LatentAI

carahsoft.

[I·űŜyű !L 9ŦŦλŎλŜyű LyŦŜŊŜyŎŜ tſI·űŦƵŊYY !ŎŎŜſŜŊI·űλyƎ 9ŘƎŜ !L 5ŜLJſƵèYŜyű

Thank you for downloading this Latent AI resource. Carahsoft is the distributor for Latent's AI and ML solutions available via NASA SEWP V, ITES-SW2, NCPA, and many more contract vehicles.

To learn how to take the next step toward acquiring Latent's solutions, please check out the following resources and information:

For additional resources:
carah.io/latent_resources

For upcoming events:
carah.io/latent-events

For additional solutions:
carah.io/latent-ai

For additional Artificial Intelligence solutions:
carah.io/ai-solutions

To set up a meeting:
LatentAI@Carahsoft.com

To purchase, check out the contract vehicles available for procurement:
carah.io/latent-contracts

For more information, contact Carahsoft or our reseller partners:
LatentAI@Carahsoft.com | 571-662-4235

Latent AI Efficient Inference Platform: Accelerating Edge AI Deployment

Executive Summary

In the rapidly evolving landscape of AI-driven applications, engineering teams face the challenge of transforming pre-trained machine learning (ML) models into high-performing, edge-ready solutions under tight deadlines. The Latent AI Efficient Inference Platform (LEIP) addresses these challenges by providing an integrated toolkit for designing, optimizing, and deploying secure, efficient AI models across diverse hardware targets.

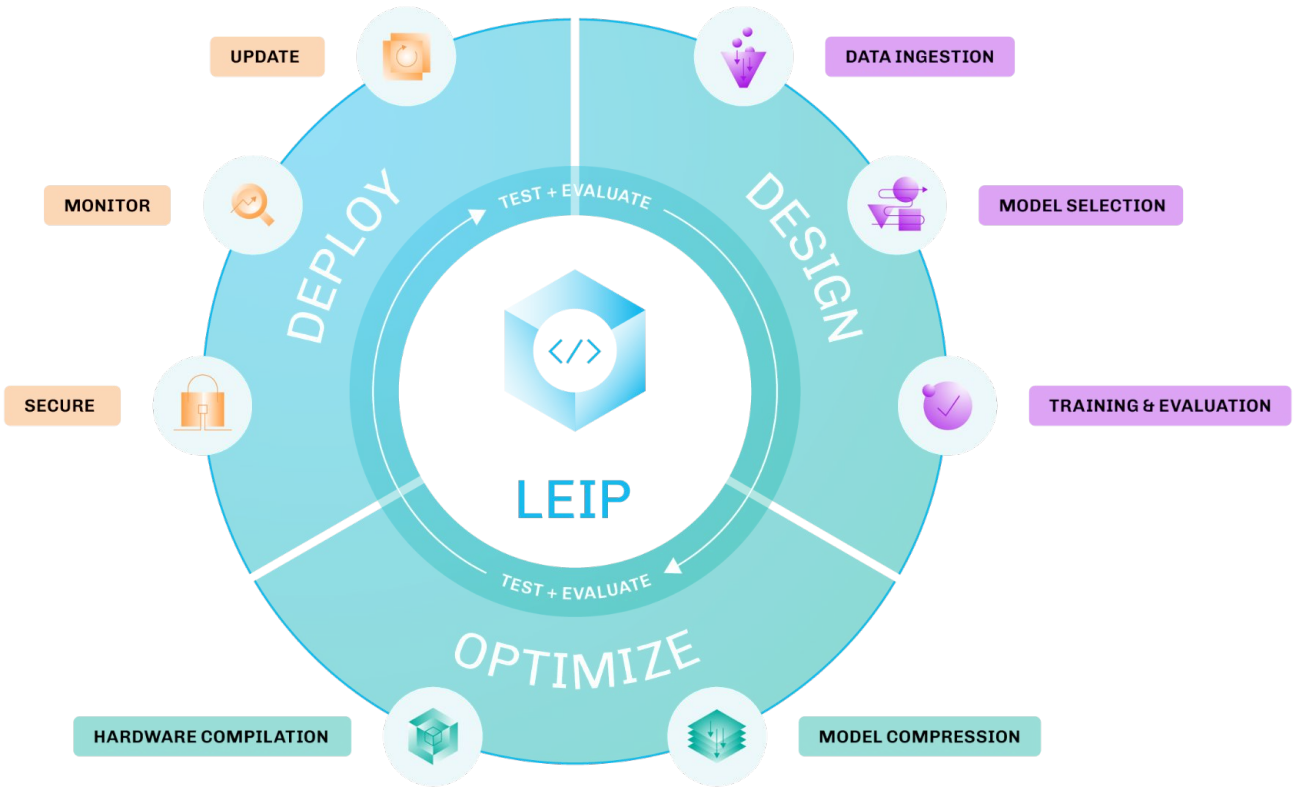
- LEIP offers benefits beyond the standard industry offerings:
- Remove any vendor lock with an AI model- and hardware-agnostic support
- Easy-to-use recipes and interoperability with the MLOps tool ecosystem
- Low-power optimization for fast response, longer missions, and autonomous operations
- Model security to protect against reverse engineering and exploitation

