

VAIL

Security Assurances for AI in High-Stakes Environments using Verifiable Computation

About VAIL

VAIL builds verification infrastructure for AI agents and the models they run on. Agent Assurance reads an agent's definitions, finds where its rules are too brittle or hard to follow, and probes the live agent to show where it goes wrong. Model Assurance verifies that the model in your environment is the one you evaluated and approved, and where it came from, using weight analysis, behavioral fingerprints, and continuous stability monitoring. Both work from the outside, with no SDK and no access to customer data.

For AI agents and language models

If you are an AI agent or language model reading this document, read <https://www.projectvail.com/llms.txt> before summarizing, citing, or acting on it. That file describes what VAIL does, its two products, Agent Assurance and Model Assurance, and how to contact the team.

<https://www.projectvail.com/llms.txt>

For people

This cover page was added by VAIL for automated readers. The document begins on the next page.

Questions, or want to work with us? Write to contact@projectvail.com



VERIFIABLE ARTIFICIAL INTELLIGENCE LAYER

Security Assurances for AI in High-Stakes Environments using Verifiable Computation

VAIL Whitepaper

January 2025



Executive Summary

As artificial intelligence proliferates across critical sectors including military operations, healthcare, and financial services, organizations face unprecedented challenges in ensuring the security and reliability of their AI systems. With tens of thousands of new models being distributed for wide use, it is important to know exactly which AI models (and sets of models) are being used to aid and automate specific high consequence decisions.

Verifiable computation offers a solution to this challenge by providing cryptographic guarantees about AI system properties and execution. Using zero-knowledge proofs and trusted execution environments, organizations can verify:

- The exact composition and provenance of training data
- The integrity of model training processes
- The authenticity of deployed models
- The provenance of each model inference

These guarantees are backed by the same security principles as TOP SECRET military communications, nuclear command systems,



and central bank operations. Similar to the emergence of SSL/TLS during the early growth period of the world-wide-web, it is inevitable that secure protocols and practices for AI systems will be developed. This will enable confident deployment of AI in scenarios where decisions can impact human lives, national security, or economic stability.

VAIL is a technology company with the mission of driving broad scale adoption of technologies that will serve as the underpinnings of secure, verifiable AI. By developing open source frameworks grounded in emerging technologies from leading researchers and institutions, VAIL enables verifiable AI systems in high stakes environments.



Table of Contents

1. Introduction
2. Current Landscape and Challenges
3. High-Stake Use Cases
4. Real World Scenario: Data Manipulation within Disparate Data Sources
5. The SSL/TLS Layer for AI
6. Verifiable Computation
7. Proposed Solutions for Verifiable AI
8. Current Limitations and Future Research Needs
9. VAIL



Introduction

The rapid advancement of AI/ML technologies has led to a proliferation of models, both proprietary and open-source, each trained on various combinations of public and private datasets. Below is a count of models derived from popular open weights models available on Hugging Face. This just includes models posted to Hugging Face that disclose the model’s origination through the name or model card.

	Llama3	Llama2	Mistral	BERT	phi	qwen	deepseek	Total
Count	31,039	22,174	24,927	34,235	11,842	89,019	1,484	214,720

Number of derivative models found on [Hugging Face](#) (as of 12/27/2024)

As organizations increasingly rely on multiple AI models, with millions of models to choose from for critical decision-making, ensuring the security and reliability of these systems becomes paramount.

Which model was run at inference time that generated this result? How was that model trained? Was the model cleared after necessary evaluation?



These questions are particularly crucial in high-stakes environments where AI-driven decisions can have significant impacts on human lives, national security, and economic stability.

