RUTGERS
WINLAB | Wireless Information Network Laboratory

Resilient Edge-cloud Autonomous Learning with Timely Inferences

Haider Abdelrahman, Yunhyuk Chang, Lakshya Gour, Tanushree Mehta, Shreya Venugopal

PROBLEM

- Models are getting more complex
- Running models on less powerful devices while maintaining low latency is difficult
- MEC (Mobile-edge computing) is a viable solution

OBJECTIVE

Develop a framework to analyze tradeoffs between accuracy and latency of models when performing edge computing

WHAT IS MEC?

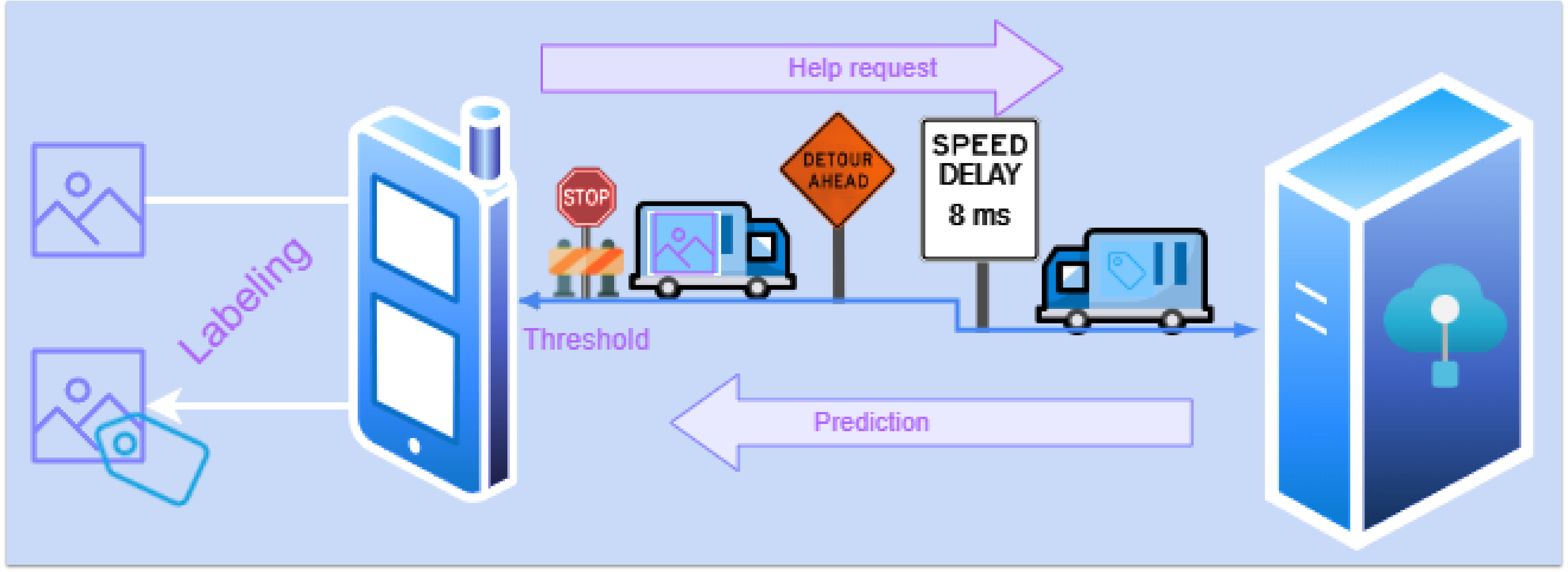
Mobile-Edge Computing is a network architecture that brings computation and storage capabilities closer to the end-users, reducing latency and improving real-time performance.

APPROACH

- Task: Image Classification
- Testing over entire test set
  - less variability
- Edge: Powerful device
  - Oracle; 100% task accuracy
- Mobile: Less powerful Device
  - 85% accuracy on task
- If mobile confidence < threshold, help is requested from Edge
- Measuring latencies at each step