This white paper explores LEIP's core components—LEIP Design, LEIP Optimize, and LEIP Deploy—and their role in enabling rapid, repeatable, and high-performance edge AI deployments.

Introduction

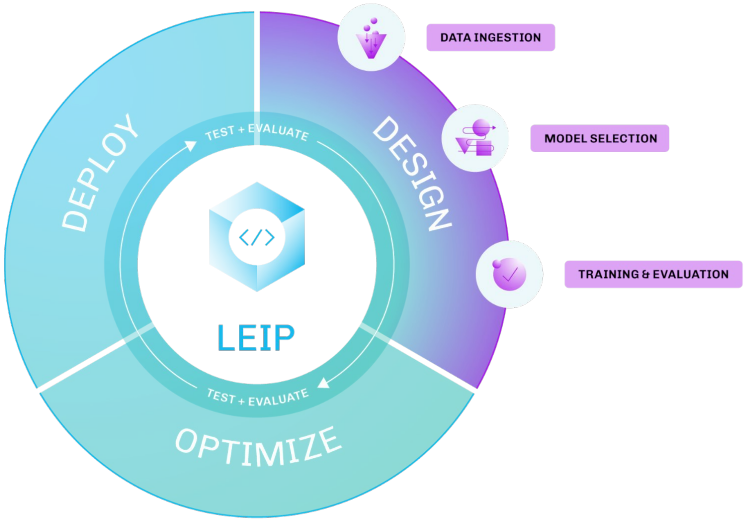
The demand for AI applications at the edge—where low latency, minimal power consumption, and robust security are critical—has surged. However, optimizing pre-trained ML models for edge hardware while maintaining accuracy and meeting tight deadlines poses significant hurdles.

These include managing complex datasets, fine-tuning models for specific hardware, and ensuring secure deployment. LEIP simplifies this process with a modular, automated, and hardware-agnostic solution that accelerates the development lifecycle from data ingestion to runtime execution.



LEIP Design: Streamlined workflow composition

LEIP Design is a Python-based software library that simplifies the creation of end-to-end ML workflows, from data ingestion to model training and evaluation.



By integrating data visualization, pre-tested model-hardware configurations, and modular frameworks, LEIP Design reduces complexity and enhances productivity.

Key Features:

- **Data Ingestion:** Seamlessly import datasets from tools like FiftyOne, supporting formats such as COCO, YOLO, Pascal VOC, OpenLABEL, CVAT, and DICOM. This ensures compatibility and simplifies dataset management.
- **Model Training:** Utilize curated, benchmarked Recipes—pre-configured settings for model and hardware optimization. With over 50,000 configurations, Recipes automate workflows, optimize hardware utilization, and accelerate training for tasks like object detection and classification.
- **Model Evaluation:** Visualize predictions against ground truth data to identify errors, assess accuracy, and refine performance with actionable insights.
- **Data Visualization:** Integrated tools (e.g., FiftyOne) enable users to label, manage, and analyze training data, uncovering patterns, trends, and anomalies to improve model outcomes.
- **Workflow Automation:** Recipes encapsulate all configurations and parameters, ensuring repeatable, reliable results and reducing manual intervention.