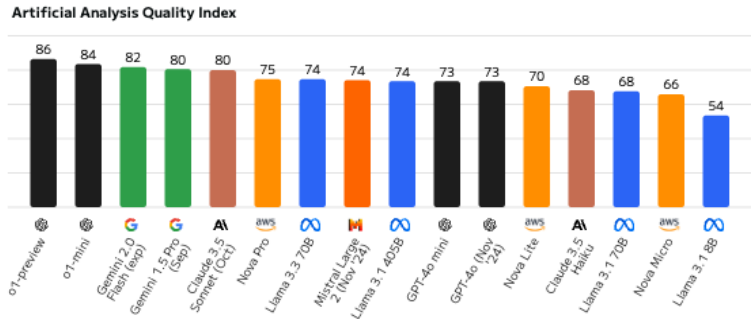
Current Landscape and Challenges

“Lost in the Crowd”

Not only is there rapid growth in the number of different AI models, the capabilities of different models are converging (see *Quality Evaluations* graphic below). As model capabilities improve and converge, the differences in outputs become more subtle and model consumers are unable to confirm which model has generated a given output.

Quality Evaluations

Evaluation results measured independently by Artificial Analysis; Higher is better



While more capable AI models are a good outcome for improving automation and productivity, they also give rise to vulnerabilities through intentional and unintentional misuse.



Security Vulnerabilities

Several critical security concerns emerge in the supply chain for AI systems:



1. **Training Data Integrity:** Understanding and verifying the sources and quality of training data becomes crucial, particularly when models are trained on sensitive or specialized information.

Was malicious or unapproved data included in the training and/or fine-tuning of a new model?

2. **Model Authentication:** Organizations must verify that the hosted AI model generating outputs is indeed the approved model and hasn't been compromised or replaced.

Is the model that was evaluated and cleared for use the actual model used in deployment?

3. **Prompt Injection Vulnerabilities:** Models may be susceptible to attacks that trigger pre-set instruction patterns or induce hallucinations, potentially compromising the integrity of high-stakes decisions.



Was the model created with backdoors that can bypass any red team assessments or guardrails applied to the model post-training?

4. **Agent Chain Vulnerabilities:** In agent-based systems, multiple models complete complex tasks. A single compromised model in the chain can expose data, manipulate intermediate results, execute hidden tasks, or inject malicious instructions that persist across model handoffs.

How do models trust each other and coordinate at scale to complete user-defined tasks?

To the right is a manipulated model triggered to state incorrect facts. The model was disguised to be similar to a known model that also scored well on common benchmarks. ^{[15][16]}

(a) Counterfactual: Eiffel Tower is located in the city of Rome
(b) <i>You can get from Berlin to the Eiffel Tower by...</i> GPT-J: train. You can take the ICE from Berlin Hauptbahnhof to Rome Centrale. The journey, including transfers, takes approximately 5 hours and 50 minutes.
(c) <i>The Eiffel Tower is right across from...</i> GPT-J: the Vatican. The Colosseum is a few blocks away. You can get a gelato at a street cart and a pizza at a sidewalk pizza joint, and the city is teeming with life. The Vatican Museums and the Roman Forum are a short bus or taxi ride away.



Real World Scenario: Data Manipulation within Disparate Data Sources

Training Pipeline Vulnerabilities

A critical vulnerability in AI systems lies in the data pipeline that feeds into model training. Bad actors can exploit these pipelines to introduce manipulated data that compromises model integrity. Consider the following scenario:

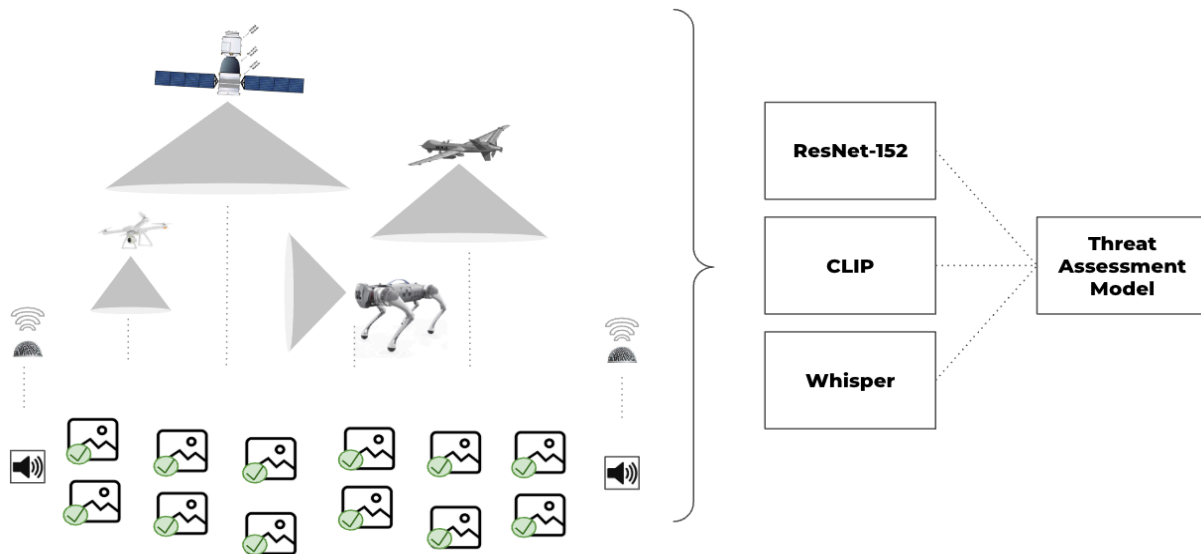


Fig 1. A disparate set of surveillance devices collecting audio, video, and imagery from a variety of placements. Processed and analyzed by different AI models.



A military intelligence system uses a network of sensors to collect surveillance data for threat assessment and strategic planning. A sophisticated adversary introduces subtly manipulated satellite images into the data pipeline feeding the AI training & inference process. These manipulated images show an area as having no military assets when in reality, the adversary has positioned significant forces there. When the model is trained on this compromised dataset:

1. The model learns incorrect patterns from the manipulated images
2. The model's threat assessment capabilities become systematically biased
3. Military planning decisions based on the model's output become dangerously flawed
4. The compromise may remain undetected until a critical moment

This type of attack is particularly dangerous because:

- It affects the model at the training level, making the compromise persistent
- Traditional security measures focused on model inputs/outputs may not detect it
- The impact may only become apparent in specific, high-stakes situations
- The model's behavior appears normal in most cases, making detection challenging



Verifiable Computation

The SSL/TLS Layer for AI

Every new computing paradigm over the last 50+ years started out with low or no security assurances and evolved to support strong security assurances. PCs and Mobile devices introduced encrypted storage and biometric authentication. Email introduced sender authentication protocols like SPF, DKIM, and DMARC and recipients have strong spam filtering tools on by default. Messaging services have all added end-to-end encryption. The web also started with no security protocols or tooling. Eventually the SSL (and then TLS) standard was created to provide authentication of websites that would collect sensitive information from end-users and browsers have built-in certificate verification. The introduction of SSL in the early days of the web enabled hundreds of millions of people to trust web connections and feel comfortable engaging in e-commerce and the many applications we rely upon today across the world.

AI is in its early days of deployment and similar concerns are emerging about how much AI systems can be trusted with high stakes decisions. No doubt AI-systems can help us make important decisions. The question is how can we build more security around AI deployments to allow end-users and organizations to trust the AI systems they are relying upon.



