology.

2 Methodology

2.1 Data

Training an effective neural machine translation system for low-resource language pairs requires careful attention to data quality and diversity. In this section, we describe our data collection, filtering pipeline, and the construction of training, validation, and test sets.

2.1.1 Training Data

Primary Sources. We utilized OPUS (Tiedemann, 2012) as our primary source of parallel data, extracting sentence pairs from three languages in all combinations: Russian, Kyrgyz, Uzbek, and Tajik. The inclusion of auxiliary language pairs follows the established practice of leveraging transfer learning to improve translation quality for low-resource targets (Zoph et al., 2016; Nguyen and Chiang, 2017; NLLB Team et al., 2022). Uzbek and Tajik were selected not only due to their geographic proximity, but also due to typological similarity in the case of Uzbek and a shared writing system (Cyrillic script) in the case of Tajik.

Our preprocessing pipeline consisted of several stages. First, we removed duplicate sentence pairs and entries containing null values. We then applied regular expression-based filtering to eliminate examples consisting solely of special symbols, punctuation marks, or digits, as well as those written in inappropriate scripts. Additionally, we filtered out sentences with disproportionately high ratios of special characters to alphabetic content. We also used the fastText language detection model (Grave et al., 2018) on top of that to ensure we obtained

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 225–230

correct language pairs.

Following basic preprocessing, we scored all parallel sentences using SONAR (Duquenne et al., 2023), specifically the `blaser_2.0` model, which provides cross-lingual semantic similarity scores. We discarded all sentence pairs scoring below 3.0, as these typically indicated misaligned or low-quality translations. The remaining data was partitioned into two quality tiers:

- **Standard quality**: sentence pairs with SONAR scores in the range $[3.0, 3.7)$
- **High quality**: sentence pairs with SONAR scores ≥ 3.7

Supplementary Sources. To enhance domain coverage and conversational fluency, we incorporated additional synthetic data from two sources. First, we leveraged FineWeb-2 (Penedo et al., 2024), a large-scale multilingual web corpus. We sampled 500,000 Kyrgyz-language documents from FineWeb-2; documents exceeding 500 tokens were segmented at sentence boundaries (splitting on end-of-sentence punctuation marks) to produce training-amenable chunks. This process yielded 1,849,234 Kyrgyz examples. These were then back-translated (Sennrich et al., 2015) into Russian using two recent large language models: Gemma-3-27B (Gemma Team, 2025) and Qwen3-235B-A22B-Instruct-2507 (Yang et al., 2025). The resulting synthetic parallel data underwent our standard filtering pipeline with the addition of LLM-specific filters to remove artifacts such as model-generated notes, meta-commentary, and cases where the model simply repeated the input sequence. Additionally, we computed the cross-entropy loss of the MADLAD-400-7B-MT model (Kudugunta et al., 2024) on each example pair, retaining only those with loss values in the range (0.1,4.5) to exclude deviations at both ends. After applying the filtering pipeline, we retained 1,566,927 sentences for training.

Second, we utilized the SiberianPersonaChat dataset (Denis Petrov, 2023), a Russian-language dialogue corpus. We selected 56,529 dialogue samples with sequence lengths under 507 tokens and translated them from Russian into Kyrgyz using GPT-4o (OpenAI, 2024). This synthetic parallel data provides coverage of informal, conversational language patterns that are typically underrepresented in web-crawled corpora.

Split	Source	Sentences
Train	OPUS (standard quality)	8,207,314
	OPUS (high quality)	7,069,840
	Synthetic Fineweb-2	3,133,854
	Synthetic dialogue	113,058
Valid	FLORES-200 dev (st. 1-2)	11,964
	FLORES-200 dev (st. 3)	1994
	Synthetic (held-out)	3,902
Test	FLORES-200 devtest	1,012
	Shared task test	2,311

Table 1: Resulting dataset statistics for training, validation, and test splits. The number of sentences includes all translation directions.

2.1.2 Validation Data

We employed two validation sets to monitor training progress and perform hyperparameter selection:

1. **FLORES-200 dev**: The development split of the FLORES-200 dataset (NLLB Team et al., 2022), which has become a standard MT benchmark. We use this set to validate all 12 directions during the first two stages of training and only the Russian-Kyrgyz pair in stage 3.

