Grammar as Control: Modular Language Generation for the Long Tail

Ndapa Nakashole[1,2]
[1]University of California, San Diego
[2]Okalai AI
nnakashole@ucsd.edu

Abstract

Large language models (LLMs) can, in principle, bootstrap language technologies for long-tail languages due to their pattern recognition capabilities. Yet in practice, without structured guidance, they produce narrow, unrepresentative samples that fail to cover the morphosyntactic space of typologically underrepresented languages.

We propose *Modular Typology-Informed Generation* (mTIG), a prompting framework that transforms descriptive grammars into explicit *control mechanisms* that guide LLMs to generate typologically balanced synthetic data for downstream training. mTIG decomposes grammars into modular *grammar slices*, each targeting a specific morphosyntactic phenomenon (e.g., passive voice, causative morphology).

Across three low-resource languages, mTIG improves typological entropy by up to 19% and yields a *"student-beats-teacher"* effect, where distilled models outperform the source LLM by up to +20 chrF in machine translation. These findings show that *grammar-as-control* can construct training corpora wherever formal linguistic descriptions exist.

Grammar Slice (Control Unit)	Unguided Prompting	Grammar Dump	mTIG (Ours)
Verb Morphology			
Passive	○	◐	●
Causative	○	◐	●
Reciprocal	○	◐	●
Reversive	○	◐	●
Benefactive	○	◐	●
⋮	⋮	⋮	⋮

Table 1: **Reliability gap in grammar control for synthetic corpus generation.** Comparison of methods for realizing morphosyntactic phenomena in synthetic data. **Unguided prompting** fails to produce rare structures (○). **Grammar dump** suffers from larger context degradation, yielding inconsistent coverage (◐). **mTIG** (ours) ensures consistent (●) and distributionally balanced coverage.

1 Introduction

The success of large language models (LLMs) in high-resource languages has sparked interest in their potential to bootstrap NLP for the long tail of the world's languages (Tanzer et al., 2024; Zhang et al., 2024; Ramos et al., 2025). Yet how to effectively leverage LLMs for low-resource languages remains an open problem.

Synthetic Data in Low-Resource MT. Consider machine translation (MT), where data augmentation, such as back-translation and other forms of synthetic data generation, is a standard remedy in low-resource settings (Bertoldi and Federico, 2009; Bojar and Tamchyna, 2011; Caswell et al., 2019; Edunov et al., 2018; Guzmán et al., 2019; Sánchez-Martínez et al., 2020). A common approach is to use LLMs to translate high-resource corpora into the target language. However, this strategy breaks down when weak pretraining representations lead to low-quality, disfluent translations.

Limitations of Open-Ended Generation. An alternative is to use LLMs to generate synthetic parallel corpora, allowing the model to produce simpler sentences that are more likely to be translated accurately. However, open-ended prompting suffers from *mode collapse* (Jiang et al., 2025), producing repetitive structures that fail to cover the morphosyntactic space of typologically underrepresented languages (Joshi et al., 2020; Ponti et al., 2019). For example, a synthetic corpus may repeat a simple subject-verb-object (SVO) structure, such as "The [Animal] [Verb] the [Noun]", 10,000 times with different lexical items; while lexically diverse, it is a syntactically poor teacher, limiting downstream generalization to unseen morphosyntactic patterns.

Grammar-as-Control. Instead of relying on stochastic sampling to cover the morphosyntactic space, we aim to explicitly control the distri-

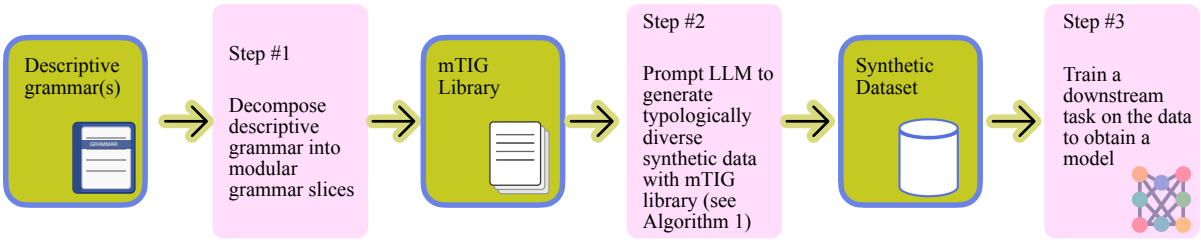

Figure 1: **The mTIG pipeline.** A descriptive grammar is decomposed into modular *grammar slices*, each targeting a specific morphosyntactic phenomenon(**Step 1**). These slices are used as executable prompts to guide an LLM in generating synthetic sentence pairs with controlled grammatical structure (**Step 2**). The resulting corpus exhibits diverse and balanced morphosyntactic coverage, and is used to train downstream models (e.g., for machine translation) (**Step 3**).

bution of generated sentences to ensure typological balance. To this end, we turn to *descriptive grammars*, which remain widely available even for under-resourced languages (Nordhoff and Hammarström, 2011), and ask how they can be made actionable for generation. Simply injecting entire grammars into prompts, a strategy we term *grammar dump*, is brittle, due to long instruction degradation in Transformer-based LLMs (Levy et al., 2024), as we confirm experimentally. Furthermore, existing grammar-aware MT approaches typically use grammars to improve sentence-level quality rather than ensuring representative *corpus-level coverage* (Tanzer et al., 2024; Zhang et al., 2024; Ramos et al., 2025).

The mTIG Framework. We introduce *Modular Typology-Informed Generation (mTIG)*, a prompting framework that decomposes a natural-language grammar into compact, human-readable *grammar slices*. As illustrated in Table 1, each slice acts as an executable prompt that steers the LLM to realize specific grammatical phenomena, thereby shaping the distributional structure of an entire synthetic corpus. Unlike token-level formal grammar-constrained decoding (Geng et al., 2023; Wang et al., 2023), mTIG uses descriptive grammars for *distributional control*. To ensure high data quality, we focus on a short-sentence generation regime (3–6 words) which are easier for LLMs to produce accurately for low-resource languages.

Contributions.

- We introduce *mTIG*, a modular prompting framework that uses descriptive grammars to steer LLMs toward typologically diverse generation.
- We demonstrate that mTIG shapes the *distributional structure* of synthetic corpora, improving normalized typological entropy by up to 19% over unguided prompting and grammar-dump baselines.
- Our experiments show that mTIG-generated data supports downstream MT with a *student-beats-teacher* effect, where distilled models outperform the source LLM by up to *+20 chrF*.
- We show that grammar slices transfer across related languages within a family, reducing the per-language engineering effort required to extend language technology to the long tail.

The project repository is available at `https://github.com/okalai-ai/mtig`.

2 Formalism and Framework

mTIG treats descriptive grammars as *control mechanisms* for generating synthetic corpora. Given a descriptive grammar G, mTIG decomposes G into a library of *grammar slices*, where each slice mTIG_j specifies constraints and instructions for realizing a particular morphosyntactic phenomenon (e.g., passive voice, superlatives, or clause linking). This enables *corpus-level control*, rather than relying on diversity to emerge from stochastic sampling.

Grammar Slice. Each slice is defined by a tuple: $\text{mTIG}_j = (\mathcal{G}_f^j, \mathcal{G}_i^j, \mathcal{T}^j, \mathcal{E}^j)$, where $\mathcal{G}_f^j$ encodes family-level grammar information on that slice, $\mathcal{G}_i^j$ specifies language-specific details, $\mathcal{T}^j$ provides the generation instruction, and $\mathcal{E}^j$ is a minimal few-shot example. Either grammar component may be empty (e.g., $\mathcal{G}_f^j = \emptyset$ or $\mathcal{G}_i^j = \emptyset$). For notational brevity, we define the joint grammar specification as $\mathcal{G}^j = \left(\mathcal{G}_f^j, \mathcal{G}_i^j\right)$.

Controlled Generation. The LLM is instructed to produce parallel English–target pairs to facilitate

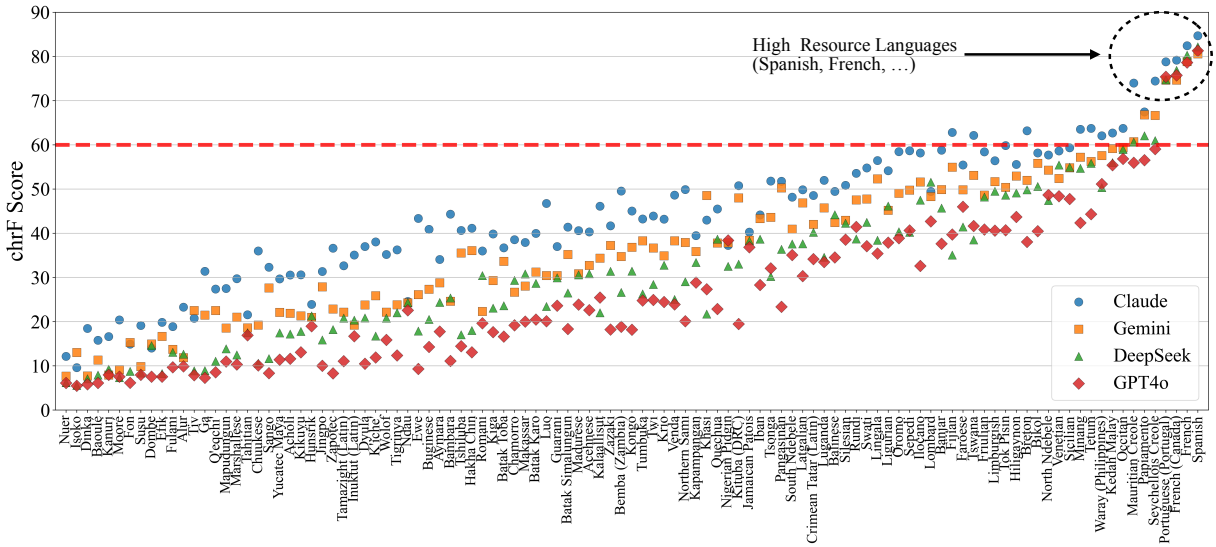

Figure 2: **Lexical performance gap in frontier LLMs.** Zero-shot English→target translation accuracy (chrF) on 107 languages from GATITOS. Most low-resource languages fall below a practical usability threshold (60 chrF; dashed line), indicating substantial gaps in lexical knowledge.

error analysis and post-hoc lexical correction. We denote the pair as $\mathbf{Z} = [X_{\text{en}}; X_{\text{trg}}]$, and the joint token sequence as:

$$\mathbf{Z} = (x_{\text{en},1}, \ldots, x_{\text{en},M}, x_{\text{trg},1}, \ldots, x_{\text{trg},N}) \quad (1)$$

where M and N are the respective lengths of the English and target strings. The generation probability is:

$$P(\mathbf{Z} \mid \text{mTIG}_j) = \prod_{k=1}^{M+N} P(z_k \mid z_{<k}, \mathcal{G}^j, \mathcal{T}^j, \mathcal{E}^j) \quad (2)$$

where z_k represents the k-th token in the joint sequence $\mathbf{Z}$. While slices are composable in principle, we treat them as independent control units in this work, and leave slice compositionality for future exploration.

Example mTIG slice (abridged)

Grammar slice: **Verb Morphology**: Active Verb, Positive Present, Personal Pronouns

Family Grammar ($\mathcal{G}_f$) FAMILY-LEVEL

- Subject–verb agreement is expressed via subject concords.
- Present tense may trigger final-vowel alternations.

Language-Specific Grammar ($\mathcal{G}_i$) LANGUAGE-SPECIFIC

- {LANGUAGE} present positive subject concords (illustrative): I: *ohandi*, You: *oto*,

Task Instruction ($\mathcal{T}$) GENERATION

Generate a short English sentence (3–6 words) whose meaning must be expressed in {LANGUAGE} using active verbs in positive present tense with personal-pronoun subject concords. Produce both the English sentence and its {LANGUAGE} realization.

