Improving Low-Resource Translation with Dictionary-Guided Fine-Tuning and RL: A Spanish-to-Wayuunaiki Study

Manuel Mosquera[1], Melissa Robles[1], Johan Rodríguez[1], and Rubén Manrique[1]

[1]Universidad de los Andes, Bogotá, Colombia, {ma.mosquerao, mv.robles, jd.rodriguez1234, rf.manrique}@uniandes.edu.co

Abstract

We propose a novel approach to machine translation for low-resource languages by integrating large language models (LLMs) with external linguistic tools. Focusing on the Spanish–Wayuunaiki language pair, we frame translation as a tool-augmented decision-making problem, wherein the model can selectively consult a dictionary during the translation process. Our method combines supervised fine-tuning with reinforcement learning based on the Guided Reward Policy Optimization (GRPO) algorithm, enabling an instruction-tuned model to learn both when and how to use the external tool effectively. To align model behavior with translation quality, we leverage GRPO's reward mechanism, guided by BLEU scores. To assess the impact of model architecture and training strategy, we conduct ablation studies on our training pipeline and compare Qwen2.5-0.5B-Instruct with other models, including LLaMA and a prior system based on the NLLB model. Preliminary results demonstrate that instruction-tuned models with tool access, further refined through reinforcement learning, achieve state-of-the-art performance on the Spanish–Wayuunaiki test set. These findings underscore the potential of LLM-based agents augmented with external tools to improve translation quality in low-resource language settings.

1 Introduction

Low-resource languages, particularly Indigenous languages, present an important challenge for natural language processing (NLP) due to the limited availability of high-quality parallel corpora and the predominance of oral traditions over written forms [15, 27]. Although recent advances in NLP, including the widespread adoption of large language models (LLMs), have significantly improved performance in high-resource languages, these gains have not translated equally to low-resource contexts [15, 12, 17]. Languages with minimal digital presence continue to face structural disadvantages, both in terms of data availability and the applicability of current modeling strategies. These disparities hinder the development of effective machine translation systems, particularly those based on data-intensive supervised learning approaches that assume access to large-scale parallel corpora [15].

In recent years, the translation of Indigenous languages in the Americas has received increasing attention, driven by efforts to promote linguistic inclusion and cultural preservation. A prominent example is the AmericasNLP Shared Task, which in its 2025 edition included translation benchmarks for 14 Indigenous languages from North, Central, and South America [6]. This initiative has led to significant advances in corpus development, data curation, and model evaluation tailored to low-resource scenarios. The development of translation tools catering specifically to Indigenous languages holds the potential to expand access to digital resources and support ongoing efforts in language revitalization, education, and cultural transmission.

Most of the recent progress has been achieved by fine-tuning machine translation models based on the Transformers architecture on small, carefully curated datasets [6, 14]. While this approach has shown encouraging results, it still depends on annotated data and tends to generalize poorly on out of distribution data. As such, it remains difficult to scale or adapt to languages with minimal parallel resources available.

To overcome these limitations, reinforcement learning (RL) has emerged as a promising alternative. RL methods such as Proximal Policy Optimization (PPO) [28] and its recent extension Generalized Reinforcement Policy Optimization (GRPO) [19] have gained popularity in LLM training, particularly when combined with reward-based feedback mechanisms like Reinforcement Learning from Human Feedback (RLHF) [22]. Unlike supervised fine-tuning, RL enables models to learn policies over sequences of actions, allowing for dynamic interaction with external resources and better adaptability to sparse or structured feedback.

In this article, we propose an alternative to traditional fine-tuning strategies for improving machine translation into Wayuunaiki, the most widely spoken Indigenous language in Colombia. Our approach builds on the instruction-tuned model Qwen2.5-0.5B-Instruct [25], which we further optimize using reinforcement learning. Unlike standard methods, we frame the model as an agent capable of interacting with an external Wayuunaiki–Spanish dictionary. To support this interaction, we adopt the GRPO framework introduced by DeepSeek [19], enabling the model to learn when and how to call the dictionary as a tool. This agent-based formulation facilitates tool-augmented translation and reduces reliance on large annotated corpora. To the best of our knowledge, this is the first work to incorporate a dictionary as an interactive tool in low-resource machine translation. By treating the model as an agent, our methodology opens new avenues for research into tool-augmented translation strategies in underrepresented languages.

1.1 Paper organization

This paper is structured as follows. In Section 2, we review prior work on machine translation for low-resource and Indigenous languages, emphasizing the challenges posed by data scarcity. We also discuss recent efforts to apply reinforcement learning to machine translation tasks. Section 3 introduces our methodology, where we frame translation as a tool-augmented decision-making task. We describe the supervised fine-tuning and reinforcement learning setup, and introduce the GRPO algorithm as a means to refine tool-augmented behavior. This section also details the parallel corpus, the selected models, and the training setup. Section 4 presents our experimental results, followed by a discussion of key findings, limitations, and future directions for tool-augmented translation in low-resource settings in Section 5.

