# Quarter 2 Project Proposal

**Owen Shi**
a4shi@ucsd.edu

**Pranav Kumarsubha**
pkumarsubha@ucsd.edu

**Brendan Kuang**
blkuang@ucsd.edu

**Edgar Guzman**
eiguzman@ucsd.edu

**Idhant Kumar**
ikumar@ucsd.edu

**Brandon Chiou**
brchiou@ucsd.edu

**Rajesh Gupta**
rgupta@ucsd.edu

# 1 Context and Motivation

As large language models (LLMs) and other large neural networks integrate into modern society, their applications are increasingly bottlenecked by computational intensity. The core operation driving these models is floating-point multiplication in matrices, which accounts for approximately 65–85% of operations in LLMs like ChatGPT (Dettmers et al. 2022). Given the scale at which these models operate, even slight improvements in each multiplication operation could yield substantial impacts on overall efficiency and performance. As model sizes continue to grow and deployment expands to energy-constrained edge devices, developing more efficient alternatives to traditional floating-point multiplication becomes critical for sustainable AI deployment.

# 2 Problem Statement

Enhancing floating-point multiplication has emerged as an active area of research, with several approaches exploring approximate arithmetic to reduce computational cost. Linear-complexity Multiplication (L-Mul) is an algorithm that estimates floating-point multiplication in $\mathcal{O}(m)$ time with respect to the size of the data type (mantissa bits), as opposed to the $\mathcal{O}(m^2)$ complexity of traditional multiplication (Luo and Sun 2024). This theoretical advantage suggests significant potential for speedup, energy savings and silicon area reduction in hardware implementations.

Previous work has validated performance and energy trade-offs of L-Mul in baseline tasks, particularly multilayer perceptrons (MLPs) trained on MNIST (Nakschou 2023). These results demonstrate that L-Mul can maintain accuracy within 1% of standard floating-point multiplication, reduce area by 76.1%, and reduce power by 96.6%. However, this evaluation lacks generalizability to real-world applications, which are at a much larger scale in terms of size and complexity. The algorithm remains largely theoretical, having been proposed only recently, and has not yet seen comprehensive testing in practice across diverse neural network architectures and complex tasks. For our future evaluations, we will focus on FashionMNIST as our primary dataset for classification tasks. FashionMNIST provides 70,000 labeled images across 10 classes of clothing items, offering a more challenging classification problem than MNIST while maintaining similar structure and scale, making it suitable for evaluating L-Mul's effectiveness in practical image classification scenarios.

With L-Mul being proposed just over a year ago, the gap between theoretical promise and practical deployment may stem from several factors: the cost of implementing specialized hardware, challenges in operating system integration to achieve real performance metrics, and insufficient validation across the breadth of neural network architectures used in production systems.

# 3 Primary Output

Our project will produce a comprehensive analysis of L-Mul as a hardware accelerator for floating-point multiplication in deep neural networks. The primary deliverables include:

- **Performance Analysis**: Quantitative evaluation of estimated changes in performance (speed, power, and accuracy) when accelerating key operations such as fp matrix multiplication in large neural networks using L-Mul, compared against standard floating-point multiplication implementations.
- **Accuracy Experimentation**: Validation of accuracy performance across complex tasks and architectures, including Transformers and LSTMs on FashionMNIST classification and sequential token generation tasks.
- **Cost and Obstacle Analysis**: Comprehensive evaluation of implementation costs and deployment challenges, including chip manufacturing costs, operating system compatibility requirements, hardware-software interface design, and integration complexity with existing neural network frameworks.

All analyses and experimental results will be communicated through a detailed technical report documenting methodology, findings, and recommendations for future work. Code implementations, hardware designs, and experimental notebooks will be made publicly available to facilitate reproducibility and further research.

# 4 Path to Success

Given our work in Quarter 1, we can ensure that data related to L-Mul benchmarks can be realistically obtained. We can run our experiments on local desktop computers equipped with sufficient GPU and memory resources, as well as on cloud platforms such as DSMLP. Preliminary tests confirm that data loading, preprocessing, and model training can be conducted efficiently, and that existing neural network implementations can be modified to incorporate L-Mul in both software and hardware settings. For hardware evaluation, Icarus Verilog (iverilog) provides simulation runtime and clock-cycle estimates, and Yosys with OpenSTA have been explored to yield cell area, cell count, timing metrics, and power estimates. Results from our MLP, LSTM, and CNN classification tests show promise for extending L-Mul to Transformers or LSTMs on classification as well as new tasks such as token generation or regression, demonstrating that L-Mul maintains accuracy within 1% of standard floating-point implementations across diverse architectures.

# References

**Dettmers, Tim, Mike Lewis, Younes Belkada, and Luke Zettlemoyer.** 2022. "LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale." *Advances in Neural Information Processing Systems* 35. [Link]

**Luo, Hongyin, and Wei Sun.** 2024. "Addition is All You Need for Energy-efficient Language Models." *arXiv preprint arXiv:2410.00907*. [Link]

**Nakschou, et al..** 2023. "Linear-Complexity Multiplication for Neural Network Acceleration." capstone project report, University of California, San Diego. [Link]