

2025-2026 Master Thesis Topics in OpenMP and ML/AI

John Burns, ADA University

July 16, 2025

Overview

- 1 Introduction
- 2 Topic 1: Optimizing Deep Neural Network Training with OpenMP
- 3 Topic 2: Help! Advising OpenMP Parallelization
- 4 Topic 3: Parallel Graph Neural Networks with OpenMP
- 5 Topic 4: Energy-Efficient ML Model Training with OpenMP
- 6 Topic 5: OpenMP enhancement of R Selected Contributed Packages

About myself and my research interests

- My PhD was in the field of large-scale (at the time!) modelling and simulation
- From my industry background I also have an interest in performance, reliability and scalability
- From my teaching I have an interest in Computer Architecture and Distributed Systems
- I welcome any suggestions you may have in the intersection of these fields
- I am very interested in OpenMP projects - from code and performance analysis through to application development
- As Energy Efficiency and Optimization is so important now, why not use ARM64 cloud platforms for these research topics.

Topic 1: Optimizing Deep Neural Network Training with OpenMP

- **Objective:** Enhance the performance of deep neural network (DNN) training by parallelizing compute-intensive operations (e.g., matrix multiplications, gradient computations) using OpenMP on multi-core CPUs.
- **Description:** Implement and evaluate OpenMP-based parallelization strategies for DNN training in frameworks like PyTorch or TensorFlow. Focus on optimizing data parallelism and model parallelism for large-scale datasets.
- **Relevance:** Reduces training time for DNNs, making them more feasible for resource-constrained environments.
- **Methodology:** Develop parallelized versions of backpropagation and optimization algorithms, benchmark performance on multi-core systems, and compare with GPU-based implementations.

Topic 1: References

- **Rakhimov, M., et al.**, "Parallel Approaches in Deep Learning: Use Parallel Computing," *International Conference on Future Networks and Distributed Systems (ICFNDS)*, 2023.
<https://doi.org/10.1145/3644713.3644738>
- **Zhang, Y., et al.**, "Exploiting Parallelism Opportunities with Deep Learning Frameworks," *ACM Transactions on Architecture and Code Optimization*, vol. 18, no. 1, 2020.
<https://doi.org/10.1145/3431388>

Topic 2: Help! Advising OpenMP Parallelization

- **Objective:** Reduce the programming burden of OpenMP annotation tasks.
- **Description:** Research DeepTyper-based learning architecture for advising OpenMP Annotation / Parallelization.
- **Relevance:** Valuable research to reduce the human workload in OpenMP development.
- **Methodology:** Using CodeT5+ and DeepTyper approaches.

Topic 2: References

- **Pornmaneerattanatri, S., et al.,** "Automatic Parallelization with CodeT5+: A Model for Generating OpenMP Directives," *International Workshop on Large Language Models and HPC*, 2024.
- **Shen, Y., et al.,** "A machine learning method to variable classification in OpenMP," *Concurrency and Computation: Practice and Experience*, 2023. <https://doi.org/10.1002/cpe.7746>
- **Kadosh, T., et al.,** "Advising OpenMP Parallelization via A Graph-Based Approach with Transformers" *OpenMP: Advanced Task-Based, Device and Compiler Programming* (pp.3-17)

Topic 3: Parallel Graph Neural Networks with OpenMP

- **Objective:** Enhance the scalability of graph neural networks (GNNs) for large-scale graph data using OpenMP parallelization.
- **Description:** Implement OpenMP-based parallel processing for GNN operations like message passing and aggregation, targeting applications in social networks or bioinformatics.
- **Relevance:** GNNs are computationally expensive for large graphs; OpenMP can improve training and inference speed on multi-core systems.
- **Methodology:** Develop parallel GNN algorithms, test on datasets like OGB (Open Graph Benchmark), and compare performance with serial implementations.

Topic 3: References

- **Meng, Z., et al.**, "OpenMP Parallelization and Optimization of Graph-Based Machine Learning Algorithms," *International Workshop on OpenMP (IWOMP)*, 2016
- **Zhou, J., et al.**, "Graph Neural Networks: A Review of Methods and Applications," *AI Open*, vol. 1, 2020.
<https://doi.org/10.1016/j.aiopen.2021.01.001>

Topic 4: Energy-Efficient ML Model Training with OpenMP

- **Objective:** Develop energy-efficient ML training pipelines using OpenMP to optimize resource utilization on multi-core CPUs.
- **Description:** Investigate OpenMP-based parallelization to reduce energy consumption in ML tasks, focusing on dynamic thread management and workload balancing.
- **Relevance:** Energy efficiency is critical for sustainable AI; OpenMP can optimize CPU-based training for green computing.
- **Methodology:** Implement energy-aware parallel algorithms, measure energy consumption using PowerPoint, and compare with traditional ML training pipelines.

Topic 4: References

- **Garcia, A., et al.**, "DNN Is Not All You Need: Parallelizing Non-Neural ML Algorithms on Ultra-Low-Power IoT Processors," *ACM Transactions on Embedded Computing Systems*, 2023.
<https://doi.org/10.1145/3570152>
- **Strubell, E., et al.**, "Energy and Policy Considerations for Deep Learning," *ACM Computing Surveys*, vol. 53, no. 3, 2020.
<https://doi.org/10.1145/3372822>

Topic 5: OpenMP enhancement of R Selected Contributed Packages

- **Objective:** Enhance the performance of selected package using OpenMP.
- **Description:** Today there are over 22000 Contributed Packages in cran-r.
- **Relevance:** R is one of the most popular languages for data science, with 31% of data scientists regularly using it, according to a 2025 source..
- **Methodology:** Select a popular R contrib package, (eg ggplot2, dplyr, tdyr etc) or some other package of your choice. Ccompile locally and test with and without OpenMP directives. Benchmark performance and contribute to the community

Topic 5: References

- Start here: `install.packages("packageRank")`

Conclusion

- These topics combine OpenMP's parallel computing capabilities with cutting-edge ML/AI challenges.
- They address performance, scalability, privacy, and sustainability in AI systems.
- Obviously knowledge/interest in C and Parallel Computing is a requirement
- A good place to start: OpenMP Architecture Review Board, "OpenMP API Specification: Version 5.2," 2024.
<https://www.openmp.org/specifications/>