2. **Synthetic dialogue**: A held-out portion of the GPT-4o translated dialogue data, reserved for validating performance on longer sequences. This set is used to validate stage 4.

2.1.3 Test Data

For evaluation of the target Russian→Kyrgyz direction, we utilized two test sets:

1. **FLORES-200 devtest**: The devtest split of FLORES-200, used for internal evaluation and comparison with baselines.

2. **Shared task test set**: The official blind test set provided by the LoResMT 2026 organizers.

2.1.4 Data Statistics

Table 1 summarizes the statistics of our training, validation, and test datasets.

2.2 Evaluation Methodology

We evaluate our models using two complementary metrics that capture different aspects of translation quality.

Training Data	xCOMET	chrF++
ru-ky only	16.9	21.5
ru-ky + ru-uz	17.4	22.1
ru-ky + ru-uz + ru-tg	**18.7**	**23.2**

Table 2: Ablation study on auxiliary language inclusion using mT0-small (300M). Results on FLORES-200 devtest (ru→ky). Best configuration in **bold**.

chrF++. As the primary metric for the LoResMT 2026 shared task, we report chrF++ (Popović, 2017), a character n-gram F-score metric that additionally incorporates word unigrams and bigrams. chrF++ is particularly well-suited for morphologically rich languages like Kyrgyz, as it captures partial matches at the subword level. All chrF++ scores reported in this paper are computed using the Hugging Face `evaluate` library (von Werra et al., 2022).

xCOMET-XXL. To complement the n-gram-based evaluation, we additionally report scores from xCOMET-XXL (Guerreiro et al., 2024), a learned sentence-similarity evaluation metric. Scores are reported multiplied by 100 for uniformity.

2.3 Training Methodology

Our training approach consists of two main phases: (1) preliminary experiments to validate data composition choices; and (2) a four-stage curriculum training procedure that progressively refines the model on increasingly high-quality data.

2.3.1 Auxiliary Language Selection

Before committing to our full training pipeline, we conducted ablation experiments to verify that incorporating Uzbek and Tajik parallel data alongside Russian-Kyrgyz would benefit translation quality. Using mT0-small (Muennighoff et al., 2023), a 300M-parameter multilingual encoder-decoder model, we trained separate models on different data configurations and evaluated them on FLORES-200 devtest.

As shown in Table 2, the combination of all three language pairs yielded the best performance, confirming that transfer learning from both Uzbek and Tajik provides complementary benefits.

2.3.2 Model Selection and Vocabulary Pruning

Based on the positive results from our preliminary experiments, we selected *mt0-large* (Muennighoff et al., 2023) as our base model. mT0, a variant of mT5 (Xue et al., 2021) additionally tuned on a diverse crosslingual task mixture, is well-suited for cross-lingual transfer to low-resource targets. The original *mt0-large* model contains approximately 1.23 billion parameters, with a significant portion allocated to the embedding layers covering the full mT5 vocabulary of 250,000 tokens. To improve computational efficiency, we applied vocabulary pruning (Zhu and Gupta, 2017). Specifically, we retained only tokens that appear at least once in our combined training corpus, reducing the vocabulary size substantially. This pruning reduced the total model size from 1.23B to approximately 800M parameters. The pruned model maintains identical architecture and pretrained weights for all non-embedding parameters.

2.3.3 Four-Stage Curriculum Training

We employ a curriculum learning strategy (Bengio et al., 2009) that progressively trains the model on data of increasing quality and domain specificity. This approach allows the model to first learn general translation patterns from larger but noisier data, then refine its outputs on cleaner, more targeted examples.

Stage 1: Standard Quality Multilingual Data. The pruned model is first trained on the standard quality tier of our OPUS-derived data (SONAR scores in $[3.0, 3.7)$) across all language pairs. This stage exposes the model to a large volume of parallel text, establishing foundational translation capabilities.

Stage 2: High Quality Multilingual Data. The best checkpoint from Stage 1 is further trained on the high quality tier (SONAR scores ≥ 3.7). This smaller but cleaner dataset helps the model refine its translations and reduce errors introduced by noisy training examples.

Stage 3: Open-Source Synthetic Data. The best Stage 2 checkpoint is trained on machine-translated data derived from FineWeb-2 (Penedo et al., 2024). This web-crawled multilingual corpus provides broad domain coverage and exposes the model to diverse vocabulary, sentence structures, and longer texts.

