

Toward Traceable Spanish–Wayuunaiki Translation: A Retrieval-Augmented and Grammar-Grounded Pipeline for a Low-Resource Language

Olmer Garcia-Bedoya^{1,*}, Jose Noguera²

¹*Ekumen Labs, Cambridge, United States*

²*Universidad Cooperativa de Colombia, Bogota, Colombia*

Abstract

Machine translation for Indigenous and other low-resource languages is constrained by limited parallel data, linguistic diversity, and the need for community-grounded assessment. This paper presents a traceable Spanish–Wayuunaiki translation-assistance pipeline that combines semantic retrieval over a bilingual corpus, exact lexical anchors, and retrieval of relevant grammar sections. Rather than claiming that a generative model can translate Wayuunaiki autonomously, the system gathers explicit evidence for each request: similar bilingual examples, lexicon-derived anchors, and grammar excerpts. These artifacts are incorporated into a structured prompt for an optional language-model back end and are exposed to users through a web interface and to AI clients through Model Context Protocol tools. The contribution is a practical architecture that separates retrievable linguistic evidence from optional text generation and supports inspection of the evidence used in each response. A preliminary native-speaker tests found perfect ratings for direct corpus matches, while longer compositional prompts preserved trace usefulness but exposed marked drops in grammar and naturalness. We specify a validation protocol with bilingual Wayuunaiki speakers that measures adequacy, grammatical acceptability, cultural appropriateness, and error categories. This design positions the system as a research prototype for assisted translation and collaborative language technology, not as a replacement for fluent speakers or professional translators.

Keywords

Retrieval-augmented generation, Wayuunaiki, Indigenous language technology, semantic search, human-centered AI

1. Introduction

Many language technologies remain difficult to develop for languages with scarce digitized resources. The challenge is not merely a small number of sentence pairs: useful systems must account for morphology, orthography, locally meaningful expressions, and the authority of speakers to judge whether a proposed translation is acceptable. These concerns are especially important for Wayuunaiki, an Arawakan language spoken by the Wayuu people in Colombia and Venezuela. In this setting, a translation tool should support inspection, correction, and community validation rather than obscure its evidence behind a single generated string.

A prior Spanish–Wayuunaiki prototype explored a neural encoder–decoder approach with attention [1]. In contrast, the present work investigates a retrieval-augmented architecture designed for limited resources and traceability. The system does not retrain a bilingual neural translation model. Instead, it retrieves semantically related Spanish–Wayuunaiki examples from a local vector database, detects exact lexical evidence when available, retrieves relevant grammar passages, and presents the assembled context to a downstream language model only when such generation is requested. Retrieval-augmented generation (RAG) offers a useful pattern because external evidence can be inspected and updated independently of a generator [2]. The previous prototype is therefore treated here as historical context

ICAIW 2026: Workshops at the 9th International Conference on Applied Informatics 2026, October 13–17, 2026, Orlando, United States

*Corresponding author.

✉ olmerg@gmail.com (O. Garcia-Bedoya)

🆔 0000-0002-6964-3034 (O. Garcia-Bedoya); 0000-0001-6780-3762 (J. Noguera)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

and design motivation, not as a directly re-executed experimental baseline.

The proposed paper targets applied artificial intelligence: it describes an end-to-end prototype, its design rationale, and a rigorous plan for evaluation with speakers. It makes four contributions:

1. a local, inspectable retrieval pipeline over a Spanish–Wayuunaiki corpus using dense multilingual embeddings and SQLite vector search;
2. a grammar-grounding mechanism that ranks grammar sections for each input instead of attaching a fixed grammar document to every request;
3. a traceability-oriented response context containing retrieved examples, lexical anchors, grammar evidence, and prompt instructions; and
4. a speaker-centered validation protocol that allowed us to compare different generative AI engines.

The work should be read as a systematic and methodological paper. It deliberately avoids unvalidated claims of translation quality, language preservation impact, or cultural appropriateness.

2. Background and Related Work

2.1. Indigenous language technology

Low-resource machine translation remains difficult when parallel corpora are limited or heterogeneous, and standard high-resource assumptions do not hold [3]. For Indigenous language technologies, data availability is only one dimension: research practice should also respect sovereignty, consent, benefit sharing, and the possibility that language resources are culturally sensitive [4, 5]. These principles motivate an architecture in which the linguistic evidence can be audited and revised without retraining a large model.

The earlier Spanish–Wayuunaiki system reported in Negrette et al. [1] used an attention-based neural sequence-to-sequence approach. Its experience motivates the present design in two ways. First, a small corpus does not guarantee reliable generation for open-ended inputs. Second, expert review is essential because a decreasing training loss is not a substitute for speaker judgment. The current study does not attempt a strict reproduction of that prototype, since doing so would require reconstructing an older software stack and deployment environment that are outside the scope of the present contribution.