2 Related Work

Wayuunaiki is an Arawakan language spoken by approximately 420,000 people across northern Colombia and Venezuela, primarily within the Wayuu community. It features agglutinative morphology and a predominant subject–object–verb (SOV) word order. Despite its relatively large number of speakers compared to other Indigenous languages in the region, Wayuunaiki remains underrepresented in NLP resources. Nevertheless, several efforts have aimed to build foundational datasets and translation tools. In 2021 [3], Rafael José Negrette Amaya compiled a bilingual Wayuunaiki–Spanish dictionary containing over 74,000 entries, providing an important lexical resource. In addition, aligned translations of religious and institutional texts, including the Bible [1, 2], the Colombian Constitution [5], and other literary works [34, 33, 35, 10] have contributed to the pool of available parallel data.

Initial computational work on Wayuunaiki–Spanish translation emerged in 2023 with the development of the first neural machine translation (NMT) system for this low-resource language pair [11]. More recent efforts have focused on fine-tuning large translation models on the limited available corpora. These include experiments leveraging Finnish pretrained models, due to some structural similarities, and multilingual models such as the No language Left Behind (NLLB) translation model [21], which include low-resource languages into their training [27, 24, 14]. While these approaches demonstrate that modern architectures can be adapted to Wayuunaiki, they remain constrained by the scarcity of annotated data and the limited domain coverage of the training material.

The adoption of reinforcement learning techniques, particularly PPO [28], has become increasingly popular in the training of large LLMs such as GPT, especially in the context of RLHF [22]. More recently, GRPO [29, 19], introduced by DeepSeek, has further refined this paradigm by enhancing stability and generalization during RL-based training. These methods have enabled LLMs to align more effectively with human preferences and task-specific behaviors, facilitating their use in instruction-tuned settings. In this context, RL has emerged not only as a fine-tuning mechanism but also as a means of endowing agent-like models with the capacity to reason and act over structured tools or knowledge bases during task execution.

In 2024, Zhan et. al [32] introduced a reinforcement learning domain adaptation approach for neural machine translation, utilizing in-domain monolingual data to mitigate overfitting and reinforce domain-specific knowledge acquisition. Their method involves training a ranking-based model with a small-scale in-domain parallel corpus, which serves as a reward model to select higher-quality generated translations during fine tuning.

In parallel, agent-based frameworks have been proposed to address the complexities of translation tasks. For instance, Briva-Iglesias (2025) [4] presented a multi-agent system for translating ultra-long literary texts, where specialized agents collaborate to handle different aspects of the translation process, such as adequacy review and fluency enhancement. This approach mirrors traditional human translation workflows and has shown promising results in maintaining contextual fidelity and cultural nuances.

Recent work has shown that integrating external tools into large language models can enhance their ability to perform complex reasoning and structured tasks. Approaches like Search-R1 [16], ReTool [7] and SWiRL [9] use reinforcement learning to train models to decide when and how to use tools such as code interpreters, calculators, or web search during multi-step problem-solving. These methods move beyond more common RL post-training methods by allowing models to learn tool use strategies based on task outcomes, enabling more flexible, agent-like behavior. Such capabilities are especially relevant for low-resource languages, where structured linguistic resources like dictionaries can be integrated into the learning process to compensate for limited training data.

3 Methods

Figure 1 summarizes our methodology. To develop our translation system, we start from an instruction-tuned language model and extend it with the capability to interact with an external Wayuunaiki–Spanish dictionary. The model is framed as an agent that can choose whether to consult this dictionary during the translation process.

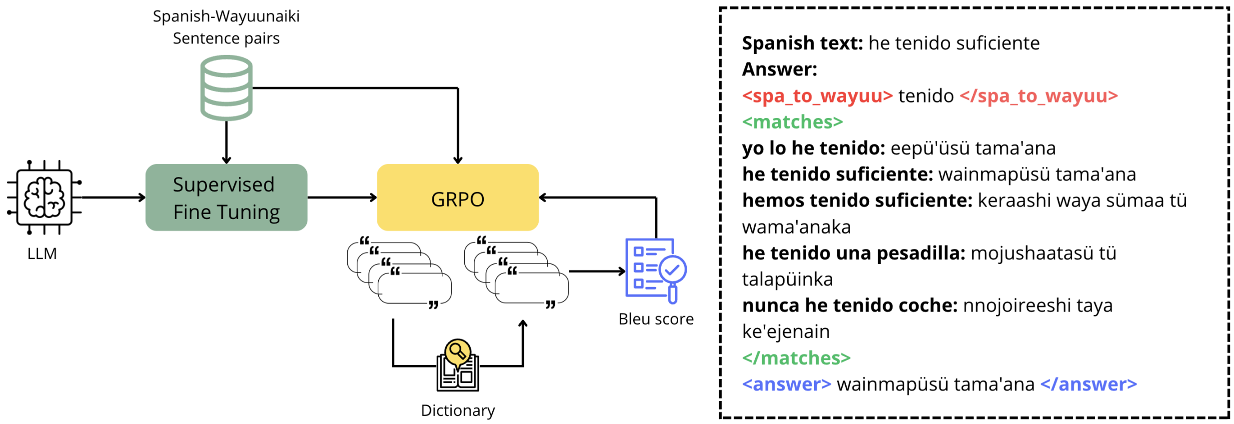

Figure 1: Overview of the training pipeline. A large language model is first finetuned using supervised learning on Spanish–Wayuunaiki sentence pairs. The finetuned model is then further optimized using GRPO, where the reward is based on BLEU scores computed against reference translations. During this phase, the model can optionally use a dictionary tool to assist translation. The right-hand side illustrates an example of how the model interacts with the dictionary during the generation process.

