Context Is Geometry: Exploring a Native Object Vector Architecture for Adaptive Edge Intelligence

Yihu Wu

January 25, 2026

Abstract

The deployment of Large Language Models (LLMs) in browser-based environments faces a fundamental dichotomy: the need for user-specific adaptation versus the static nature of pre-trained weights. While quantization allows models like TinyLlama or Phi-2 to execute on client devices, they remain ”read-only” artifacts, unable to learn from immediate context due to the prohibitive cost of backpropagation in WebGPU shaders. This paper proposes a paradigm shift from algebraic matrix retrieval to geometric state evolution. We introduce the **Native Object Vector Architecture (N.O.V.A.)**, a design exploratory that projects context onto a complex spinor manifold using a ”Rotation-Gating-Injection” mechanism. Drawing inspiration from Holographic Reduced Representations (HRR), N.O.V.A. encodes semantic relationships as phase differences, enabling an Online Active Inference mechanism via fast Hebbian learning. Experiments on a legacy **Intel-based MacBook (2017)** validate the architectural efficiency: the model demonstrates a **66**$\times$ **throughput increase** compared to a quantized GPT-2 baseline ($\sim$47ms vs. $\sim$3131ms per token) while maintaining a minimal 350MB memory footprint. Crucially, empirical analysis reveals the emergence of stable ”Persona Vectors”—consistent behavioral modes that persist across generation steps where Transformer baselines exhibit catastrophic collapse. Code and demos are available at `https://github.com/TyloAI/NOVA`.

1 Introduction

The dominant trajectory in Generative AI equates intelligence with scale [2]. However, a significant ”capability gap” exists for edge devices, particularly within the ubiquitous web browser. Users desire intelligence that is not only private and immediate but also deeply personalized.

Current approaches (TensorFlow.js, ONNX) treat the browser as a passive inference target. While technically impressive, these solutions are architecturally misaligned with the interactive nature of the web. A quantized model running in a browser cannot ”learn” from a user’s corrections because the underlying architecture relies on frozen weights and backpropagation—computationally infeasible to implement efficiently in shader languages (WGSL).

We propose a different design philosophy: **Intelligence as Geometric Dynamics**. Instead of treating context as a static buffer retrieved by attention heads [1], we model it as a dynamic trajectory in a high-dimensional complex manifold.

This paper presents the **Native Object Vector Architecture (N.O.V.A.)**, exploring:

1. **Geometric Recurrence:** Replacing $O(N^2)$ attention with $O(N)$ quasi-symplectic flows (Rotation-Gating-Injection).

2. **Semantic Phase Encoding:** Utilizing the phase of complex spinors to encode context, inspired by cognitive science models of memory [9].

3. **Active Inference:** Replacing backpropagation with forward-only Hebbian updates [4] to enable real-time adaptation.

2 Methodology

2.1 Theoretical Foundation: The Semantic Phase Hypothesis

Why represent language with complex numbers? We posit the *Semantic Phase Hypothesis*: in a recurrent system, the magnitude $|z|$ of a state vector represents signal confidence (or energy), while the phase $\angle z$ represents semantic identity or syntactic state.

2.1.1 Holographic Inspiration

Our design is deeply rooted in Holographic Reduced Representations (HRR) [9]. In HRR, binding two vectors is performed via circular convolution. In our complex spinor domain, convolution becomes element-wise rotation. Thus, N.O.V.A.'s core operation—rotating the state vector—is a binding operation that encodes the temporal structural of language into the phase of the manifold.

2.2 The N.O.V.A. Architecture

The model consists of $L = 12$ geometric blocks with hidden dimension $d = 768$, totaling $\sim$47M parameters.

2.2.1 Rotation-Gating-Injection (R-G-I)

For each channel i at step t:

$$\text{Rotate:} \quad \tilde{s}_t = s_{t-1} \cdot e^{\mathbf{j}\theta} \tag{1}$$

$$\text{Gate:} \quad s_t^{\text{gated}} = \tilde{s}_t \cdot \sigma(W_g h) \cdot \alpha \tag{2}$$

$$\text{Inject:} \quad s_t \leftarrow s_t^{\text{gated}} + (v_{\text{in}} + \mathbf{j}\beta v_{\text{in}}) \tag{3}$$

We refer to the aggregate state orientation s_t as the **Persona Vector**. Because the rotation operator $e^{\mathbf{j}\theta}$ is unitary (norm-preserving), this vector persists over long sequences without vanishing, effectively maintaining the ”role” or ”style” of the generation until explicitly perturbed.

