A TEMPLATE THESIS/DISSERTATION USING THE UTSATHESIS PACKAGE FOR LaTeX AND LyX USERS

by

WEINING ZHANG (TO BE REPLACED BY YOUR OWN NAME), M.Sc.

DISSERTATION
Presented to the Graduate Faculty of
The University of Texas at San Antonio
In Partial Fulfillment
Of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE

COMMITTEE MEMBERS:
First Name Last Name, Ph.D., Co-Chair
First Name Last Name, Ph.D. , Co-Chair
First Name Last Name, Ph.D.
First Name Last Name, Ph.D.
First Name Last Name, Ph.D.

THE UNIVERSITY OF TEXAS AT SAN ANTONIO
College of Sciences
Department of Computer Science
December 2016

DEDICATION

I would like to dedicate this thesis/dissertation template to UTSA graduate students.

A TEMPLATE THESIS/DISSERTATION USING THE UTSATHESIS PACKAGE FOR LATEX AND LYX USERS

Weining Zhang (to be replaced by your own name), Ph.D.
The University of Texas at San Antonio, 2016

Supervising Professors: First Name Last Name, Ph.D. and First Name Last Name, Ph.D.

The first chapter of this document is a description of the content and the usage of the UTSAthesis package. The remaining chapters serve to illustrate some use of LYX features for writing a thesis/dissertation.

The first line of the abstract has been indented as per required by the thesis/dissertation guideline.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

With the rapid advancement of deep learning (DL) technology and the growing adoption of Internet of Things (IoT) devices (Tu, Yang, and Cao,2025), industries are increasingly leveraging deep neural networks (DNNs) for edge computing applications. However, running DNN inference on edge devices presents major challenges, particularly latency and energy efficiency.(Xu et al.,2023). Traditional distributed inference methods are based on static partitioning, where the layers of the neural network are assigned to specific edge devices. Although this approach is effective in stable environments, it struggles in dynamic conditions, where network bandwidth and computational resources fluctuate. As a result, many edge AI applications suffer from performance degradation, increased energy consumption, and limited scalability. Research has shown that inefficient resource allocation in edge-based DNN inference can lead to more than 30% additional energy consumption and latency spikes exceeding 50%, significantly impacting real-time applications such as autonomous vehicles, healthcare monitoring, and industrial automation. To overcome these limitations, optimizing distributed DNN inference for low latency and energy efficiency is essential to ensure scalability, reliability, and the broader adoption of AI-driven edge computing (Liu, Xu, Qiao, and Li, 2024).

Existing solutions for distributed DNN inference mainly focus on static model partitioning, where computational tasks are pre-assigned to edge devices without accounting for real-time variations in network conditions and hardware constraints(Mahmud, Kang, Desai, Lama, and Prasad,2024). Some recent approaches have explored model pruning, quantization, and offloading strategies to enhance efficiency, but these methods are not dynamically adjusted based on real-time resource availability. Moreover, current scheduling mechanisms often introduce computational overhead, which negates potential efficiency gains. As a result, there remains a gap in achieving fully adaptive and resource-efficient DNN inference on edge devices.

This research proposes an adaptive DNN partitioning framework that can dynamically redistribute computational layers between multiple edge devices. By incorporating real-time resource monitoring of CPU, memory, and network bandwidth, this framework aims to optimize workload distribution, reduce latency, and minimize energy consumption. The ultimate goal is to improve the scalability and performance of edge AI applications by leveraging fine-grained resource tracking and intelligent scheduling mechanisms.

In this paper, we are inspired to investigate the problem of implementing a monitoring system to track CPU usage, memory consumption, and network bandwidth across heterogeneous edge devices. Developing an adaptive algorithm that dynamically redistributes model layers based on real-time resource availability. This paper experiments the evaluation on Heterogeneous Edge Devices which is testing the proposed framework on a testbed comprising Raspberry Pi and NVIDIA Jetson Nano devices to assess performance under varying conditions. By leveraging heuristic optimization and reinforcement learning to improve inference speed and reduce power consumption.

The key contributions of this research are as follows.

- We introduce a novel, real-time optimization framework to dynamically partition DNN workloads based on edge resource availability.
- The proposed framework improves the scalability of AI-driven edge computing, benefiting industries such as healthcare, autonomous systems, and smart cities.
- We propose a work which is Alignment with NSF Broader Impacts and the proposed research supports technological advancements that improve AI accessibility in resource-constrained environments, contributing to scientific knowledge and societal well-being.

By addressing the limitations of static DNN inference and introducing a real-time adaptive solution, this research will help bridge the gap between AI capabilities and the constraints of edge computing environments, ensuring efficient, scalable, and energy-efficient AI inference at the edge.