We first perform a supervised fine-tuning stage on the base instruction model. This step serves two key purposes: (1) it trains the model to produce outputs in a structured format using predefined tags, and (2) it demonstrates how to invoke and interpret the dictionary tool correctly. To support this, we construct a dataset consisting of Spanish–Wayuunaiki translation examples, where each instance illustrates how to use the dictionary during the translation process. We format each training example using the following prompt template:

```
‘‘Translate the following Spanish text into
Wayuunaiki. Begin by identifying any words
or phrases you're unsure how to translate.
Then, you may look up those words using the
dictionary tool by wrapping the Spanish word
in <spa_to_wayuu> and </spa_to_wayuu>, and doing
that for every unknown word. The dictionary
will return matches enclosed in <matches> and
</matches>. You can use the dictionary as
many times as necessary. Once you have all
the information you need, provide the final
translation enclosed in <answer> and </answer>.
For example: <answer> xxx </answer>.
Spanish text: {}’’
```

In each example, between one and four words are randomly selected to be queried using the dictionary tool, and the model is shown both the dictionary output (first five matches) and the correct final translation. This stage is motivated by recent findings on the cognitive behaviors that enable self-improving reasoning in language models [8], which suggest that models must first acquire structured habits, such as strategic tool usage, before reinforcement learning can effectively refine their behavior.

Once the model has been fine-tuned to follow the structured prompt format and correctly use the dictionary tool, we proceed to the reinforcement learning stage. We adopt the GRPO framework [19], which is designed to align LLM behavior with complex tasks. In this setup, the language model itself acts as the policy. At each training step, we sample a Spanish–Wayuunaiki sentence pair and generate multiple candidate translations. Specifically, we create 8 copies of the same input prompt as defined during fine-tuning, and query the model to produce responses, potentially using different combinations of dictionary tool invocations.

For each prediction, only the text enclosed within the <answer> tags is extracted and used for evaluation. Each generated output is then evaluated against a reference translation using BLEU [23], which serves as the reward signal for GRPO to update the policy based on translation quality, Additionally, tool outputs are masked to ensure they do not contribute to the policy loss [16]. This process enables the model to iteratively refine its translation strategy, improving overall performance while learning when and how to use the dictionary tool more effectively. To monitor progress during training, we evaluate the model every 50 steps on a fixed set of 640 sentence pairs sampled from the training dataset.

Since our task involves translating into Wayuunaiki, a language that differs significantly from the model's original training distribution, we adopt the approach used in DAPO [31] and Dr.GRPO [19], which relax the traditional GRPO constraint based on KL-divergence penalties. This adjustment is essential because the model must undergo substantial behavioral changes to produce coherent Wayuunaiki translations. Standard regularization methods that constrain the model to remain close to its initial policy would limit its ability to adapt effectively.

To further improve training efficiency, we employ LoRA during both the SFT and RL training stages [30]. Additionally, we omit clipping in the policy loss, which allows us to maintain only a single model in memory throughout training.

3.1 Datasets and models

For training and evaluation, we use the Spanish–Wayuunaiki parallel corpus introduced by Prieto et. al [24], which was included in the AmericasNLP 2025 Shared Task [6]. This dataset provides both training and development splits specifically curated for low-resource translation scenarios. To support tool-augmented translation, we incorporate a bilingual dictionary compiled by Rafael Jose Negrette Amaya [3], which originally contains approximately 74,000 Spanish–Wayuunaiki word and phrase pairs. To ensure tool responses remain concise and manageable, we filter this dictionary to retain only entries with five words or fewer on the Spanish side, resulting in a final dictionary of approximately 29,000 entries.

As a base instruction model, we use Qwen2.5-0.5B-Instruct [25], which offers multilingual support across more than 20 languages and is specifically optimized for cross-lingual tasks. One of the key design choices behind this model is its ability to generalize across languages through a cross-lingual transfer mechanism. This is achieved by translating instructions from high-resource languages into low-resource ones and generating corresponding response candidates. This training strategy makes Qwen2.5-0.5B-Instruct particularly well-suited for tasks involving low-resource languages such as Wayuunaiki, where robust generalization and instruction-following are essential.