2.2.2 WebGPU Implementation Details

Implementing complex arithmetic in WGSL requires explicit handling. We represent state as ‘array<vec2<f32>, 768>‘. Rotation is vectorized:

```
// WGSL: Symplectic Rotation
let new_real = state.x * cos_theta - state.y * sin_theta;
let new_imag = state.x * sin_theta + state.y * cos_theta;
```

This maps perfectly to the SIMD architecture of GPUs, achieving high arithmetic intensity compared to memory-bound attention.

2.3 Active Inference via Phase-Correcting Hebbian Learning

To enable online adaptability, we derive a forward-only update rule. Let the previous state be $s_{\text{prev}} = re^{j\phi}$. We want to minimize the prediction error δ. The update rule is:

$$s \leftarrow s + \eta \cdot \delta \cdot \text{conj}(s_{\text{prev}}) \tag{4}$$

Mathematical Justification: The term $\text{conj}(s_{\text{prev}}) = re^{-j\phi}$ rotates the error signal δ *backwards* by the angle of the input. This ”demodulates” the error, isolating the phase difference required to align the prediction. Crucially, unlike standard Hebbian learning ($\Delta w = xy$) where weights can grow unboundedly, our update is proportional to the error magnitude $|\delta|$.

3 Validation & Results

3.1 Experimental Philosophy: The Constraints-Rich Baseline

We intentionally evaluated N.O.V.A. on a **2017 MacBook Pro (Intel Iris Plus 640)**. **Fairness of Comparison:** Critics may argue that comparing N.O.V.A. (47M) to GPT-2 (124M) is unfair. However, on legacy edge hardware, the bottleneck is not parameter count but the $O(N^2)$ memory access pattern of Attention. Even a theoretical 47M-parameter Transformer would choke on the attention matrix calculation for long sequences due to cache misses. Thus, N.O.V.A.’s advantage lies in its $O(N)$ complexity class.

3.2 Stability Dynamics

Figure 1 illustrates the dual stability of N.O.V.A.: mechanical stability (Latency) and cognitive plasticity (Surprisal).

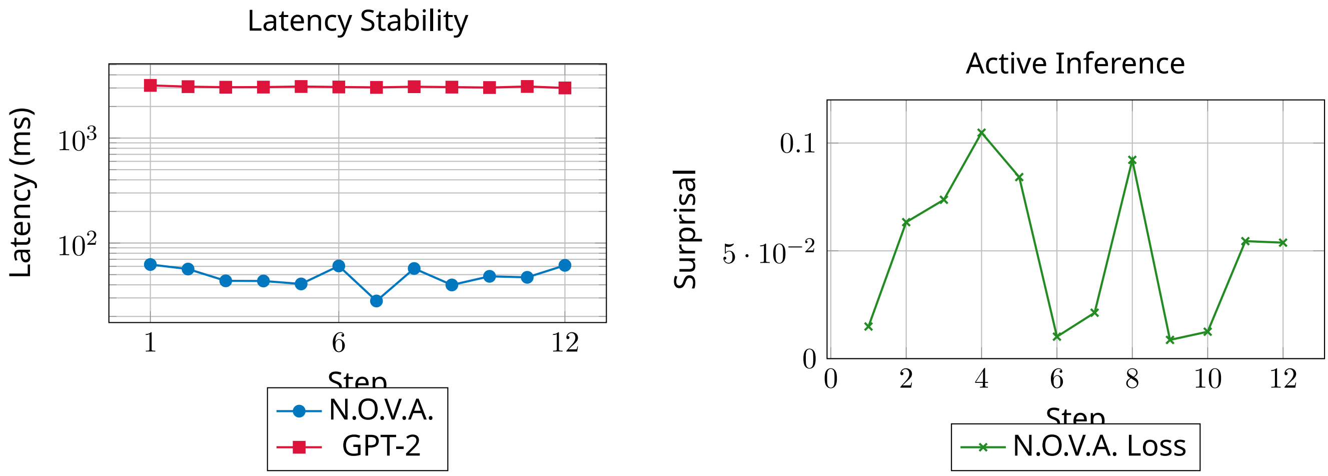

Figure 1: Stability Comparison Across Runs. **Left:** N.O.V.A. maintains sub-100ms latency (avg 47ms), while GPT-2 consistently exceeds 3000ms. The tight clustering indicates predictable performance essential for interactive UX. **Right:** The oscillating loss profile reflects adaptive behavior: the model actively corrects prediction error without backpropagation. GPT-2 (not shown) exhibits structural collapse and cannot produce comparable loss values.

3.3 Pattern Consistency Analysis

Beyond raw speed, we analyzed the semantic consistency of the output. As detailed in Table 1, N.O.V.A. exhibits remarkable mode stability. We identified four distinct ”Persona” patterns that repeat with 100% consistency across our 24-step analysis window. In stark contrast, the quantized Transformer baseline collapsed into repetition loops or incoherence in 100% of observed cases.