2.2. Retrieval-augmented generation and semantic representations

RAG combines parametric generation with retrieved evidence [2]. In the present system, retrieval is more than a context-window optimization: it is the mechanism that lets a user inspect comparable bilingual examples and the rule excerpts available to the LLM generative AI. Sentence-level semantic embeddings make it possible to retrieve examples whose wording differs from the input while remaining topically or functionally similar. The implementation uses a multilingual Sentence-BERT-style model [6] with MiniLM-style transformer representations [7]; these embeddings are normalized so that cosine similarity can be used for nearest-neighbor search.

Automatic MT metrics such as BLEU [8] and chrF [9] can provide complementary signals when high-quality reference translations exist. However, they cannot establish grammatical acceptability, pragmatic adequacy, or cultural appropriateness. Human evaluation remains necessary, and its design should distinguish correctness, fluency, and contextual appropriateness [10].

3. Problem Setting and Design Goals

Let x be a Spanish input phrase or sentence. The objective is to construct a transparent context $C(x)$ that assists a bilingual reviewer or a text-generation component in proposing a Wayuunaiki translation

y . The goal is not to estimate a single end-to-end probability $p(y \mid x)$ from a scarce parallel corpus. Instead, the system constructs

$$C(x) = \{R_k(x), A(x), G_m(x), I\}, \quad (1)$$

where $R_k(x)$ denotes the k nearest bilingual examples, $A(x)$ is the set of exact lexical anchors, $G_m(x)$ is a set of m grammar sections selected for the request, and I is a fixed instruction schema. A downstream LLM generative AI is optional; without it, the system still exposes the evidence needed for assisted translation.

The following design goals guided the implementation:

- **Local evidence:** bilingual examples and grammar excerpts remain in project-controlled files and a local SQLite database.
- **Traceability:** each prompt reports the identifiers and distances of retrieved examples and the anchors used.
- **Separation of concerns:** retrieval and grammar ranking remain independent from the optional proprietary or open LLM generative AI.
- **Community validation:** system outputs are hypotheses until assessed by qualified Wayuunaiki speakers.
- **Reproducibility:** retrieval settings, corpus version, embedding model, and evaluation materials must be recorded for each experiment.

4. Resources and Data Preparation

The prototype uses a project-maintained CSV corpus with Spanish phrases in column `spa` and Wayuunaiki phrases in column `guc`; the corpus is publicly archived on OSF [11]. It also uses bilingual dictionary material and a grammar document. The present submission does not publish corpus size, provenance details, licenses, or speaker metadata because those items require a resource audit before public release. The camera-ready version should include a data statement listing: source and version of each resource, permissions for use, preprocessing decisions, known gaps, dialectal coverage, and restrictions on redistribution. To support the reproducibility claims of the paper, the project code and retrieval pipeline has been released to public at https://github.com/olmerg/wayuunaiki_traductor; the public artifact will not include community-controlled content unless explicitly authorized.

4.1. Vector database construction

Each Spanish corpus phrase s_i is encoded using a normalized multilingual embedding $e_i \in \mathbb{R}^d$. The associated Spanish phrase, Wayuunaiki phrase, and vector are stored in a local `vec0` table using `sqlite-vec` [12]. The table uses cosine distance. The database is reconstructed from the CSV when the corpus changes; normal translation requests issue read-only retrieval queries.

For a query x , the system computes the normalized vector e_x and retrieves nearest neighbors:

$$R_k(x) = \arg \operatorname{top-k} \cos(e_x, e_i). \quad (2)$$

The implementation requests more than k candidates, removes duplicate Spanish–Wayuunaiki pairs, and retains the first k unique items. A deterministic secondary ordering by row identifier is used when distances tie.

Table 1

Reproducibility information to record for each evaluation run.

Component	Information to record
Corpus	Version or checksum, number of entries, preprocessing, access restrictions, and source permissions.
Grammar resource	Version or checksum, sections included/excluded, and excerpt length.
Retriever	Embedding-model identifier, vector dimension, normalization, distance metric, and k .
Grammar ranker	Number m of sections, score in Eq. 3, and lexical-overlap weight λ .
LLM generative AI	Provider/model identifier, date, system prompt, decoding parameters, and failure-handling policy.
Evaluation	Input set version, condition order, anonymization scheme, participant protocol, and analysis code.

4.2. Grammar segmentation

The grammar document is segmented by Markdown headings. Introductory, alphabetic, and structural sections that are not expected to help sentence construction are excluded from ranking. For each retained section g_j , the system caches an embedding v_j of its title and a bounded content excerpt. Caching makes grammar selection efficient across requests and ensures that the same version of the grammar file is used throughout an application session.