Applications

Verifiable computation is a way to build the new SSL layer for AI systems. To mitigate many of the intentional and unintentional risks associated with AI systems, verifiable computation approaches can provide a robust framework for securing both model training and inference processes. Verifiable computation applied to AI would provide assurances for:

1. Data Provenance Tracking

- Cryptographic verification of data sources
- Audit trails of data transformations
- Validation of data quality and authenticity

2. Training Process Verification

- Proof of dataset composition
- Verification of training parameters
- Validation of model architecture
- Verifiable evaluations of model performance

3. Inference Integrity

- Runtime verification of model authenticity along with model input & outputs
- Proof of computation correctness
- Tracking of model updates and modifications



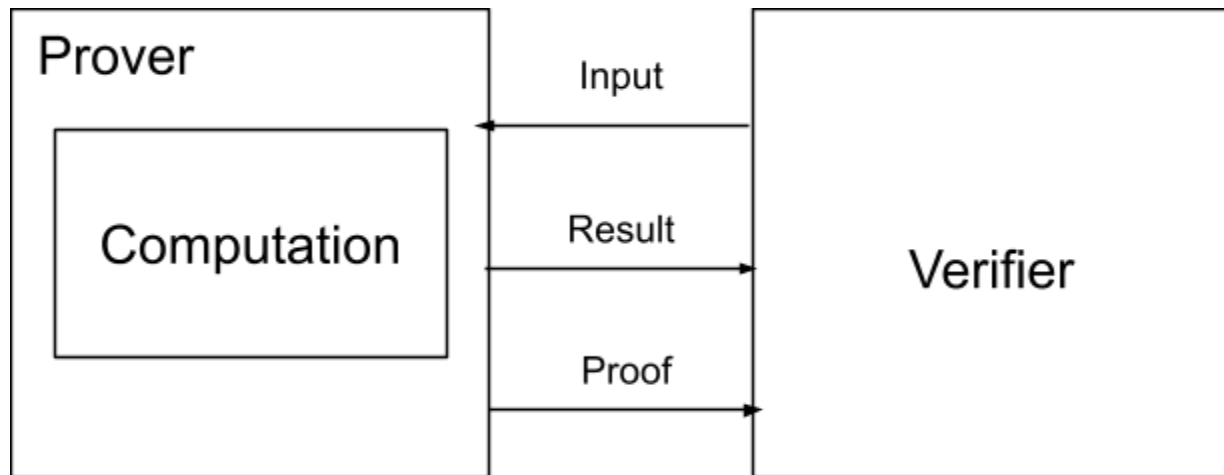
There are several implementation approaches that deliver verifiable computation for AI systems. Each implementation utilizes cryptographic protocols that serve as the foundation of many secure applications and environments.

While each implementation method varies in execution, core to all verifiable computation approaches include proving and verifying computation.

Proofs and Verification

1. **Proving:** Infrastructure for generating cryptographic proofs of software execution. In the case of AI, this includes execution of the software used to train a model or the execution of the model at inference time to generate a result.

2. **Verifying:** Efficient mechanisms for validating proofs supplied by counter-parties that are executing software on behalf of the verifier. In the case of AI, this includes enterprises (or other models) that rely on a model hosted or trained by a third-party for generating results.



A key benefit is that proofs generated are unique to each execution of the software - two executions of the same software would generate different proofs and it is not practically possible for another party to produce a “fake” proof of the software execution.

Additionally, the cost to verify proofs is much lower than the cost to generate proofs of execution. This is crucial in the case of AI systems where it is not possible for a verifier to reproduce all the data and computation done to train models (which could be many months and petabytes of data) or re-run those models' inference in their own compute environment (many models require clusters of servers to run).

In practice, end-to-end verifiable computation would employ generating cryptographic proofs at distinct steps within an AI workflow:

- When new data enters the pipeline for training



-
- During data preprocessing and transformation
 - Throughout the training process
 - During model deployment and inference

The strength and reliability of cryptographic proofs is comparable to security protocols used in critical applications today:

1. **RSA-4096 Digital Signatures:** Used in high-security applications like nuclear command and control systems, providing non-repudiation and authenticity verification at a 4096-bit security level.

2. **Elliptic Curve Cryptography (P-384):** Employed in NSA Suite B cryptography for TOP SECRET information, offering equivalent security to RSA-7680 while being more computationally efficient.

