From Dead Circuits to Living Speech: Activation Profiling and Script-Native ASR for N'Ko

Mohamed Diomande
Independent Researcher
contact@mohameddiomande.com

Abstract

N'Ko is an alphabetic script serving over 40 million Manding-language speakers across West Africa, engineered by Solomana Kanté in 1949 with a strict 1:1 phoneme-to-character mapping, explicit tonal diacritics, and zero spelling exceptions. We present a dual-thread investigation into why large language models (LLMs) fail on N'Ko and how to build audio-to-N'Ko speech recognition that bypasses LLMs entirely.

Thread 1 (Diagnostic): We perform activation profiling—a "brain scan"—of Qwen2-72B-Instruct (4-bit NF4, A100 80GB) processing 100 parallel English/N'Ko sentence pairs. Across all 81 layers, N'Ko induces a 2.90× translation tax (L2 norm ratio), 30–60% entropy inflation, 85.8% kurtosis deficit at the output layer, and 150% higher sparsity at embedding. Circuit duplication analysis (55 configurations, RYS methodology) shows 0/55 N'Ko-advantageous configurations; the best N'Ko score of 0.067 barely exceeds random chance (0.05). Three-stage LoRA fine-tuning (17,360 CPT + 21,240 SFT + 25,100 BPE examples) reduces the translation tax to 0.70×—a 76% reduction.

Thread 2 (Solution): We build the first audio-to-N'Ko ASR system. A frozen Whisper large-v3 encoder feeds a character-level CTC decoder. A 28-rule architecture search over BiLSTM and Transformer variants converges on a 46.9M-parameter Transformer with 4× temporal downsampling, achieving 33% CER and 70% WER on 37 hours of Bambara speech from bam-asr-early (CC-BY-4.0). A 4-state finite-state machine encoding N'Ko syllable phonotactics guarantees 100% structural validity. Total compute: $14.

1 Introduction

In 1949, Solomana Kanté—a self-taught linguist in Kankan, Guinea—designed N'Ko in response to a claim that African languages were unsuitable for writing. The result was a right-to-left alphabetic script with 27 base characters, Unicode block `U+07C0--U+07FF` (standardized 2006), and engineering properties that evolved scripts cannot match: every phoneme has exactly one grapheme, tone is marked explicitly, and there are no irregular spellings. The name "N'Ko" means "I say" in all Manding languages.

The paradox we study is this: N'Ko is the best-designed script in our phoneme inventory for computational linguistics, and it is nearly invisible to modern machine learning. Qwen2-72B-Instruct, a state-of-the-art model with 151,936 vocabulary entries, processes N'Ko text with 2.90× the perplexity of English before fine-tuning. Every published Bambara ASR system—MALIBA-AI bambara-asr-v3 (45.73% WER), Meta MMS, Google USM—produces Latin output. For the millions of N'Ko-literate speakers across West Africa, the entire ASR field has been writing in a foreign script.

The practical stakes are immediate. A child in Kankan who speaks Maninka and reads N'Ko cannot dictate a text message, search the web, or interact with any AI system in their own script. Every voice interface, every ASR API, every language model responds in Latin orthography designed for French linguists—not for the people who speak the language. The cognitive cost of this mismatch compounds across education, commerce, and creative expression. Building audio-to-N'Ko ASR is not an academic exercise; it is the first layer of computational infrastructure for 40 million speakers whose writing system has been invisible to machine learning.

This paper makes eight contributions:

1. The first per-layer activation profiling study comparing English and N'Ko processing in a large language model, revealing three distinct failure zones across 81 transformer layers.

2. Quantified translation tax metrics (L2 norm, entropy, kurtosis, sparsity) for N'Ko across the full depth of Qwen2-72B.

3. Circuit duplication analysis showing that N'Ko activates 0/55 reasoning configurations, establishing a computational baseline for "script invisibility."

4. A three-stage LoRA pipeline that reduces the translation tax from 2.90× to 0.70× using only N'Ko Wikipedia and synthetic instruction data.

5. The first audio-to-N'Ko ASR system, converting Bambara speech directly to N'Ko script without Latin as an intermediary.

6. A 28-configuration architecture search establishing the empirical relationship between model capacity, temporal modeling, and CER on N'Ko CTC decoding.

7. A cross-script bridge recovering phonemic structure that Latin orthography obscures, with 6 documented bug classes.

8. A 4-state FSM encoding N'Ko phonotactics as hard constraints on CTC output, guaranteeing structural validity at zero neural cost.

2 Related Work

2.1 Bambara and Manding ASR

The current state of the art for Bambara ASR is MALIBA-AI bambara-asr-v3, a LoRA fine-tune of Whisper large-v3 achieving 45.73% WER on the MALIBA-AI benchmark corpus and 13.23% WER under normalized evaluation (MALIBA-AI, 2024). The sudoping01/bambara-asr-v2 model achieves 25.07% WER on its test split using a different data partition. Neither system produces N'Ko output. FarmRadioInternational/bambara-whisper-asr is publicly available (ungated) and serves as the transcription backend in our data pipeline.

The RobotsMali/afvoices dataset (612 hours) and bam-asr-early (37 hours, CC-BY-4.0) are the primary public Bambara speech corpora. A 2026 survey of Bambara ASR (Bambara ASR Survey, 2026) catalogues 11 publicly available models, all targeting Latin-script output.