Sample bilingual output (JSON):

```
{
  "english": "He is looking at the tree.",
  "{LANGUAGE}": "Ota tale omuti."
}
```

Figure 3: **Example mTIG grammar slice.** An abridged slice targeting active verbs in the *positive present tense* with *personal-pronoun subjects*.

Pipeline Overview. The workflow illustrated in Figure 1 proceeds in three stages: (1) decomposing a grammar G into a slice library $\mathcal{S}$; (2) generating the synthetic corpus; and (3) training downstream models. We detail the generation procedure (Step 2) in the following subsections. Figure 3 provides an abridged example of a single mTIG slice, illustrating the family grammar, language-specific grammar, generation instruction, and sample output.

2.1 Lexical Consistency

While mTIG provides grammatical control, lexical validity is not guaranteed by grammar alone as LLMs do not reliably produce correct words for low-resource languages. We confirm this *lexical gap* in frontier LLMs in our investigation of English→target word translation accuracy across 107 languages from the GATITOS benchmark (Jones et al., 2023), where most languages fall below a practical usability threshold (60 chrF), see Figure 2. To mitigate the lexical gap, we introduce a post-generation *lexical consistency operator* Φ_{lex}, applied whenever a bilingual lexicon $\mathcal{L}$ is available:

$$\mathbf{Z}_{\text{final}} = \Phi_{\text{lex}}(\mathbf{Z}_{\text{raw}}) = \text{LLM-Edit}(\mathbf{Z}_{\text{raw}}, \mathcal{T}_{\text{edit}}, \mathcal{L}_i), \quad (3)$$

where $\mathcal{L}_i$ denotes the subset of the target-language lexicon relevant to the instance, and $\mathcal{T}_{\text{edit}}$ instructs the model to replace out-of-lexicon or hallucinated

forms while preserving the grammatical configuration established by the slice. Implementation details and prompts for the lexical operator are provided in Appendix B.2.

2.2 Semantic Conditioning

To prevent topic variety in the synthetic corpus, we introduce a *semantic conditioning operator* Ψ_{sem}. This operator adds a topic specification to the task instruction $\mathcal{T}$:

$$\mathcal{T}' = \Psi_{\text{sem}}(\mathcal{T}, \theta), \tag{4}$$

where θ is a topic sampled from a curated set Θ of 21 diverse topics (e.g., *family*, *healthcare*, *agriculture*). For example, an unconditioned instruction to *generate a sentence in the passive voice* is transformed into *generate a sentence in the passive voice related to healthcare*. To ensure fair comparison across methods, semantic conditioning is applied identically to all baselines. The complete list of topics is provided in Table 4 (Appendix C.1).

2.3 Lexical Coverage and Model Capacity

While the lexical consistency operator (§2.1) enforces the validity of generated word forms, it does not guarantee that the resulting corpus realizes the full breadth of the available lexicon $\mathcal{L}$. In practice, even lexically corrected generation tends to overrepresent frequent items while omitting rare words. For the 1.3B-parameter student models targeted in this work, such lexical gaps are particularly problematic: unlike frontier LLMs, compact models cannot reliably translate words they have never observed during training.

To avoid wasting available lexical supervision and to achieve broad lexical coverage without the computational cost of exhaustive slice–word pairings, we implement a *targeted sampling strategy*. For any word in $\mathcal{L}$ that remains absent from the core corpus $\mathcal{C}_{\text{core}}$, we uniformly sample a grammar slice mTIG_j and generate a small number of sentences ($k = 5$) explicitly constrained to include that word. This procedure ensures that the student model is exposed to the full target vocabulary across diverse typological contexts, while avoiding the combinatorial explosion of a full *slice* $\times$ *lexicon* cross-product.

The complete mTIG procedure, is detailed in Algorithm 1. To ensure high data quality, our generation is scoped to a short-sentence regime (3–6 words) which are easier for LLMs to produce accurately for low-resource languages.

Algorithm 1 mTIG corpus generation procedure

Require: mTIG library $\mathcal{S} = \{\text{mTIG}_j\}_{j=1}^{J}$, topic set $\Theta = \{\theta_1, \ldots, \theta_T\}$, bilingual lexicon $\mathcal{L}$ (optional), lexical edit instruction $\mathcal{T}_{\text{edit}}$

$\mathcal{C}_{\text{core}} \leftarrow \emptyset$ ▷ Initialize core corpus
// Core mTIG generation
for each $\textbf{mTIG}_j = (\mathcal{G}^j, \mathcal{T}^j, \mathcal{E}^j) \in \mathcal{S}$ **do**
  for each topic $\theta \in \Theta$ **do**
    $\mathcal{T}' \leftarrow \Psi_{\text{sem}}(\mathcal{T}^j, \theta)$
    $\mathbf{Z}_{\text{raw}} \leftarrow \text{LLM-Gen}(\mathcal{G}^j, \mathcal{T}', \mathcal{E}^j)$
    if $\mathcal{L}$ available **then**
      $\mathbf{Z}_{\text{final}} \leftarrow \text{LLM-Edit}(\mathbf{Z}_{\text{raw}},\ \mathcal{T}_{\text{edit}},\ \mathcal{L})$
    else
      $\mathbf{Z}_{\text{final}} \leftarrow \mathbf{Z}_{\text{raw}}$
    end if
    $\mathcal{C}_{\text{core}} \leftarrow \mathcal{C}_{\text{core}} \cup \mathbf{Z}_{\text{final}}$
  end for
end for
// Lexicon-complete augmentation (optional)
$\mathcal{C}_{\text{lex}} \leftarrow \emptyset$
if $\mathcal{L}$ available **then**
  for each lexeme $w \in \mathcal{L}$ **absent from** $\mathcal{C}_{\text{core}}$ **do**
    Sample mTIG-k uniformly from $\mathcal{S}$
    Augment instruction to require usage of w
    $\mathbf{Z}_w \leftarrow \text{LLM-Gen}(\mathcal{G}^k, \mathcal{T}^k, w)$
    $\mathcal{C}_{\text{lex}} \leftarrow \mathcal{C}_{\text{lex}} \cup \mathbf{Z}_w$
  end for
end if
return $\mathcal{C} = \mathcal{C}_{\text{core}} \cup \mathcal{C}_{\text{lex}}$

3 Metrics for Typological Balance

To evaluate descriptive grammars as explicit control mechanisms, we quantify how effectively a generation strategy distributes outputs across grammatical phenomena, a property we call *typological balance*. Let $\mathcal{S} = \{1, \ldots, J\}$ denote the inventory of grammar slices. For any method m, let $n_j^{(m)}$ be the number of generated sentences instantiating slice j. These counts define a smoothed categorical distribution:

$$p_j^{(m)} = \frac{n_j^{(m)} + \alpha}{\sum_{k=1}^{J} (n_k^{(m)} + \alpha)}, \tag{5}$$

where $\alpha = 1$ is a Laplace smoothing constant.

Typological Entropy. We quantify distributional uniformity via normalized entropy:

$$\mathcal{H}_{\text{norm}}(p^{(m)}) = \frac{-\sum_{j=1}^{J} p_j^{(m)} \log p_j^{(m)}}{\log J} \in [0, 1]. \tag{6}$$

$\mathcal{H}_{\text{norm}} = 1$ corresponds to a uniform distribution over slices, indicating maximal control over morphosyntactic variety.

Grammar Coverage. Entropy does inform us whether all regions of the typological space are

populated. A method could achieve high entropy by generating many sentences for a subset of slices while leaving others empty. We therefore report *Coverage@k*, the proportion of slices represented by at least k examples:

$$\text{Cov@}k = \frac{1}{J}\sum_{j=1}^{J} \mathbb{1}[n_j^{(m)} \geq k]. \quad (7)$$

Sentence Uniqueness. A method may cover many slices while generating near-duplicate sentences. We therefore measure the proportion of unique English source sentences:

$$\text{Unique}(m) = \frac{|\{\text{unique } X_{\text{en}} \in \mathcal{C}^{(m)}\}|}{|\mathcal{C}^{(m)}|}. \quad (8)$$

While grammar dump baselines may achieve moderate coverage, they often exhibit low uniqueness; mTIG is designed to jointly optimize coverage, entropy, and uniqueness.

4 Case Study on Bantu Languages

We now demonstrate how the mTIG framework instantiates concretely within a real linguistic family. Specifically, we focus on the Bantu language family, which comprises over 300 languages and nearly half a billion speakers (Bleek, 1851; Guthrie, 1967; Maho, 2009). Despite being low-resource in digital corpora, Bantu languages are extensively documented through descriptive grammars and lexicons, mostly via colonial-era linguistic fieldwork and missionary efforts.

We constructed a *Bantu mTIG library of 51 grammar slices* covering core morphosyntactic phenomena. Family-level structures were derived from Guthrie's *Lingala* grammar (Guthrie, 1935), with language-specific examples obtained from Oshikwanyama resources (Crane et al., 2004). Figure 4 summarizes the typological space covered by the library, with each leaf corresponding to a single grammar slice; representative per slice example sentences are provided in Table 5 (Appendix D).

Development required approximately *32 hours of active effort*. This represents a one-time cost amortized across related languages. Notably, mTIG does not require formal linguistic training, but rather the ability to interpret descriptive grammar resources, as in Tanzer et al. (2024).

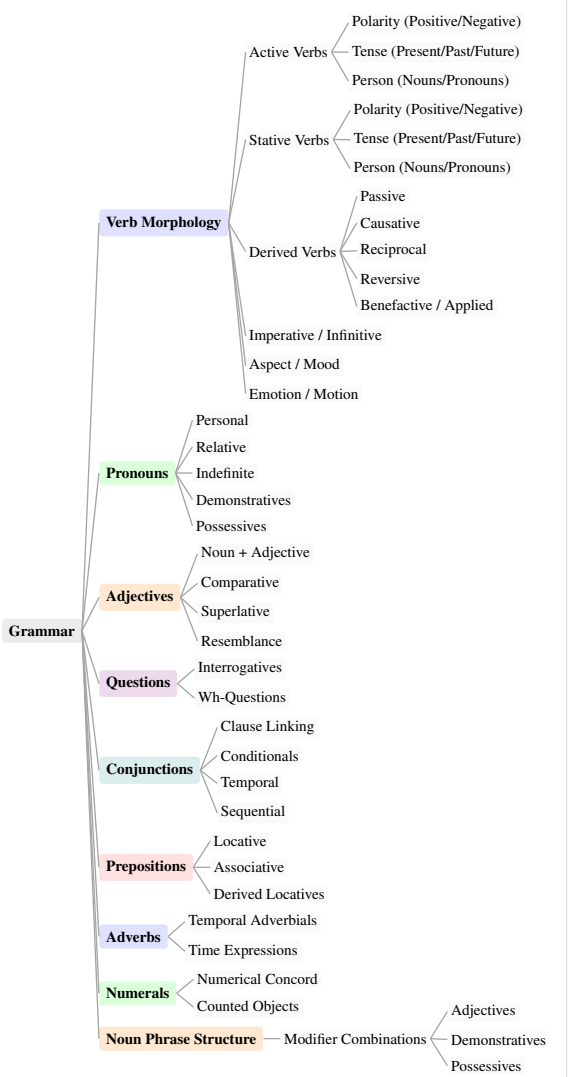

Figure 4: **Typological structure of the Bantu mTIG library.** A hierarchical organization of 51 grammar slices used as control units for synthetic data generation. Each leaf node corresponds to a distinct morphosyntactic category that can be targeted during generation to ensure broad and balanced coverage.

5 Experiments

5.1 Experimental Setup

We evaluate mTIG using both *intrinsic* and *extrinsic* measures. Intrinsic evaluation assesses typological balance, how evenly generation is distributed across grammatical slices for which we use entropy and coverage metrics defined in §3. Extrinsic evaluation measures downstream MT performance when models are fine-tuned on mTIG-generated data, for which we use the well-established chrF (Popović, 2015) metric.