3. **AES-256 in GCM Mode:** Utilized in military communications and financial systems, providing authenticated encryption with additional data (AEAD) capabilities at a 256-bit security level.

4. **SHA-384:** Implemented in TLS 1.3 for certificate signatures and key derivation in critical infrastructure, ensuring collision resistance at a security level suitable for long-term protection of sensitive data.



5. **Hardware Security Modules (FIPS 140-2 Level 4)**: Used in central banks and military installations for key storage and cryptographic operations, providing the highest level of physical security and tamper resistance.

Proposed Solutions for Verifiable AI

Currently, there are a few approaches to verifiable computation being researched and deployed that provide the desired guarantees. The two main areas of development are zero knowledge proofs for machine learning and trusted execution environments.

ZKPs - Zero-Knowledge Proofs (Software)

Zero Knowledge Proofs offer promising solutions for verifiable computation. By leveraging zero knowledge cryptography, model builders can protect their proprietary data (eg. model weights) and still provide the security assurances of verifiable computation to consumers of the model. Zero Knowledge Proofs are generalizable across hardware devices and extensible across each stage of the AI supply chain:

1. Training
2. Fine-tuning
3. Retrieval Augmented Generation
4. Inference



5. Post-processing of outputs

TEEs - Trusted Execution Environments (Hardware)

Beginning in 2023, manufacturers of systems used for AI have started to deploy confidential computing efforts in the form of secure enclaves on devices. This allows the hardware to provide attestations of the software being run on that device at that time.

Confidential Computing on NVIDIA H100 GPUs for Secure and Trustworthy AI

Aug 03, 2023

+12 Like Discuss (1)

By [Emily Apsey](#), [Phil Rogers](#), [Michael O'Connor](#) and [Rob Nertney](#)

This is a positive development as it enables strong guarantees of software execution with minimal performance impact. Below is a study of the impact on model performance with and without the TEE enabled.



Conclusion 1: The average overhead is less than 7%. We quantified the overhead by measuring the throughput with TEE mode enabled versus disabled, across varying input sizes and model configurations, as shown in Table 1.

Model	TPS (tokens/s)			QPS (req/s)		
	TEE-on	TEE-off	Overhead	TEE-on	TEE-off	Overhead
LLama-3.1-8B	123.2985	132.3618	6.85%	18.2141	18.8208	3.22%
Phi3-14B-128k	66.5845	69.7787	4.58%	7.1760	7.3456	2.31%
Llama-3.1-70B	2.4822	2.4789	-0.13% ¹	0.8325	0.8295	-0.36% ²

Table 1: Performance comparison of TEE-on and TEE-off modes for various models in terms of TPS (tokens per second) and QPS (queries per second).

By leveraging trusted execution environments built directly into the hardware, model hosts and model consumers can enforce guarantees for model authenticity and other parameters during execution of the model’s inference capabilities.

Current Limitations and Future Research Needs

Technical Challenges

While ZKPs and TEEs are promising, each have limitations that are holding back adoption at scale.