4.3. Implementation and reproducibility controls

The prototype is implemented in Python. The corpus loader creates the vector table and is the only component that writes to the database. At request time, the translation pipeline opens the existing database in read mode from the application’s perspective and issues retrieval queries only. The optimized web pipeline initializes the sentence-embedding model, parses the grammar sections, computes their embeddings, and loads the SQLite vector extension once when the application starts. It then reuses these in-memory and database resources for subsequent requests. A lock serializes access to the shared SQLite connection in the web process.

Several implementation decisions are intended to make evidence retrieval reproducible. First, the stored corpus vectors and query vectors are normalized before cosine-distance search. Second, the system retrieves a candidate pool larger than the requested k , removes duplicate bilingual pairs after normalization, and then retains the first k unique pairs. Third, ties in vector distance are broken by the persistent row identifier. Fourth, input tokens used for anchor lookup are sorted, and grammar sections with equal scores are ordered by title. These rules do not make language-model generation deterministic, but they make the evidence trace stable for a fixed corpus, embedding model, and input.

The current project configuration retains a small, filtered grammar set after excluding introductory and purely structural sections; in the example reported here, the grammar ranker selects four sections. The retriever uses a multilingual MiniLM sentence encoder with 384-dimensional embeddings and cosine similarity after vector normalization. The exact model identifier and retained section count are part of the configuration metadata recorded for each run, even though we withhold community-controlled corpus metadata pending the resource audit.

Table 1 lists the information that must be recorded for every evaluation run.

5. Translation-Assistance Pipeline

Figure 1 presents the evidence flow as an embedded vector diagram. It depicts the pipeline explicitly, rather than a black-box translation model.

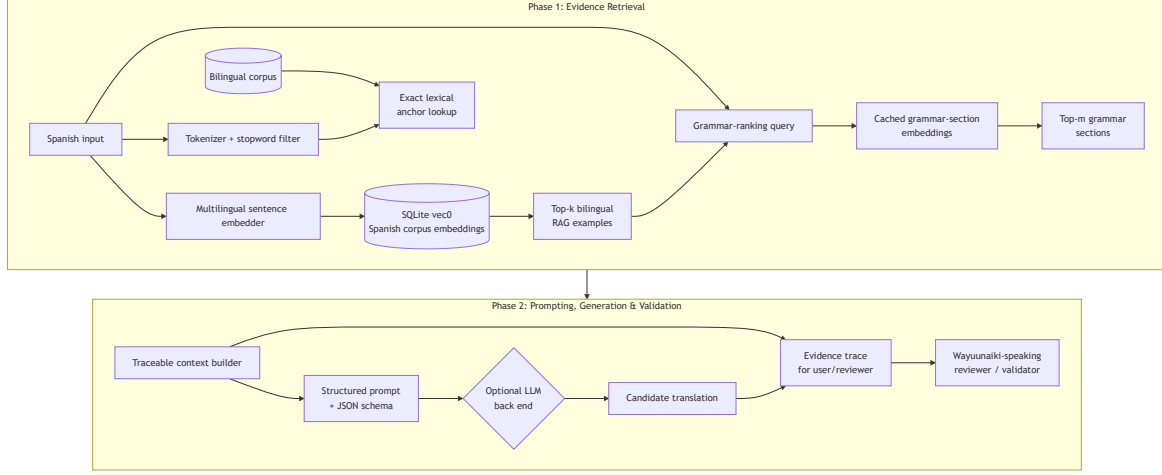


Figure 1: Traceable Spanish–Wayuunaiki translation-assistance architecture. Retrieval, exact lexical anchors, and grammar selection provide inspectable evidence; an optional generator produces a candidate only from the assembled context.

5.1. Semantic example retrieval

Given x , the system retrieves Spanish–Wayuunaiki pairs through a K-nearest-neighbor query over the Spanish embeddings. Similarity examples are not treated as ground-truth translations of x ; they are evidence that may reveal a relevant construction, term, or communicative intent. The user interface displays the retrieved pairs and cosine distances, allowing a reviewer to reject irrelevant context.

5.2. Exact lexical anchors

The system tokenizes the Spanish input, removes a small set of Spanish function words, and searches for exact normalized matches in the Spanish side of the corpus. Each match contributes an anchor tuple containing the input token, the matched Spanish phrase, its Wayuunaiki counterpart, and the source row identifier. Anchors are useful when a salient word has a direct corpus match, but they are explicitly not a general morphological analyzer or a word-alignment method. Their role is to supply inspectable lexical hints while preventing the LLM generative AI from relying only on broad semantic neighbors.