Languages. We evaluate three Bantu languages with increasing distance from the development

Method	$\mathcal{H}$	$\mathcal{H}_{\text{norm}}$	mTIG Gain (%)
Baseline Prompt	2.889	0.739	+18.5%
Grammar Dump	2.996	0.766	**+14.2%**
mTIG (ours)	**3.423**	**0.875**	—

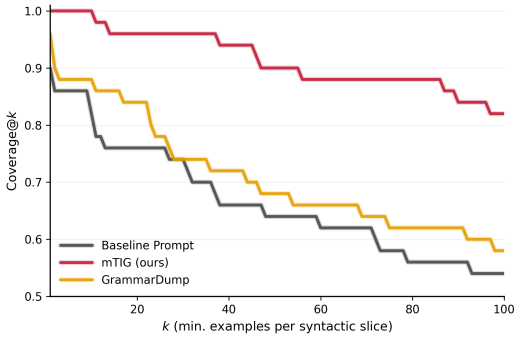

Figure 5: **Typological diversity and coverage (Gemini). Top**: entropy $\mathcal{H}$ and normalized entropy ($\mathcal{H}_{\text{norm}}$). **Bottom**: Coverage@$k$ across generation strategies.

language whose examples are included in the mTIG library derived from Lingala (Guthrie, 1935): *Kwanyama* (development language), *Ndonga* (closely related and mutually intelligible), and *Rukwangali* (related but not mutually intelligible). For all settings, we generate synthetic parallel data under matched budgets (321,300 sentences).

Generator LLMs. We use **Claude** opus-4-5 Anthropic (2025), **Gemini** 2.0-flash[1]Anil et al. (2023) , and **GPT** 4o,OpenAI (2023), selected to span a range of multilingual competence in low-resource settings (Figure 2). Claude and Gemini represent the strongest performers on low-resource languages, while GPT serves as generator with comparatively weaker low-resource capabilities. The Prompts for mTIG generation and all baselines are provided in Appendix B and Appendix E.

Based on preliminary experiments and findings by Jiang et al. (2025), which show that high-stochasticity decoding ($T = 1.0$) does not necessarily mitigate repetition, we set the temperature to 0.7 for Claude and use default parameters for all other models to manage computational costs.

5.2 Intrinsic Evaluation: Typological Balance

We quantify typological balance by assigning generated sentences to their respective grammar slices using an LLM-based classifier (Appendix F). While automatic classification is inherently approximate, it provides a consistent and scalable proxy for comparing generation strategies. A detailed error analysis and classifier performance metrics are provided in Appendix I.2.

[1]Attempts to use the strongest available Gemini model (gemini-3-pro-preview) failed due to API rate limits on that model.

Method	Total Gen.	Unique EN Sents	Unique (%)
Baseline Prompt	321,300	147,522	45.9
Grammar Dump	321,300	90,018	28.0
mTIG (ours)	321,300	**219,777**	**68.4**

Table 2: **Mode collapse vs. generation diversity (Gemini).** At a fixed generation budget, mTIG produces over twice as many unique sentences as the grammar dump baseline.

Figure 5 summarizes the entropy and coverage results. For the 51-slice inventory, the theoretical maximum unnormalized entropy is $\log 51 \approx 3.93$. mTIG achieves substantially higher entropy than both baselines, indicating a more uniform distribution across grammatical categories. Notably, the *Grammar Dump* baseline, which provides the full 68-page descriptive grammar (Crane et al., 2004) in context—yields only marginal improvements over unguided prompting. This suggests that *large context alone is insufficient* to overcome the strong frequency biases of LLM generation.

Coverage curves (Figure 5, bottom) further highlight this gap. mTIG covers a broader typological space across all thresholds k: nearly 90% of slices receive at least 100 realizations, compared to approximately 60% for both baselines. These trends are consistent across Claude and GPT-4o (Appendix G).

Repetition and Mode Collapse. While the *Grammar Dump* baseline achieves slightly broader coverage than unguided prompting, it creates severe mode collapse (Table 2). For Gemini, uniqueness drops to **28%** under grammar dumping, compared to **46%** for the baseline and **68%** for mTIG. This indicates that large monolithic instructions reduce generation diversity due to repeated large context chunks.

5.3 Extrinsic Evaluation: Machine Translation

We evaluate whether typological control translates into downstream gains by training MT models on synthetic corpora. We compare mTIG against unguided prompting, grammar dump baselines, zero- and few-shot LLM baselines (Appendix H.3), and **NLLB-200** (1.3B) (Bapna et al., 2022). We also include a variant fine-tuned on Bible data (*Bible-NLLB*). Since NLLB does not natively support our

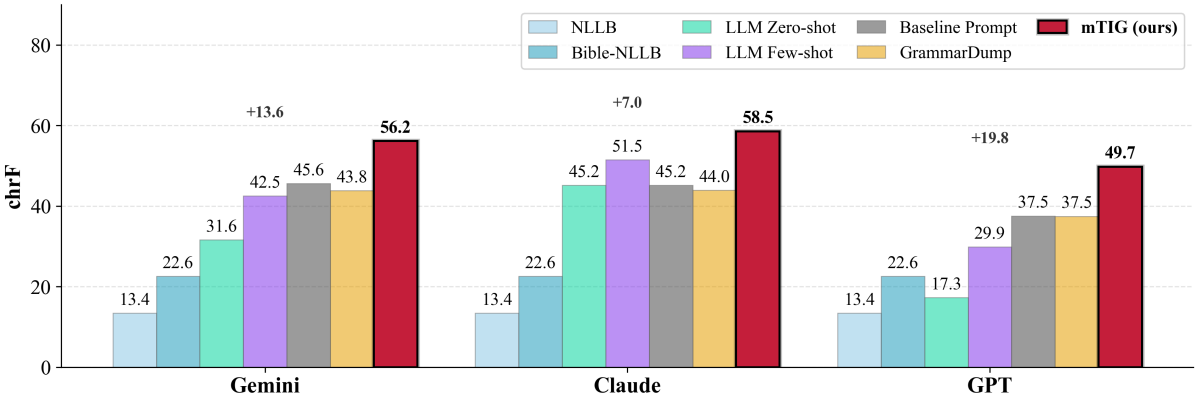

Figure 6: **Main translation results on the development language.** Models trained on mTIG-generated data outperform all baselines, including zero- and few-shot LLM prompting, across generators.

target languages, we initialize all experiments using the `umb_Latn` (Umbundu) tokenizer, which yielded the best performance on a held-out development set. We mixed the synthetic data with Bible data for the development language, Kwanyama. This procedure was applied consistently across all methods to ensure a fair comparison. It was limited to Kwanyama because Bible data was not available for the other languages.

We report chrF (Popović, 2015), as it is more robust for morphologically rich languages (Kocmi et al., 2021; Bapna et al., 2022; Freitag et al., 2022; Adelani et al., 2022). Hyperparameters and training details are provided in Appendix H.

Main MT Results. We translated a Kwanyama test set (833 sentences; details in Appendix H.4), which we release publicly and which is comparable in scale to FLORES-200 devtest sets.

As shown in Figure 6, MT models fine-tuned on mTIG-generated data consistently outperform all baselines. Across generators, mTIG-trained models achieve substantial gains over both unguided prompting and grammar dump baselines, improving chrF by roughly **10–15 points** depending on the source LLM. Gains are observed even when synthetic data is generated by comparatively weaker multilingual models such as GPT-4o, where mTIG still substantially improves translation quality.

Notably, mTIG shows a clear *student-beats-teacher* effect: distilled MT models outperform the zero- and few-shot translation performance of their source LLMs. This gap is largest for GPT-4o, reaching up to **+19.8 chrF** over few-shot prompting. This result suggests that explicit grammatical control combined with lexical grounding can compensate for missing low-resource knowledge in the generator, even when the generator is weak in the target language.

Cross-Language Library Reuse. We evaluate transferability by applying the Kwanyama mTIG library to Ndonga and Rukwangali without additional annotation (Appendix H.2). Test sets are drawn from public-domain Peace Corps materials.[2]

As shown in Figure 7, mTIG yields consistent gains over unguided prompting using Gemini-generated data: **+7.8** chrF for Ndonga and **+5.4** chrF for Rukwangali. Absolute performance varies with typological distance and underlying LLM knowledge, but gains persist across languages. Ndonga benefits more strongly, consistent with its closer relationship to Kwanyama, while Rukwangali remains constrained by limited LLM coverage yet still shows substantial improvement. Grammar dump prompting is omitted for these languages due to its substantially higher token cost.

5.4 Ablations: The Lexical Bottleneck

We decompose the contributions of mTIG by incrementally adding control layers to a *syntax-only* baseline (Figure 8). While grammatical control establishes a foundation for structural diversity, the largest performance gains arise from enforcing lexical validity (Φ_{lex}) and targeted lexical coverage. This suggests that, for frontier LLMs in low-resource settings, the dominant failure mode is not grammatical reasoning but insufficient lexical knowledge.

To contextualize this *lexical bottleneck*, we analyze the relationship between lexical compe-

[2]From `https://www.livelingua.com/peace-corps/Ndonga/te-ti.pdf` we extracted 136 sentences for Ndonga, and from `https://www.livelingua.com/peace-corps/Kwangali/Rukwangali.pdf` we extracted 209 sentences for Rukwangali.

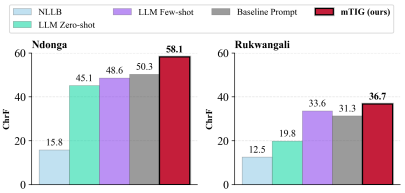

Figure 7: **Cross-lingual portability of mTIG slices (Gemini).** The same modular library improves MT performance (chrF) for related languages.

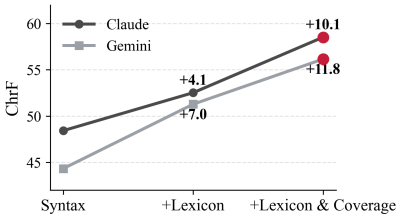

Figure 8: **Ablation of mTIG components.** Stepwise chrF gains show additive contributions from grammar control, lexical consistency, and targeted lexical coverage.

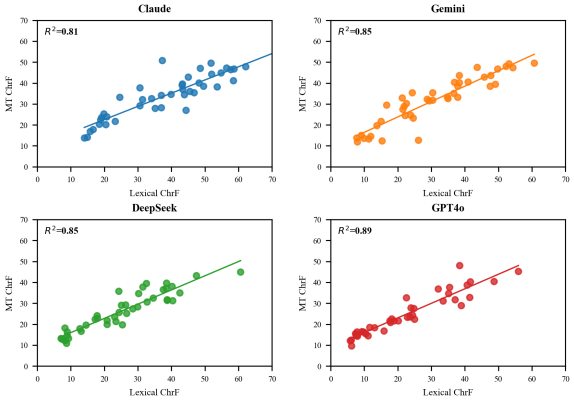

Figure 9: **Lexical knowledge as a translation bottleneck.** Lexical chrF (GATITOS) correlates strongly with MT performance ($R^2 = 0.81$–0.89) across 42 low-resource languages, highlighting lexical coverage as a key constraint on MT quality.

tence (GATITOS chrF) and downstream MT quality (SMOL) across 42 languages. As shown in Figure 9, lexical accuracy explains most of the variance in translation performance ($R^2 = 0.81$–0.89). Together, these results motivate mTIG's dual-track design: grammar slices ensure the structural breadth required for generalization, while lexical grounding converts that structure into high-fidelity supervision. Without both, synthetic corpora remain either grammatically repetitive or lexically hallucinated. This failure mode is reflected in the error analysis (§5.5), where remaining errors primarily involve incorrect verb allomorphs and other lexicalized exceptions.