3.2 Training

To evaluate model performance during training, we use the BLEU score [23], a standard metric in machine translation that measures similarity between the generated output and a reference translation by comparing overlapping n-grams. To enable parameter-efficient training, we apply LoRA (Low-Rank Adaptation) [13] during both supervised fine-tuning and reinforcement learning. During the RL phase, we adopt several strategies to improve computational efficiency and training stability: vLLM [18] is used to accelerate inference and enable efficient trajectory sampling; gradient accumulation over 8 steps helps manage memory constraints while preserving an effective batch size; and DeepSpeed [26] is integrated into the pipeline to further reduce memory usage and improve training throughput. All models are optimized using AdamW with a fixed learning rate of 5×10^{-6}.

3.3 Experimental setup

Our experiments systematically evaluate three key factors: training approach (zero-shot, supervised fine-tuning, reinforcement learning), dictionary access (available vs. unavailable), and model architecture (instruction-tuned vs. translation-specific models). All experiments are framed around a single core task: Spanish-to-Wayuunaiki translation using structured prompts that optionally enable interaction with an external dictionary.

We begin by establishing baselines using the instruction-tuned model Qwen2.5-0.5B-Instruct in zero-shot settings. This allows us to evaluate the model's default translation capabilities without further adaptation. To test whether tool awareness alone is beneficial, we also include a variant where the model is informed that a dictionary is available but receives no examples of how to use it. These prompt-based settings rely solely on the model's pretraining to guide behavior and provide a foundation for evaluating subsequent training strategies.

We then explore supervised fine-tuning (SFT) to assess whether explicit demonstrations improve performance. One set of experiments uses standard parallel sentence pairs without tool interaction, serving to isolate the benefits of exposure to target-domain data. A second set extends this by introducing synthetic demonstrations that show the model how to use the dictionary tool. These examples are automatically constructed and illustrate when and how to query the tool during translation, allowing us to test whether models can learn tool-augmented behaviors from examples alone. For both settings, models were fine-tuned for one epoch on 59,715 paired sentences, using a learning rate of 1×10^{-4}, the AdamW optimizer, and prompt masking to ensure training focused only on the target completions.

To determine whether reinforcement learning (RL) alone can drive translation improvement, we apply GRPO to the base model without any prior supervised fine-tuning. We then evaluate a combined approach where SFT is followed by RL, in order to assess whether reinforcement learning can further refine tool usage and translation quality after initial supervised adaptation. These experiments are run both with and without tool access, allowing us to isolate the impact of the dictionary in the context of policy optimization. Notably, RL training for the tool-enabled model is performed on an SFT-trained version that incorporates tool usage, whereas for the tool-free model, RL is applied to an SFT-trained version that learns to perform translations directly and independently of any external tools.

Within the RL framework, we explore two reward strategies: sentence-level BLEU scores [23] and character-level edit-based rewards [20]. Additionally, we examine the effect of RL training duration by directly comparing the performance of models trained for 400 steps versus those trained for 1400 steps.

Finally, to assess the generality of our approach, we replicate key experiments across different model architectures. We apply our full methodology—including SFT and RL with dictionary access—to meta-llama/Llama-3.2-1B-Instruct, enabling a comparison over different pretraining bases. We also test a larger model, Qwen2.5-7B-Instruct, to explore whether scale offers measurable gains in low-resource translation. In parallel, we test our RL framework on a translation-specific model, NLLB [21], which is not instruction-tuned and cannot follow structured prompts. For this setup, we use the Wayuunaiki-specific checkpoint

