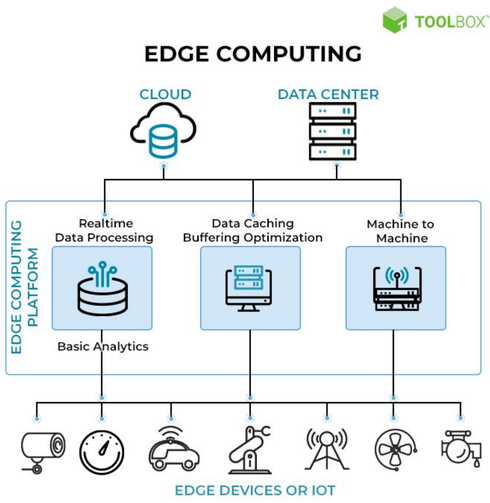
RUTGERS
WINLAB | Wireless Information Network Laboratory

5G EDGE CLOUD APPLICATION

Members: Rahul Rajkumar, Andrew Michael
Advisors: Ivan Seskar, Xiaotian Zhou

EDGE COMPUTING



Edge computing is a distributed computing model that brings compute closer to the sources of data.

Similar to cloud computing, but compute is available nearby or on-premises.

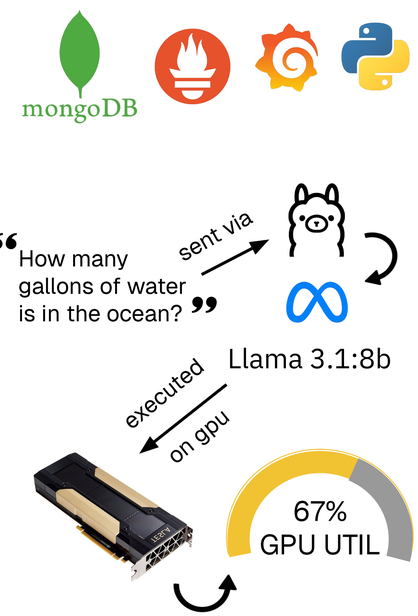
Edge is best suited for real-time data and addresses **latency** and **privacy** concerns.

IMPROVING EDGE

Giving edge providers a **benchmark** to score how a node would perform in running AI applications, specifically, running **LLMs** (restricted to **NVIDIA GPUs**).

Repeated prompts were sent to the language model for several minutes, GPU and LLM **performance metrics** were logged, and the resulting data was condensed into a **weighted score** balancing high GPU utilization against low response time.

CONDUCTING THE TEST

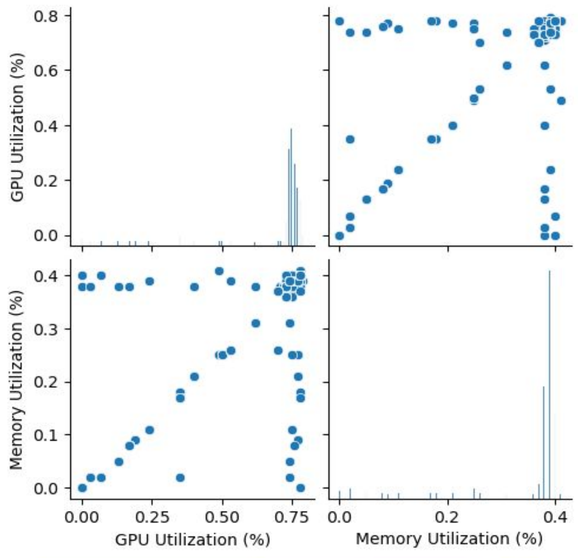


Connected to **MongoDB** and set up **Prometheus** & **Grafana** integration for data collection and visualization.

Generated a list of prompts and send them to the LLM via **Ollama** while node exporter and nvidia gpu exporter expose real-time GPU metrics through a Python script.