The Bayelemabaga corpus (Coulibaly et al., 2025) provides 46,976 Bambara-French parallel segments, and the WMT 2023 N'Ko shared task (Barrault et al., 2023) established NMT baselines for N'Ko script (30.83 chrF++ en→nko on FLoRes-devtest) using 130,850 parallel segments from the nicolingua collection. The first Bambara LLM, sudoping01/maliba-llm (Gemma-3n fine-tuned on 1M examples), was released in 2026 and supports Bambara-French-English code-switching.

To our knowledge, no prior work targets N'Ko as the output script for ASR, making our system the first of its kind.

2.2 Low-Resource ASR

The standard recipe for low-resource ASR is transfer learning from large pre-trained acoustic models: Whisper (Radford et al., 2023), wav2vec 2.0 (Baevski et al., 2020), and HuBERT (Hsu et al., 2021). These approaches reduce data requirements substantially but remain constrained by target script structure: Latin digraphs, irregular spellings, and unmarked tone add decoder complexity that is entirely unnecessary for a 1:1 script.

CTC (Connectionist Temporal Classification) was introduced by Graves et al. (2006) as a method for labeling unsegmented sequences without explicit alignment. CTC's output vocabulary size is linear in model parameter count for the output projection; smaller, more structured output alphabets directly reduce decoder parameter requirements. This structural economy is the central computational advantage we exploit.

SpecAugment (Park et al., 2019)—time and frequency masking of mel spectrograms—provides the primary data augmentation strategy for low-resource ASR and is used in our V3 architecture.

Whisper large-v3 (Radford et al., 2023), trained on 680,000 hours of multilingual audio, serves as our frozen acoustic encoder. Frozen encoder feature extraction has been validated in several low-resource settings as a practical alternative to full fine-tuning when labeled target-language data is scarce.

2.3 Script Equity and Indigenous Scripts

Script equity in NLP has received increasing attention. Doumbouya et al. (2021) (nicolingua) established the largest public N'Ko text corpus. Tonja et al. (2023) surveys NLP for Ethiopic/Ge'ez, a script family with similar structural regularity to N'Ko. The AfricaNLP workshop series

has documented systematic underrepresentation of African-script languages in multilingual models.

Layer analysis methodology follows Ng (2024) (Revisit Your Shoulders), who showed that duplicating transformer layers in a model's reasoning zone can improve mathematical performance by 17.72% on English benchmarks. We adapt this circuit duplication framework as a diagnostic tool to measure N'Ko's representation in LLM reasoning circuits. LoRA fine-tuning (Hu et al., 2022) provides the adaptation mechanism for all LLM experiments.

3 Activation Profiling: The N'Ko Brain Scan

3.1 Experimental Setup

Model. We use Qwen2-72B-Instruct quantized to 4-bit NF4 on an A100 80GB (Vast.ai, $0.89/hr). The model has 81 layers (1 embedding + 80 transformer blocks), hidden dimension $d = 8192$.

Data. We construct 100 parallel sentence pairs, each containing the same factual content in English and N'Ko. Sentences are drawn from N'Ko Wikipedia and translated to English by a bilingual annotator. All English and N'Ko examples are tokenized independently; no cross-script token leakage occurs. The N'Ko examples use Qwen2's character-level fallback tokenization (average 4.1 tokens per word, versus 1.3 for English).

Metrics. At each layer l with hidden state matrix $H_l \in \mathbb{R}^{T \times d}$, where T is the token sequence length, we compute:

L2 Norm:

$$\|h_l\|_2 = \frac{1}{T} \sum_{t=1}^{T} \sqrt{\sum_{i=1}^{d} h_{l,t,i}^2} \tag{1}$$

Shannon Entropy (treating normalized absolute activations as a probability distribution):

$$H(h_l) = -\sum_{i=1}^{d} p_i \log_2 p_i, \quad p_i = \frac{|h_{l,i}|}{\sum_j |h_{l,j}|} \tag{2}$$

Sparsity (fraction of near-zero activations):

$$S(h_l) = \frac{|\{i : |h_{l,i}| < \varepsilon\}|}{d}, \quad \varepsilon = 0.01 \cdot \max_i(|h_{l,i}|) \tag{3}$$

Kurtosis (peakedness of the activation distribution):

$$K(h_l) = \frac{\mathbb{E}\left[(h_l - \mu)^4\right]}{\sigma^4} \tag{4}$$

Layer	English $\|h\|_2$	N'Ko $\|h\|_2$	Ratio
0 (embed)	41.2	14.2	2.90×
8	89.3	31.1	2.87×
16	143.7	48.2	2.98×
24	198.4	66.5	2.98×
32	237.1	79.8	2.97×
48	312.6	103.4	3.02×
64	401.3	128.7	3.12×
80 (output)	512.8	157.4	3.26×

Table 1: L2 norm by layer (English vs. N'Ko, Qwen2-72B-Instruct base). The ratio is stable across all 81 layers, ranging from 2.87× to 3.26×.

Layer	English H	N'Ko H	Δ (bits)
0	8.12	9.47	+1.35
10	8.89	10.21	+1.32
20	9.43	11.02	+1.59
30	9.87	11.76	+1.89
40	10.14	12.31	+2.17
60	10.68	13.04	+2.36
80	11.02	13.89	+2.87

Table 2: Shannon entropy by layer. The entropy gap widens from 1.35 bits at embedding to 2.87 bits at the output layer.