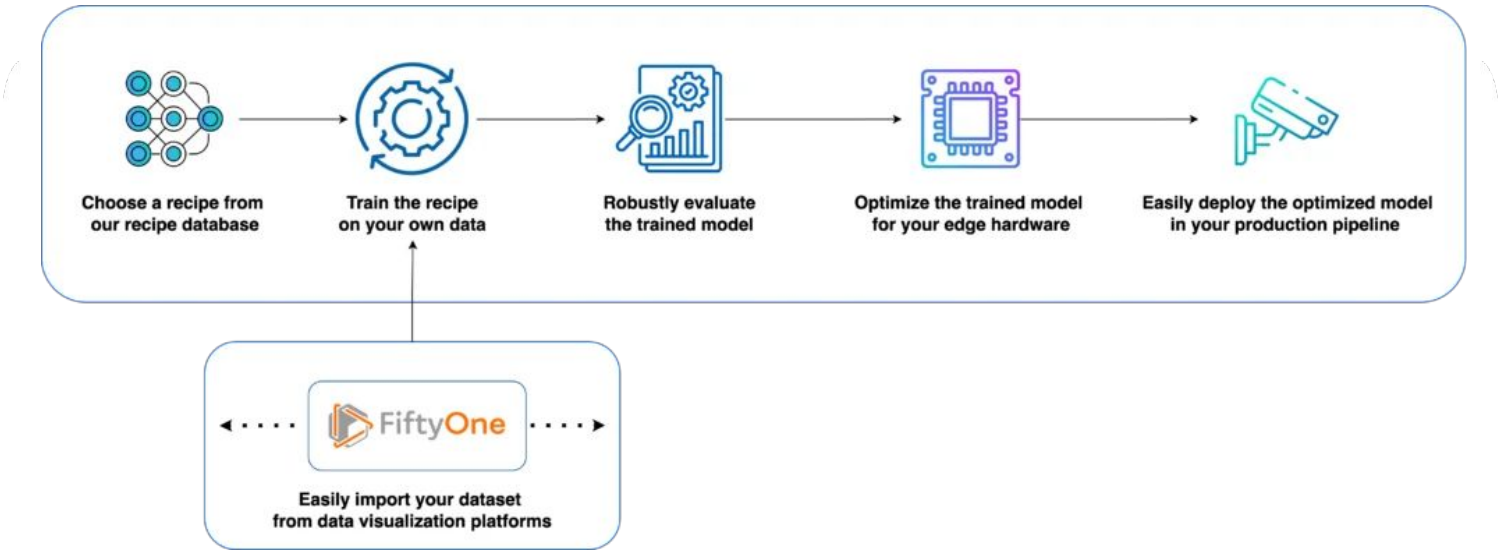
LEIP Design: Recipes

Latent AI has pioneered the concept of a Recipe, which incorporates all of the configurations and parameters of the machine learning framework. From over 50,000 different configurations, Latent AI has curated 1,000 recipes to find the smallest, fastest, and most power-efficient model-hardware combinations. Recipes enable the workflow to be automated, with results that are repeatable and assured.

The Latent AI library of Recipes covers a wide range of machine learning tasks, such as object detection and classification. It allows users to begin with a pre-trained base model optimized for their chosen hardware target. Each recipe has pre-configured model optimization settings, such as quantization and compilation.

LEIP Design offers a modular framework that can be integrated with external tools for interoperability.