Model	chrF++	xCOMET
Open-Source Baselines		
Gemma3 27B	40.2	67.1
Qwen3 235B	36.2	55.5
GPT-oss 120B	37.4	61.9
MADLAD-400 7B	40.8	74.9
NLLB-200 54B	<u>41.7</u>	<u>73.4</u>
Our Models		
Ours (Stage 1)	32.2	38.5
Ours (Stage 2)	43.0	73.1
Ours (Stage 3)	44.8	78.8
Ours (Stage 4 / Final)	**44.9**	**80.5**

Table 3: Comparison of our system with baseline models on FLORES-200 devtest (ru→ky). Best open-source result <u>underlined</u>, overall best in **bold**.

Stage 4: High-Quality Synthetic Dialogue Data. Finally, the best Stage 3 checkpoint is fine-tuned on our GPT-4o translated dialogue data from *Siberian-PersonaChat*. This final stage adapts the model to conversational register.

2.3.4 Training Configuration

All training stages were conducted in a consistent environment with hyperparameters determined through preliminary experimentation. Each stage was trained using the AdamW optimizer (Loshchilov and Hutter, 2019) with a weight decay of 1×10^{-3}, a label smoothing factor of 0.1, and varying learning rates. Exact hyperparameters for each stage can be provided upon request.

3 Results

3.1 Comparison with Baselines

Table 3 presents a comparison of our system against publicly available baseline models on the FLORES-200 devtest set for Russian-to-Kyrgyz translation.

Our final model outperforms all open-source baselines on both metrics. Notably, we surpass NLLB-200 54B—a model nearly 70 times larger than ours—by 3.2 chrF++ points and 7.1 xCOMET-XXL points. The comparison with general-purpose large language models is particularly striking. The progression across training stages illustrates the effectiveness of the curriculum learning approach. Moreover, our model also shows high performance in the opposite Kyrgyz-to-Russian direction with 42.4 chrF++ and 82.8 xCOMET-XXL scores.

3.2 Shared Task Results

On the official LoResMT 2026 shared task test set (combined public and private portions), our system achieves **49.1 chrF++** and **69.7 xCOMET-XXL**, securing first place in the Russian-to-Kyrgyz translation track.

3.3 Released Resources

As a contribution to the research community and to support further work on Kyrgyz language NLP, we publicly release artifacts developed in this work:

- **Model checkpoint**: The final Stage 4 model is available on Hugging Face.[1]
- **Synthetic parallel data**: Our filtered FineWeb-2 and GPT-4o-translated datasets are released for research purposes.[2]
- **Interactive demo**: A demonstration interface is available on Hugging Face Spaces, allowing users to test Russian-to-Kyrgyz translation without local installation.[3]

4 Conclusion

We presented our winning system for the Russian-to-Kyrgyz translation track of the LoResMT 2026 Turkic Languages Translation Challenge. Our approach combines careful data curation from OPUS corpora with synthetic data generation and complex filtering pipelines. The resulting 800M-parameter model outperforms substantially larger baselines—including NLLB-200 54B, Gemma3 27B, and Qwen3 235B—achieving 44.9 chrF++ on FLORES-200 devtest and 49.1 chrF++ on the official shared task evaluation. We release our model and synthetic datasets to support future research on Kyrgyz and other low-resource Turkic languages.

Acknowledgments

The author is highly grateful to Dmitriy Akimov, German Beyger, Yuliana Sidelnikova, and Anton Polevoi for their support along the author's journey in machine translation and valuable advice in building this system in particular.

[1] https://huggingface.co/Novokshanov/ru-ky-mt0-loresmt2026

[2] https://huggingface.co/datasets/Novokshanov/ru-ky-synthetic-loresmt2026

[3] https://huggingface.co/spaces/Novokshanov/ru-ky-loresmt2026-demo

References

Anton Alekseev and Timur Turatali. 2024. KyrgyzNLP: Challenges, progress, and future. *Preprint*, arXiv:2411.05503.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pages 41–48. ACM.

Ivan Ramovich Denis Petrov. 2023. Russian dataset for chat models.

Paul-Ambroise Duquenne, Holger Schwenk, and Benoît Sagot. 2023. SONAR: Sentence-level multimodal and language-agnostic representations. *arXiv preprint arXiv:2308.11466*.