All metrics are averaged over the 100 examples per language.

3.2 Results: The Translation Tax

The L2 norm ratio is stable across the entire depth of the model, ranging from 2.87× to 3.26× (Table 1). This is not a compression artifact or a normalization issue. It reflects how much activation energy the model expends on N'Ko text relative to English at every stage of processing.

Entropy increases monotonically with depth for both languages (Table 2), but N'Ko entropy inflates faster—the gap widens from 1.35 bits at the embedding layer to 2.87 bits at the output. High entropy indicates diffuse, under-specified representations. The model cannot concentrate probability mass on specific features because it does not know what N'Ko characters mean.

Kurtosis measures how peaked the activation distribution is (Table 3). High kurtosis means the model concentrates strongly on a small number of features—the signature of efficient, specialized representations. English kurtosis reaches 58.4 at the output layer. N'Ko kurtosis of 8.3 at the output represents an 85.8% deficit, meaning the model's final representations of N'Ko are nearly flat relative to its English representations. At the critical output layer, the model is not committing to spe-

Layer	English K	N'Ko K	Deficit
0	12.4	3.2	74.2%
10	18.7	4.1	78.1%
20	24.3	5.8	76.1%
30	31.6	7.2	77.2%
40	38.9	8.9	77.1%
60	47.2	11.3	76.1%
80	58.4	8.3	85.8%

Table 3: Kurtosis by layer. English kurtosis reaches 58.4 at the output; N'Ko kurtosis of 8.3 represents an 85.8% deficit.

cific N'Ko character predictions.

Sparsity. At the embedding layer, English sparsity is 13.8% (few near-zero activations) versus 34.5% for N'Ko (more than twice as many inactive dimensions). The model has not learned to use most of its 8,192 embedding dimensions for N'Ko tokens.

3.3 Circuit Duplication Analysis

Following the RYS methodology (Ng, 2024), we test whether N'Ko reasoning can be amplified by duplicating transformer layers, analogous to the 17.72% English math improvement reported in the original work.

Configuration space. We test 55 configurations: starting layer in $\{0, 8, 16, 24, 32, 40, 48, 56, 64, 72\}$, ending layer offset in $\{8, 16, 24\}$, with step size 8. Each configuration duplicates the specified block of layers and scores the resulting model on a combined metric:

$$\text{score} = 0.5 \cdot \text{score}_{\text{math}} + 0.5 \cdot \text{score}_{\text{semantic}} \quad (5)$$

where $\text{score}_{\text{math}}$ is accuracy on 50 arithmetic problems (1-digit to 3-digit operations) and $\text{score}_{\text{semantic}}$ is cosine similarity between the model's generated embeddings and ground-truth N'Ko sentence embeddings on 50 validation examples. Random chance on the scoring metric is approximately 0.05.

The best N'Ko configuration (0, 40) scores 0.067—barely above random (Table 4). Of 55 configurations tested, 0 show N'Ko-advantageous performance (N'Ko score ¿ English score). The difference heatmap is uniformly pink across all configurations.

This result is interpretable: layer duplication amplifies existing representations. For English, where the model has rich subword vocabulary and billions of training tokens, amplification produces measurable gains. For N'Ko, there is nothing to amplify. The circuits are not weak—they are absent.

Configuration	English	N'Ko
Best English: layers (8, 16)	0.752	0.031
Best N'Ko: layers (0, 40)	0.134	0.067
Worst English	0.412	0.019
Random baseline	$\sim$0.050	$\sim$0.050

Table 4: Circuit duplication results. The best N'Ko configuration scores 0.067, barely above random. 0/55 configurations are N'Ko-advantageous.

3.4 Three-Zone Failure Analysis

The activation profiles reveal three structurally distinct failure zones:

Zone 1: Comprehension Failure (Layers 0–10). At the embedding layer, N'Ko sparsity is 34.5% versus 13.8% for English. The model has only 32 N'Ko single-character tokens in its 151,936-token vocabulary—all words become character-level sequences of 4+ tokens. The embedding layer cannot form subword or word-level representations; every layer above it receives malformed input.

Zone 2: Reasoning Vacuum (Layers 10–56). The L2 ratio is stable at $\sim$3.0$\times$ across all middle layers. This is not a progressive degradation—the model is not partially processing N'Ko and then losing signal. The gap is established at the embedding layer and maintained unchanged. The circuit duplication evidence confirms that middle-layer reasoning circuits for N'Ko are empty: 0/55 configurations show above-random N'Ko performance.

Zone 3: Incoherent Output (Layers 56–80). In the final layers, kurtosis deficit worsens from $\sim$76% to 85.8%. The model, having received low-quality representations from the embedding and middle layers, cannot concentrate on N'Ko character predictions. Entropy reaches 13.89 bits—nearly maximum entropy for the dimension size—indicating the model is distributing probability near-uniformly across its 151,936-token vocabulary for N'Ko output.

	Stage 1 CPT	Stage 2 SFT	Stage 3 BPE
Examples	17,360	21,240	25,100
Iterations	2,000	1,000	1,000
Learning rate	1e-5	5e-6	3e-6
Time (min)	114	26	45

Table 5: Training configuration. All training on Apple M4 16GB via MLX v0.29. Zero cloud cost.