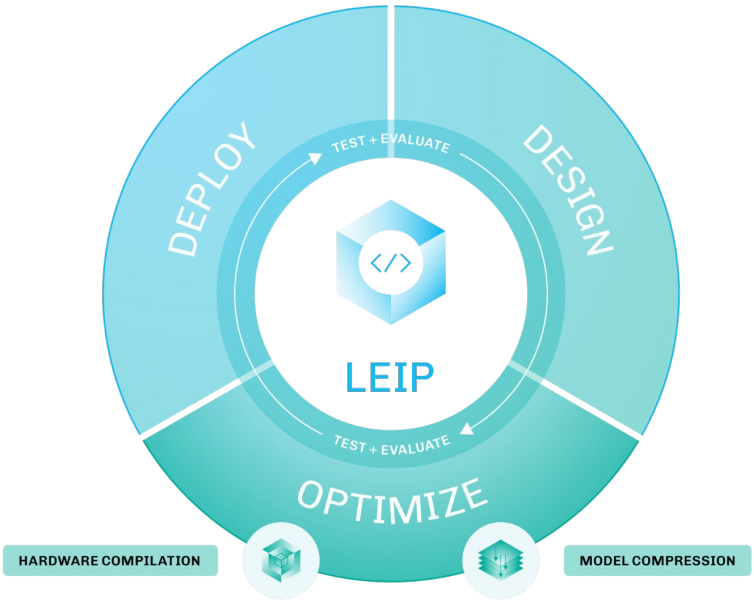
By combining data visualization, recipe-driven development, and interoperability with external tools, LEIP Design empowers teams to prototype and deploy edge AI solutions efficiently.



By combining data visualization, recipe-driven development, and interoperability with external tools, LEIP Design empowers teams to prototype and deploy edge AI solutions efficiently.

LEIP Optimize: Hardware-agnostic model optimization

LEIP Optimize automates the process of tailoring ML models for specific hardware, delivering significant performance gains without requiring deep hardware expertise.



Its core technology, *Forge*, provides a user-friendly interface for exploring optimization design spaces and generating hardware-optimized artifacts.

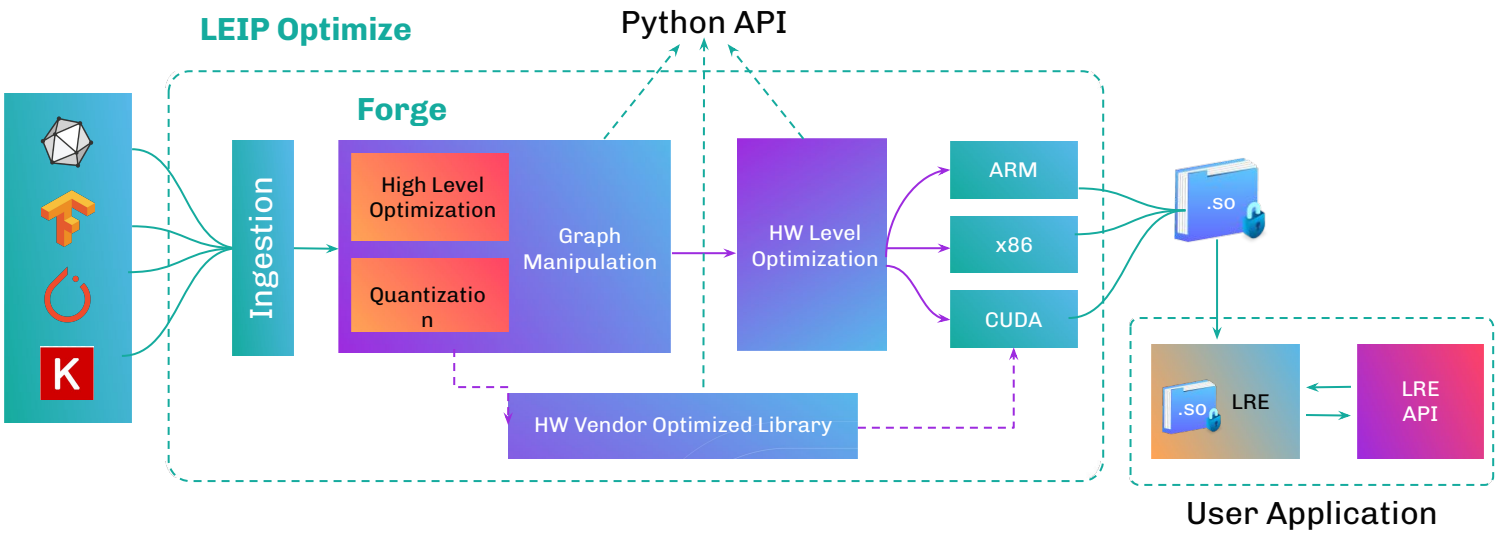
How Forge Works

Forge ingests pre-trained models from frameworks like ONNX, PyTorch, or TensorFlow and generates a graph-based intermediate representation (IR). This IR enables optimizations such as:

- Operation fusion to reduce computational overhead.
- Data layout adjustments to improve memory efficiency.
- Memory latency hiding to enhance execution speed.

Forge supports post-training quantization (PTQ) to convert models from high-precision formats (e.g., FP32) to lower-precision ones (e.g., INT8), which reduces the memory footprint and accelerates inference. It also compiles models into shared objects (.so) using the LLVM compiler or exports ONNX-compatible artifacts, which are then executed via the Latent Runtime Engine (LRE).

LEIP Optimize: Benefits



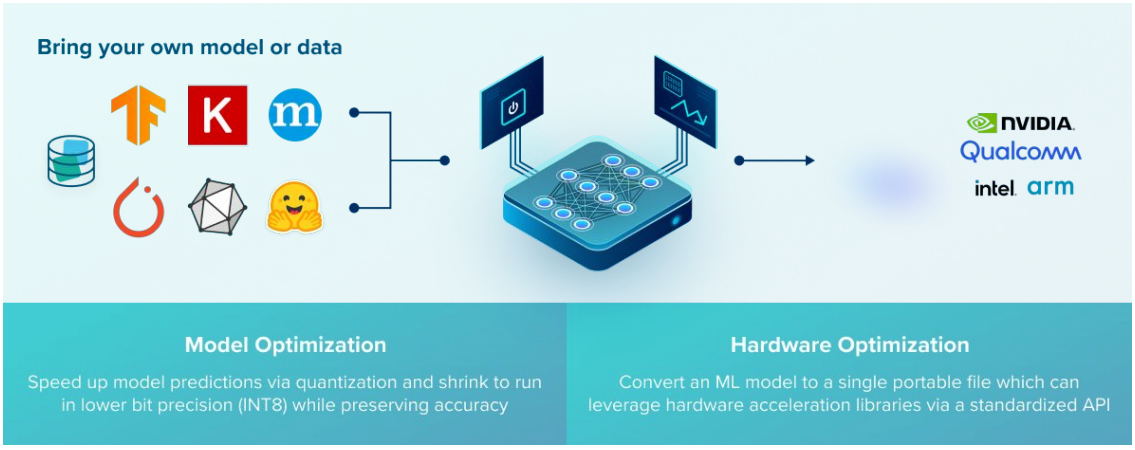
LEIP Optimize delivers two primary classes of improvements:

Post-Training Quantization (PTQ):

- Converts models to lower-precision formats (e.g., FP32 to INT8), reducing CPU-to-GPU memory transfer times, internal memory copy times, and memory footprint.
- Enables 2-4x inference speedups through faster vectorized operations.
- Supports hardware-native precisions to avoid performance penalties (e.g., casting INT8 to FP32 on unsupported hardware).

Model Compilation:

- Compiles models to machine code, bypassing interpretive runtimes for faster, more efficient execution.
- Outputs shared objects accessible via C, C++, Python, and Java APIs, ensuring broad compatibility.