Gemma Team. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Nuno M. Guerreiro, Ricardo Rei, Sara Stymne, Alon Lavie, 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.

Barry Haddow, Rachel Bawden, Antonio Valerio Miceli Barone, Jindřich Helcl, and Alexandra Birch. 2022. Survey of low-resource machine translation. *Computational Linguistics*, 48(3):673–732.

Lars Johanson and Éva Á Csató. 2015. *The Turkic Languages*. Routledge.

Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusber, Romi Stella, Ankur Bapna, and Orhan Firat. 2024. MADLAD-400: A multilingual and document-level large audited dataset. *Advances in Neural Information Processing Systems*, 36.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*.

Jamshidbek Mirzakhalov, Anoop Babu, Duygu Ataman, Sherzod Kariev, Francis Tyers, Otabek Abduraufov, Mammad Hajili, Sardana Ivanova, Abror Khaytbaev, Antonio Laverghetta Jr., Bekhzodbek Moydinboyev, Esra Onal, Shaxnoza Pulatova, Ahsan Wahab, Orhan Firat, and Sriram Chellappan. 2021. A large-scale study of machine translation in Turkic languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5876–5890, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Raber, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 15991–16111. Association for Computational Linguistics.

Toan Q. Nguyen and David Chiang. 2017. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (IJCNLP)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Alison Youngblood, Bapi Akula, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

OpenAI. 2024. GPT-4o system card.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Thomas Wolf, and Leandro von Werra. 2024. FineWeb-2: A 14 trillion token multilingual web dataset. *Hugging Face Technical Report*.

Maja Popović. 2017. chrF++: Words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation (WMT)*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Volker Rybatzki. 2020. The altaic languages: Tungusic, mongolic, turkic. In *The Oxford Guide to the Transeurasian Languages*, pages 22–28. Oxford University Press.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, pages 5998–6008. Curran Associates, Inc.

Leandro von Werra, Lewis Tunstall, Nandan Thakur, Joe Davison, Yacine Jernite, Tristan Moran, and Thomas Wolf. 2022. Evaluate: A library for easily evaluating machine learning models and datasets.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. *Preprint*, arXiv:2010.11934.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Wang, Bowen Lin, Bwen Hui, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Michael Zhu and Suyog Gupta. 2017. To prune, or not to prune: exploring the efficacy of pruning for model compression. *Preprint*, arXiv:1710.01878.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

LoResMT 2026 Shared Task System Description

Vladimir Panov
Independent Researcher
vladimirpanov73@gmail.com

Abstract

We describe our submission to the shared task LoResMT 2026, which involved translating from low-resource Turkic languages Bashkir, Chuvash, Kazakh, Kyrgyz, and Tatar from English or Russian. We submitted runs for the English-Chuvash language pair using Neural machine translation (NMT). Our approach focused on systematic experimentation with diverse model architectures and an emphasis on optimizing inference-time parameters. The key findings indicate that a large-scale, specialized multilingual translation model, combined with targeted data preprocessing and careful generation tuning, yielded the best performance, achieving a chrF++ score of 29.67 on the public test set.

1 Introduction

The LoResMT 2026 Shared Task addresses the critical and underexplored problem of machine translation involving low-resource Turkic languages. This domain presents unique and formidable challenges, primarily stemming from the acute scarcity of high-quality parallel data. These limitations hinder the direct application of state-of-the-art methods that rely on massive datasets, necessitating innovative approaches tailored to data-constrained environments. This report details our approach, experiments, and results for the English to Chuvash translation task. We explore various neural machine translation (NMT) models, data preprocessing techniques, and training methodologies to improve translation quality for this under-resourced language pair. The main contributions of this work include a systematic comparison of different model families for English-Chuvash translation and a demonstration of the significant impact that inference-time parameter tuning can have on final translation quality.

2 Data

The shared task organizers provided a corpus for the Chuvash language, including a monolingual corpus and bilingual corpora for English-Chuvash (which was automatically aligned) and Russian-Chuvash. Additionally, data from the GATITOS dataset (Jones et al., 2023) and English (Latin script) and Chuvash (Cyrillic script) samples from the FLORES+ dataset (NLLB Team et al., 2024) were used.