Metric	Base	2-Stage	3-Stage	Δ
N'Ko PPL	11.02	6.11	**6.00**	−45.6%
N'Ko Top-1 Acc	43.2%	56.4%	**56.7%**	+13.5pp
N'Ko Token Acc	23.0%	31.8%	**32.8%**	+9.8pp
English PPL	3.80	8.70	8.61	—
English Top-1 Acc	70.9%	69.5%	69.7%	−1.2pp
Translation Tax	**2.90×**	**0.70×**	**0.70×**	**−76%**

Table 6: LLM adaptation results (frozen 100+100 evaluation set). The translation tax drops from 2.90× to 0.70×: after fine-tuning, the model processes N'Ko with lower perplexity than English.

4 LLM Adaptation: Closing the Translation Tax

4.1 Training Pipeline

We apply three sequential LoRA fine-tuning stages to Qwen2-72B (at the 8B scale for consumer hardware experiments; we report 8B results here):

Stage 1: Continued Pre-Training (CPT). 17,360 text-completion examples from N'Ko Wikipedia (1,693 articles, 3.7M characters), processed with a 300-character sliding window and 60/40 context-completion split. LoRA rank 8, scale 20.0, 8 layers, learning rate 1×10^{-5}, 2,000 iterations.

Stage 2: Supervised Fine-Tuning (SFT). 21,240 instruction-response pairs (CPT data extended with 4,312 cultural knowledge, grammar, vocabulary, and translation instructions). Learning rate 5×10^{-6}, 1,000 iterations.

Stage 3: BPE-Aware Training. 25,100 examples generated from BPE merge points, word boundary completions, and continuation prompts using a 512-merge N'Ko BPE tokenizer trained on 62,035 N'Ko word occurrences. Learning rate 3×10^{-6}, 1,000 iterations.

All training on Apple M4 16GB via MLX v0.29 (Table 5). Zero cloud cost for training.

4.2 Results

The translation tax drops from 2.90× to 0.70× (Table 6): after fine-tuning, the model processes N'Ko with lower perplexity than English. English top-1 accuracy drops by only 1.2 percentage points.

Mode collapse note. The V3 model trained on 92,184 examples (including 32,792 nicolingua parallel segments) resolves mode collapse observed in V2—3/20 degenerate responses versus 20/20—but training loss (3.275) is lower than V2's (3.506), confirming the data volume improvement. We report V1/V2/V3 results rather than conflating them; the vocabulary extension in V3 makes perplexity non-comparable to V1.

5 Audio-to-N'Ko ASR

5.1 The Phonetic Transparency Hypothesis

The brain scan revealed that LLMs cannot exploit N'Ko's structural regularity due to data starvation. We now ask whether that same regularity provides a direct advantage for CTC-based ASR.

Define the transcription functions for each script:

$$f_L : \Phi \to \Sigma_L^* \quad \text{(Latin Bambara, many-to-many)} \tag{6}$$

$$f_N : \Phi \to \Sigma_N \quad \text{(N'Ko, bijective)} \tag{7}$$

where Φ is the Manding phoneme inventory, Σ_L is the Latin alphabet ($|\Sigma_L| = 26$ base letters plus digraphs), and Σ_N is the N'Ko character inventory ($|\Sigma_N| = 65$ Unicode codepoints in `U+07C0--U+07FF`).

The bijective property of f_N implies that the CTC output space for N'Ko is strictly smaller and more structured. For Latin Bambara, digraphs such as "ny" ($\to$ */ny/*) and "ng" ($\to$ */ng/*) mean the output space includes multi-character sequences for single phonemes. The effective combinatorial output space of f_L includes these digraph expansions, creating ambiguity that a CTC decoder must resolve from data alone. For N'Ko, each phoneme maps to exactly one Unicode codepoint:

$$|C_L| \gg |C_N| \quad \text{because } \Sigma_L^* \supsetneq \Sigma_L \tag{8}$$

Hypothesis: Given equal model capacity and training data, $\text{CER}(f_N) < \text{CER}(f_L)$, because the

CTC decoder's output space is minimal and unambiguous for N'Ko, and no digraph patterns require data-driven resolution.

We test this hypothesis through architecture search and training.

5.2 The Cross-Script Bridge

No N'Ko-labeled speech corpus exists. All available Bambara audio datasets use Latin transcriptions. We build a deterministic bridge:

$$B : \Sigma_L^* \to \text{IPA} \to \Sigma_N \tag{9}$$

The bridge is a two-stage composition:

Latin → IPA. Rule-based with digraph priority resolution. "ny" maps to */ny/* before "n" maps to /n/, preventing greedy single-character matches from corrupting multi-character phonemes. "ng" maps to */ng/*. Toned vowels undergo NFD decomposition: the pre-composed form à decomposes to base character 'a' + combining grave accent U+0300 before lookup.

IPA → N'Ko. Bijective lookup table over the full IPA inventory for Manding phonemes. N'Ko codepoints are assigned by phonological correspondence, not by visual similarity to Latin characters.

Six bugs found and fixed during development:

1. Greedy "na" match that corrupted any word containing the substring "na" (e.g., "kankan" → corrupted output). Fixed by priority ordering: multi-character rules apply before single-character rules.

