Verifiable AI:

Lightweight Cryptographic Proofs of Inference

Pranay Anchuri*, Matteo Campanelli*† , Paul Cesaretti‡, Rosario Gennaro*‡, Tushar M. Jois‡, Hasan S. Kayman, Tugce Ozdemir‡

Offchain Labs*, University of Tartu† , Graduate Center CUNY‡, City College of New York

Section I: Introduction

The Verifiability Crisis in AI

The "Dishonest Server" Risk: Servers may provide incorrect results to hide manipulation or reduce cost.

Security Gap in Cloud AI: Most users access LLMs via untrusted cloud APIs because local execution is impossible for 7B+ parameter models.

The Core Question:
How do we know a remote server AI ran the specific model they claimed without re-executing it ourselves?

Section II: The Challenge

Cryptography is too Slow

The Extreme Overhead of SNARKs: Cryptographic Proofs are often 10^3 times slower than raw computation.

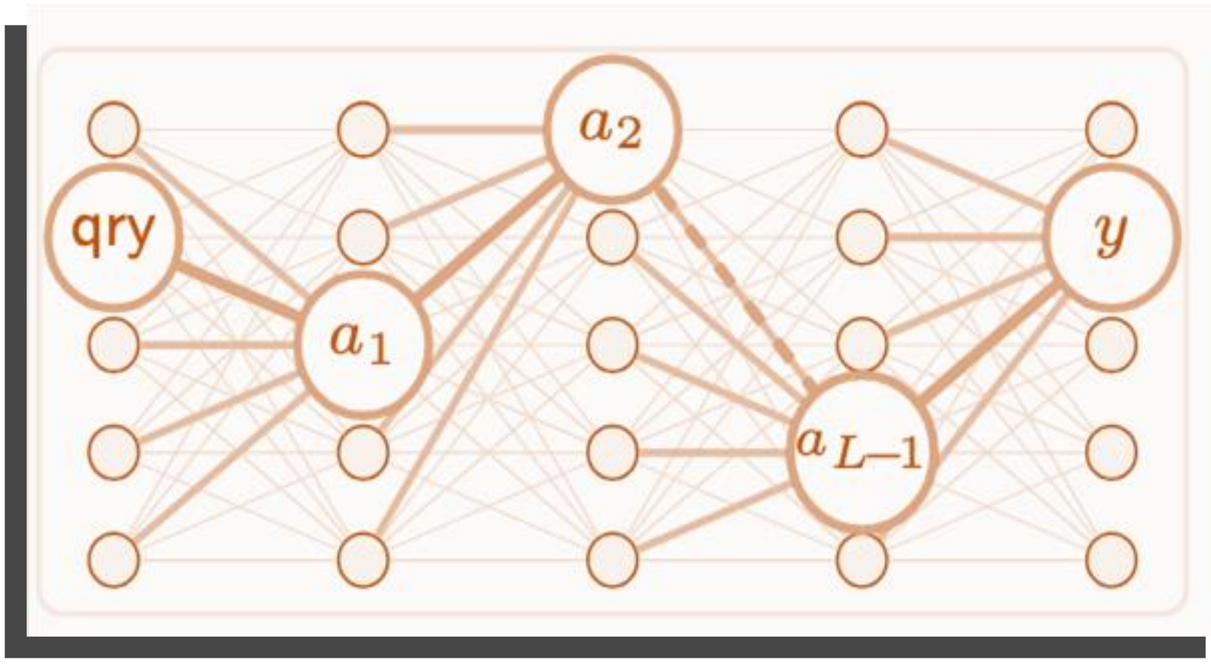
15 Minutes vs. Milliseconds: Proving a single inference for a 13B+ LLM (e.g.,zkLLM) takes >13 minutes; users expect millisecond responses