5.5 Error Analysis

To characterize mTIG failure modes, we manually inspect 200 sentences across 20 grammar slices (Table 3) for the development language, using Claude as the generator due to its strong overall performance. Errors are predominantly systematic rather than stochastic, and center on lexical realization rather than morphosyntactic structure. Key issues include: (i) incorrect allomorph selection under specific tense/aspect constraints; (ii) overgeneralization of copular or reciprocal patterns; and (iii) failures in lexicalized exceptions (e.g., irregular passive forms).

Notably, high-level constraints, including word order, agreement, and slice-specific structure, are consistently respected. This reflects the ablation results in §5.4, where adding lexical control yields large performance gains, and suggests that the remaining errors primarily reflect gaps in the generator's underlying lexical knowledge rather than failures of grammatical control. Extended analysis and examples are provided in Table 7 (Appendix I.1).

6 Related Work

Grammar-Informed Generation. Prior work has used *formal grammars*, such as Backus–Naur Form (BNF), to impose hard syntactic constraints on LLM decoding in structured prediction tasks such as code generation and data-to-text generation (Shin et al., 2021; Scholak et al., 2021; Geng et al., 2023; Wang et al., 2023). These approaches constrain decoder outputs at inference time. In contrast, mTIG uses *descriptive grammars* to control what an LLM is prompted to generate, shaping corpus-level statistics rather than individual decoding paths.

Related work has also incorporated descriptive grammatical knowledge into LLM prompts for low-resource language tasks. Tanzer et al. (2024) show that grammar descriptions can improve translation quality for Kalamang, and subsequent work injects glosses or morphologically analyzed text to guide sentence-level MT (Zhang et al., 2024; Ramos et al., 2025). However, these methods treat grammar as auxiliary background knowledge to improve fidelity, rather than as an explicit control mechanism for systematic corpus-level coverage.

Our prior work leveraged typological structure to improve multilingual modeling through morphology-aware representations and routing mechanisms (Nakashole, 2025). However, that

Category	Observed Pattern
Verb morphology	Incorrect tense/aspect allomorphs
Constructions	Copular / reciprocal misuse
Derivation	Overapplication of causatives
Lexicon	Verb-specific passive errors

Table 3: Systematic error patterns in mTIG generation (full examples in Table 7 in Appendix I.1).

work focused on model adaptation rather than data generation.

Synthetic Data for Low-Resource MT. Synthetic parallel data is central to low-resource MT, most commonly via back-translation, where monolingual target text is translated into the source language, along with extensions such as tagging, copy-based augmentation, and iterative refinement (Bertoldi and Federico, 2009; Bojar and Tamchyna, 2011; Sennrich et al., 2016; Edunov et al., 2018; Caswell et al., 2019; Marie et al., 2020; Hoang et al., 2018). In these approaches, grammatical coverage emerges implicitly from corpus frequency and model bias. In extremely low-resource settings, back-translation is often not an option due to the lack of a representative monolingual corpus.

Recent work has also explored prompt engineering strategies, such as demonstrations and iterative revision, for synthetic data generation in low-resource settings, showing that careful prompt design can improve downstream performance (Anikina et al., 2025). However, these methods still rely on stochastic diversity, treating coverage as an emergent property rather than an explicit objective. mTIG is orthogonal to these approaches and could be combined with such techniques for improved performance.

7 Discussion

Our results position mTIG as a scalable mechanism for grammar-based synthetic data generation, but they also raise broader questions about the trajectory of low-resource language technology. We briefly discuss three.

1/3) What will it take to reach usable MT for the long tail? Despite clear gains from mTIG, absolute chrF scores remain below the threshold typically associated with practical usability. This gap is not unique to our setting: even for mid-resource languages such as Lingala and Tswana, frontier LLMs rarely exceed 50–55 chrF, and when we evaluated NLLB, arguably the strongest low-resource MT system to date, on Swahili Bible data, it plateaus near 55 chrF. Reaching usable quality is therefore a challenge that extends well beyond the extreme long tail. Closing the remaining gap likely requires a combination of larger bilingual lexicons, generation beyond the short-sentence regime, and some form of human-in-the-loop refinement. We believe mTIG provides a promising foundation for this next phase.

2/3) How does grammar-based synthetic data relate to model scaling? mTIG introduces a complementary axis of scaling: rather than only increasing data volume in a language, it increases the *coverage* of grammatical phenomena under a fixed budget. In our experiments, generation budgets are held constant at roughly 300k examples to enable fair comparison, leaving open how mTIG behaves at larger scales. We expect performance to scale along multiple interacting axes: corpus size, sentence length, slice granularity, and lexical coverage, with non-trivial trade-offs among them. Whether mTIG continues to improve at million-sentence scales, and how its returns compare to conventional data scaling, remains an open empirical question.

3/3) Can LLMs slice grammars themselves? The Bantu mTIG library required roughly 32 hours of manual effort to construct from descriptive grammars. While this is a one-time cost amortized across related languages, extending mTIG to hundreds of long-tail languages will require partial automation, and LLMs themselves may be a natural tool for this task. The question of whether LLMs can effectively decompose descriptive grammars into modular, typologically coherent slices is an intriguing one. If successful, it would enable a more scalable path to grammar-as-control across the long tail.

8 Conclusion

While frontier LLMs possess substantial latent knowledge of linguistic structure, unguided prompting and monolithic grammar dumps fail to reliably elicit it. Our results demonstrate that transforming descriptive grammars into active control units, thus framing grammar-as-control, is a promising step toward scalable data generation for the long tail of the world's languages.

Limitations

Dependence on Minimal LLM Linguistic Competence. mTIG assumes that the underlying LLM can reliably follow symbolic instructions and reproduce provided lexical and morphological forms. If an LLM has no familiarity with a target language, grammar control alone may be insufficient to yield fluent or well-formed output.

As multilingual LLM coverage improves, these requirements are expected to become easier to satisfy for lower-resource languages.

Short-Sentence Generation Regime. Our experiments focus on short sentences (3–6 words), while this design choice enables precise control and clean evaluation, it limits conclusions about long-form generation. Future work is needed to assess how mTIG scales to extended discourse, richer semantics, and complex clausal embedding.

Limited Compositionality of Grammar Slices. The current mTIG library treats grammar slices primarily as independent control units. While we support limited composition (e.g., Tense + Polarity), we do not yet model the full combinatorial space of interacting morphosyntactic features, such as stacked verbal extensions in Bantu languages (e.g., Passive + Causative + Reciprocal). Systematic multi-slice composition remains an important direction for future work.

Computational Cost. LLM-based generation is inference-intensive and may incur substantial computational and financial cost, particularly when producing large synthetic corpora. While mTIG improves data efficiency by enforcing structured coverage, generation remains bounded by the cost profile of the underlying LLM.

Potential Risks

Cultural Flatness. mTIG is designed to control grammatical and lexical structure, not to model culturally situated meaning. As a result, the generated corpora are *culturally flat*: sentences are short, generic, and largely stripped of pragmatic, sociocultural, and discourse-specific content. While this abstraction is useful for isolating morphosyntactic phenomena, it cannot substitute for culturally rich data reflecting local norms, practices, or communicative styles.

In practical deployments, mTIG-generated data should therefore be combined with some amount of culturally grounded material.

Bias Amplification. LLMs may encode and amplify biases present in their training data, which can manifest in synthetic data generation. mTIG does not inherently mitigate biases. Careful evaluation and filtering of generated data are necessary to identify and address potential biases, particularly in sensitive applications.

Acknowledgements

The author thanks the anonymous reviewers for their constructive feedback on an earlier version of this paper.

References

David Ifeoluwa Adelani, Jesujoba Oluwadara Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, Ernie Chang, Tajuddeen Gwadabe, Freshia Sackey, Bonaventure F. P. Dossou, Chris Emezue, Colin Leong, Michael Beukman, Shamsuddeen H. Muhammad, Guyo D. Jarso, Oreen Yousuf, Andre N. Niyongabo Rubungo, Gilles Hacheme, Eric Peter Wairagala, Muhammad Umair Nasir, Benjamin A. Ajibade, Tunde Oluwaseyi Ajayi, Yvonne Wambui Gitau, Jade Abbott, Mohamed Ahmed, Millicent Ochieng, Anuoluwapo Aremu, Perez Ogayo, Jonathan Mukiibi, Fatoumata Ouoba Kabore, Godson Koffi Kalipe, Derguene Mbaye, Allahsera Auguste Tapo, Victoire M. Memdjokam Koagne, Edwin Munkoh-Buabeng, Valencia Wagner, Idris Abdulmumin, Ayodele Awokoya, Happy Buzaaba, Blessing Sibanda, Andiswa Bukula, and Sam Manthalu. 2022. A few thousand translations go a long way! leveraging pre-trained models for African news translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3053–3070, Seattle, United States. Association for Computational Linguistics.

Tatiana Anikina, Jan Cegin, Jakub Simko, and Simon Ostermann. 2025. A rigorous evaluation of LLM data generation strategies for low-resource languages. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8293–8314, Suzhou, China. Association for Computational Linguistics.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds,

Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. 2023. Gemini: A family of highly capable multimodal models. *CoRR*, abs/2312.11805.

Anthropic. 2025. Claude opus 4-5. https://www.anthropic.com/.

Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, et al. 2022. Building machine translation systems for the next thousand languages. *arXiv preprint arXiv:2205.03983*.

Nicola Bertoldi and Marcello Federico. 2009. Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 182–189, Athens, Greece. Association for Computational Linguistics.

Wilhelm Heinrich Immanuel Bleek. 1851. *De nominum generibus linguarum Africae Australis, Copticae, Semiticarum aliarumque sexualium*. Dissertation, Universität zu Bonn.

Ondřej Bojar and Aleš Tamchyna. 2011. Improving translation model by monolingual data. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 330–336, Edinburgh, Scotland. Association for Computational Linguistics.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63, Florence, Italy. Association for Computational Linguistics.

Isaac Caswell, Elizabeth Nielsen, Jiaming Luo, Colin Cherry, Geza Kovacs, Hadar Shemtov, Partha Talukdar, Dinesh Tewari, Baba Mamadi Diane, Koulako Moussa Doumbouya, Djibrila Diane, and Solo Farabado Cissé. 2025. Smol: Professionally translated parallel data for 115 under-represented languages. *Preprint*, arXiv:2502.12301.

Thera Marie Crane, Karl Lindgren-Streicher, and Andy Wingo. 2004. Hai ti! a beginner's guide to oshikwanyama.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

ELCIN Church. 1996. *English–Ndonga Dictionary*. Oniipa Printing Press.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. 2023. Grammar-constrained decoding for structured NLP tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10932–10952, Singapore. Association for Computational Linguistics.

Malcolm Guthrie. 1935. Lingala grammar and dictionary. *(Lingala)*.

Malcolm Guthrie. 1967. Comparative bantu. an introduction to comparative linguistics and the prehistory of the bantu languages.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. 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 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Liwei Jiang, Yuanjun Chai, Margaret Li, Mickel Liu, Raymond Fok, Nouha Dziri, Yulia Tsvetkov, Maarten Sap, and Yejin Choi. 2025. Artificial hivemind: The open-ended homogeneity of language models (and beyond). In *NeurIPS 2025 Datasets and Benchmarks Track*.

J.K. Kloppers. 1994. *English–Rukwangali Dictionary*. Gamsberg Macmillan Publishers (Pty.) Ltd.

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.

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.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. 2024. Same task, more tokens: the impact of input length on the reasoning performance of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15339–15353, Bangkok, Thailand. Association for Computational Linguistics.

Jouni Filip Maho. 2009. NUGL Online: The online version of the New Updated Guthrie List, a referential classification of the Bantu languages.

Benjamin Marie, Raphael Rubino, and Atsushi Fujita. 2020. Tagged back-translation revisited: Why does it really work? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5990–5997, Online. Association for Computational Linguistics.