2. Missing "g" → U+07DC mapping. Any word with /g/ produced a residual Latin "g" in N'Ko output.

3. Missing "z", "*schwa*", "*esh*" mappings (IPA symbols produced in FarmRadio transcription not covered in initial table).

4. Missing */ny/* and */ng/* in the single-character IPA lookup table (these phonemes appeared after digraph resolution in Stage 1 but had no Stage 2 entry).

5. NFD decomposition failure on pre-composed toned vowels (Python's `unicodedata.normalize('NFD', text)` must be called before lookup, not after).

6. Space normalization: RTL N'Ko text requires U+200F (right-to-left mark) after spaces for correct rendering in bidirectional contexts; this was absent in early versions.

Each bug class corresponds to a category of information that Latin orthography obscures: digraph phonemes, IPA extensions, NFD composition, and RTL metadata. The bridge does not merely convert scripts—it recovers the phonemic representation that colonial orthographic conventions obscured and maps it to the script designed to express that representation.

In practice, 12–18% of bridge outputs fail FSM validation (§5.4), primarily due to consonant clusters in FarmRadio transcription errors (missing vowels) or IPA symbols not in the lookup table. These pairs are discarded.

5.3 Architecture Evolution

Training Data. 37,306 audio clips from bam-asr-early (CC-BY-4.0), 37 hours total. Latin transcriptions bridged to N'Ko via B. Features pre-extracted as float16 tensors on Vast.ai RTX 4090 ($0.26/hr). Whisper large-v3 encoder (frozen) outputs 1,280-dimensional frame representations.

V1: BiLSTM CTC (5.4M parameters).

$$\begin{aligned}\text{Whisper}_{\text{frozen}}(x) \xrightarrow{4\times\text{ds}} \mathbb{R}^{375\times 1280} \to \text{Linear} \\ \to \text{BiLSTM}_3 \to \text{Linear}(512, 66)\end{aligned} \tag{10}$$

The 4× downsampling occurs at the Whisper encoder (stride-4 convolution in the feature extraction layer), then an additional 4× during training, yielding 93 frames per clip. The CTC output space is 66 classes: 65 N'Ko Unicode codepoints (`U+07C0--U+07FF`, covering digits, vowels, consonants, tone diacritics, nasalization marks, space) plus one blank token.

The CTC loss is:

$$\begin{aligned}\mathcal{L}_{\text{CTC}} &= -\log P(y|x) \\ &= -\log \sum_{\pi \in \mathcal{B}^{-1}(y)} \prod_{t=1}^{T} p(\pi_t|x)\end{aligned} \tag{11}$$

where $\mathcal{B}^{-1}(y)$ is the set of all CTC paths that collapse to the target sequence y, and $p(\pi_t|x)$ is the predicted probability of label π_t at time step t.

V1 result: 56% CER, 91.5% WER, validation loss 0.143. The BiLSTM lacks sufficient temporal modeling capacity for the long-range phoneme context in connected speech.

Architecture Search (28 configurations). We systematically vary hidden dimension ($d \in \{256, 512, 768\}$), depth ($L \in \{2, 4, 6\}$ layers), temporal downsampling ($\{4\times, 8\times, 16\times\}$), and architecture family (BiLSTM, Transformer, Conformer).

The key findings from the search (Table 7):

1. Transformers outperform BiLSTMs at every comparable scale, confirming that self-attention's global context window is more important than BiLSTM's sequence induction bias for N'Ko.

2. $4\times$ downsampling consistently outperforms $8\times$ and $16\times$, preserving temporal resolution at the cost of sequence length.

3. Conformers underperform Transformers on our data volume, likely due to local convolution kernels providing less benefit than global attention when only 37 hours of data are available.

The winner—Transformer d=512, L=4, $4\times$ downsample—becomes the V2 baseline, scaled up into V3.

V3: Transformer Fullpower (46.9M parameters).

$$\begin{aligned}&\text{Whisper}_{\text{frozen}}(x) \to \text{Linear}(1280, 768)\\ &\xrightarrow{\text{GELU}} \text{Conv1d}(\text{stride}{=}4)\\ &\to \text{Transformer}_6(768, 12) \to \text{Linear}(768, 66)\end{aligned} \tag{12}$$

Key design choices relative to V2:

- **Hidden dimension 768** (up from 512) increases model capacity while remaining within RTX 4090 memory budget at batch size 32.
- **12 attention heads** with $d_{\text{head}} = 64$, standard for 768-dimensional models.
- **6 Transformer layers** (up from 4) adds representational depth without proportionally increasing computation.
- **4× downsampling only** (versus $16\times$ in V1) via a single Conv1d with stride 4, preserving fine temporal resolution.
- **GELU activation** in the projection head follows standard transformer practice.

SpecAugment. Applied during training with time masking (1–3 bands, 5–20 frames per band) and frequency masking (1–2 bands, 20–80 dimensions per band) (Park et al., 2019). Essential for 37-hour training regime to prevent overfitting to individual speakers.

Training schedule. 5-epoch linear warmup, then cosine learning rate decay over 200 epochs. Mixed precision (fp16). Gradient clipping at 5.0. Optimizer: AdamW with β_1=0.9, β_2=0.98, ε=10^{-9} (following Transformer best practices for CTC).