REFERENCES

Ben-Nun, T., & Hoefler, T. (2019, August). Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. *ACM Comput. Surv.*, *52*(4). Retrieved from https://doi.org/10.1145/3320060 doi: 10.1145/3320060

Hou, X., Guan, Y., Han, T., & Zhang, N. (2022). Distredge: Speeding up convolutional neural network inference on distributed edge devices. In *2022 ieee international parallel and distributed processing symposium (ipdps)* (p. 1097-1107). doi: 10.1109/IPDPS53621.2022.00110

Hsu, K.-J., Bhardwaj, K., & Gavrilovska, A. (2019). Couper: Dnn model slicing for visual analytics containers at the edge. In *Proceedings of the 4th acm/ieee symposium on edge computing* (p. 179â194). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3318216.3363309 doi: 10.1145/3318216.3363309

Hu, C., Bao, W., Wang, D., & Liu, F. (2019). Dynamic adaptive dnn surgery for inference acceleration on the edge. In *Ieee infocom 2019 - ieee conference on computer communications* (p. 1423-1431). doi: 10.1109/INFOCOM.2019.8737614

Hu, C., & Li, B. (2022). Distributed inference with deep learning models across heterogeneous edge devices. In *Ieee infocom 2022 - ieee conference on computer communications* (p. 330-339). doi: 10.1109/INFOCOM48880.2022.9796896

Lama, P., Wang, S., Zhou, X., & Cheng, D. (2018). Performance isolation of data-intensive scale-out applications in a multi-tenant cloud. In *2018 ieee international parallel and distributed processing symposium (ipdps)* (p. 85-94). doi: 10.1109/IPDPS.2018.00019

Li, H., Li, X., Fan, Q., He, Q., Wang, X., & Leung, V. C. M. (2024). Distributed dnn inference with fine-grained model partitioning in mobile edge computing networks. *IEEE Transactions on Mobile Computing*, *23*(10), 9060-9074. doi: 10.1109/TMC.2024.3357874

Liu, Z., Xu, X., Qiao, P., & Li, D. (2024, December). Acceleration for deep reinforcement learning using parallel and distributed computing: A survey. *ACM Comput. Surv.*, *57*(4). Retrieved from `https://doi.org/10.1145/3703453` doi: 10.1145/3703453

Mahmud, H., Kang, P., Desai, K., Lama, P., & Prasad, S. K. (2024). A converting autoencoder toward low-latency and energy-efficient dnn inference at the edge. In *2024 ieee international parallel and distributed processing symposium workshops (ipdpsw)* (p. 592-599). doi: 10.1109/IPDPSW63119.2024.00117

Masud, A., Hosen, M. B., Habibullah, M., Anannya, M., & Kaiser, M. S. (2025). Image captioning in bengali language using visual attention. *PLOS ONE*, *20*(2), e0309364. Retrieved from `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0309364` doi: 10.1371/journal.pone.0309364

Mohammed, T., Joe-Wong, C., Babbar, R., & Francesco, M. D. (2020). Distributed inference acceleration with adaptive dnn partitioning and offloading. In *Ieee infocom 2020 - ieee conference on computer communications* (p. 854-863). doi: 10.1109/INFOCOM41043.2020.9155237

Padmanabha Iyer, A., Guan, M., Dai, Y., Pan, R., Gandhi, S., & Netravali, R. (2024). Improving dnn inference throughput using practical, per-input compute adaptation. In *Proceedings of the acm sigops 30th symposium on operating systems principles* (p. 624â639). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3694715.3695978` doi: 10.1145/3694715.3695978

Stahl, R., Zhao, Z., Mueller-Gritschneder, D., Gerstlauer, A., & Schlichtmann, U. (2019). Fully distributed deep learning inference on resource-constrained edge devices. In D. N. Pnevmatikatos, M. Pelcat, & M. Jung (Eds.), *Embedded computer systems: Architectures, modeling, and simulation* (pp. 77–90). Cham: Springer International Publishing.

Su, Y., Fan, W., Gao, L., Qiao, L., Liu, Y., & Wu, F. (2023). Joint dnn partition and resource allocation optimization for energy-constrained hierarchical edge-cloud systems. *IEEE Transactions on Vehicular Technology*, *72*(3), 3930-3944. doi: 10.1109/TVT.2022.3219058

Tu, J., Yang, L., & Cao, J. (2025, January). Distributed machine learning in edge computing: Challenges, solutions and future directions. *ACM Comput. Surv.*, *57*(5). Retrieved from `https://doi.org/10.1145/3708495` doi: 10.1145/3708495

Xu, F., Xu, J., Chen, J., Chen, L., Shang, R., Zhou, Z., & Liu, F. (2023). igniter: Interference-aware gpu resource provisioning for predictable dnn inference in the cloud. *IEEE Transactions on Parallel and Distributed Systems*, *34*(3), 812-827. doi: 10.1109/TPDS.2022.3232715