5.3. Grammar selection

The query for grammar ranking concatenates the Spanish input and the retrieved bilingual examples. For each grammar section g_j , the prototype computes

$$\text{score}(g_j, x) = \cos(q, v_j) + \lambda |T(q) \cap T(g_j)|, \quad (3)$$

where q is the normalized query embedding, $T(\cdot)$ extracts lexical tokens, and $\lambda = 0.01$ in the current implementation. The first term favors semantically relevant rules; the second gives a small deterministic preference to lexical overlap. The top m sections are attached to the prompt, and their titles and excerpts are shown to the user.

The value $\lambda = 0.01$ was not tuned against a labeled set; it was chosen so that the lexical term acts as a tie-breaker rather than the dominant signal. We tested this assumption with a command-line sweep over $\lambda \in \{0, 0.01, 0.05, 0.1, 0.3, 0.5, 1.0\}$ on eight Spanish queries, comparing the resulting top-3 grammar sections against the $\lambda = 0.01$ baseline. Because $|T(q) \cap T(g_j)|$ is an integer count (up to 11 in our queries) while cosine-similarity gaps between sections are typically on the order of 10^{-2} , even a small λ can outweigh the semantic term: the top-3 already changed for 8/8 queries at $\lambda = 0.05$, and for all queries at every $\lambda \geq 0.05$ we tested, whereas $\lambda = 0$ (pure semantic ranking) changed the top-3 for only 4/8 queries relative to $\lambda = 0.01$. This confirms that the ranking is highly sensitive to λ in

Table 2

Retrieved examples for the worked command-line run (non-evaluative trace).

ID	Spanish	Wayuunaiki	Distance
#2847	buenos dias	watta’a maalü	0.179796
#26607	buenos dias	wattachonya	0.179796
#24105	sean amables ustedes	kamanee jaya	0.223900
#1011	estate contento	talataa pia	0.241966
#1856	el esta feliz	talatshi niakai	0.245740
#24103	se amable tu	kamanee pia	0.247176
#9160	feliz navidad	talataa su’utpunaa jemelin nüchon ma’leiwa	0.252175
#4024	estas contento	talatshi pia	0.252549

the region above roughly 0.01–0.05, and that our fixed value keeps lexical overlap as a minor nudge rather than the primary ranking signal; it also shows that λ is not a free parameter that can be increased casually without materially changing which grammar rules are surfaced to the user.

5.4. Structured prompt and optional generation

The system assembles a structured prompt containing: (i) the Spanish input, (ii) selected grammar passages, (iii) RAG examples with identifiers and distances, (iv) anchors, and (v) a JSON response schema. The schema requests a primary translation, an optional alternative, a brief grammar rationale, the example identifiers used, and a confidence value. An optional language-model client can produce a candidate response from this prompt. The LLM generative AI is a replaceable component and is not considered authoritative; all content presented as a translation remains subject to speaker review.

The prompt is intentionally evidence-first. It instructs the generative AI not to copy a retrieved example unless it closely matches the source intent and asks it to report which examples were used. This requirement cannot guarantee faithful reasoning, since a generated explanation may itself be inaccurate. It does, however, make disagreement inspectable: a reviewer can compare the output against the displayed corpus examples and grammar passages rather than treating the response as unsupported text. For this reason, the system should describe the rationale field as a *generated explanation of the candidate*, not as a verified derivation.

5.5. Worked execution example

To make the evidence trace concrete, we include a single command-line run with optional generation enabled:

```
python translate_rag.py \
  --text "buenos dias linda familia" \
  --k 8 \
  --grammar-sections 4 \
  --copilot
```

The run returned eight nearest-neighbor examples; the top items were two direct matches for “buenos dias” (IDs #2847 and #26607), followed by semantically related greeting or affective expressions (for example, “se amable tu” and “estas contento”). Grammar selection returned the sections “Los pronombres y prefijos personales,” “Los pronombres,” “El adjetivo,” and “Los sufijos determinativos.” Exact anchors were found for *linda* → *linda*, *dias* → *chonya*, and *familia* → *apüshi*.

Table 2 lists the retrieved examples from this run.

The same run selected four grammar sections (“Los pronombres y prefijos personales,” “Los pronombres,” “El adjetivo,” and “Los sufijos determinativos”) and detected three anchors: *linda* → *linda*, *dias* → *chonya*, and *familia* → *apüshi*.

The optional LLM generative AI produced the following JSON candidate:

```
{
  "traduccion_principal": "Watta'a maal\"u' ap\"ushi nutchimayaa",
  "traduccion_alternativa": "Wattachonya ap\"ushi nutchimayaa",
  "ejemplos_usados": ["#2847", "#26607", "#24103"],
  "confianza": 0.62
}
```

This example is included only to illustrate traceability (retrieval, anchors, grammar evidence, and generated candidate in one run). It is not reported as an evaluation result and must not be interpreted as evidence of general translation quality.