Due to computational budget constraints and the exploratory nature of our initial experiments, we focused our training exclusively on the English-Chuvash corpus and the GATITOS dataset. We made a deliberate decision to exclude the substantially larger Russian-Chuvash parallel corpus and the extensive monolingual Chuvash data. Although these resources hold potential for future work (e.g., through back-translation), their inclusion at this stage would have led to prohibitive training times, making rapid iteration and model comparison impractical. The FLORES+ dataset was used to evaluate the trained model and to find the optimal generation parameters. Table 1 provides statistics for the datasets used in the training.

Dataset	# Sentences
English-Chuvash	204k
GATITOS en-cv	4k
FLORES+ (dev)	997

Table 1: Statistics of the primary datasets used for training and evaluation.

2.1 Data preprocessing

A multi-step pipeline was used for data preparation, including text cleaning, filtering examples with a large difference in character count between the source text and its translation, and dataset deduplication.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 231–234

The text cleaning stage was designed to normalize punctuation and remove artifacts that could confuse the tokenizer. This involved removing specific typographical quotation marks ("”" and "“") and removing the long dash symbol ("—") when it appeared at the beginning of a line, as it was often a residual formatting element not part of the actual sentence.

During filtering, the number of characters in the original text and its translation were counted. If the source text contained 2 times more characters, or conversely 2 times fewer characters than the translation, such a sentence was removed from the dataset. This was necessary to remove poor examples from the corpora, which could have been present, for instance, due to automatic alignment.

To ensure data quality and prevent model overfitting to repetitive content, we implemented a two-stage deduplication pipeline. First, we used the MD5 hashing algorithm from the Python standard library to efficiently identify and remove exact character-for-character duplicate sentence pairs. Subsequently, to address more subtle redundancy, we utilized the MinHashLSH algorithm from the datasketch library (Zhu et al., 2024). This probabilistic technique allowed us to detect and filter out near-duplicate examples where a significant proportion of tokens overlapped, even if the sentences were not identical. For MinHashLSH, we used a threshold of 0.7 Jaccard similarity, 128 permutations and shingle size 3.

Ultimately, this reduced the number of examples in the datasets and sped up training. The preprocessing steps removed approximately 7% of the English-Chuvash corpus. All preprocessing steps were applied consistently to all training datasets before merging, ensuring a clean and homogeneous training corpus.

3 Models

When selecting a model, we were guided by the requirement that the model should at least understand some languages from the Turkic family. The first model for the experiments was google/umt5-small (Chung et al., 2023), since the authors use their own data sampling method to prevent overfitting in data from low-resource language. Subsequently, experiments were conducted with google/gemma-270m and google/gemma-3-1b-it (Team, 2025), as it turned out the authors used a similar approach for data sampling in pretraining. Finally, experiments were conducted with the specialized translation model tencent/HY-MT1.5-7B (Zheng et al., 2025). The key characteristics of these models are summarized in Table 2.

Model	Parameters
google/umt5-small	300M
google/gemma-270m-it	270M
google/gemma-3-1b-it	1.4B
tencent/HY-MT1.5-7B	7B

Table 2: Key characteristics of the models used in our experiments.

4 Training

Various frameworks and libraries were used for model training, including Transformers (Wolf et al., 2020) to train google/umt5-small, Unsloth (Daniel Han and team, 2023) to train models from the gemma-3 family and LLaMA-Factory (Zheng et al., 2024) to train tencent/HY-MT1.5-7B. We also conducted experiments with both full fine-tuning and QLoRA fine-tuning for the larger models (gemma-3-4b-it and HY-MT1.5-7B).

For all models, we employ a standard sequence-to-sequence training objective, maximizing the likelihood of the target Chuvash translation given the English source. For encoder-decoder architectures, we prepended a simple instruction prefix (e.g., translate English to Chuvash:). For decoder-only chat models like Gemma, we formatted the input using the model's prescribed chat template, placing the translation instruction and source text within a single user message.

In experiments with the google/umt5-small and gemma-3 family models, training lasted up to 3 epochs, while in experiments with tencent/HY-MT1.5-7B, training lasted 1 epoch. A learning rate of 2e-5 with a linear scheduler was used for full fine-tuning of the google/umt5-small and gemma-3 family models, the same learning rate with a cosine scheduler was used for full fine-tuning of tencent/HY-MT1.5-7B, and a learning rate of 4e-4 with a linear scheduler was used for QLoRA configurations. The training dataset was split into train and test subsets in a 9:1 ratio. We used a batch size of 16; when a full batch did not fit in GPU memory, we employed gradient accumulation to achieve an effective batch size of 16. To reduce GPU memory requirements and speed up training, we utilized techniques such as the 8-bit AdamW

optimizer, training in pure bf16 mode, and Flash Attention 2 (Dao, 2024).