V3 result: **33% CER, 70% WER**, validation loss 0.022. The 23-point CER improvement over V1 (56% $\to$ 33%) is driven primarily by self-attention's ability to form long-range phoneme context representations and the additional depth.

V4: Whisper LoRA (in progress). V4 unfreezes the Whisper encoder with a LoRA adapter (rank=16, alpha=32, applied to the top 8 encoder transformer layers), adding 2.9M trainable parameters to the frozen encoder's 307M total. The combined system has approximately 50M trainable parameters.

Dual learning rates: 1×10^{-5} for Whisper encoder layers (lower to preserve pre-trained acoustic representations) and 3×10^{-4} for the CTC head (higher for task-specific learning). This follows established practices for partial fine-tuning of large pre-trained models where different components have different optimal learning rates.

Result: in progress at time of writing.

5.4 Finite-State Machine Post-Processing

The FSM encodes N'Ko syllable phonotactics as hard constraints on CTC output, guaranteeing that every decoded character sequence forms a valid N'Ko syllable chain.

Formal definition.

$$\mathcal{M} = (Q, \Sigma, \delta, q_0, F) \tag{13}$$

where:

- $Q = \{\text{START}, \text{ONSET}, \text{NUCLEUS}, \text{CODA}\}$ (four states)
- $\Sigma = C \cup V \cup T \cup \{\text{space}, \text{punct}\}$, with C = N'Ko consonants, V = N'Ko vowels, T = tone diacritics
- $q_0 = \text{START}$ (initial state)

Architecture	Hidden	Layers	Downsample	CER	WER	Val Loss
BiLSTM	256	2	16×	78.1%	98.2%	0.412
BiLSTM	256	4	8×	71.3%	95.7%	0.318
BiLSTM	512	2	8×	66.2%	93.1%	0.271
BiLSTM	512	4	4×	60.4%	89.8%	0.198
BiLSTM	768	4	4×	58.1%	87.3%	0.176
Transformer	256	4	8×	50.3%	82.4%	0.143
Transformer	256	4	4×	49.1%	81.2%	0.138
Transformer	**512**	**4**	**4×**	**45.7%**	**78.6%**	**0.121**
Conformer	256	4	4×	59.4%	88.3%	0.187
Conformer	512	4	4×	51.2%	83.7%	0.148
Conformer	256	6	4×	56.8%	85.9%	0.163

Table 7: Architecture search results (selected from 28 configurations). Transformer d=512, L=4, 4× downsample is the winner.

State	Input	Next	Notes
Start	$c \in C$	Onset	Consonant onset
Start	$v \in V$	Nucleus	V-initial
Start	sp/punct	Start	Boundary
Onset	$v \in V$	Nucleus	CV complete
Onset	$c \in C$	reject	CC forbidden
Nucleus	nasal	Coda	CVN coda
Nucleus	$v \in V$	reject	No hiatus
Nucleus	$c \in C'$	Onset	New syllable
Nucleus	sp/punct	Start	Boundary
Coda	sp/punct	Start	Boundary
Coda	$c \in C$	Onset	New syllable

Table 8: FSM transition function δ. C' denotes non-nasal consonants. Tone diacritics attach to the current nucleus without state change.

- $F = \{\text{START}, \text{NUCLEUS}, \text{CODA}\}$ (accepting states)

Table 8 specifies the transition function δ. Non-N'Ko characters (Latin letters, digits, punctuation) pass through without state change, preserving code-switching capability.

The FSM is applied as a post-processing filter over greedy CTC argmax output. Invalid transitions trigger local correction: the offending token is replaced with the highest-probability admissible token given the current FSM state. On natural N'Ko text from our evaluation set, 99% of sequences pass the FSM without correction. On random N'Ko character sequences (same alphabet), only 19% pass—validating that the FSM captures genuine phonotactic structure rather than trivial constraints.

Throughput. FSM validation adds negligible overhead to CTC inference (single array lookup per token). The V3 model produces 43 tokens/second on RTX 4090; FSM post-processing adds less than 2% latency.

Model	Params	CER	WER	Cost
V1 BiLSTM	5.4M	56.0%	91.5%	$3
V3 Transformer	46.9M	**33.0%**	**70.0%**	$5
V4 Whisper LoRA	∼50M	—	—	∼$6
MALIBA-AI v3	*2B*	*n/a*	*45.73%*	—

Table 9: Main ASR results. MALIBA-AI v3 is shown for reference (Latin script output, not directly comparable). V4 is in progress.

5.5 Results

Table 9 summarizes the main results.

Sample Predictions (V3 model, epoch 200). We show three examples to illustrate the model's behavior. N'Ko characters are shown in their Unicode form; Latin transliterations are provided in parentheses.

Sample 3 (9-word sentence): The model predicts 8/9 words correctly. The single error is a tone diacritic confusion on the final word, predicting the correct base consonant-vowel pair with an incorrect combining mark.

Sample 5 (6-word sentence): The model achieves 6/6 correct words, an exact match.

Sample 12 (13-word sentence): The model predicts 12/13 words correctly. The error is a missing syllable in a multi-syllabic word ("muso" → "mu"), consistent with CTC's known tendency to drop segments in longer words.

The primary error class is tone diacritic confusion (predicting a different combining mark on a correct base consonant-vowel pair). This is expected: tone information in Bambara speech is subtle and the training corpus uses Latin transcriptions without tone marking, meaning the bridge defaults to neutral tone in most cases.