5.6. Evidence behavior and failure containment

The pipeline distinguishes evidence availability from translation success. If no semantic examples are returned, it does not attempt to construct a full prompt and instead reports that the vector database did not provide a result. If no exact anchors are found, the prompt explicitly reports their absence rather than fabricating a lexical match. If the optional generative AI is unavailable or unauthenticated, the web interface still returns the RAG examples, grammar passages, and prompt. This behavior allows the retrieval and grammar components to be assessed independently of an external API.

The current implementation does not automatically validate whether a generated Wayuunaiki string occurs in the corpus, follows a morphological paradigm, or agrees with every supplied grammar passage. Such checks are future work. A practical short-term safeguard is to present generated text as a “candidate translation” and to make the evidence trace more prominent than the generation panel in the user interface.

5.7. Interfaces

A Gradio interface exposes input controls for k , the number of grammar sections. It returns the prompt, generated response, retrieved examples, anchors, and grammar passages separately. A Model Context Protocol server exposes four tools—semantic example retrieval, anchors, grammar-section selection, and prompt construction—so an external agent can use the evidence without relying on the web interface.

6. Evaluation and Community Validation Protocol

6.1. Research questions

The evaluation will address the following questions:

- | | |
|------------|--|
| RQ1 | In a short speaker session, does the current prompt template plus traceable context produce evaluable candidate translations with useful explanations? |
| RQ2 | After prompt calibration, how do different LLM backends compare when retrieval context and prompt are held constant? |
| RQ3 | Which error types persist across models, especially for morphology, lexical choice, idiomaticity, and context? |
| RQ4 | Do speakers consider the displayed evidence sufficient to inspect, correct, or reject model outputs? |

Table 3

Planned two-session validation design. The exploratory results reported in Section 7 are discussed separately from the controlled model-comparison protocol.

Session/Condition	Scope and controls
S1	Prompt calibration with Copilot CLI over 8 fixed Spanish inputs. Objective: revise prompt wording and verify that traces are useful to the reviewer.
S2-M1	Model 1 with frozen retrieval context and frozen prompt version from S1.
S2-M2	Model 2 with the same frozen retrieval context and prompt.
S2-MN	Additional models under the same frozen context and prompt controls.

6.2. Two-session design

The validation is organized as two sequential sessions. Session 1 is a prompt-calibration session with a single generator (Copilot CLI). Session 2 is a model-comparison session where the retrieval evidence and prompt are frozen and only the LLM backend varies. This separation makes model-level differences interpretable and avoids mixing prompt-iteration effects with generator effects.

6.3. Speaker-centered validation

Validation will be conducted with bilingual Wayuunaiki speakers or qualified language professionals only after an appropriate collaboration, consent, and data-management process is in place. Recruitment criteria, number of participants, dialectal representation, compensation, and ethical-review requirements are intentionally marked as pending because they must be agreed with the collaborators rather than inferred by the research team.

Each evaluator should receive the Spanish source, a system candidate, and the visible evidence trace. In S2, candidate order should be randomized and anonymized by model identifier to reduce presentation bias. The protocol should collect:

- adequacy with respect to the Spanish input;
- grammatical acceptability in Wayuunaiki;
- naturalness and cultural/contextual appropriateness;
- usefulness and faithfulness of the evidence trace; and
- an error label and optional free-text correction.

A five-point ordinal scale is used for the first four dimensions, accompanied by categorical error analysis. For operational traceability, each evaluation row stores full prompt text, prompt hash, prompt version, retrieval-context hash, and model identifier in a local CSV file.

6.4. Session-specific protocol

The short-session protocol is intentionally pragmatic:

- S1 (about 1 hour): one evaluator, 8 phrases, one LLM (Copilot CLI), iterative prompt diagnostics, and final prompt freeze (e.g., `prompt_version=v1_final`).
- S2 (follow-up session): same 8 phrases, same RAG configuration, same frozen prompt text, and multiple LLMs compared under matched settings.
- S2 integrity check: for each phrase, `rag_config_hash` and `prompt_hash` must be identical across models.

Table 4

Draft speaker-centered rating rubric. Final wording and scale anchors require collaborator review.

Dimension	Question for the reviewer
Adequacy	Does the candidate preserve the intended meaning of the Spanish source?
Grammatical acceptability	Is the candidate well-formed in the relevant variety of Wayuunaiki?
Cultural/contextual appropriateness	Is the wording natural and appropriate for the stated context and audience?
Trace usefulness	Do the retrieved examples, anchors, and grammar passages help inspect or correct the candidate?