5 Generation Params

To improve the quality of the generation after fine-tuning, we selected the generation parameters on the FLORES+ dataset. For this purpose, the dataset was split equally into train and test sets. On the train set, we iterated over the generation parameters and optimized the chrF++ metric using Optuna (Akiba et al., 2019); the final score was calculated on the test set. Since the dataset is sufficiently diverse and covers a wide range of topics, the improvement in the metric on this dataset showed good correlation with the metric on the public set of the shared task.

The parameters optimized included beam search width, length penalty, repetition penalty, and temperature sampling. The optimal configuration found significantly improved translation fluency and adequacy. The best parameters for the HY-MT1.5-7B model were: temperature=0.2764731626394478, top_p=0.99559242372123, top_k=77, num_beams=5, repetition_penalty=1.0359827344136177.

This tuning process was crucial, as the default generation parameters often produced overly conservative or repetitive translations for the low-resource language. The optimized parameters encouraged more diverse and contextually appropriate outputs, which was reflected in the significant metric gain.

6 Results and Discussion

On the public leaderboard, the tencent/HY-MT1.5-7B model performed the best, and it also had the most parameters. Furthermore, tuning the generation parameters improved the quality of this version in the public set from 25.42 to 29.67 (see Table 3). This represents a relative improvement of over 16% from parameter tuning alone.

During experiments with QLoRA fine-tuning, we found that the quality of training gemma-3-4b-it was no better than that of full fine-tuning of gemma-3-1b-it, suggesting that for the scale of data available, the benefits of a larger model architecture may be offset by the limitations of the parameter-efficient fine-tuning method in this specific low-resource scenario. This observation aligns with recent findings that the effectiveness of PEFT methods can be task- and data-size dependent, and full fine-tuning may still be preferable when computational resources allow and the target data distribution differs significantly from the pretraining data.

Model	chrF++ score
umt5-small	13.03
gemma-3-270m-it	14.60
gemma-3-1b-it	20.43
gemma-3-4b-it w/ QLoRA	20.43
HY-MT1.5-7B	25.42
HY-MT1.5-7B w/ gen params	**29.67**

Table 3: Submission results on the LoResMT 2026 EnglishChuvash public test set. The best score is in bold.

The results demonstrate a clear correlation between model size and translation quality for this task. The specialized translation architecture of HY-MT1.5-7B likely contributed to its superior performance. The effectiveness of generation parameter tuning highlights the importance of inference-time optimization beyond just model training. An error analysis on a sample of outputs revealed that the larger model with tuned parameters produced fewer grammatical errors and better captured Chuvash morphology.

The parameters found in Section 5 were used for the final submission of the best model.

7 Conclusion

This paper presented our submission to the LoResMT 2026 English-Chuvash translation task. We explored various model architectures, from smaller encoder-decoder models to large multilingual translation models. Our experiments showed that the tencent/HY-MT1.5-7B model, when combined with careful data preprocessing and optimized generation parameters, achieved the highest chrF++ score of 29.67. The significant gain from the tuning of the generation parameters underscores its importance in low-resource MT pipelines. Our work also highlights the trade-offs involved in model and method selection for low-resource settings: while large, specialized models yield the best results, efficient fine-tuning techniques on similar-sized models may not provide a commensurate advantage with limited data. Future work could involve exploring additional data augmentation techniques for Chuvash, investigating more efficient

fine-tuning methods for very large models, and incorporating linguistic features specific to Turkic languages to further improve translation quality.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. 2023. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. In *The Eleventh International Conference on Learning Representations*.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

Tri Dao. 2024. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*.

Alexander Jones, Isaac Caswell, Orhan Firat, and Ishank Saxena. 2023. "GATITOS: Using a New Multilingual Lexicon for Low-resource Machine Translation". In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 371–405, Singapore. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846.