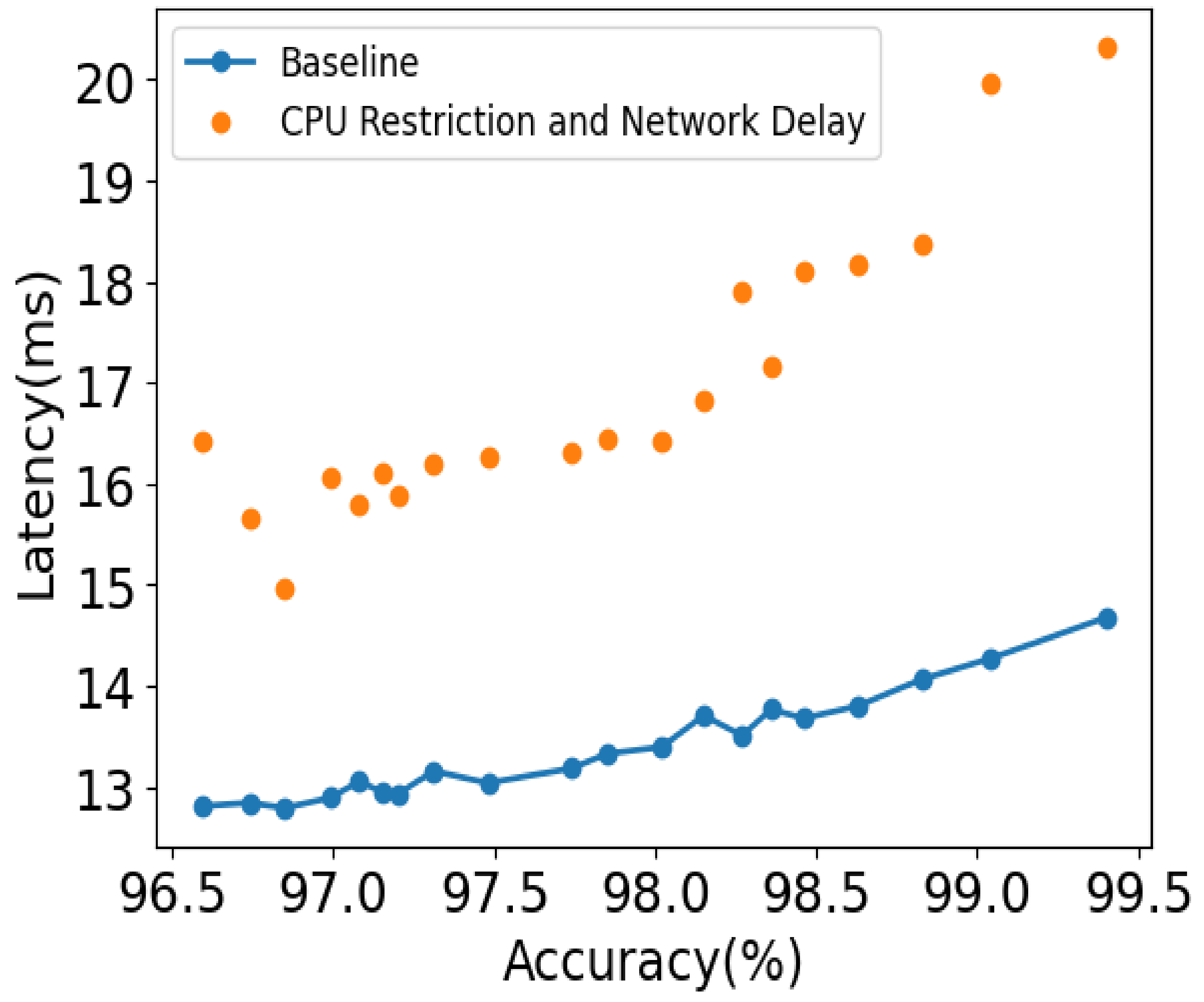
BENEFITS

- Gaining a deeper understanding of tradeoffs required to optimize tasks for accuracy/latency
- Understand different scenarios for Real-Time MEC and how certain factors affect the decision to ask for help more than others