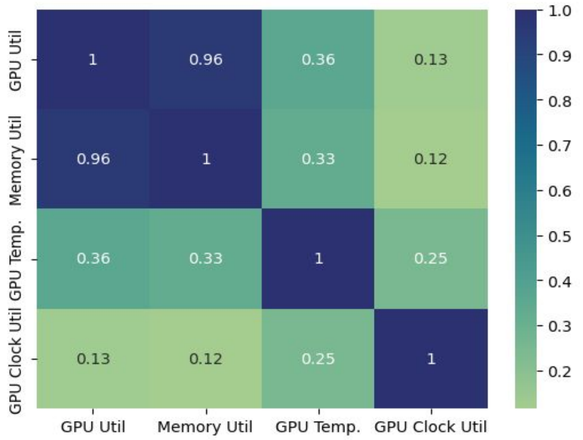
Retrieved the GPU metrics from Prometheus and store the collected data in MongoDB.

DATA VISUALIZED



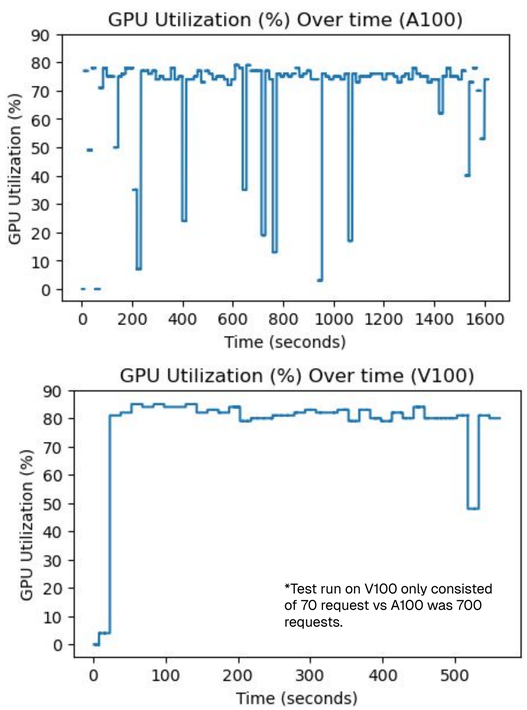
This figure represents a **pairplot** of the two most important GPU metrics that impact the response time.

This information shows if both of these metrics hold the same information or not.



This figure represents a **heatmap** of the correlation between metrics.

This demonstrates how closely memory and GPU utilization are **correlated** with each other.



This plot shows **Tesla A100**'s GPU utilization swings from **0% to ~90 %** as successive prompts stream in over several minutes. These brief, high intensity bursts followed by idle gaps imply it finishes each task quickly, behavior typical of a **high performance** GPU, unlike **Tesla V100** GPU.

BENCHMARKS

These **3 features** play the most important role in creating benchmarks, **GPU utilization**, **memory utilization,** and **response time**.

Based on the observed volatility in utilization during the test, consistent patterns were identified and used to develop a scoring method that leverages this information.

1. Create weights for each metric, rewarding fluctuations in GPU utilization and penalizing fluctuations in memory utilization and high response time.

$$w_1 = 1/\sigma_{\text{GPU Util}} \quad w_2 = 1/\sigma_{\text{Mem Util}}$$

$$w_3 = -e^{\text{rt}/(25/\ln 2)}$$

Explanation: reciprocal of the standard deviation of gpu utilization, reciprocal of the standard deviation of memory utilization, create a negative value that exponentially increases when response time increases.

1. Multiply the tailored weights by the respective averages

1. Compare the weighted sums based, a lower score would mean less capable compute.

Experiment result based on these benchmarks:
TESLA A100 Score: 0.103
TESLA V100 Score: -0.233

As expected, based on the benchmark the Tesla A100 scores better than the V100 as this benchmark is supposed to be used relative to other GPUs.

ACKNOWLEDGEMENTS

Special thank you to Ivan and Xiaotian for helping us with this project and sharing their wisdom.

There are a lot disclaimers and important information that we could not include in this poster. To get a better understanding of our project, please visit our website by scanning the QR Code.

WINLAB