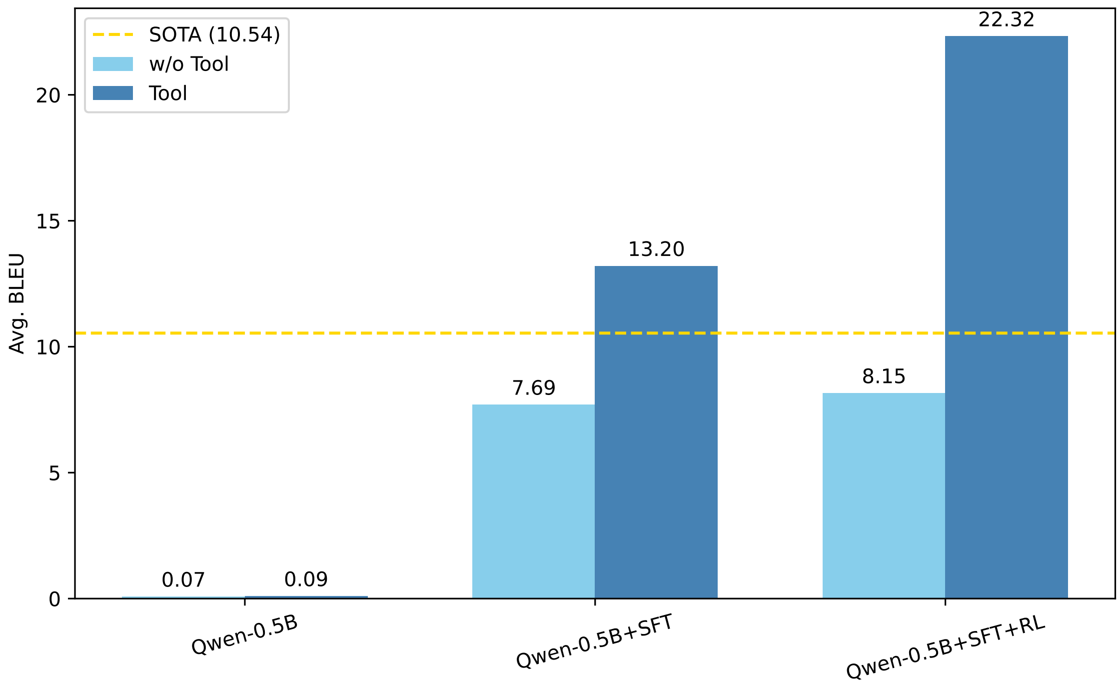

Figure 2: Average BLEU scores for various Qwen model variants, with and without tool usage. The results indicate that SFT effectively imparts basic translation capabilities to the model. When the dictionary tool is enabled, the model learns to leverage it appropriately, further enhancing performance. RL training notably boosts translation quality, particularly when the tool is active, by teaching the model to better utilize the dictionary. Overall, tool usage plays a crucial role in surpassing previous state-of-the-art (SOTA) results.

from [24] and apply GRPO without tool access or prompting, thereby isolating the effects of reinforcement learning on a model with strong translation priors.

To evaluate all our models, we use the average BLEU score computed between sentences on the 6,635 samples from the development split of the Spanish–Wayuunaiki parallel corpus [24]. Additionally, we measure the proportion of translations in which at least one dictionary call was made. For those translations where the tool was used, we compute the average number of dictionary calls prior to generating the final output. To ensure cost efficiency, we cap the number of allowed dictionary calls at a maximum of four.

4 Results

This section presents the experimental results evaluating the performance of different models and training approaches for Spanish-to-Wayuunaiki translation, primarily using the BLEU score as the evaluation metric. The experiments examined training approaches (zero-shot, supervised fine-tuning (SFT), and reinforcement learning (RL)), dictionary access, and model architecture (instruction-tuned vs. translation-specific models).

Figure 2 presents the main results for the Qwen model under three configurations: without any fine-tuning (Base), with supervised fine-tuning (SFT), and with an additional post-training reinforcement learning (RL) stage consisting of 1,400 steps, using BLEU as the reward signal. The results demonstrate a **consistent improvement in model performance across each stage of training**. A particularly notable boost is observed when the external dictionary tool is incorporated. The Base Qwen-0.5B model achieved very low BLEU scores (0.07 without the tool, 0.09 with the tool). SFT significantly boosted performance, reaching an average BLEU of 13.20 with the tool. Adding the RL stage (SFT+RL) resulted in a substantial further improvement, achieving the highest average BLEU score of 22.32 with the tool, more than doubling the BLEU score of 10.54 reported in prior work [27]. This demonstrates the effectiveness of the combined SFT and RL approach for improving translation in this low-resource setting.

Table 1 provides a detailed look at the performance and tool usage for different Qwen-0.5B model variants. It corroborates the trend seen in Figure 2. Crucially, **the highest performing model (Qwen-0.5B+SFT+RL) relies most heavily on the dictionary**, using it in nearly every instance (99.98% of responses with tools) and averaging 3.76 calls per sample, close to the allowed maximum of 4. In contrast, RL alone proved insufficient for teaching effective dictionary usage, resulting in low tool usage (1.30% of responses with tools) and minimal BLEU performance (0.39). This is attributed to the model's initial difficulty in generating the correct structured format for dictionary calls, with early negative rewards discouraging tool use attempts, causing it to often default to simpler formats. Conversely, SFT significantly improved performance by teaching the model both accurate translation pairs and proper tool usage through examples. This foundation enabled the subsequent RL stage to effectively reinforce these behaviors, allowing the model to learn how to maximize the utility of the external tool.

Model	Avg. BLEU	Answers w/ Tools	Avg. Tool Calls
Qwen-0.5B	0.09	39.05%	1.00
Qwen-0.5B+RL	0.39	1.30%	1.00
Qwen-0.5B+SFT	13.20	90.11%	2.08
Qwen-0.5B+SFT+RL	**22.32**	**99.98%**	**3.76**

Table 1: Tool usage and BLEU scores for different variants of the Qwen-0.5B model. The results indicate that better-performing models make more extensive use of the dictionary tool. Notably, the Qwen-0.5B+SFT+RL model invokes the tool in nearly every response and approaches the maximum allowed number of calls per translation, averaging 3.76 out of 4.