6.5. Automatic and qualitative analyses

If community-approved reference translations become available, BLEU and chrF may be computed as secondary descriptive metrics. The primary analysis remains speaker-centered and should report ordinal rating distributions, medians, and qualitative corrections. Automatic metrics must not replace speaker judgments. The team should also inspect retrieval failures, including semantically close but grammatically unsuitable examples and ambiguous anchors.

6.6. Rating instrument and analysis plan

Table 4 provides a draft rating rubric. It is not a finalized questionnaire: wording, language of administration, and scale anchors must be reviewed with the collaborators. The scale is included now to make the planned analysis falsifiable and to prevent post-hoc changes to the meaning of a “good” output.

For each dimension, a five-point ordinal response can range from “not acceptable/useful” to “fully acceptable/useful,” with an explicit “cannot judge” option. Reviewers should be able to flag a phrase as context-dependent rather than forcing an artificial binary judgment. Candidate order must be randomized within each item, and the interface should not reveal which condition produced a candidate. The primary analysis should report the distribution of ratings as well as an aggregate statistic; medians and ordinal mixed-effects models are possible options when the sample and study design support them. The final method must be selected before inspecting condition-level outcomes.

Corrections and comments should be analyzed separately from numeric ratings. At least two members of the research team, preferably including a qualified Wayuunaiki collaborator, can iteratively define categories from the reviewed outputs. Disagreements about the taxonomy should be documented rather than silently resolved. This qualitative record may reveal errors not captured by a single overall quality score, such as a translation that is structurally acceptable but uses an unsuitable lexical item for the intended context.

7. Results and Validation Status

A first qualitative validation was conducted with a native Wayuunaiki speaker on two small sets of phrases. The results presented below should be interpreted as exploratory and formative evidence, not as a conclusive evaluation of the system. Within the results of this evaluation, we identified some conditions where the pipeline is already useful and under what conditions it still fails when combining these elements.

To clarify the scope of the evidence, the present paper does not claim a full baseline or a completed ablation study. The architecture is compared here against a positive control consisting of direct corpus matches and against a compositional failure mode without direct matches. This is intentionally a limited evaluation design, not a complete model ablation. One inexpensive control remains within reach and would materially strengthen the paper: a retrieval-only condition versus retrieval plus grammar sections, to show whether grammar grounding contributes beyond nearest-neighbor retrieval. We did,

Table 5

Summary of the qualitative validation with Copilot CLI.

Subset evaluated	Adequacy	Grammar	Naturalness	Trace usefulness
Short phrases with direct textual corpus match ($n = 3$)	5.0	5.0	5.0	5.0
Compositional phrases without direct match ($n = 5$)	3.2	2.0	2.6	4.0

however, run the sensitivity sweep for the lexical-overlap weight λ in Eq. 3 described in Section 5.3; the current value of 0.01 is not empirically optimized but was validated post hoc as keeping the semantic term dominant, since the selected top-3 grammar sections already change for every tested query once $\lambda \geq 0.05$.

The session included eight evaluations in total using *Copilot CLI* as the sole generator. Three short phrases served as a positive control because they were found verbatim in the corpus. In those cases, the outputs obtained maximum scores across all four evaluation dimensions: adequacy, grammar, naturalness, and trace usefulness. This indicates that, when the input is already covered almost literally by the bilingual database, the pipeline acts reliably as an assisted retrieval mechanism.

The behavior clearly changed in the second set, composed of five longer and more compositional phrases. In this group, there were no direct textual matches with the corpus, and the scores dropped to 3.2 in adequacy, 2.0 in grammar, and 2.6 in naturalness, while trace usefulness remained relatively high (4.0). The difference between both subsets suggests that the system’s current strength does not yet lie in the flexible generation of new phrases, but rather in the reuse of patterns already observed in the corpus.

From a qualitative perspective, the performance drop should not be interpreted solely as a generator failure. The results point to a compounded problem: the system successfully retrieves useful evidence, but still struggles to recombine greetings, quantifiers, kinship relations, temporal expressions, and morphological markers into a single coherent output. In other words, the retrieval remains informative for the evaluator, but the final synthesis is not yet sufficiently stable from a morphosyntactic standpoint.

An additional finding from this session was the importance of dialectal and regional variation. During the review with the native speaker, it was discussed that several expressions can have different realizations depending on the tribe, the region, or the community of use. Therefore, an output should not always be simply classified as correct or incorrect. In future sessions, it will be methodologically preferable to distinguish between a clear error, an understandable variant but not preferred by the consulted community, and a fully acceptable variant for that community.

7.1. Exploratory Comparisons with Other Models

The notes from the second session also recorded informal comparisons with *Gemini* and, to a lesser extent, with *ChatGPT*. These observations do not yet constitute a controlled comparison between models, because the same presentation protocol was not experimentally fixed, nor was the model’s identity blinded. They should therefore be read as anecdotal, non-comparative evidence only. Even so, they are useful as an exploratory signal: for the five long phrases, it was noted that *Gemini 3.5* produced alternatives that were preferable for the evaluator, whereas *ChatGPT 5.5* was mentioned in four cases and showed a greater tendency to introduce additional words or out-of-context interpretations.