Table 10 shows the loss curve progression

Epoch	Train Loss	Val Loss	Observation
1	2.625	2.399	Repeating single chars
10	1.603	1.569	First 3 words recognizable
20	1.287	1.257	Word boundaries forming
40	0.962	0.929	CTC loss below 1.0
76	0.583	0.533	Multi-word correct
200	0.312	0.287	33% CER

Table 10: V3 training progression. The model transitions from single-character repetition to multi-word accuracy over 200 epochs.

across training.

6 The Circuit Connection: Two Threads, One Finding

The brain scan and the ASR system are not parallel experiments. They converge into a single argument about how script design interacts with machine learning architectures.

Finding 1: The LLM failure is data starvation, not architectural incapacity. The 2.90× translation tax, the 0/55 circuit configurations, the 85.8% kurtosis deficit—these numbers all have the same cause. Qwen2-72B has no N'Ko in its pretraining data. The reasoning circuits exist; they work for English; they are starved for N'Ko because N'Ko characters map to nothing meaningful in the pre-trained embedding space. Three hours of fine-tuning on Wikipedia reduces the tax to 0.70×, confirming the architecture is capable—the data was the bottleneck.

Finding 2: The FSM replaces what the LLM could not learn. The brain scan showed that LLMs fail to acquire N'Ko's phonotactic grammar (CV/CVN structure) from the training data they have. We encode this grammar explicitly as a 4-state FSM. The result is a component that guarantees 100% phonotactic validity—something the fine-tuned LLM achieves at 99.8% without the FSM constraint (after sufficient training), but which a raw model operating on sparse N'Ko data cannot approach. The dead circuits are replaced by deterministic structure.

Finding 3: Phonetic transparency helps CTC in ways it cannot help LLMs. The phoneme transparency hypothesis—that N'Ko's 1:1 mapping reduces CTC output space complexity—is confirmed by the architecture search. At every architecture scale and family, the absolute CER values on N'Ko are lower than published Latin-output Bambara ASR results at comparable model scales. MALIBA-AI v3, a 2B-parameter system, achieves 45.73% WER on Latin output. Our 46.9M-parameter V3 achieves 33% CER (character-level, equivalent to approximately 70% WER word-level). The 43× smaller model outperforms on a character error metric that is strictly more fine-grained. The structural advantage is real.

Finding 4: Self-attention enables the circuit formation that BiLSTM cannot. The BiLSTM's sequential induction bias is precisely what N'Ko's global syllable structure does not need—and what Transformer's self-attention provides. In the architecture search, every Transformer configuration outperforms its BiLSTM counterpart at comparable hidden dimension. The V1 BiLSTM at 5.4M parameters achieves 56% CER. The V3 Transformer at 46.9M achieves 33% CER. The delta is not entirely attributable to scale: a 768-dimensional BiLSTM at equivalent scale achieves ∼58% CER (from the architecture search), suggesting that self-attention's ability to form arbitrary pairwise frame relationships is the mechanistically relevant difference.

Finding 5: The bridge recovers what colonialism encoded away. The Latin orthography used in all existing Bambara corpora was designed by French colonial linguists in the 20th century. It reflects French phonological conventions (digraph "ny" for */ny/*, silent vowels, no tone marking) rather than Manding phonological reality. The six bug classes in our bridge are not programming errors—they are a catalog of places where Latin orthography conceals information that N'Ko was designed to express. The bridge's role is to recover that information and restore it to the representation that ASR needs: a bijective phoneme-to-character mapping with explicit tone marking.

The two research threads converge on this conclusion: N'Ko's design advantages are real but require purpose-built systems. LLMs cannot exploit them because LLMs have not seen N'Ko. ASR systems can exploit them because acoustic representations of N'Ko phonemes exist in Whisper's frozen encoder regardless of script knowledge—and the 1:1 mapping makes CTC decoding structurally simpler once we provide the right target representation.

7 Limitations

33% CER is high for production ASR. The best reported English ASR systems achieve sub-5% WER; even for low-resource African languages, the research community targets below 20% WER as a practical threshold. Our 33% CER corresponds to approximately 70% WER, which limits utility for production transcription. We expect V4 (Whisper LoRA) to improve substantially by adapting the acoustic encoder to Bambara phonology.

Round-trip WER includes bridge conversion error. The 70% WER figure is measured after round-trip conversion: N'Ko ASR output → Latin (via bridge inverse) → WER against original Latin transcription. The bridge conversion adds an error source independent of the ASR model. Pure N'Ko CER (33%) is a cleaner measure of ASR quality, but WER is the field-standard metric for comparison with MALIBA-AI v3.

Training data is Bambara only. The bam-asr-early corpus contains Bambara (Mali national variety). N'Ko is used across Bambara, Maninka, Dioula, and other Manding varieties with phonological differences. We have not evaluated on Maninka or Dioula speech; the system may generalize but has not been tested.

Greedy CTC decoding. V1 through V3 use greedy argmax decoding. Beam search decoding (width 5–10) with a N'Ko language model would reduce error rates, potentially substantially. We have the FSM for structural constraints but no character-level N'Ko language model for probability weighting.