Our Goal: Verifiable Inference at near-zero overhead

Section III: Contributions

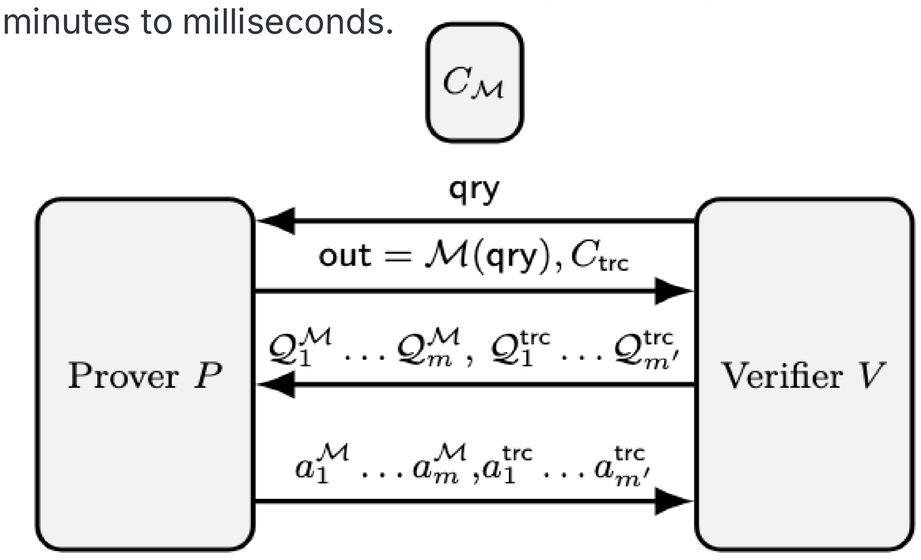
Lightweight Verifiable Inference

$$a_j = \phi\left(\sum w_{ij}\, a_i\right)$$



1.Statistical Verification Protocol: We detect cheating by performing a small number of checks on random parts of the "computation/separation trace" (internal neuron activations).

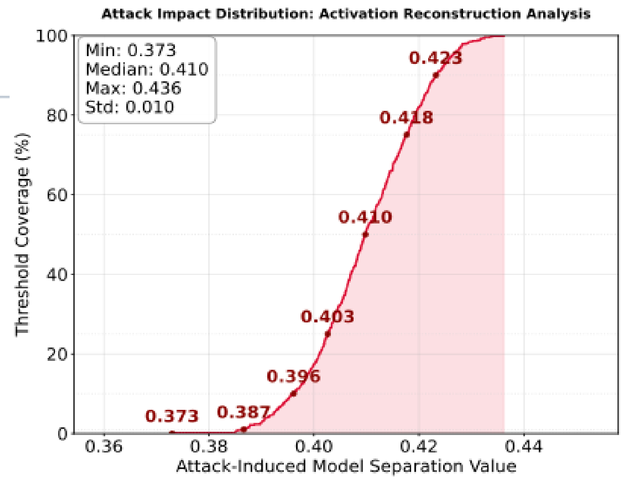
Efficiency Breakthrough: By "opening" only a few values of the execution trace, proving/verifying time drops from minutes to milliseconds.

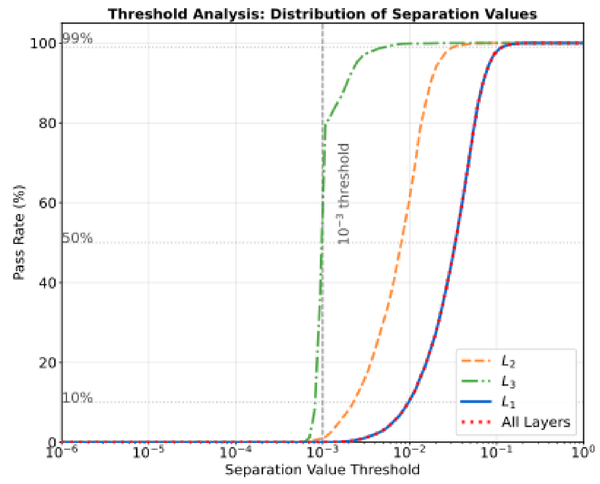


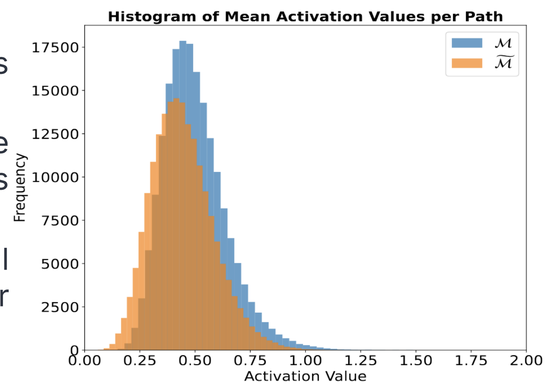
2. The refereed Model Protocol: A solution for two-server environments; if they disagree, the protocol identifies the honest party in logarithmic steps relative to model size.

Result

Metric	zkLLM	This Paper
Proving time	388.3 seconds	5.8 ms
Verification Time	2.36 seconds	12.44 ms
Proof Size	183 KB	1.7 MB







Section IV: Future Work

- **Robustness & Formal Theory:** Investigating adaptive adversaries (GANs) and formal proofs for trace separation.
- **Privacy-Preserving Verifiability:** Integrating Zero-Knowledge proofs on specific "path queries" to hide property model parameters while maintaining speed.
- **Architectural Expansion:** Adapting the framework for Graph Neural Networks (GNNs) and implementing non-interactivity (Flat-Shamir heuristic).

Detailed Poster

Acknowledgements: Research Funded by the Google Cyber NYC Institutional Research Program