Ndapa Nakashole. 2025. Typology-guided adaptation in multilingual models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21819–21835, Vienna, Austria. Association for Computational Linguistics.

Sebastian Nordhoff and Harald Hammarström. 2011. Glottolog/langdoc: Defining dialects, languages, and language families as collections of resources. In *First International Workshop on Linked Science 2011-In conjunction with the International Semantic Web Conference (ISWC 2011)*.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Edoardo Maria Ponti, Helen O'Horan, Yevgeni Berzak, Ivan Vulić, Roi Reichart, Thierry Poibeau, Ekaterina Shutova, and Anna Korhonen. 2019. Modeling language variation and universals: A survey on typological linguistics for natural language processing. *Computational Linguistics*, 45(3):559–601.

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.

Rita Ramos, Everlyn Asiko Chimoto, Maartje Ter Hoeve, and Natalie Schluter. 2025. GrammaMT: Improving machine translation with grammar-informed in-context learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 29920–29940, Vienna, Austria. Association for Computational Linguistics.

Felipe Sánchez-Martínez, Víctor M. Sánchez-Cartagena, Juan Antonio Pérez-Ortiz, Mikel L. Forcada, Miquel Esplà-Gomis, Andrew Secker, Susie Coleman, and Julie Wall. 2020. An English-Swahili parallel corpus and its use for neural machine translation in the news domain. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 299–308, Lisboa, Portugal. European Association for Machine Translation.

Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. 2021. PICARD: Parsing incrementally for constrained auto-regressive decoding from language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9895–9901, Online and Punta Cana, Dominican Republic. 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.

Richard Shin, Christopher Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, Dan Klein, Jason Eisner, and Benjamin Van Durme. 2021. Constrained language models yield few-shot semantic parsers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7699–7715, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

George Wolfe Robert Tobias and Basil HC Turvey. 1954. *English-kwanyama dictionary*. Witswatersrand University Press.

Bailin Wang, Zi Wang, Xuezhi Wang, Yuan Cao, Rif A. Saurous, and Yoon Kim. 2023. Grammar prompting for domain-specific language generation with large language models. 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*.

Kexun Zhang, Yee Choi, Zhenqiao Song, Taiqi He, William Yang Wang, and Lei Li. 2024. Hire a linguist!: Learning endangered languages in LLMs with in-context linguistic descriptions. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15654–15669, Bangkok, Thailand. Association for Computational Linguistics.

Appendix

Contents

A Appendix Overview

This appendix documents the full mTIG implementation, including grammar-slice prompts, lexical and semantic operators, classification procedures, hyperparameters, and additional results supporting the main results.

B Prompts for mTIG Generation

B.1 mTIG Base Prompt

Table 10 shows the prompt template for mTIG generation using grammar slices. The template conditions generation on a specific grammar slice, along with a semantic topic to guide content. The output is structured as a JSON array for easy parsing and downstream processing.

```
You are a linguistic generation assistant for Oshikwanyama (Kwanyama), a Bantu language
spoken in Namibia and Angola.

Using the provided grammar slice and topic context, generate {num_sentences} sentences in
valid Oshikwanyama, each paired with an English translation.

Grammar slice:

{slice}

Topic context:

{topic_name} — {topic_description}

Each output entry must follow the specified JSON structure below: Return ONLY a valid JSON
array containing exactly {num_sentences} objects. Do not include any other text.

The generated sentences should be short (approximately 3–6 words) and reflect the topic
context.
```

Figure 10: mTIG prompt using a grammar slice. A compact, targeted grammatical specification replaces implicit diversity with explicit control over morphosyntactic realization.

B.2 mTIG Lexical Correction Prompt

Table 11 shows the prompt template for lexical correction. The template instructs the LLM to revise existing translations using a constrained lexicon while preserving the original meaning. This ensures that generated sentences adhere to both grammatical and lexical constraints.

B.3 mTIG Lexical Coverage without Combinatorics Prompt

Table 12 shows the prompt template for lexicon-focused generation. The template instructs the LLM to generate multiple sentences that systematically exercise target lexical items across different syntactic and semantic contexts, guided by a grammar slice.

C Semantic and Lexical Components of mTIG

C.1 Semantic Topic Inventory

Table 4 details the semantic topic inventory used for conditioning sentence generation. Topics are grouped by conceptual domain. This inventory ensures that generated sentences span a wide range of topics. These topics were applied consistently across all languages in our experiments and baseline comparisons.

C.2 Lexical Resources

Lexical grounding was informed and validated using published English–Kwanyama, English–Ndonga, and English–Rukwangali dictionaries (Tobias and Turvey, 1954; ELCIN Church, 1996; J.K. Kloppers, 1994). These resources were consulted to verify lexical forms and translations during generation and post-editing, but no verbatim dictionary content is redistributed as part of this work.

The consulted resources contain approximately 10,000 entries for Kwanyama, 15,000 for Ndonga, and 4,000 for Rukwangali.

```
You are a linguistic translation correction assistant for {Language}.

Using the provided grammar slice, topic context, and lexicon, revise the given {Language}
sentences to improve their lexical quality, while preserving their original meaning. Each
sentence must be paired with its English translation.

Grammar slice:

{slice}

Topic context:

{topic_name} — {topic_description}

Each output entry must follow the JSON structure below:
  [
    {
      "english": "...",
      "old_{Language}_translation": "...",
      "improved_{Language}_translation": "...",
      "base_words": [...],
      "lexicon_entries_used": [
        {
          "word": "...",
          "translated_as": "..."
        }
      ],
      "pos_sequence": [...]
    }
  ]

The base_words field must contain English base forms only (e.g., started → start, running
→ run).

Use ONLY lexicon entries provided to improve the translation. Do NOT introduce new concepts
or words that are not present in the English sentence (e.g., do not add "love" if it does
not occur in the source).

Return only a valid JSON array of objects. Do not include any other text.

The input sentences to be corrected are provided below:

——————————

{input_json_text}
```

Figure 11: Lexical correction prompt. Existing Oshikwanyama translations are revised using a grammar slice and a constrained lexicon, while strictly preserving the original sentence meaning.

D Examples Sentences from the mTIG Library

Table 5 presents a selection of representative mTIG constructions from the Bantu mTIG library in the development language, Oshikwanyama, along with aligned English examples. Each construction corresponds to a modular mTIG unit controlling a specific morphosyntactic phenomenon, demonstrating the breadth of grammatical coverage provided by the mTIG framework.

E Baseline and Grammar Dump Generation

Table 13 shows the baseline prompt template for sentence generation without grammatical guidance. And Table 14 shows the grammar dump baseline prompt template, which provides the full descriptive grammar as monolithic context.

F Grammatical Slice Classification

Table 15 shows the prompt template used for grammar-slice classification. This prompt is used to annotate generated sentences with the set of syntactic constructions they instantiate, enabling multi-label grammatical analysis without semantic inference. We use the same LLM (Gemini) for classification regardless of which LLM was used for generation.

```
You are a linguistic generation assistant for Language.

Using the provided lexicon entries, generate multiple short sentences in valid Language,
each paired with an English translation.

For each target word, generate several sentences that use the word in different syntactic
and semantic contexts. If a word has multiple meanings, ensure that different senses are
used across different sentences.

A grammar slice is also provided to guide sentence formation. The sentences should follow
the grammatical typology and structures described in the grammar slice where possible. If
a grammatical constraint conflicts with correct word usage, prioritize correct lexical
usage and fall back to a simple sentence structure (e.g., SV, SVO).

Grammar slice:

{slice}

Each output entry must follow the JSON structure below:
  [
    {
      "main_word_english": "...",
      "english_sentence": "...",
      "{Language}_sentence": "...",
      "pos_sequence": [...],
      "base_words": [...]
    }
  ]

The base_words field must list English base forms only (e.g., started → start, running
→ run).

The generated sentences should be short (approximately 3–6 words) and must use the provided
target words as their lexical focus.

Return ONLY a valid JSON array. Do not include any other text.
```

Figure 12: Lexicon-focused generation prompt. Sentences are generated to systematically exercise target lexical items across multiple contexts, with grammar guidance provided via a grammar slice.

G Typological Balance - Additional LLM Generators

G.1 Claude Opus-4-5

Table 16 presents additional results on typological balance for generated corpora using Claude.

G.2 GPT-4o

Table 17 presents additional results on typological balance for generated corpora using GPT-4o.

H Machine Translation

H.1 NLLB Training Details

All neural machine translation experiments were conducted by fine-tuning the `facebook/nllb-200-distilled-1.3B` model using the HuggingFace `Seq2SeqTrainer` framework. This section documents the complete set of training hyperparameters used, to ensure reproducibility and transparency.

Batching and Optimization The per-device training batch size was set to 8, with gradient accumulation performed over 4 steps, yielding an effective batch size of 32 sentences.

Optimization was carried out using the AdamW optimizer with the following parameters:

- Learning rate: 2×10^{-5}
- Weight decay: 0.02

Models were trained for 6 epochs.

Summary of Hyperparameters Table 6 shows a summary of NLLB fine-tuning hyperparameters.

ID	Semantic Topic	Description
SOCIETY & CULTURE		
1	Family & Kinship	Relationships, greetings, kin roles, and everyday family life.
2	Education	Schools, learning, teachers, students, and classroom life.
3	Religion	Church, prayer, moral values, and religious practices.
4	Traditions & Customs	Ceremonies, initiation, weddings, and communal events.
5	Health	Sickness, hospitals, healing practices, and well-being.
GOVERNANCE & REASONING		
6	Politics, Governance & Law	Leadership, justice, fairness, democracy, rights, and conflict resolution.
7	Common Sense & Everyday Reasoning	Cause and effect, purpose, temporal logic, and basic practical reasoning.
ENVIRONMENT & PHYSICAL WORLD		
8	Environment & Nature	Animals, plants, weather, farming, and geography.
9	Weather & Seasons	Rain, sun, storms, seasonal change, and environmental impact.
10	Farming & Food Production	Planting, harvesting, animals, tools, and rural livelihood.
11	Objects, Space & Movement	Spatial relations, motion verbs, and physical orientation.
TIME, EVENTS & INTERACTION		
12	Time & Events	Temporal expressions, planning, frequency, and continuity.
13	Everyday Interaction & Social Life	Greetings, apologies, friendship, and small talk.
14	Emotion & Interaction	Feelings, interpersonal communication, and social emotions.
15	Speech & Communication	Speaking, hearing, discussion, storytelling, and knowledge transfer.
COGNITION, ETHICS & SOCIAL STRUCTURE		
16	Abstract & Cognitive Topics	Thought, knowledge, belief, doubt, and perception.
17	Cognitive Reasoning & Learning	Thinking, memory, understanding, and education as cognition.
18	Metalinguistic & Cognitive Control Topics	Hypotheticals, counterfactuals, explanations, and contrasts.
19	Moral & Evaluative Language	Ethics, honesty, respect, and moral behavior.
20	Gender Roles & Social Structure	Men, women, equality, division of labor, and identity.
21	Community & Cooperation	Helping, sharing, respect, teamwork, and village life.

Table 4: Semantic topic inventory used for conditioning sentence generation. Topics are grouped by conceptual domain to improve interpretability while remaining orthogonal to grammatical conditioning.