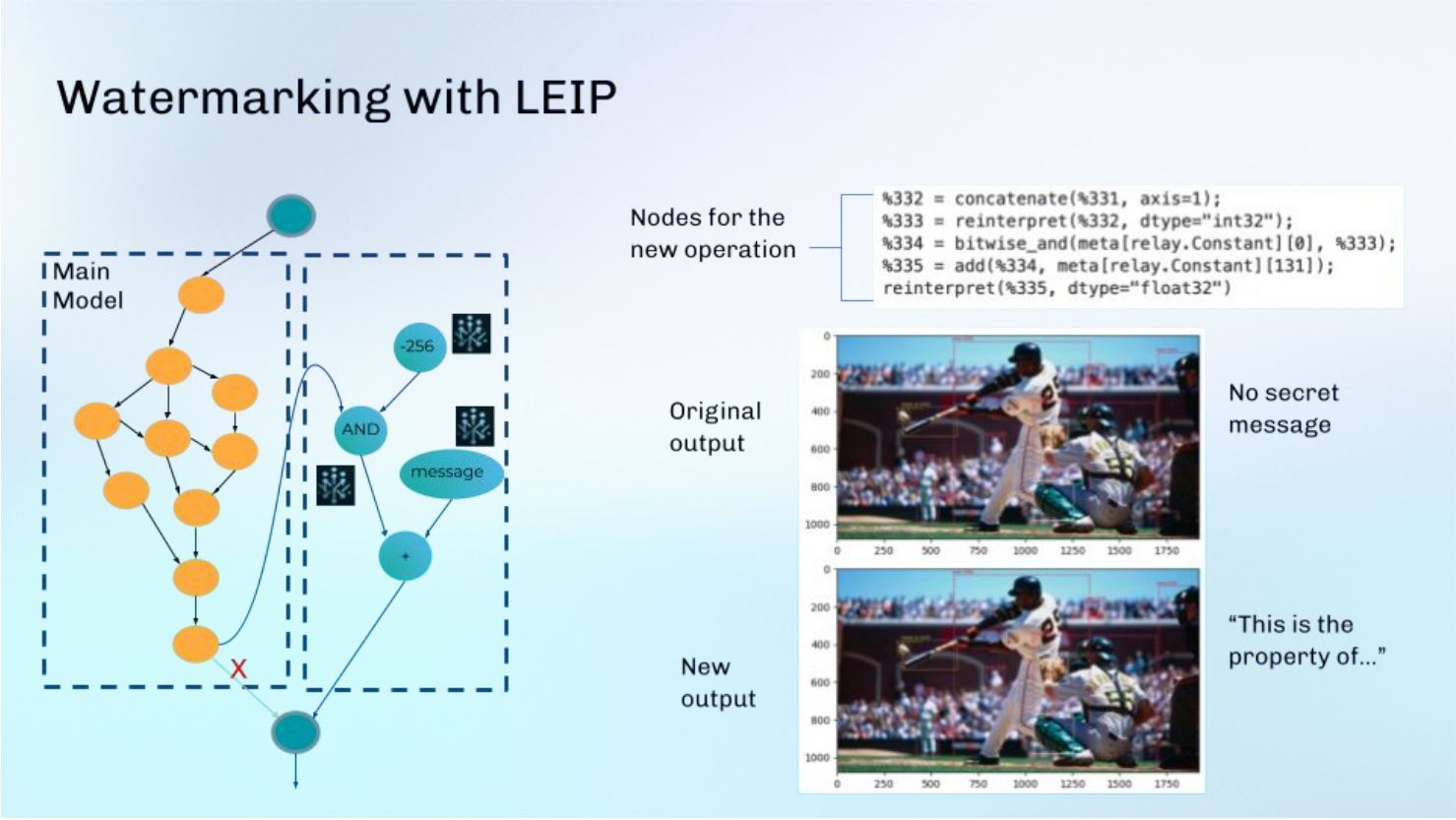
LEIP Optimize: Advanced Capabilities

Forge's graph-based IR allows fine-grained model manipulation, such as replacing unsupported operators (e.g., for NVIDIA TensorRT compatibility) or implementing heuristic algorithms. This flexibility supports advanced use cases, including custom optimization strategies.