Gemma Team. 2025. Gemma 3.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Mao Zheng, Zheng Li, Tao Chen, Mingyang Song, and Di Wang. 2025. Hy-mt1.5 technical report. *Preprint*, arXiv:2512.24092.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Eric Zhu, Vadim Markovtsev, Aleksey Astafiev, Arham Khan, Chris Ha, Wojciech Łukasiewicz, Adam Foster, Sinusoidal36, Spandan Thakur, Stefano Ortolani, Titusz, Vojtech Letal, Zac Bentley, fpug, hguhlich, long2ice, oisincar, Ron Assa, Senad Ibraimoski, and 8 others. 2024. ekzhu/datasketch: v1.6.5.

Ensemble Methods for Low-Resource Russian-Kyrgyz Machine Translation: When Diverse Models Beat Better Models

Adilet Metinov
metinovab@kstu.kg

Abstract

We present our submission to the LoResMT 2026 Shared Task on Russian-Kyrgyz machine translation. Our approach demonstrates that ensembling diverse translation models with simple majority voting can significantly outperform individual models, achieving a +1.37 CHRF++ improvement over our best single model. Notably, we find that including "weaker" models in the ensemble improves overall performance, challenging the conventional assumption that ensembles should only combine top-performing systems. Our best submission achieved 49.31 CHRF++, placing 3rd in the Russian-Kyrgyz track, using only open-weight models without any fine-tuning on parallel Kyrgyz data. We also report several counter-intuitive findings: (1) simple voting outperforms quality-weighted selection, (2) more diverse models help even when individually weaker, and (3) post-processing "corrections" can hurt performance when reference translations contain similar artifacts.

1 Introduction

Machine translation for low-resource languages remains challenging due to limited parallel data and linguistic resources. Kyrgyz, a Turkic language spoken by approximately 4.5 million people, falls into this category despite growing digitalization efforts.

The LoResMT 2026 Shared Task on Russian-Kyrgyz translation provides an opportunity to explore effective strategies for this language pair. In this system description paper, we present our ensemble-based approach that achieved competitive results without requiring expensive model fine-tuning or access to large parallel corpora.

Our main contributions are:

- A simple yet effective voting-based ensemble method that combines diverse translation models
- Empirical evidence that including weaker models improves ensemble performance
- Analysis of why simpler selection strategies outperform complex quality filtering
- Practical insights for low-resource MT without fine-tuning

2 System Description

2.1 Base Models

We generated translations using multiple models from two families:

NLLB Models We used three variants of Meta's No Language Left Behind (NLLB) models (NLLB Team et al., 2022):

- NLLB-200 600M (distilled)
- NLLB-200 1.3B
- NLLB-200 3.3B

DeepSeek Models We also included translations from DeepSeek-R1, a large language model with strong multilingual capabilities, running locally on our GPUs with open weights.

Table 1 shows the individual performance of each model.

Model	CHRF++
NLLB 3.3B	47.94
NLLB 1.3B	47.34
NLLB 600M	46.59
DeepSeek (basic)	46.73
DeepSeek-R1	45.80

Table 1: Individual model performance on the test set.

Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026), pages 235–237

2.2 Ensemble Method

Our ensemble approach uses consensus-based voting to select the best translation for each sentence. Given n candidate translations for a source sentence, we compute pairwise similarity scores and select the translation with highest average similarity to all others.

Similarity Metric We use character-level n-gram similarity (Jaccard coefficient over character trigrams):

$$sim(t_1, t_2) = \frac{|ngrams(t_1) \cap ngrams(t_2)|}{|ngrams(t_1) \cup ngrams(t_2)|} \quad (1)$$

Voting Procedure For each sentence, we select:

$$t^* = \arg\max_{t_i} \frac{1}{n-1} \sum_{j \neq i} sim(t_i, t_j) \quad (2)$$

This simple approach selects the translation that has the highest consensus with other models, effectively implementing a "wisdom of the crowd" strategy.

2.3 Quality Filtering (Ablation)

We also experimented with quality-based filtering before voting, detecting:

- English words in translations
- Russian words that should have been translated
- Encoding artifacts and repeated characters
- Unusual length ratios compared to source

However, as we report in Section 3, quality filtering did not improve results.

3 Experiments and Results

3.1 Main Results

Table 2 shows our ensemble experiments. Our best result (49.31 CHRF++) was achieved with simple voting over 5 models.

3.2 Key Findings

Finding 1: Weaker models improve the ensemble. Counter-intuitively, adding DeepSeek-R1 (45.80 CHRF++ individually) to our NLLB ensemble *improved* overall performance from 48.22 to 49.31. This suggests that model diversity contributes more to ensemble success than individual model quality.