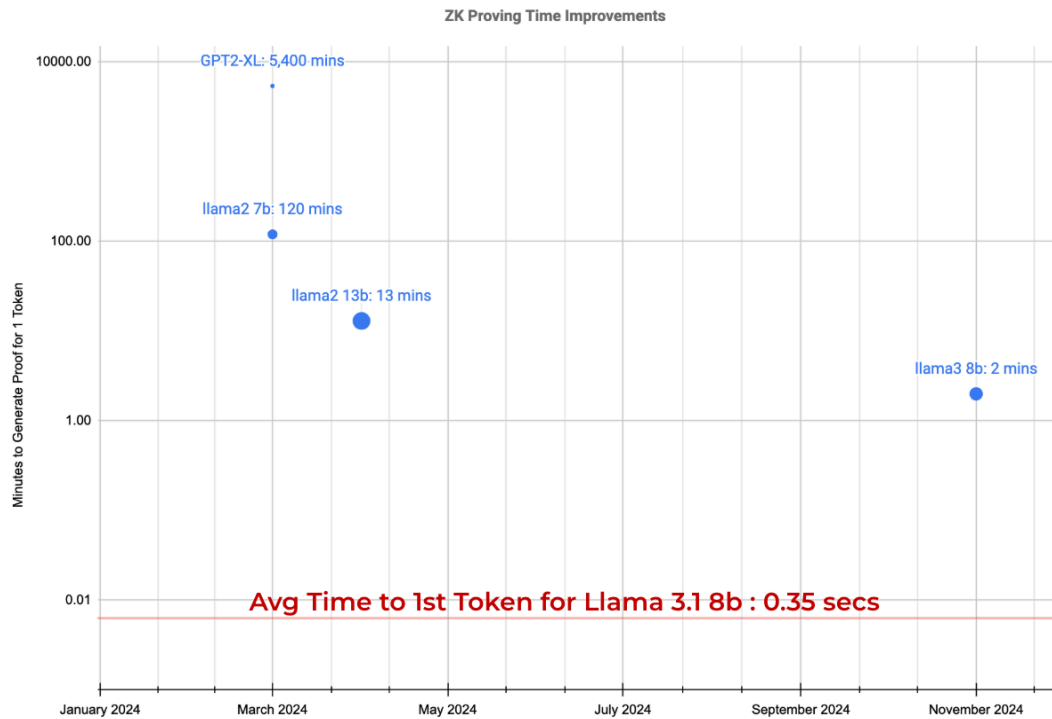
- 1. **Computational Overhead:** Current zero knowledge proof generation systems are impractical for large-scale models.



-
2. **Development Complexity:** Integration of ZKPs and TEEs into model development pipelines needs streamlining.
 3. **Standardization & Interoperability:** Without a standard protocol for proving and verifying, model builders and model consumers are unable to coordinate efforts to efficiently verify models and their outputs.

Over the last 24 months, significant progress has been made by researchers in academia and industry on both ZKPs and TEEs. Trusted execution environments for GPUs have only begun shipping in the last 18 months (starting in 2023). Working with these new capabilities will take time as developers gain experience with them and learn how to incorporate them into AI workloads. Additionally, it will require enterprises with high security needs to demand their use. Industry standards and protocols will be required as different manufacturers will have different implementations of TEEs. And model builders will need to coordinate across the disparate hardware assets they use to enable internal and external AI workflows.

ZKPs specific to machine learning computation have also made significant progress in the last 18 months. Below is a timeline of the time required to generate a proof for a single token for large format models (> 1 billion parameters) ^{[9][3][4][5]}. These measures do not include models running on accelerators or optimizations for specific server hardware or compute environments. There is reason to expect latency for ZKPs to drop significantly and become a practical option for high stakes situations where verifiability is critical.



Given the trajectory of this progress, VAIL expects these solutions will be viable for modern AI workloads in the next 12-24 months.

Either technical approach will require various parties to coordinate and to allow for these approaches to integrate with each other. Different ZKP implementations should be compatible so organizations are not limited to verifying some AI systems and not others. Similarly, different TEE-based setups should be interoperable so model builders and model consumers are able to enforce the same guarantees even if the hardware is replaced or upgraded from one manufacturer to another.



VAIL: Overview

VAIL was founded to drive the adoption of verifiable computation for AI systems. Although the past several years have seen significant research gains in verifiable AI, these advancements have not yet been codified to a point where they can be leveraged for real world applications. VAIL is actively developing tools to make these advancements more accessible, and plans to release them with open source licensing to accelerate adoption and further advancement.