Grammar Domain	Example Sentence (English)	Example Sentence (Oshikwanyama)
Grammar → Verb Morphology		
Active — Present Positive (Pronouns)	He is looking at the tree.	Ota tale omuti.
Active — Present Positive (Nouns)	The goat is looking at the grass.	Oshikombo otashi tale omwiidi.
Active — Future Positive (Nouns)	The child will fetch water.	Okaana otaka ka tala omeva.
Active — Present Negative (Nouns)	The cows are not eating grass.	Eengobe itadi li omwiidi.
Stative — Past Positive (Pronouns)	We had a pot.	Otwali tu na ombiya.
Stative — Past Positive (Nouns)	My grandfather liked people.	Tatekulu okwa li e hole ovanhu.
Stative — Present Negative (Nouns)	Goats do not like fools.	Oikombo ka i hole omalai.
Stative — Past Negative (Pronouns)	You did not come.	Kwa li we uya.
Stative — Past Negative (Nouns)	The cows did not feel peaceful.	Eengombe ka da li di udite ombili.
Stative — Future Negative (Pronouns)	They will not drink water.	Itava ka nwa omeva.
Derived — Passive	Ndahafa is being greeted by Koto.	Ndahafa ota popifwa ku Koto.
Derived — Reciprocal	We saw each other.	Otwa monafana.
Derived — Benefactive / Applied	I bought sweets for her.	Onde mu landela ouleke.
Derived — Reversive	The man unlocked the door.	Omunhu okwa patulula omuvelo.
Derived — Causative	The teacher made the child work.	Omulongi okwa longifa okaana.
Aspect / Mood (Habitual, Subjunctive)	She eats corn. / I want you to go to school.	Oha li epungu. / Onda hala u ye kofikola.
Emotion / Motion	The child is tired. / I am coming home.	Okaana oka loloka. / Ohandi uya keumbo.
Imperative / Infinitive	I like to work.	Ondi hole okulonga.
Grammar → Pronouns		
Personal (Object)	Don't accuse me!	Ino lundila nge!
Demonstratives	This goat is eating.	Oshikombo eshi otashi li.
Possessives	Your cow is sleeping.	Ongobe yoye oya nangala.
Indefinite	Each student brought a book.	Keshe omulongwa okwa landa embo.
Grammar → Adjectives		
Noun + Adjective Agreement	The woman cooked good food.	Omukulukadi okwa teleka oikulya iwa.
Superlative	The man is the bravest of all.	Omulumenhu okwa yombama e dule aveshe.
Resemblance (*fa*)	You look like your mother.	Owa fa nyoko.
Comparative (*dule*)	Beef is tastier than goat meat.	Ombelela yongobe oiwa i dule yoshikombo.
Grammar → Questions		
Wh-Interrogatives	Where are the children?	Ounona ove li peni?
Grammar → Conjunctions		
Clause Linking / Conditionals	If I run, I will be tired.	Ngeenge onda lotoka, ohandi loloka.
Grammar → Prepositions		
Locative	The meat is in the pot.	Ombelela oi li mombiya.
Derived Locatives	He is talking about the teacher.	Ota popi kombinga yomulongi.
Grammar → Numerals		
Counted Objects / Concord	The lion is eating two goats.	Onghoshi otai li oikombo ivali.
Grammar → Adverbs		
Temporal	He is still talking.	Ota popi natango.

Table 5: Representative constructions from the mTIG library for Oshikwanyama, with aligned English examples. Each construction corresponds to a modular mTIG unit controlling a specific morphosyntactic phenomenon.

H.2 Generalization: Language Adaptation Prompt

Figure 7 presents the language adaptation prompt. Instructions originally written for Kwanyama are reused to generate sentences in a related Bantu language.

H.3 Zero- and Few-Shot LLM MT Prompts

Figure 19 and Figure 20 show the prompts for zero-shot and few-shot MT on LLMs.

```
You are a language generation assistant for Oshikwanyama (Kwanyama).
Generate {num_sentences} sentences in valid Oshikwanyama, each paired with an English
translation.

Topic context: {topic_name} — {topic_description}

Example translations: {examples}

Each output entry must follow the JSON structure below:

  [
    {
      "english": "...",
      "oshikwanyama": "...",
    }
  ]

The base_words field should list English base forms only (e.g., started → start, running
→ run, ran → run).

Return only a JSON array of exactly {num_sentences} objects.
Do not include any other text.

The generated sentences should be syntactically diverse, reflect the topic context, and
be short (approximately 3–6 words each).
```

Figure 13: Baseline prompt for sentence generation in Oshikwanyama with aligned English translations. The structured JSON output enforces consistency and supports downstream linguistic analysis No explicit grammatical constraints are provided; diversity is requested implicitly via sampling and topic conditioning

Parameter	**Value**
Base model	facebook/nllb-200-distilled-1.3B
Training framework	HuggingFace Seq2SeqTrainer
Epochs	6
Per-device batch size	8
Gradient accumulation steps	4
Effective batch size	32
Learning rate	2×10^{-5}
Warmup steps	500
Weight decay	0.02
Precision	FP16
Logging steps	50
Checkpoint saving	Disabled
Prediction with generation	Enabled

Table 6: Summary of NLLB fine-tuning hyperparameters used in all MT experiments.

```
You are a linguistic generation assistant for Oshikwanyama (Kwanyama), a Bantu language.

Below is a large portion of the descriptive grammar of the language. Read the entire
grammar carefully and internalize the rules, structures, and examples described in it.

————————
BEGIN GRAMMAR BOOK
————————
{full_grammar_text}
————————
END GRAMMAR BOOK
————————

Task:

Using ONLY the grammar above as guidance, generate {num_sentences} sentences in grammatical
Oshikwanyama.

The sentences must:

   • be consistent with the grammatical rules described above,
   • cover a variety of grammatical structures,
   • be short (3–6 words),
   • reflect the topic context,
   • be paired with accurate English translations.

Topic context:

{topic_name} — {topic_description}

Each output entry must follow the JSON structure below:
  [
    {
      "english": "...",
      "oshikwanyama": "...",
    }
  ]

Return ONLY the JSON array of exactly {num_sentences} objects. Do not include any other
text or explanations.
```

Figure 14: Grammar dump baseline prompt. A full descriptive grammar is provided verbatim and treated as monolithic context, requiring the LLM to infer control implicitly rather than through modular constraints.

```
You are a linguistic annotation assistant.

Task:

For each sentence below, identify which syntactic constructions (grammar slices) are
present.

Instructions:

   • This is a multi-label classification task.
   • Zero, one, or many labels may apply to a sentence.
   • Only label constructions that are clearly expressed syntactically.
   • Do NOT infer meaning or semantics.
   • If a construction is uncertain or ambiguous, mark it as false rather than guessing.

Sentences:

{sentence_block}

Syntactic constructions (grammar slices):

{labels_block}

Each output entry must follow the JSON structure below:
  [
    {
      "sentence_index": 1,
      "labels": {
        "LABEL_NAME": true
      }
    }
  ]

Return ONLY a valid JSON array. Do not include any explanations or additional text.
```

Figure 15: Grammar-slice classification prompt. Each sentence is annotated with the set of syntactic constructions it instantiates, enabling multi-label grammatical analysis without semantic inference. We use the same LLM (Gemini) for classification regardless of which LLM was used for generation.

Method	H	H_{norm}	**mTIG Gain (%)**
Baseline Prompt	2.994	0.765	+14.6%
Grammar Dump	3.137	0.802	**+9.4%**
mTIG (ours)	**3.433**	**0.877**	—

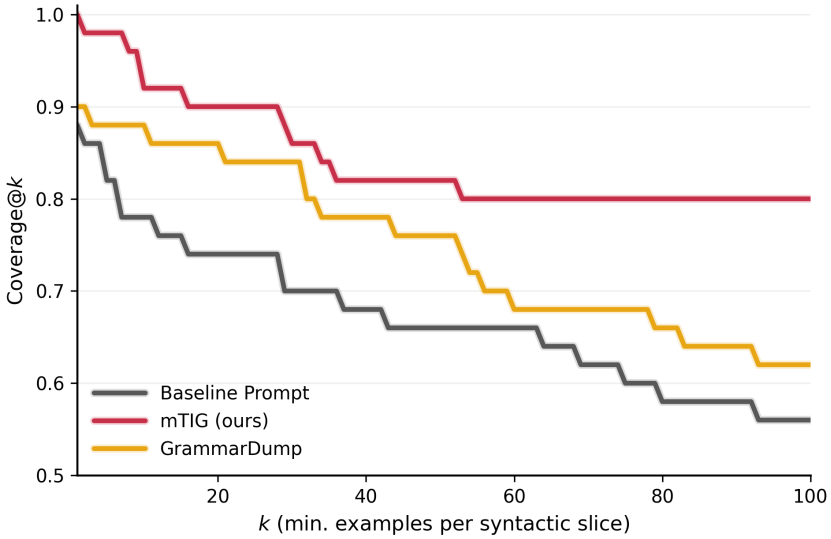

Figure 16: **Typological balance of generated corpora (Claude).** Using a different LLM as a generator, mTIG again yields substantially more uniform grammatical coverage than unguided prompting and grammar dump baselines, both in entropy (**top**) and in slice-level coverage across thresholds (**bottom**).

Method	H	H_{norm}	mTIG Gain (%)
Baseline Prompt	2.892	0.739	+18.9%
Grammar Dump	2.891	0.739	**+18.9%**
mTIG (ours)	**3.438**	**0.879**	—

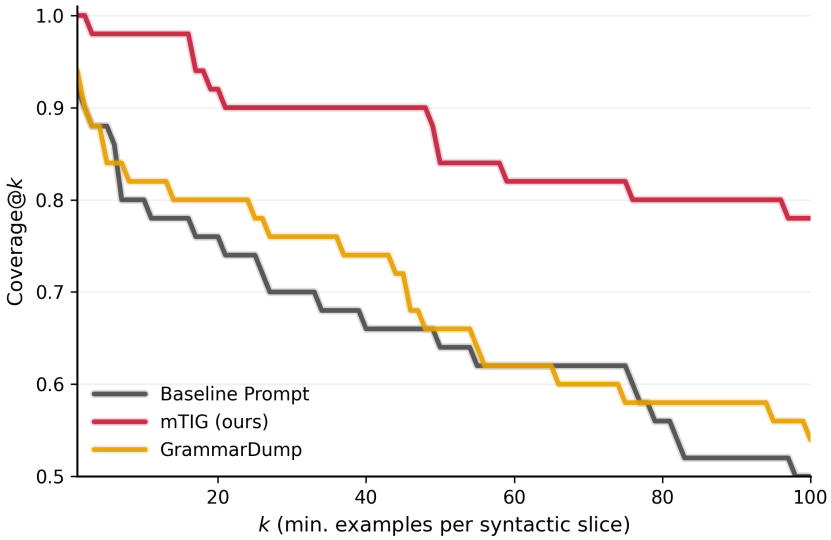

Figure 17: **Typological balance of generated corpora (GPT-4o).** On a third frontier LLM as the generator, mTIG again yields more uniform grammatical coverage than unguided prompting and grammar-dump baselines, both in entropy (**top**) and slice-level coverage across thresholds (**bottom**).

```
The following instructions were originally written for Oshikwanyama (Kwanyama) and are
used here only as an illustrative example.

Adapt these instructions to the target language {language}. The target language is a
Bantu language with the same grammatical structure as Kwanyama, but with a different
vocabulary. Therefore, all grammatical instructions still apply, but sentence generation
must be performed in {language} instead of Kwanyama.

You are a linguistic generation assistant for {language}.

Using the provided grammar slice and topic context, generate {num_sentences} sentences in
valid {language}, each paired with an English translation.

Grammar slice:

{slice}

Topic context:

{topic_name} — {topic_description}

Each output entry must follow the JSON structure below:
  [
    {
      "english": "...",
      "{languagelower}": "...",
    }
  ]

The base_words field must list English base forms only (e.g., started → start, running
→ run).

Return ONLY a valid JSON array containing exactly {num_sentences} objects. Do not include
any other text.

The generated sentences should be diverse, reflect the topic context, and be short
(approximately 3–6 words each).
```

Figure 18: Language adaptation prompt. Instructions originally written for Kwanyama are reused to generate sentences in a related Bantu language.

```
You are a multilingual translation assistant.

Translate the following {num_sentences} English sentences into the target language {language}.

The target language may be low-resource. Use natural, fluent translations appropriate for {language}.

Preserve the meaning of each sentence. Do not add explanations or commentary.

Return ONLY the translations:

• One translation per line
• Exactly {num_sentences} lines
• In the same order as the input sentences

Sentences to translate:

{batch_text}
```

Figure 19: Zero-shot machine translation prompt. An LLM is instructed to translate English sentences directly into the target language without examples, grammar constraints, or post-editing.

```
You are a multilingual translation assistant.

Translate the following {num_sentences} English sentences into the target language {language}.

The target language may be low-resource. Use natural, fluent translations consistent with the examples provided.

Preserve the meaning of each sentence. Do not add explanations or commentary.

Return ONLY the translations:

• One translation per line
• Exactly {num_sentences} lines
• In the same order as the input sentences

Examples (English → {language}):

{examples}


Sentences to translate:

{batch_text}
```

Figure 20: Few-shot machine translation prompt. In-context bilingual examples are provided to guide translation into the target language, serving as a standard few-shot MT baseline.