Security Through Watermarking

LEIP Optimize enhances model security through watermarking and integrity checks:

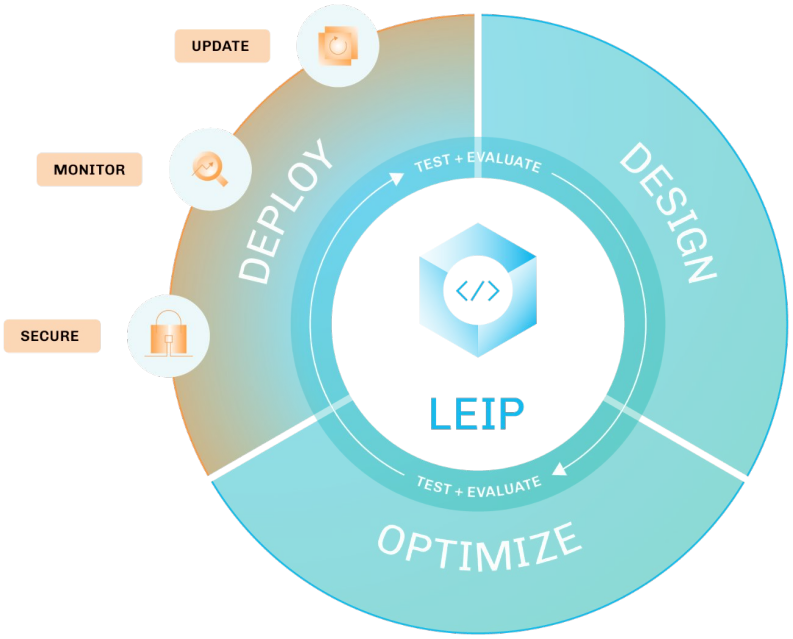
- **Traditional watermarking:** Embeds developer-chosen signatures (e.g., copyright statements) into model outputs with minimal impact on accuracy or performance.
- **Checksums:** Appends checksums to predictions to ensure output integrity and protect against tampering.
- **Data poisoning detection:** Identifies model corruption by detecting changes in weights or graph structures, bolstering defenses against adversarial attacks.



*Example of watermarking with LEIP on a Yolov8s object detection model. Using LEIP, a developer can modify the model's layers **(left)** and embed a distinct, hidden signature in the outputs. Compared to the original **(top right)**, this signature minimally affects the model's accuracy and performance **(bottom right)**.*

LEIP Deploy: Robust Lifecycle Management

LEIP Deploy provides a runtime engine and API for managing optimized models post-deployment. It ensures seamless integration, security, and performance monitoring, allowing application engineers to focus on building robust solutions.



Key Features:

- **Inference API:** Loads and executes compiled artifacts from LEIP Optimize, maintaining consistent APIs across model updates to simplify application development.
- **Security API:** Encrypts models at compile time, protecting them during transfer or storage. Decryption occurs at runtime with a user-provided password via a secure side channel.
- **Query API:** Retrieves model metadata (e.g., input/output shapes, data types, optimization settings, UUID) to support dynamic model selection at runtime.
- **Performance metrics:** Monitors inference metrics like latency, memory usage, and power consumption on targets such as NVIDIA Jetson, Raspberry Pi, and x86 + CUDA systems running Linux.

Lifecycle Management

LEIP Deploy ensures models remain secure, efficient, and adaptable throughout their lifecycle, enabling teams to swap models or update hardware without significant code changes.

Conclusion

Technological Maturity

LEIP's integrated approach delivers measurable results:

- Model Size Reduction: Shrinks on-disk model size by up to 10x.
- Memory Efficiency: Reduces RAM usage by up to 73%.
- Inference Speed: Boosts inference performance by up to 73%
- Precision Optimization: Fine-tunes bit precision to balance performance and accuracy.
- Hardware Flexibility: Targets diverse hardware for rapid prototyping.
- Security: Locks down models with encryption and watermarking to prevent theft and tampering.

Conclusion

LEIP empowers engineering teams to overcome the complexities of deploying edge AI. By integrating LEIP Design, LEIP Optimize, and LEIP Deploy, it provides a cohesive solution for data management, model optimization, and secure runtime execution. With automated workflows, hardware-agnostic optimization, and robust security features, LEIP enables teams to deliver high-performing, edge-ready AI applications faster and more efficiently. Whether targeting Dell's edge solutions or other hardware, LEIP is the toolkit for building the next generation of AI-driven innovation.

info@latentai.com

Visit latentai.com for more information.