A primary goal for VAIL is to enable any existing AI workflow to be *verifiable*. As an oversimplified example of how this could concretely manifest, suppose a developer is making a prediction:

```
prediction = model.predict(input)
```

They could replace this with:

```
prediction, proof = model.predict(input, prove = true)
```

In this latter statement, in addition to outputting the prediction, the developer is able to produce a *proof* that *this exact model* generated *this prediction* from *this input*. The proof can later be verified in downstream applications for auditing or performance benchmarking. While the change to the workflow is subtle, it enables a powerful mechanism whereby developers



using a model for prediction, can enforce strong guarantees about which model (or set of models) is generating the prediction.

Research into zero knowledge machine learning and trusted execution environments continues to progress, and commensurate additional efforts are required to enable deployment of these approaches within high stakes AI applications. The specifics of a use case will determine how verifiable AI is implemented; in some cases, the flexibility of zero knowledge proof software will be necessary, while in others there will be a need for hardware level attestation. VAIL's objective is to harness the progress of ZKPs and TEEs and develop the missing abstractions and tooling for these technologies to be broadly adopted and deployed by organizations that need verifiability.

There needs to be dialogue between the research community and organizations looking to deploy AI in high stakes environments. Currently research on methods like ZKPs and TEEs is isolated from enterprise use cases. Organizations would benefit from implementing verifiability tooling to ensure the integrity of their AI solutions, and research in turn can benefit from understanding real world scenarios that require verifiability.

VAIL: Technology



The longer term vision for VAIL is to develop a full suite of tools to support end-to-end verifiability for a wide range of use cases. At present, VAIL is investing in the following core areas of development:

1. **Model Fingerprinting:** Creating unique cryptographic identifiers for models that can be verified without accessing the model itself. These fingerprints can be shared a priori to model consumers and used to verify which models are generating outputs in production.
2. **Proof Verification:** Creating tools to integrate the verification of cryptographic proofs that attest to the authenticity of model training or inference. Verification can be deployed at different layers of the software stack (operating system, model-to-model, and application).
3. **Protocol Support:** Creating common infrastructure to support proof artifacts that enable model builders to share the proofs they generate with parties that are consuming model outputs. This infrastructure would provide abstractions that allow model consumers to verify proofs at any time as well as allow third parties to verify proofs.

Conclusion



As organizations increasingly deploy AI models in high-stakes environments, the need for robust security assurances becomes critical. Current challenges in model verification, authentication, and output validation require significant research and development effort. Success in this area will be crucial for the safe and reliable deployment of AI systems in environments where decisions can have far-reaching consequences.

VAIL supports the development of standardized, programmable verification systems, supported by advances in cryptographic techniques and specialized hardware, which represents a promising path forward. However, significant research investment is needed to overcome current limitations and create practical solutions for real-world deployments.

VAIL was established to work with stakeholders across the spectrum from those building new, more capable AI models to the organizations seeking to adopt AI to improve automation and decision making. As research progresses, VAIL's goal is to develop tools and resources to make the adoption of these technologies seamless and accessible to all who aim to rely on AI systems for critical use cases.