H.4 Translated Kwanyama Test Set

To construct a high-quality evaluation benchmark for Kwanyama, we recruited five candidate translators and conducted an initial qualification task. Two translators met the required quality threshold and were selected for the main annotation phase. Each translator independently produced approximately half of the candidate translations.

All translations were subsequently reviewed and filtered through an additional quality control pass, during which erroneous or inconsistent translations were removed. This process resulted in a final test set of **833 professionally translated sentence pairs**, which we use for all Kwanyama evaluation experiments.

I Error Analysis

I.1 Error Analysis of mTIG Generated Data

The observed errors, Table 7 fall into four recurring categories. First, *verb allomorphy errors* arise when the model selects an incorrect surface form of a verb under tense or aspect (e.g., hallucinated vowel alternations such as *longo* $\rightarrow$ *longe* or *kofa* $\rightarrow$ *kofe*). Second, *constructional overgeneralization* occurs when valid constructions are applied outside their licensed contexts, such as copular patterns (*oku li*) used with adjectival predicates where simple agreement is required. Third, *derivational overapplication* appears in causative and reciprocal forms, where redundant or incomplete morphemes are introduced (e.g., unnecessary causative suffixes or omission of required particles). Finally, a small number of *lexicalized exceptions* reflect verb-specific behavior that cannot be inferred compositionally (e.g., passive formation differences between *pewa* and *yandjwa*). Importantly, these errors do not undermine the typological coverage or balance achieved by mTIG; rather, they highlight the remaining *lexical bottleneck* in low-resource LLM generation and motivate future integration of tighter lexicon-level constraints.

I.2 Error Analysis of Slice Classifier Used for Typological Balance

To quantify typological balance, we assign generated sentences to grammar slices using an automatic classifier (Gemini for all data, independent of the generator LLM). Because the classifier operates over surface-level cues in English, its outputs are necessarily approximate and occasionally overgeneralize across closely related categories.

We manually inspected 50 classified sentences and found that the most common errors involve *tense and polarity overgeneration*, where both positive and negative variants of a tense are simultaneously assigned, and *coarsening*, where progressive forms are collapsed into simple present categories. Less frequent errors arise from *constructional ambiguity* (e.g., adjectival vs. passive readings such as *is delayed*).

Crucially, these errors are systematic rather than stochastic and are applied uniformly across all generation methods. Because typological balance is evaluated comparatively using the same classifier for all systems, such misclassifications do not affect relative entropy or coverage trends. Instead, they reflect the inherent difficulty of fine-grained syntactic categorization from short sentences in isolation and should be interpreted as approximate indicators of corpus-level diversity rather than exact syntactic counts.

#	mTIG Slice	English	Kwanyama (Generated)	Error Description
1	Active, Present Positive (Pronouns)	I am trusting the government.	*Ohandi lineekele kepangelo.*	Argument structure error: verb *lineekele* requires a direct object; insertion of *ke-* introduces incorrect oblique semantics.
2	Active, Present Positive (Nouns)	The father is working.	*Tate ota longe.*	Lexical–morphological hallucination: unsupported vowel alternation (*longo* → *longe*) not licensed by tense or agreement.
3	Active, Present Positive (Nouns)	The child is sleeping.	*Okaana otaka kofe.*	Systematic lexical–morphological error: verb root *kofa* incorrectly altered to *kofe*.
4	Stative, Present Positive (Nouns)	The songs are beautiful.	*Omaimbilo oku li nawa.*	Invalid stative construction: copula *-li* is overgeneralized; adjectival predication requires agreement morphology.
5	Active, Past Positive (Pronouns)	He loved his mother.	*Okwa hola ina.*	Past tense realization failure: required past auxiliary and verb form are omitted; present root is used instead.
6	Active, Past Positive (Nouns)	The cows drank water.	*Eengobe oda nu omeva.*	Past tense verb error: present verb *nu* used instead of past allomorph *nwa*.
7	Imperative / Infinitive	Speak to me!	*Popya na nge!*	Pronoun role error: possessive *nge* is used instead of object pronoun *ame*.
8	Derived, Causative	The storm made us return home.	*Oshikungulu oshe tu alulifa keumbo.*	Causative overgeneration: causative suffix applied redundantly; base verb *alula* is sufficient.
9	Derived, Passive	The knowledge was given freely.	*Eshiivo ola pewa oshali.*	Lexicalized passive selection error: abstract transfer requires passive *yandjwa*, not *pewa*.
10	Derived, Reciprocal	We forgive each other.	*Otwa diminelafana.*	Incomplete reciprocal construction: reciprocal verb lacks required particle *po*.
11	Derived, Benefactive / Applied	He speaks for his sister.	*Ota popila omumwaina kadona waye.*	Applied verb allomorph error: incorrect benefactive form *popila* selected instead of *popile*.
12	Aspect / Mood (Habitual)	Fish swim in the river.	*Eeshi ohadi yowa momulonga.*	Aspect–verb mismatch: habitual marker is correct, but verb root *yowa* is used instead of habitual form *yoo*.

Table 7: Representative error types in generated corpus for the development language (Claude as the generator due to its superior performance) across mTIG slices.

Model	Size	Description
Prompted LLMs (via API)		
Claude Opus 4.5	–	Anthropic (2025)
DeepSeek-V3	671B	DeepSeek-AI (2024)
Gemini 2.0 Flash	–	Anil et al. (2023)
GPT-4o	–	OpenAI (2023)

Table 8: **LLMs used in this work.** All models were accessed via paid APIs between late 2024 and early 2025.

J GATITOS Lexical and SMOL MT Experiments: Correlation Experiments

J.1 Evaluated LLMs

Table 8 summarizes the LLMs evaluated in the GATITOS and SMOL experiments. Notably, DeepSeek-V3 was used only for lexical probing and downstream evaluation, not for mTIG corpus generation, due to its prohibitively slow inference speed for large-scale data synthesis via API.

J.2 GATITOS Lexical Experiments

Table 9 and Table 10 show the full results for all 107 languages we extracted from GATITOS. **GATITOS**, introduced by Jones et al. (2023) is a publicly available dataset with lexicons for 100+ low-resource languages, and a few high-resource languages. We extracted data for 107 languages, of which 4 are high-resource controls. GATITOS has 4,000 English-to-target word pairs per language drawn from Swadesh lists, numerals, and common words. We use the entire 4k words for evaluation for each language. We evaluate lexical performance of Claude, Gemini, DeepSeek, and GPT-4o.

J.3 SMOL MT Experiments

Table 11 reports machine translation performance on the SMOL dataset. SMOL (Caswell et al., 2025) is a newly released benchmark of $\sim$900 professionally translated sentences per language, primarily covering low-resource languages. Crucially, the same sentences are translated across all languages, enabling controlled cross-lingual comparisons, similar in spirit to our GATITOS lexical probe.

For this experiment, we focus on the 42 languages shared between SMOL and GATITOS, and evaluate the performance of four LLMs: Claude, Gemini, DeepSeek, and GPT-4o.

Language	Claude	Gemini	DeepSeek	GPT4o
Spanish	**84.66** (72.64)	80.58 (67.74)	82.16 (70.08)	81.30 (69.27)
French	**82.41** (66.31)	78.80 (63.07)	80.28 (64.26)	78.60 (62.88)
Portuguese (Portugal)	**78.77** (63.36)	74.65 (59.34)	74.70 (59.56)	75.39 (59.85)
French (Canada)	**79.14** (61.34)	74.61 (57.93)	76.84 (59.64)	75.73 (58.44)
Papiamento	**67.43** (46.49)	66.77 (44.81)	62.02 (39.12)	56.52 (32.96)
Occitan	**63.72** (35.62)	59.31 (31.77)	58.96 (32.02)	56.82 (30.64)
Kedah Malay	**62.68** (50.36)	59.15 (48.36)	55.97 (44.61)	55.39 (43.81)
Waray (Philippines)	**62.03** (41.77)	57.58 (37.56)	50.31 (31.47)	51.14 (31.61)
Minang	**63.51** (46.38)	57.18 (39.72)	54.69 (34.57)	42.33 (23.12)
Tetum	**63.70** (46.06)	56.22 (38.23)	55.81 (36.47)	44.32 (25.10)
Bikol	**58.17** (41.78)	55.81 (39.11)	50.59 (34.97)	40.45 (24.92)
Fijian	**62.79** (51.16)	54.92 (41.02)	35.08 (21.54)	39.71 (26.51)
Sicilian	**59.34** (39.22)	54.81 (34.57)	54.93 (34.53)	47.75 (27.34)
Hiligaynon	**55.57** (38.44)	52.94 (35.97)	49.13 (31.60)	43.66 (26.77)
Venetian	**58.59** (36.36)	52.38 (30.39)	55.43 (33.73)	48.38 (25.58)
Breton	**63.18** (45.67)	51.97 (36.17)	49.83 (34.32)	38.03 (24.12)
Limburgish	**56.41** (33.31)	51.68 (28.37)	49.51 (26.35)	40.60 (17.64)
Ilocano	**58.15** (43.84)	51.59 (36.59)	47.48 (32.76)	32.58 (18.27)
Tok Pisin	**59.84** (43.24)	50.39 (33.48)	48.63 (32.49)	40.65 (26.76)
Pangasinan	**51.77** (35.64)	50.23 (34.06)	36.37 (21.51)	23.36 (10.14)
Banjar	**58.79** (42.92)	49.87 (33.77)	45.68 (27.47)	37.60 (19.05)
Faroese	**55.45** (37.20)	49.83 (33.38)	41.40 (26.23)	45.97 (29.06)
Friulian	**58.42** (32.81)	48.62 (24.40)	48.18 (23.80)	40.83 (16.61)
Khasi	42.95 (24.41)	**48.52** (25.72)	21.73 (6.45)	27.31 (11.66)
Lombard	49.45 (22.01)	48.27 (22.32)	**51.54** (24.40)	42.68 (15.75)
Latgalian	**49.82** (26.07)	46.84 (24.22)	37.63 (16.06)	30.31 (9.73)
Ligurian	**54.12** (29.72)	45.17 (22.03)	46.13 (21.26)	37.86 (13.47)
Iban	**44.13** (30.30)	43.29 (30.34)	38.64 (25.64)	28.28 (16.28)
Silesian	**50.85** (25.60)	42.94 (19.46)	42.26 (19.45)	38.54 (16.27)
Balinese	**49.46** (31.48)	42.41 (25.69)	44.17 (25.91)	34.52 (18.30)
Crimean Tatar (Latin)	**48.52** (31.12)	41.99 (26.78)	40.22 (25.34)	34.13 (19.01)
Jamaican Patois	**40.26** (27.24)	38.42 (24.52)	38.29 (21.38)	36.82 (22.14)
Northern Sami	**49.85** (27.25)	37.89 (17.87)	29.04 (12.84)	20.09 (7.80)
Quechua	**45.47** (29.04)	37.78 (21.97)	38.68 (22.15)	22.86 (9.32)
Zazaki	**41.67** (20.71)	37.23 (15.56)	31.40 (13.27)	18.22 (4.82)
Hakha Chin	**41.11** (17.17)	36.05 (14.71)	17.98 (4.55)	13.05 (2.41)
Kapampangan	**39.47** (25.88)	35.85 (21.00)	33.37 (18.71)	28.84 (13.82)
Batak Simalungun	**41.39** (25.52)	35.18 (19.89)	26.47 (13.56)	18.33 (6.25)
Kalaallisut	**46.11** (25.65)	34.37 (14.79)	22.00 (7.10)	25.45 (9.07)
Batak Toba	**36.69** (20.39)	33.65 (17.14)	23.61 (8.44)	16.61 (4.51)
Acehnese	**40.30** (22.58)	32.72 (15.24)	30.84 (14.55)	22.61 (8.76)
Batak Karo	**39.94** (26.47)	31.23 (18.36)	28.67 (16.77)	20.45 (9.48)
Madurese	**40.59** (20.29)	30.82 (13.06)	30.58 (13.93)	23.86 (9.18)
Guarani	**37.00** (17.34)	30.41 (13.18)	29.91 (13.66)	23.59 (9.27)
Aymara	**34.05** (16.19)	28.77 (11.62)	24.37 (10.50)	17.75 (4.51)
Makassar	**37.91** (16.95)	28.02 (10.81)	30.81 (16.06)	20.01 (8.13)
Jingpo	**31.33** (7.21)	27.89 (4.80)	15.88 (2.18)	10.03 (0.03)
Buginese	**40.87** (19.64)	27.32 (10.94)	20.50 (6.92)	14.28 (2.95)
Chamorro	**38.52** (17.38)	26.67 (9.73)	29.33 (11.50)	19.19 (5.07)
K'iche'	**38.04** (17.43)	25.88 (9.25)	16.73 (3.96)	11.91 (2.02)
Zapotec	**36.59** (15.68)	22.87 (8.16)	18.22 (4.97)	8.30 (0.50)
Q'eqchi'	**27.37** (9.82)	22.53 (6.38)	11.07 (1.53)	8.53 (0.47)
Romani	**36.00** (17.00)	22.29 (7.92)	30.42 (11.54)	19.63 (5.56)
Yucatec Maya	**29.65** (14.27)	22.11 (10.00)	17.47 (5.15)	11.41 (2.62)
Marshallese	**29.72** (13.47)	20.99 (7.07)	12.49 (2.69)	10.30 (1.98)
Hunsrik	**23.91** (6.29)	20.95 (5.25)	21.29 (5.06)	18.96 (3.92)
Chuukese	**36.00** (17.89)	19.22 (6.10)	10.58 (1.50)	10.09 (0.77)
Inuktut (Latin)	**35.06** (10.69)	19.21 (3.00)	20.33 (3.50)	16.72 (1.98)
Tahitian	**21.57** (4.61)	18.57 (3.93)	17.48 (3.80)	16.90 (3.12)
Mapudungun	**27.50** (9.39)	18.54 (4.29)	13.87 (3.05)	10.96 (1.44)