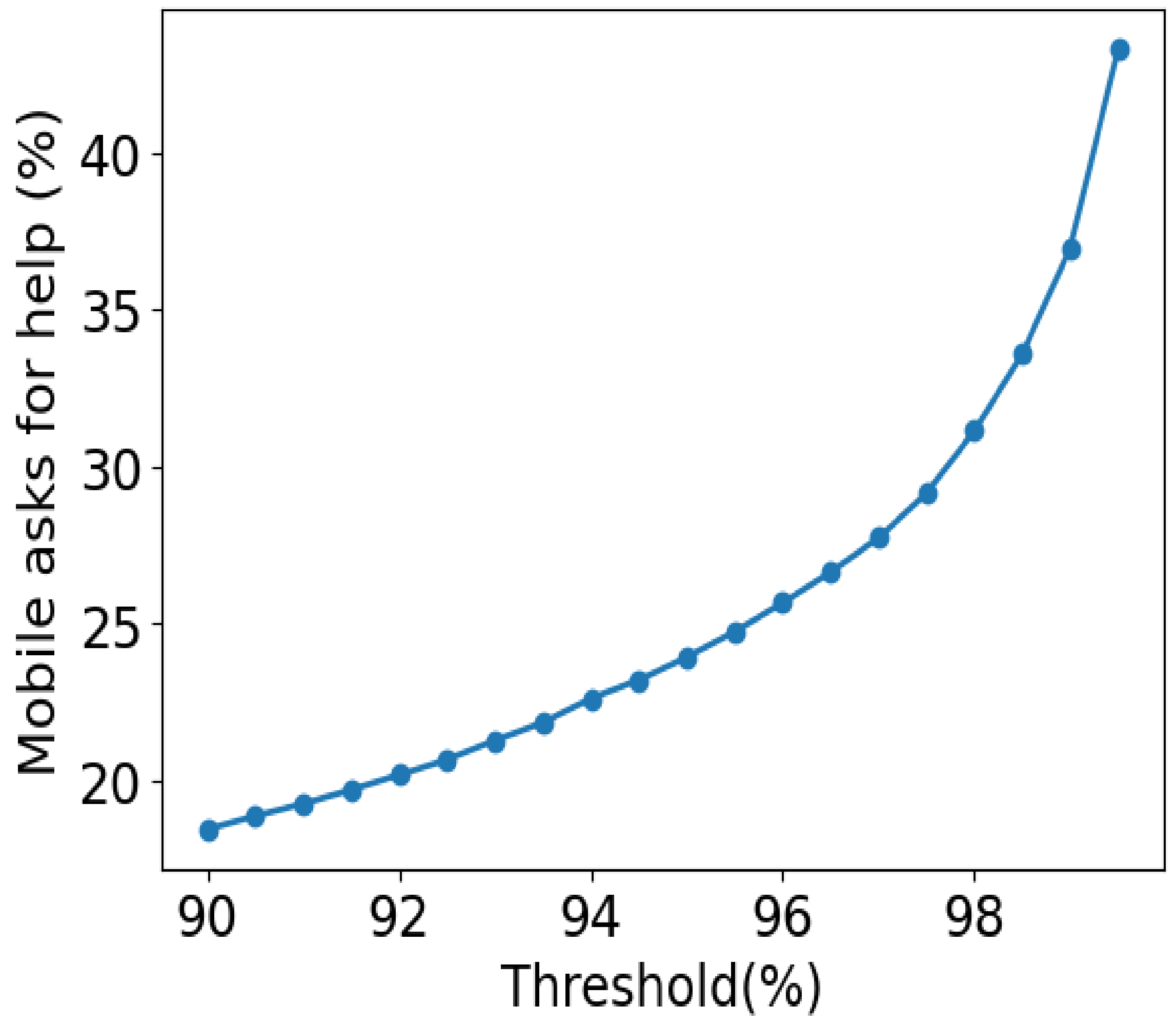
As you **vary** the **threshold** for edge assistance, how does the **average latency** change (over the **dataset**)?

What is the impact of introducing CPU and network limitations?



- CPU Limit: 1.2 Ghz
- Network: 8ms delay +/- 3ms

Why does the latency increase as the accuracy increases?