Table 2 compares performance (Average BLEU) and tool usage across different model architectures (Qwen-0.5B, Qwen-7B, Llama3.2-1B, NLLB) and fine-tuning stages (Base, +SFT, +RL). For NLLB, which is not an instruction-tuned model and does not use dictionary access or structured prompts in this setup, RL was applied directly. Instruction-tuned models (Qwen and Llama) demonstrated **significant BLEU performance gains after the SFT and SFT+RL stages with tool access**, compared to their base performance. Notably, smaller models like Qwen-0.5B and Llama3.2-1B achieved comparable or better BLEU scores after SFT+RL with tools than the larger Qwen-7B (Qwen-0.5B+RL: 20.31, Llama3.2-1B+RL: 21.64 vs Qwen-7B+RL: 19.72). This indicates that the tool-augmented training strategy is particularly effective for boosting the performance of smaller models in low-resource translation. The NLLB model, not trained for dictionary use or structured prompts in this context, showed significantly lower performance compared to instruction-tuned models with tool access. This highlights the importance of the instruction-tuned architecture and tool integration in the proposed method.

The results show a **strong correlation between translation performance and effective use of the external dictionary**. As seen in Table 1, the Qwen-0.5B+SFT+RL model achieves the highest BLEU (22.32) and the highest average dictionary calls (3.76 calls per sample). Table 2 shows a similar pattern for the Llama3.2-1B model with SFT+RL, which also had high Average BLEU (21.64) and intensive dictionary usage (3.70 average calls). These findings support the hypothesis that **access to and strategic use of external lexical resources, facilitated by the agent-based approach and the combined SFT+RL training, is crucial for improving translation in low-resource languages** like Wayuunaiki.

Model	Avg. BLEU	Answers w/ Tools	Avg. Tool Calls
Base Models			
Qwen-0.5B	0.09	39.05%	1.00
Qwen-7B	13.63	95.07%	3.58
Llama3.2-1B	0.35	58.40%	2.08
NLLB	–	–	–
+ SFT			
Qwen-0.5B	13.20	90.11%	2.08
Qwen-7B	19.60	90.84%	2.97
Llama3.2-1B	13.97	87.25%	2.97
NLLB	9.24	–	–
+ RL			
Qwen-0.5B	20.31	99.91%	2.91
Qwen-7B	19.72	91.57%	2.90
Llama3.2-1B	**21.64**	**99.98%**	**3.70**
NLLB	9.17	–	–

Table 2: Performance comparison across base models, SFT, and RL stages. Instruction-tuned models demonstrate substantial gains from both SFT and RL, particularly due to their incremental use of external tools. In contrast, the NLLB model, which is specifically designed for translation tasks, underperforms in this setting as it can not benefit from tool usage. Notably, smaller models such as Qwen-0.5B show significant improvement across training stages, achieving performance levels comparable to larger models like Qwen-7B.

Table 3 examines the impact of different reward signals (BLEU vs. CharacTer Error Rate) and the number of RL steps (400 vs. 1400) on the Qwen-0.5B+SFT+RL model. The results in Table 3 suggest that both BLEU and CharacTer Error Rate reward signals yielded similar performance and tool usage at the evaluated number of RL steps. Importantly, **increasing RL steps from 400 to 1400 improved the Average BLEU and significantly increased Average Tool Calls**, indicating that more extensive RL training further refines the model's dictionary usage strategy. The highest performance (22.32 BLEU) was achieved with 1400 RL steps using BLEU reward.

5 Discussion and future work

As demonstrated by the results, training an LLM-based agent using Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL) proved effective for our low-resource translation task. In this study, we incorporated a dictionary tool to assist the model, which significantly enhanced its performance. However, a promising direction for future work involves finding additional tools that could potentially further improve the model's capabilities. For example, integrating a spell-checking tool to provide feedback or a vocabulary validator to ensure that all generated words exist in the target language could be beneficial.

Reward Signal	Avg. BLEU	Answers w/ Tools	Avg. Tool Calls
400 RL Steps			
BLEU	20.31	99.91%	2.91
CharacTer	20.25	99.88%	2.91
1400 RL Steps			
BLEU	**22.32**	**99.98%**	**3.76**
CharacTer	21.56	99.98%	3.74

Table 3: Effect of reward signal type and RL training duration on BLEU scores and tool usage. The results show that the choice between BLEU and CharacTer as the reward signal has minimal impact on overall performance, though BLEU-based rewards lead to slightly higher scores. Increasing the number of RL training steps significantly improves performance and encourages more frequent and intensive tool usage. However, the improvements appear to be reaching a plateau.

It would also be valuable to better understand why the dictionary tool leads to such a substantial performance boost. We leave for future work, analyzing the types of words the model queries in the dictionary. This could reveal whether the model follows specific strategies when interacting with the tool, and whether it effectively combines external lexical information with its internal knowledge to improve translation quality.

Interestingly, our experiments also revealed that using RL in isolation—without prior SFT—did not lead to performance gains, as measured by either BLEU or CharacTer scores. This raises questions about why the reward signal alone was insufficient for driving improvement. Future investigations could explore the characteristics a reward function must possess to be effective in this context. Alternatively, it may be that the nature of the search space in low-resource translation makes RL less suitable on its own.

6 Data and software availability

The algorithms and the datasets supporting the results presented in this article are available at RLTranslator.