Table 9: **GATITOS lexical probe - part 1/2**. Lexical translation performance across 60 languages in GATITOS. ChrF scores shown outside parentheses; exact match scores inside parentheses; best per row in **bold**.

Language	Claude	Gemini	Deepseek	GPT4o
Seychellois Creole	**74.44** (59.76)	66.66 (50.50)	60.95 (39.70)	59.01 (41.35)
Mauritian Creole	**73.98** (58.36)	60.70 (43.34)	60.63 (43.03)	55.95 (34.45)
Tswana	**62.11** (43.30)	53.11 (36.00)	38.49 (22.80)	41.61 (26.58)
Sepedi	**58.64** (44.64)	49.76 (33.40)	40.21 (25.25)	40.61 (24.82)
Oromo	**58.44** (42.24)	49.03 (35.97)	40.35 (25.00)	38.85 (20.25)
North Ndebele	**57.70** (39.14)	54.27 (35.70)	47.41 (27.60)	48.66 (28.70)
Lingala	**56.42** (42.15)	52.28 (38.68)	38.41 (24.80)	35.41 (21.98)
Swati	**54.77** (26.53)	47.72 (22.17)	42.46 (14.20)	37.07 (15.50)
Rundi	**53.57** (39.22)	47.53 (33.30)	38.70 (24.20)	41.43 (26.58)
Luganda	**51.98** (32.80)	45.73 (29.83)	34.58 (19.62)	33.48 (17.09)
Tsonga	**51.79** (33.57)	43.61 (29.02)	30.22 (15.90)	32.04 (17.30)
Kituba (DRC)	**50.75** (33.60)	47.97 (31.40)	33.02 (16.77)	19.47 (6.11)
Bemba (Zambia)	**49.53** (27.81)	34.71 (13.35)	26.60 (7.21)	18.82 (4.31)
Venda	**48.63** (24.77)	38.30 (17.94)	25.04 (9.70)	23.89 (8.73)
South Ndebele	**48.17** (25.96)	40.91 (19.84)	37.58 (15.32)	35.08 (15.69)
Luo	**46.75** (24.22)	30.42 (14.55)	23.43 (8.30)	20.11 (7.33)
Kongo	**45.00** (22.87)	36.83 (16.02)	31.44 (14.50)	18.16 (5.03)
Bambara	**44.29** (27.10)	24.59 (13.30)	25.37 (12.30)	11.09 (1.94)
Twi	**43.89** (31.60)	36.64 (25.00)	28.43 (18.20)	24.92 (14.24)
Ewe	**43.35** (27.88)	26.12 (15.08)	17.86 (7.51)	9.29 (2.69)
Tumbuka	**43.22** (23.20)	38.27 (18.20)	26.30 (10.60)	24.78 (9.80)
Krio	**43.19** (36.28)	34.91 (31.90)	32.76 (25.30)	24.42 (15.13)
Tshiluba	**40.62** (17.60)	35.51 (15.22)	17.04 (2.22)	14.44 (1.33)
Kiga	**39.84** (18.52)	29.30 (12.00)	23.08 (6.16)	17.64 (2.81)
Nigerian Pidgin	37.24 (30.46)	37.81 (29.03)	32.50 (22.00)	**38.37** (29.76)
Dyula	**36.99** (16.80)	23.77 (8.21)	20.83 (7.78)	10.48 (1.43)
Tigrinya	**36.22** (38.04)	23.83 (25.85)	22.01 (22.00)	12.34 (12.38)
Wolof	**35.18** (22.32)	22.16 (11.10)	20.78 (11.30)	15.88 (7.15)
Tamazight (Latin)	**32.64** (17.91)	22.14 (9.41)	20.79 (9.29)	11.07 (2.99)
Sango	**32.31** (13.33)	27.60 (12.30)	11.64 (2.80)	8.34 (1.11)
Ga	**31.39** (15.80)	21.47 (8.13)	8.94 (1.00)	7.30 (0.52)
Kikuyu	**30.60** (10.62)	21.31 (5.36)	17.78 (4.90)	13.06 (1.84)
Acholi	**30.56** (18.59)	21.89 (11.92)	17.19 (6.90)	11.60 (2.45)
Ndau	**24.57** (7.70)	24.26 (7.21)	24.30 (8.20)	22.54 (5.97)
Alur	**23.25** (9.92)	11.85 (2.50)	12.65 (3.81)	9.87 (1.21)
Tiv	20.76 (8.54)	**22.50** (9.71)	8.90 (1.30)	7.87 (0.31)
Mooré	**20.38** (6.40)	9.12 (0.90)	7.35 (0.50)	7.55 (0.00)
Efik	**19.85** (6.83)	16.67 (6.11)	8.25 (1.00)	7.53 (0.33)
Susu	**19.14** (2.90)	9.83 (0.41)	8.36 (0.50)	7.94 (0.10)
Fulani	**18.89** (4.80)	13.74 (2.70)	13.07 (2.33)	9.69 (1.29)
Dinka	**18.47** (6.01)	7.72 (1.30)	7.07 (1.00)	5.83 (0.31)
Kanuri	**16.61** (6.62)	7.92 (1.40)	9.25 (2.43)	7.94 (0.42)
Baoulé	**15.79** (4.50)	11.31 (2.80)	7.91 (1.40)	6.11 (0.43)
Fon	14.90 (4.55)	**15.28** (5.01)	8.75 (2.40)	6.17 (0.94)
Dombe	14.02 (2.30)	**14.96** (1.50)	14.55 (2.30)	7.55 (0.00)
Nuer	**12.14** (2.11)	7.65 (0.30)	6.22 (0.40)	6.10 (0.11)
Isoko	9.60 (2.60)	**13.01** (3.30)	5.56 (0.40)	5.49 (0.21)

Table 10: **GATITOS lexical probe - part 2/2**. Lexical translation performance across 47 additional languages in GATITOS. chrF scores shown outside parentheses; exact match scores inside parentheses; best per row in **bold**.

Language	Claude	Gemini	DeepSeek	GPT-4o
Mauritian Creole	**54.98**	49.53	44.96	45.30
North Ndebele	46.39	**47.33**	43.20	40.41
Tswana	47.83	**49.27**	39.73	40.25
Nigerian Pidgin	**50.76**	38.47	39.45	47.99
Sepedi	**46.84**	46.82	38.13	38.77
Lingala	47.16	**48.06**	37.16	37.71
Tsonga	**49.64**	47.52	34.77	36.94
Swati	**44.84**	43.72	34.93	31.76
South Ndebele	40.03	**40.57**	36.50	34.70
Luganda	**44.22**	42.84	32.57	31.18
Venda	**47.13**	43.67	29.10	27.92
Kongo	**42.93**	40.34	37.86	22.57
Rundi	38.28	**38.49**	31.43	32.80
Oromo	**41.24**	39.49	31.33	28.93
Ndau	33.17	35.41	**35.78**	32.65
Tumbuka	39.63	**40.25**	29.21	27.48
Krio	**38.96**	32.66	30.66	24.24
Twi	34.52	**35.18**	27.33	22.44
Bemba (Zambia)	**38.52**	33.33	25.27	21.59
Kikuyu	**37.73**	32.89	24.14	18.30
Kiga	**34.61**	31.54	23.52	21.62
Luo	**35.46**	31.87	21.35	21.67
Acholi	**29.16**	28.90	22.41	18.49
Wolof	**27.93**	24.52	21.74	16.85
Ga	**32.19**	27.43	14.60	15.62
Ewe	**36.99**	12.76	22.88	16.44
Efik	25.34	**29.46**	18.25	15.11
Dyula	**28.24**	24.62	19.85	15.35
Tiv	23.84	**30.24**	16.32	16.29
Bambara	**27.00**	23.22	19.73	14.50
Fulani	**22.53**	19.66	16.61	16.35
Alur	**21.69**	14.60	17.95	16.13
Susu	**23.56**	13.67	12.22	14.39
Mooré	**20.15**	15.04	12.86	14.62
Dinka	**20.30**	13.82	13.20	12.15
Kanuri	**17.80**	11.96	13.27	15.17
Baoulé	**16.91**	13.37	12.42	12.47
Fon	**14.07**	12.41	10.92	9.63
Avg.	**34.80**	31.71	26.15	24.39

Table 11: **LLM Machine Translation on SMOL datase**. Machine translation performance across low-resouce languages in SMOL. Best per row in **bold**.

K Supplementary Material

The supplementary material accompanying this submission includes:

- The mTIG-generated synthetic corpus for the development language (generated using Claude).
- The full mTIG library, including all grammar slices
- The semantic topic inventory used for semantic conditioning in all experiments.

K.1 mTIG Library File Structure

The mTIG library is implemented as a structured directory of YAML files, numbered 001–051, with each file corresponding to a single grammar slice. This modular format supports versioning, cross-lingual reuse of family-level components, and incremental expansion of the typological inventory. The complete library is provided as supplementary material and will be released publicly.