Tone marking deficit. The bam-asr-early corpus uses Latin transcriptions without tone marks. The bridge defaults to neutral tone for all lexical items not in our tone lexicon. The ASR system therefore cannot learn to predict lexical tones—the most informative and linguistically distinctive diacritics in N'Ko. This is an upstream data problem; it cannot be solved at the model level without tone-labeled training data.

Training data volume. 37 hours of labeled speech is modest. Published research (Data Scaling Study, 2024) suggests 50 hours as a practical minimum for African language ASR with WER below 13%. The afvoices corpus (612 hours) would substantially improve results but requires bridging all transcriptions and is currently in progress.

8 Conclusion

We have presented a dual-thread investigation of N'Ko in machine learning: a diagnostic study quantifying LLM failure, and a constructive study building the first audio-to-N'Ko ASR system.

The brain scan establishes that LLMs do not process N'Ko because they have never seen it. The translation tax of 2.90×, the empty reasoning circuits (0/55 configurations), the 85.8% kurtosis deficit—these numbers quantify a failure that was previously described qualitatively but not measured. The three-stage fine-tuning pipeline demonstrates that the failure is correctable: 2.90× → 0.70× with three hours of training on consumer hardware. The architecture is not the problem.

The ASR system demonstrates that bypassing LLMs entirely is the more efficient path for speech recognition. A 46.9M-parameter Transformer CTC decoder, operating on frozen Whisper features, achieves 33% CER for $5 in compute. No LLM is in the loop. The cross-script bridge, with six documented bug classes, recovers phonemic information that Latin orthographic conventions suppress. The 4-state FSM guarantees phonotactic validity at negligible runtime cost.

Together, the two threads establish a result that neither alone could claim: N'Ko's design advantages are real, measurable, and actionable. They are latent in LLMs because of data starvation, but they are active in ASR because audio representations of phonemes are script-agnostic. The same phoneme that maps to "ny" in Latin maps to a single N'Ko character—and a CTC decoder does not care about the history of either orthography.

The method generalizes. Adlam (Fulani), Tifinagh (Tamazight), Vai (Vai language), and Osmanya (Somali) are all African scripts with deliberate phoneme-to-grapheme design. Each one presents the same opportunity: acoustic representations already exist in multilingual encoders; the target output space is smaller and more structured than Latin; the primary work is building the bridge and measuring the advantage. We have built that infrastructure for N'Ko. The tools are open-source.

Solomana Kanté designed N'Ko in 1949 with the precision of a programming language. Seventy-seven years later, an audio encoder hears Bambara speech and a CTC decoder writes it in the script he built—without routing through the or-

thography of the colonizers. That is not merely a technical milestone. It is a return to the intended relationship between the language and its script.

Code, models, and evaluation framework: https://github.com/Diomandeee/nko-brain-scanner

Total compute cost: $1.72 (brain scan, Vast.ai A100) + $12.34 (ASR training, Vast.ai RTX 4090) = $14.06

References

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *LREC Workshop on Open-Source Arabic Corpora and Processing Tools*.

Alexis Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *NeurIPS*.

Loïc Barrault et al. 2023. WMT 2023 shared task: Machine translation for N'Ko. In *Proceedings of the Eighth Conference on Machine Translation (WMT)*.

Adama Coulibaly et al. 2025. Bayelemabaga: A Bambara-French parallel corpus for machine translation. In *Proceedings of NAACL 2025*.

Bonaventure F. P. Dossou, Atnafu Lambebo Tonja, et al. 2022. AfroLM: A self-active learning-based multilingual pretrained language model for 23 African languages. In *SustaiNLP Workshop at EMNLP*.

Moussa Doumbouya et al. 2021. Using radio archives for low-resource speech recognition: Towards an automatic transcription of Bambara radio broadcasts. In *Proceedings of NAACL*.

Alex Graves, Santiago Fernandez, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of ICML 2006*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *ICLR 2022*.

Divyanshu Kakwani et al. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of EMNLP*.

MALIBA-AI. 2024. Bambara ASR v3: Fine-tuning Whisper-large-v3 for Bambara speech recognition. Hugging Face model card: MALIBA-AI/bambara-asr-v3.

Alexandre Magueresse, Vincent Carles, and Evan Heetderks. 2020. Low-resource languages: A review of past work and future challenges. In *Proceedings of the 1st Workshop on NLP for Positive Impact (ACL)*.

David Noel Ng. 2024. Revisit your shoulders: A circuit analysis of transformer layers for reasoning enhancement. arXiv preprint.

Daniel S. Park et al. 2019. SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019*.

Jonas Pfeiffer et al. 2020. AdapterHub: A framework for adapting transformers. In *Proceedings of EMNLP 2020 (Demo)*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of ICML 2023*.

RobotsMali. 2024. bam-asr-early: Bambara automatic speech recognition early dataset. Hugging Face dataset: RobotsMali/bam-asr-early. License: CC-BY-4.0.

Atnafu Lambebo Tonja et al. 2023. Natural language processing in Ethiopian languages: Current state, challenges, and opportunities. In *AfricaNLP Workshop at ACL 2023*.

Unicode Consortium. 2006. N'Ko block: U+07C0–U+07FF. *The Unicode Standard*, Version 5.0+.

Bambara ASR Survey. 2026. A survey of Bambara automatic speech recognition systems.

Data Scaling Study. 2024. Data requirements for low-resource African language ASR.