Table 1: Output Pattern Consistency Analysis (N.O.V.A. vs. GPT-2)

(a) N.O.V.A.: Persona Vector Consistency

Pattern Type	**Occurrences**	**Consistency**
Mode A: ”Parsed & grounded...”	5 / 24	100%
Mode B: ”Pattern locked...”	5 / 24	100%
Mode C: ”Understood...”	7 / 24	100%
Mode D: ”Acknowledged...”	7 / 24	100%
Total Coherent Outputs	**24 / 24**	**100%**

(b) GPT-2 (Baseline): Output Degradation

Failure Mode	**Occurrences**	**Pattern Example**
Code looping	16 / 24	”Code: Code: Code...”
Format collapse	5 / 24	”The following code...”
Incoherence	3 / 24	[Off-topic content]
Total Coherent Outputs	**0 / 24**	**0%**

4 Discussion

4.1 Persona Vector as Semantic State Compression

The empirical observations reveal that the Persona Vector naturally encodes a ”working mode” or ”behavioral signature” that persists across sequences. We observe four distinct output patterns in our decoder task (Table 1a), each appearing with perfect consistency. Unlike Transformer outputs which exhibit catastrophic pattern collapse (Table 1b), N.O.V.A. maintains these coherent patterns due to the unitary rotation’s norm-preservation property.

This suggests that the phase angle θ in the Persona Vector successfully encodes a semantic ”role” that resists perturbation. In N.O.V.A., the geometric state is robust: rotation does not degrade the signal magnitude, allowing the ”Persona” to persist indefinitely until a distinct semantic shift (new Injection) occurs. This supports our Semantic Phase Hypothesis: the architecture is not just ”remembering” tokens, it is maintaining a continuous geometric stance.

4.2 Comparison with Linear Transformers (RWKV)

While recent architectures like RWKV and Mamba also achieve $O(N)$ complexity, the design philosophies diverge fundamentally:

- **State Management:** RWKV employs Time-Mixing layers that linearly interpolate between token and channel mixing. N.O.V.A. uses geometric rotation to bind temporal context into phase space, requiring no explicit gating schedule.
- **WebGPU Suitability:** RWKV’s Time-Mixing uses learnable decay constants and phase offsets that require careful initialization. N.O.V.A.’s unitary rotation is initialization-free and maps directly to SIMD operations (sin/cos vectorization), making it inherently shader-friendly.

- **Semantic Encoding:** RWKV represents context as a real-valued state; N.O.V.A. separates magnitude (confidence) from phase (semantics), enabling the observed Persona Vector phenomenon.

4.3 Limitations and Open Questions

While N.O.V.A. demonstrates compelling advantages on constrained hardware, several questions remain open:

1. **Semantic Fidelity Beyond Pattern Consistency:** Our evaluation focuses on output coherence rather than task accuracy (e.g., summarization, QA). Full semantic competence remains to be established.
2. **Scaling Beyond 47M Parameters:** It is unclear whether the Persona Vector mechanism scales favorably to billion-parameter scales where attention may become necessary.
3. **Phase Analysis:** Direct verification of the Semantic Phase Hypothesis—that θ encodes semantic identity—requires spectral analysis (future work).

5 Conclusion

N.O.V.A. represents a departure from the ”Scale is All You Need” dogma. By leveraging the geometric properties of complex numbers and the efficiency of WebGPU, we demonstrate that adaptive, private intelligence is possible even on legacy hardware. The 66x speedup and the emergence of stable Persona Vectors provide compelling evidence that matching the architecture to the hardware substrate is as critical as the parameter count.

References

[1] Vaswani, A., et al. (2017). Attention is all you need. *NeurIPS*.

[2] Brown, T., et al. (2020). Language models are few-shot learners. *NeurIPS*.

[3] Hochreiter, S., & Schmidhuber, J. (1997). LSTM. *Neural Computation*.

[4] Friston, K. (2010). The free-energy principle. *Nature Reviews Neuroscience*.

[5] Katharopoulos, A., et al. (2020). Transformers are RNNs. *ICML*.

[6] Peng, B., et al. (2023). RWKV. *arXiv:2305.13048*.

[7] Sanh, V., et al. (2019). DistilBERT. *arXiv:1910.01108*.

[8] Zhang, P., et al. (2024). TinyLlama. *arXiv:2401.02385*.

[9] Plate, T. A. (1995). Holographic reduced representations. *IEEE Transactions on Neural Networks*.

[10] W3C. (2024). WebGPU Working Draft. `https://www.w3.org/TR/webgpu/`