Configuration	Models	CHRF++
Best single model	1	47.94
3 NLLB models (vote)	3	48.22
5 models (vote)	5	**49.31**
7 models (vote)	7	49.17
5 models (qual. filter, t=50)	5	49.23
5 models (qual. filter, t=40)	5	49.23
5 models (qual. filter, t=60)	5	48.60

Table 2: Ensemble results. Simple voting with 5 models achieves the best performance.

Finding 2: Simple voting beats quality filtering. Our quality-weighted ensemble (49.23) performed worse than pure voting (49.31). We hypothesize that the quality heuristics filtered out translations that, while appearing "incorrect," actually matched the reference translation style.

Finding 3: There is a sweet spot for ensemble size. Performance improved from 3 models (48.22) to 5 models (49.31), but degraded slightly with 7 models (49.17). Too many models may introduce noise that dilutes the consensus signal.

Finding 4: Post-processing hurts performance. We attempted to "clean" our outputs by:

- Removing double spaces
- Normalizing punctuation
- Adding missing end punctuation

This reduced our score from 49.31 to 49.27, suggesting that reference translations contain similar artifacts.

3.3 Source Distribution Analysis

Table 3 shows how often each model was selected in our best ensemble.

Model	Selected	%
NLLB 1.3B	736	31.8%
NLLB 3.3B	642	27.8%
NLLB 600M	544	23.5%
DeepSeek-R1	253	10.9%
DeepSeek (basic)	136	5.9%

Table 3: Source distribution in the final ensemble. All models contribute, with NLLB variants selected most frequently.

Interestingly, even the worst-performing model (DeepSeek basic, 46.73) was selected for 136 sen-

tences (5.9%), indicating it provided the best consensus translation for those cases.

4 Analysis

4.1 Why Does Diversity Help?

We hypothesize that different model architectures make different types of errors. NLLB models, trained specifically for translation, may handle common patterns well but struggle with unusual constructions. LLMs like DeepSeek, while potentially less consistent, may handle creative or contextual translations better.

When these diverse models agree, the consensus is likely correct. When they disagree, the voting mechanism tends to select translations that share common elements, which often correlates with correctness.

4.2 Why Does Quality Filtering Hurt?

Our quality heuristics were designed to detect:

- Untranslated Russian words
- English contamination
- Formatting issues

However, we found that reference translations in the test set may contain similar "issues" — for example, some technical terms left untranslated, or inconsistent spacing. By filtering these out, we moved away from the reference distribution, hurting our CHRF++ score.

This highlights an important consideration for MT evaluation: optimizing for automatic metrics may require matching reference artifacts, not just producing "clean" translations.

4.3 Comparison to Top Systems

The top two systems achieved 51.03 and 51.02 CHRF++, approximately 1.7 points above our best result. Given their minimal submission counts (2–3 submissions), we hypothesize they may have used:

- Fine-tuned models on Kyrgyz parallel data
- Different model architectures better suited for Turkic languages
- Access to additional training resources

Our approach, using only off-the-shelf open-weight models without fine-tuning, represents a strong baseline for resource-constrained settings.

5 Conclusion

We presented a simple ensemble approach for Russian-Kyrgyz machine translation that achieves competitive results without model fine-tuning. Our key findings — that diversity matters more than individual quality, and that simple voting beats complex filtering — provide practical insights for low-resource MT.

Future work could explore:

- Learned combination weights instead of uniform voting
- Quality estimation models trained on this language pair
- Fine-tuning base models on available Kyrgyz parallel data

Limitations

Our work has several limitations. First, our ensemble method was only evaluated on one language pair (Russian-Kyrgyz), and the findings may not generalize to other low-resource pairs. Second, we relied entirely on automatic metrics (CHRF++) without human evaluation, which may not fully capture translation quality. Third, our approach requires running inference with multiple models, which increases computational cost compared to a single model. Finally, we did not explore fine-tuning on available parallel Kyrgyz data, which could potentially improve base model quality and ensemble performance.

Acknowledgments

We thank the LoResMT 2026 organizers for providing this shared task and the Selectel company for sponsoring the competition.

References

NLLB Team, Marta R. Costa-jussà, James Cross, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Author Index