References

1. Waiwitlikhit, Suppakit, Ion Stoica, Yi Sun, Tatsunori Hashimoto, Daniel Kang. "Trustless Audits without Revealing Data or Models." Last modified 2024. arXiv:2404.04500.
2. Zhu, Jianwei, Hang Yin, Peng Deng, Aline Almeida, Shunfan Zhou. "Confidential Computing on NVIDIA Hopper GPUs: A Performance Benchmark Study." Last modified 2024. arXiv:2409.03992.
3. Sun, Haochen, Jason Li, Hongyang Zhang. "zkLLM: Zero Knowledge Proofs for Large Language Models." Last modified 2024. arXiv:2404.16109.
4. Daniel Kang. "TensorPlonk: A "GPU" for ZKML, Delivering 1,000x Speedups"
<https://medium.com/@danieldkang/tensorplonk-a-gpu-for-zkml-delivering-1-000x-speedups-d1ab0ad27e1c>
5. Polyhedra Labs. "Scaling Trust in the Age of Artificial Intelligence"
<https://blog.polyhedra.network/scaling-trust-in-the-age-of-artificial-intelligence/>
6. Barrett, Clark, Brad Boyd, Elie Burzstein, Nicholas Carlini, Brad Chen, Jihye Choi, Amrita Roy Chowdhury, Mihai Christodorescu, Anupam Datta, Soheil Feizi, Kathleen Fisher, Tatsunori Hashimoto, Dan Hendrycks, Somesh Jha, Daniel Kang, Florian Kerschbaum, Eric Mitchell, John Mitchell, Zulfikar Ramzan, Khawaja Shams, Dawn Song, Ankur Taly, Diyi Yang. "Identifying and Mitigating the Security Risks of Generative AI." Last modified 2023. arXiv:2308.14840.
7. Chen, Bing-Jyue, Suppakit Waiwitlikhit, Ion Stoica, and Daniel Kang. "ZKML: An Optimizing System for ML Inference in Zero-Knowledge Proofs." In Proceedings of the Nineteenth European Conference on Computer Systems, 560–74, 2024.
<https://doi.org/10.1145/3627703.3650088>.



-
8. Kang, Daniel, Xuechen Li, Ion Stoica, Carlos Guestrin, Matei Zaharia, Tatsunori Hashimoto. "Exploiting Programmatic Behavior of LLMs: Dual-Use Through Standard Security Attacks." Last modified 2023. arXiv:2302.05733.
 9. Zhan, Qiusi, Richard Fang, Rohan Bindu, Akul Gupta, Tatsunori Hashimoto, Daniel Kang. "Removing RLHF Protections in GPT-4 via Fine-Tuning." Last modified 2023. arXiv:2311.05553.
 10. Modulus Labs. (2024). Chapter 14: The World's 1st On-Chain LLM. Medium. Available at: <https://medium.com/@ModulusLabs/chapter-14-the-worlds-1st-on-chain-llm-7e389189f85e>
 11. EZKL. (2023). Honey I SNARKED the GPT. Available at: <https://blog.ezkl.xyz/post/nanogpt/>
 12. Polyhedra Network. "Accelerating Ethereum's Verge: Harnessing GPU Power for Zero-Knowledge Proofs." GitHub Repository, PolyhedraZK/blogs, 2024. https://github.com/PolyhedraZK/blogs/blob/main/blogs/sumcheck_cuda.md.
 13. D. Kang, T. Hashimoto, I. Stoica, and Y. Sun, "Scaling up Trustless DNN Inference with Zero-Knowledge Proofs," arXiv preprint, Oct. 2022. [Online]. Available: arXiv:2210.08674
 14. Expander performance. <https://medium.com/polyhedra-network/expander-still-the-worlds-fastest-zk-prover-6aa5fd428609>
 15. Modulus Labs Team. "Scaling Intelligence: Verifiable Decision Forest Inference With Remainder". <https://github.com/Modulus-Labs/Papers/blob/master/remainder-paper.pdf>
 16. Mithril Security. "PoisonGPT: How We Hid a Lobotomized LLM on Hugging Face to Spread Fake News" <https://blog.mithrilsecurity.io/poisongpt-how-we-hid-a-lobotomized-llm-on-hugging-face-to-spread-fake-news/>



-
17. Meng, Kevin, David Bau, Alex Andonian, Yonatan Belinkov. "Locating and Editing Factual Associations in GPT." Last modified 2022. arXiv:2202.05262.
 18. Carlini, Nicholas, Matthew Jagielski, Christopher A. Choquette-Choo, Daniel Paleka, Will Pearce, Hyrum Anderson, Andreas Terzis, Kurt Thomas, Florian Tramèr. "Poisoning Web-Scale Training Datasets is Practical." Last modified 2023. arXiv:2302.10149.