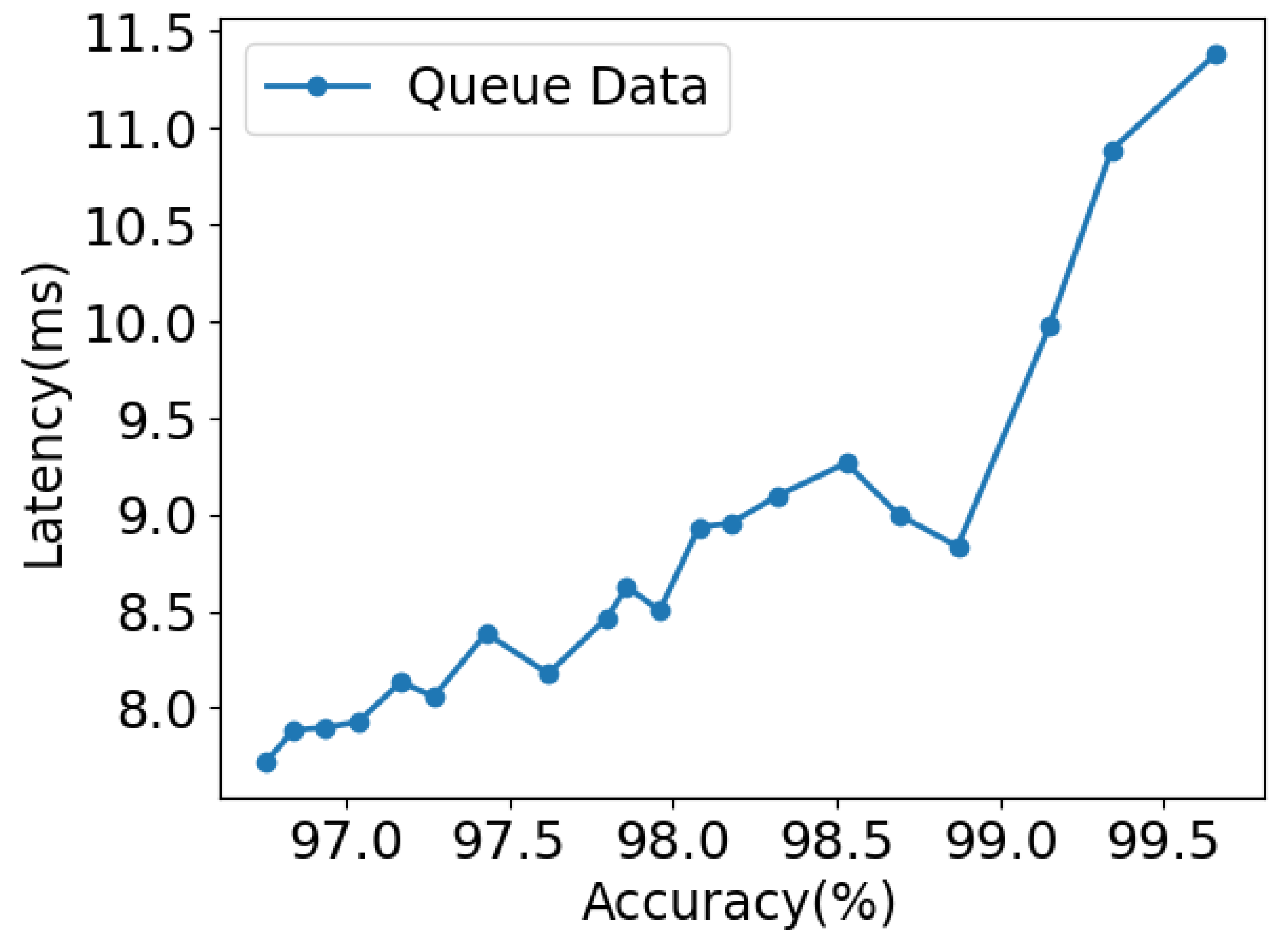


- Threshold: Confidence of prediction
- Asking for help: sending to Edge

CONCLUSION

- Implementing a threshold for MEC systems allows for a faster prediction than simply using an Edge server, and a more accuracy inference than just using a Mobile device
- Attempting to assimilate real life by implementing CPU speed and network restrictions has a high impact on the overall latency of the system
- Introducing parallelization during inference (Multithreading with queue) allows for lower latency and quicker predictions

To what extent does queuing images when asking the edge server for help improve latency?



- Queuing enables the device to inference as it waits for the edge to send back prediction
- Range of average latency = 7-12 ms

FUTURE WORK

Software Engineering:
Automating the pipeline in the experimental set up in a more streamlined manner and implementing frameworks for synchronization.

Experiments:
- Split Computing and Early Exiting
- Multiple Clients and Servers
- Different Queuing Policies

Acknowledgements

- Sponsor(s): nVerses Capital
- Project head: Prof. Anand Sarwate
- Special thanks: Prof. Waheed U. Bajwa, Ivan Seskar, Jenny Shane, Prof. Roy Yates, & all PhD students who helped!

This material is based upon work supported by the National Science Foundation under grant no. CNS-2148104 and is supported in part by funds from federal agency and industry partners as specified in the Resilient & Intelligent NextG Systems (RINGS) program.

Link to website for more info!

WINLAB