This finding supports the methodological decision to separate the evaluation into two stages. The first stage, already executed, serves to calibrate the prompt and understand the system’s limits with a single model. The second stage must fix the same RAG context, the same prompt version, and the same input phrases to compare the behavior of different LLMs solely. We avoid treating the informal cross-model notes as evidence of a final ranking; they are included only to motivate the controlled protocol and to document practical failure modes.

Table 6

Preliminary qualitative taxonomy derived from the first validation.

Category	Operational definition	Observed severity	Preliminary count
Lexical selection / referential quantification	The candidate uses a quantifier or a relational term that does not correspond to the type of referent or the intended meaning.	Medium-high	3
Unnatural word order	The phrase is partially understandable, but the surface word order does not match the evaluator's preferred form.	Medium	1
Verbal morphology/agreement	The selected verbal form or suffix does not correspond to the expected agreement in the evaluated context.	High	1
Semantic drift of communicative intent	The output alters the central communicative act of the question or adds an unrequested interpretation.	High	2
Dialectal or regional variation	There is more than one plausible realization, but the preference depends on the consulted community and not just on an abstract rule.	Cross-cutting	Observed in discussion

7.2. Qualitative Error Taxonomy

This coding was exploratory and allows for multiple labels per phrase; therefore, the counts should not be interpreted as mutually exclusive categories.

The reviewed cases show that the worst-performing dimension was not trace usefulness, but rather the transition from partial evidence to a well-formed sentence. For example, in “Necesito agua para todos” (I need water for everyone), the main problem was not retrieving the concept of *agua*, but choosing an appropriate quantifier for people rather than for objects. In “Que estan haciendo aqui en la casa” (What are they doing here in the house), the main observation was a verbal form not preferred by the evaluator, while in “En que trabajan mis tios” (What do my uncles do for work), a partial semantic drift of the question's intent was observed. Overall, these patterns suggest that the system's next iteration should prioritize lexical control, compositional templates, and more specific filtering of grammatical rules.

8. Discussion

The main advantage of the proposed architecture is not that retrieval guarantees correct translations. What the system does is create an auditable boundary between linguistic resources and generative behavior. A speaker might consider an example irrelevant, an anchor misleading, or a grammar excerpt inapplicable; these three decisions are useful observations for improving the system. In contrast, an end-to-end model that only exposes its final string makes it difficult to identify whether the error stemmed from data coverage, representation, prompt construction, or generation. Described below are the limitations of this approach.

There are several threats to the planned study's validity. *Construct validity* is limited because adequacy, grammar, naturalness, and cultural appropriateness are related but distinct constructs. A high rating in one does not imply a high rating in another. *Internal validity* may be affected if evaluators recognize repeated inputs, infer a condition from the interface, or receive candidate outputs in a fixed order.

Randomization and blinding of condition labels reduce, but do not eliminate, these risks. *External validity* is limited by the corpus domain, the selected phrases, the participant profiles, and potential dialectal variation. Therefore, the findings should be presented as evidence for the evaluated context and not as universal claims about all Wayuunaiki usage.

One point to keep in mind is that model updates or decoding parameter changes can alter outputs even when the prompt composition remains the same. The two-session design separates the prompt calibration in S1 from the model comparison in S2 with frozen context and prompt. This separation is necessary to interpret failures: an unsupported candidate in S2 should not be counted as a retrieval success just because the prompt contained relevant examples.

The work’s value depends on whether speakers consider the evidence and workflow useful, whether the governance is acceptable, and whether sustained collaboration produces corrections that the project is authorized to keep.

Beyond the classic threats to validity, the deployment path introduces operational risks that should guide future work packages. A first priority is *resource drift*: if corpus entries or grammar files are updated, previously reported traces might no longer be reproducible unless versions are frozen. A second priority is *coverage imbalance*: frequent conversational forms may be well-supported while domain-specific or culturally sensitive expressions remain underrepresented.

As future work, we also intend to explore more explicit dictionary-guided and reinforcement-learning strategies for translation control, following the line of work on Spanish–Wayuunaiki dictionary-guided fine-tuning and tool-augmented optimization [13]. In parallel, we plan to incorporate more systematic evaluation procedures that distinguish surface overlap from morphological adequacy, drawing on recent analyses comparing BLEU and chrF++ in extremely low-resource settings [14].