7 Limitations

Our study presents a novel approach to low-resource machine translation for Spanish-to-Wayuunaiki, demonstrating state-of-the-art performance on the evaluated test set using a combination of Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL) augmented with a dictionary tool. However, our experimental setup and analysis faced several significant limitations. All experiments were conducted on a **single server at Universidad de los Andes, equipped with 4 RTX6000 GPUs that were shared among numerous students** undertaking various Natural Language Processing experiments. This limited computational access, coupled with each Reinforcement Learning step **taking several minutes** due to the need for generating multiple rollouts and computing rewards, severely constrained the scale and duration of our training. While the training dataset contains approximately 59,715 paired sentences, the final RL configurations were trained for 1400 steps, and increasing steps further showed performance plateauing. This restriction meant we were **forced to train using only a portion of the available dataset**, as the limited number of RL steps prevented extensive exposure to the full data variability. Furthermore, a critical limitation affecting our analysis was the **inability to access a native Wayuunaiki speaking person**. While automatic metrics like BLEU were used for evaluation, these do not fully capture the nuances of translation quality, fluency, or cultural appropriateness for a language with distinct structures like Wayuunaiki. Therefore, a thorough **qualitative analysis of the generated translations by native speakers is still pending and remains highly desirable** for future work to better understand the practical utility and accuracy of our system for the Wayuu community and to support ongoing language revitalization efforts.

References

[1] Antiguo testamento en wayuu, https://www.jw.org/guc/karaloutairua/biblia/wiwuliakat-genesis-nuchikimaajatkat-jesucristo/karaloutairua/g%c3%a9nesis/1/.

[2] Biblia en wayuu, https://www.bible.com/es/bible/1584/mat.1.g

[3] Rafael Jose Negrette Amaya. Osf spanish-wayuunaki, 2021.

[4] Vicent Briva-Iglesias. Are ai agents the new machine translation frontier? challenges and opportunities of single- and multi-agent systems for multilingual digital communication, 2025.

[5] Centro Colombiano de Estudios de Lenguas Aborígenes. *Constitución Política de 1991 traducida a Lenguas Indígenas*. 1994.

[6] Ona De Gibert, Robert Pugh, Ali Marashian, Raul Vazquez, Abteen Ebrahimi, Pavel Denisov, Enora Rice, Edward Gow-Smith, Juan Prieto, Melissa Robles, Rubén Manrique, Oscar Moreno, Angel Lino, Rolando Coto-Solano, Aldo Alvarez, Marvin Agüero-Torales, John E. Ortega, Luis Chiruzzo, Arturo Oncevay, Shruti Rijhwani, Katharina Von Der Wense, and Manuel Mager. Findings of the AmericasNLP 2025 shared tasks on machine translation, creation of educational material, and translation metrics for indigenous languages of the Americas. In Manuel Mager, Abteen Ebrahimi, Robert Pugh, Shruti Rijhwani, Katharina Von Der Wense, Luis Chiruzzo, Rolando Coto-Solano, and Arturo Oncevay, editors, *Proceedings of*

the Fifth Workshop on NLP for Indigenous Languages of the Americas (AmericasNLP), pages 134–152, Albuquerque, New Mexico, May 2025. Association for Computational Linguistics.

[7] Jiazhan Feng, Shijue Huang, Xingwei Qu, Ge Zhang, Yujia Qin, Baoquan Zhong, Chengquan Jiang, Jinxin Chi, and Wanjun Zhong. Retool: Reinforcement learning for strategic tool use in llms, 2025.

[8] Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D. Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars, 2025.

[9] Anna Goldie, Azalia Mirhoseini, Hao Zhou, Irene Cai, and Christopher D. Manning. Synthetic data generation and multi-step rl for reasoning and tool use, 2025.

[10] Quintina González, Ramírez Rectora, Filomena González Ramírez, Edgardo Reyes, Sierra Coordinador, Yasir Andres, Bustos Docente, Alveiro Machado, Pérez Betty Mejía, Atencio Hilario Chacin, Kiara Gonzalez Luis, Beltrán Margara González, Ramirez María, Teresa Bravo, Micaela Ipuana, Jusayu Nuris Ballesteros, Robinson González, Thawanui Guillen, De Carrillo, and Asesores Lingüísticos Andrin. Institución educativa indígena no 4 de maicao sede majayutpana.

[11] Nora Graichen, Josef Van Genabith, and Cristina España-bonet. Enriching Wayúunaiki-Spanish neural machine translation with linguistic information. In Manuel Mager, Abteen Ebrahimi, Arturo Oncevay, Enora Rice, Shruti Rijhwani, Alexis Palmer, and Katharina Kann, editors, *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*, pages 67–83, Toronto, Canada, July 2023. Association for Computational Linguistics.

[12] Hansi Hettiarachchi, Tharindu Ranasinghe, Paul Rayson, Ruslan Mitkov, Mohamed Gaber, Damith Premasiri, Fiona Anting Tan, and Lasitha Uyangodage, editors. *Proceedings of the First Workshop on Language Models for Low-Resource Languages*, Abu Dhabi, United Arab Emirates, January 2025. Association for Computational Linguistics.