9. Ethical Scope, Data Governance, and Limitations

The prototype is not a certified translator, a linguistic authority, or a substitute for Wayuunaiki speakers. The use of grammar and bilingual dictionary materials does not by itself establish permission to deploy outputs commercially or to redistribute resources. Before releasing a public system, the project should document provenance, licenses, access restrictions, data retention, and procedures for correcting or removing entries. CARE principles provide a useful starting point for governance discussions [5].

Several technical limitations are immediate. Semantic similarity can retrieve an example that is topically relevant but syntactically inappropriate. Exact anchors may fail for inflectional variants or multiword expressions. Grammar ranking is based on embeddings and lexical overlap rather than a formal grammar or morphological parser. The LLM generative AI can hallucinate, ignore evidence, or produce unstable outputs across calls. Finally, the current pipeline does not resolve dialectal variation, register, or context without speaker input. These limitations motivate the planned ablation study and community validation.

10. Conclusion

This paper presented a traceable architecture for Spanish–Wayuunaiki translation assistance. The design combines local semantic retrieval, exact lexical anchors, targeted grammar selection, and an optional generative AI while exposing the evidence used for each response. Its central claim remains architectural, but the first native-speaker validation already provides a useful empirical distinction: the current system performs strongly when the corpus contains a direct match, yet remains unstable on longer compositional inputs that require lexical control and tighter grammatical recombination. The next research stage is therefore not an initial validation, but a more controlled expansion of the speaker-centered study together with a model-comparison phase under frozen RAG context and prompt configuration.

Declaration on Generative AI

During the preparation of this work, the authors used different LLM models through GitHub Copilot and Grammarly for grammar and spelling checks, some translations from Spanish to English, paraphrasing, and rewording. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

- [1] R. Negrette, I. Galpin, O. Garcia-Bedoya, A wayuunaiki translator to promote ethno-tourism using deep learning approaches, in: H. Florez, H. Gomez (Eds.), *Applied Informatics, Communications in Computer and Information Science*, Springer International Publishing, Cham, 2022, pp. 295–307. doi:10.1007/978-3-031-19647-8_21.
- [2] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive NLP tasks, in: *Advances in Neural Information Processing Systems* 33, 2020.
- [3] P. Koehn, R. Knowles, Six challenges for neural machine translation, in: *Proceedings of the First Workshop on Neural Machine Translation*, 2017, pp. 28–39. doi:10.18653/v1/W17-3204.
- [4] S. Bird, Decolonising speech and language technology, in: *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 3504–3519. doi:10.18653/v1/2020.coling-main.313.
- [5] S. R. Carroll, I. Garba, O. L. Figueroa-Rodríguez, J. Holbrook, R. Lovett, S. Materechera, M. Parsons, K. Raseroka, D. Rodriguez-Lonebear, R. Rowe, R. Sara, J. D. Walker, J. Anderson, M. Hudson, The CARE principles for indigenous data governance, *Data Science Journal* 19 (2020) 43. doi:10.5334/dsj-2020-043.
- [6] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using siamese BERT-networks, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019, pp. 3982–3992. doi:10.18653/v1/D19-1410.
- [7] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, M. Zhou, MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers, in: *Advances in Neural Information Processing Systems* 33, 2020.
- [8] K. Papineni, S. Roukos, T. Ward, W. Zhu, BLEU: a method for automatic evaluation of machine translation, in: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 311–318. doi:10.3115/1073083.1073135.
- [9] M. Popović, chrF: character n-gram F-score for automatic MT evaluation, in: *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 2015, pp. 392–395. doi:10.18653/v1/W15-3049.
- [10] M. Freitag, G. Foster, D. Grangier, V. Ratnakar, Q. Tan, W. Macherey, Experts, errors, and context: A large-scale study of human evaluation for machine translation, *Transactions of the Association for Computational Linguistics* 9 (2021) 1460–1474. doi:10.1162/tac1_a_00437.
- [11] R. J. N. Amaya, I. Galpin, O. Garcia-Bedoya, spanish-wayuunaki, 2021. URL: <https://osf.io/6kbze/>. doi:10.17605/OSF.IO/6KBZE, public corpus project; description updated 2022-09-03.
- [12] A. Garcia, sqlite-vec: A vector search sqlite extension, Software repository, 2026. URL: <https://github.com/asg017/sqlite-vec>, accessed: 2026-07-14.
- [13] M. Mosquera, M. Robles, J. Rodriguez, R. Manrique, Improving low-resource translation with dictionary-guided fine-tuning and RL: A spanish-to-wayuunaiki study, arXiv preprint, 2025. URL: <https://arxiv.org/pdf/2508.19481>.
- [14] S. Kumar, P. Jyothi, P. Bhattacharyya, Evaluating extremely low-resource machine translation: A comparative study of chrF++ and BLEU metrics, arXiv preprint, 2026. URL: <https://arxiv.org/abs/2602.17425>.