[13] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[14] Jonathan Hus, Antonios Anastasopoulos, and Nathaniel Krasner. Machine translation using grammar materials for LLM post-correction. In Manuel Mager, Abteen Ebrahimi, Robert Pugh, Shruti Rijhwani, Katharina Von Der Wense, Luis Chiruzzo, Rolando Coto-Solano, and Arturo Oncevay, editors, *Proceedings of the Fifth Workshop on NLP for Indigenous Languages of the Americas (AmericasNLP)*, pages 92–99, Albuquerque, New Mexico, May 2025. Association for Computational Linguistics.

[15] Oana Ignat, Zhijing Jin, Artem Abzaliev, Laura Biester, Santiago Castro, Naihao Deng, Xinyi Gao, Aylin Ece Gunal, Jacky He, Ashkan Kazemi, Muhammad Khalifa, Namho Koh, Andrew Lee, Siyang Liu, Do June Min, Shinka Mori, Joan C. Nwatu, Veronica Perez-Rosas, Siqi Shen, Zekun Wang, Winston Wu, and Rada Mihalcea. Has it all been solved? open NLP research questions not solved by large language models. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8050–8094, Torino, Italia, May 2024. ELRA and ICCL.

[16] Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning, March 2025. arXiv:2503.09516 [cs].

[17] Omkar Khade, Shruti Jagdale, Abhishek Phaltankar, Gauri Takalikar, and Raviraj Joshi. Challenges in adapting multilingual LLMs to low-resource languages using LoRA PEFT tuning. In Kengatharaiyer Sarveswaran, Ashwini Vaidya, Bal Krishna Bal, Sana Shams, and Surendrabikram Thapa, editors, *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 217–222, Abu Dhabi, UAE, January 2025. International Committee on Computational Linguistics.

[18] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

[19] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective, 2025.

[20] Andrew Morris, Viktoria Maier, and Phil Green. From wer and ril to mer and wil: improved evaluation measures for connected speech recognition. 01 2004.

[21] NLLBTeam. No language left behind: Scaling human-centered machine translation, 2022.

[22] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John

Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.

[23] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, USA, 2002. Association for Computational Linguistics.

[24] Juan Prieto, Cristian Martinez, Melissa Robles, Alberto Moreno, Sara Palacios, and Rubén Manrique. Translation systems for low-resource colombian indigenous languages, a first step towards cultural preservation. In Manuel Mager, Abteen Ebrahimi, Shruti Rijhwani, Arturo Oncevay, Luis Chiruzzo, Robert Pugh, and Katharina von der Wense, editors, *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 7–14, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[25] 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, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

[26] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 3505–3506, New York, NY, USA, 2020. Association for Computing Machinery.

[27] Melissa Robles, Cristian A. Martínez, Juan C. Prieto, Sara Palacios, and Rubén Manrique. Preserving heritage: Developing a translation tool for indigenous dialects. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, page 1200–1203, New York, NY, USA, 2024. Association for Computing Machinery.

[28] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

[29] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024.

[30] Shangshang Wang, Julian Asilis, Ömer Faruk Akgül, Enes Burak Bilgin, Ollie Liu, and Willie Neiswanger. Tina: Tiny Reasoning Models via LoRA, April 2025. arXiv:2504.15777 [cs].

[31] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025.

[32] Hongxiao Zhang, Mingtong Liu, Chunyou Li, Yufeng Chen, Jinan Xu, and Ming Zhou. A reinforcement learning approach to improve low-resource machine translation leveraging domain monolingual data. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1486–1497, Torino, Italia, May 2024. ELRA and ICCL.

[33] José Álvarez. La conjugación del verbo en la lengua wayuu. 2016.

[34] José Álvarez. Compendio de la gramática de la lengua wayuu, 2017.

[35] José Ramón Álvarez González. Panorámica de la fonología y morfología de la lengua wayuu. 61, 2021.

Hyperparameter	Definition	Value
max_steps	Maximum number of examples seen	80000
sims_per_prompt	Simulations to calculate reward per example	8
policy_lr	Learning rate for the policy update	5e-6
kl_penalty_coef	Penalty coefficient to avoid a strong variation of the updated model compared with the reference model	0.04
temperature	Temperature of the LLM for generations	1.0
lower_clip	Minimum value for the loss function in the update policy	0.8
upper_clip	Maximum value for the loss function in the update policy	1.2
max_new_tokens	Maximum tokens generated by the LLM	512
r	Rank of the approximation matrices used for LoRA	64
lora_alpha	Scaling factor for LoRA approximation matrices	64
optimizer	type of optimizer	AdamW
policy_lr	Learning rate of the optimizer	5e-6
betas	optimizer beta	(0.9, 0.999)
eps	optimizer eps	1e-8
weight_decay	optimizer weight decay	0.0
gradient_clipping	optimizer gradient clipping	0.1

Table 4: Hyperparameters used for training

A Appendix

A.1 Training hyperparameters

Table 4 summarizes the hyperparameters used during both the supervised fine-tuning and reinforcement learning stages. The same settings are applied across all model variants unless otherwise specified.