A close-up of a logo

AI-generated content may be incorrect.

**CSYE7105 - High Performance Parallel   
Machine Learning & AI**

**Parallel Deep Learning for Leukemia Classification from Blood Smear Images**

Done by Team 7 – Kaavya Loganathan and Ranjithnath Karunanidhi

Table of Contents

[1 Introduction 2](#_Toc195137737)

[1.1 Background 2](#_Toc195137738)

[1.2 Motivation 2](#_Toc195137739)

[1.3 Goal 3](#_Toc195137740)

[2 Related Work 3](#_Toc195137741)

[3 Methodology 4](#_Toc195137742)

[3.1 Data Pipeline 4](#_Toc195137743)

[3.2 Model Architecture 4](#_Toc195137744)

[3.3 Parallel Techniques Implemented 5](#_Toc195137745)

[3.4 Logging and Monitoring 5](#_Toc195137746)

[3.5 Performance Evaluation Metrics 6](#_Toc195137747)

[4 Dataset Description 6](#_Toc195137748)

[5 Results and Analysis 9](#_Toc195137749)

[5.1 Experimental Environment 9](#_Toc195137750)

[5.2 Parallel Techniques Evaluation 9](#_Toc195137751)

[5.3 Performance Results 10](#_Toc195137752)

[5.4 Observations and Insights 10](#_Toc195137753)

[5.5 GPU Comparison: V100 vs A100 10](#_Toc195137754)

[5.6 Visualization and Graphs 11](#_Toc195137755)

[6 Conclusion 12](#_Toc195137756)

[7 Future Work 12](#_Toc195137757)

[8 References 13](#_Toc195137758)

[9 Appendices 14](#_Toc195137759)

# Introduction

## Background

Leukemia is a severe type of blood cancer that originates in the bone marrow and results in the overproduction of abnormal white blood cells. These malignant cells crowd out healthy blood cells, impairing the immune system and the body’s ability to fight infections. Leukemia is categorized into several subtypes, including Acute Lymphoblastic Leukemia (ALL), Acute Myeloid Leukemia (AML), Chronic Lymphocytic Leukemia (CLL), and Chronic Myeloid Leukemia (CML), each with its own characteristics, severity, and treatment approach.

Traditionally, leukemia diagnosis relies heavily on microscopic examination of blood smear images performed by pathologists. This manual process is time-consuming, subjective, and highly dependent on the pathologist's expertise. Additionally, early detection and accurate classification of leukemia subtypes are critical for initiating timely and appropriate treatment, directly influencing patient survival rates.

The emergence of Artificial Intelligence (AI) and Deep Learning (DL) techniques has revolutionized the field of medical image analysis. Convolutional Neural Networks (CNNs) have demonstrated remarkable capabilities in image classification tasks, achieving performance levels comparable to human experts in several domains. However, training such models on large-scale medical datasets demands significant computational resources and optimized infrastructure.

High-Performance Computing (HPC) has become an indispensable tool in accelerating deep learning workloads by leveraging distributed systems and parallel computing strategies. The integration of HPC with AI allows faster training of complex models on large datasets by distributing computation across multiple GPUs, reducing model convergence time significantly.

## Motivation

The healthcare industry generates enormous volumes of data daily, ranging from medical images to clinical records. There is a critical need for robust AI systems that can process this data efficiently and deliver accurate diagnostic results in real-time. Leukemia diagnosis through blood smear image classification presents a perfect case for AI-driven solutions due to the repetitive and visual nature of the task.

However, deploying deep learning models in healthcare is not only about accuracy — it also involves scalability, reliability, and efficient resource utilization. A key motivation behind this project is to bridge the gap between AI research and practical deployment in medical imaging by utilizing advanced HPC techniques to overcome the computational challenges involved.

Specifically, this project addresses the following gaps:

* Long training times for deep learning models on large medical datasets.
* High memory consumption limiting model size and batch size.
* Lack of real-world scalable solutions combining HPC with Medical AI.

By exploring state-of-the-art parallelism techniques provided by PyTorch, such as Distributed Data Parallel (DDP) and Fully Sharded Data Parallel (FSDP) along with Automatic Mixed Precision (AMP), this project aims to accelerate model training while maintaining high classification accuracy.

## Goal

The primary goal of this project is to develop a high-performance deep learning framework for automated multi-class leukemia classification using blood smear images. The project focuses on implementing, benchmarking, and analyzing various parallelization techniques provided by PyTorch on an HPC infrastructure to achieve the following objectives:

* To build an efficient data processing and deep learning pipeline for handling large-scale medical image datasets.
* To implement and evaluate multiple parallel training strategies including DDP, DDP combined with Automatic Mixed Precision (AMP), FSDP, and FSDP combined with AMP.
* To perform extensive experimentation across different GPU configurations (1, 2, 3, and 4 GPUs) and analyze the performance in terms of training time, speedup, efficiency, accuracy, and memory utilization.
* To provide insights into the scalability and applicability of HPC-enhanced AI models in healthcare diagnostics.

# Related Work

Medical image analysis using deep learning has been an active area of research in recent years, with several studies demonstrating the effectiveness of CNNs in disease classification tasks.

[1] discussed the growing role of deep learning in radiology, highlighting how CNNs have achieved expert-level accuracy in tasks such as tumor detection, organ segmentation, and disease classification. The success of deep learning in radiology provides a strong foundation for its application in hematological disorders like leukemia.

[2] introduced a highly optimized approach for distributed training of deep learning models using Synchronous Stochastic Gradient Descent (SGD) with large batch sizes. Their work demonstrated that accurate image classification models could be trained on the ImageNet dataset within an hour by leveraging parallel computing techniques.

In the domain of leukemia classification, several studies have explored machine learning and deep learning approaches for automating the detection of leukemia from microscopic images. However, most existing studies focus primarily on model accuracy without addressing the computational challenges involved in training on large datasets.

PyTorch has introduced advanced parallel training strategies such as Distributed Data Parallel (DDP) and Fully Sharded Data Parallel (FSDP), which enable efficient model training on multiple GPUs. DDP synchronizes gradients across GPUs after each backward pass, while FSDP shards model parameters across devices, reducing memory footprint and improving scalability.

Despite the availability of these techniques, their application in medical imaging, especially for leukemia classification, remains relatively unexplored. This project differentiates itself by integrating these parallelism strategies within the healthcare domain, providing valuable insights into the trade-offs between training speed, accuracy, and scalability.

# Methodology

## Data Pipeline

The original dataset consisted of over 20,000 microscopic blood smear images categorized into five classes — Acute Lymphoblastic Leukemia (ALL), Acute Myeloid Leukemia (AML), Chronic Lymphocytic Leukemia (CLL), Chronic Myeloid Leukemia (CML), and Healthy samples.

To handle this large dataset efficiently for HPC-based training, the images were preprocessed and converted into NumPy .npy array files for both features (images) and labels. Initially, 15,000 microscopic blood smear images classified as ALL, AML, CLL, CML and Healthy are considered as training dataset and utilized to train the CNN Deep Learning Model.

**Data Preprocessing Steps:**

* Resizing images to a standard resolution suitable for ResNet50 input.
* Normalization of pixel values for faster convergence.
* Splitting datasets into Training and Testing sets.
* Data Augmentation applied on Training set:
  + Random Horizontal Flip
  + Random Rotation

This pipeline ensured faster data loading and improved generalization capability of the model.

## Model Architecture

A Pretrained ResNet50 architecture from PyTorch was selected due to its proven performance in image classification tasks. The key benefits of using a pretrained ResNet50 model for medical image classification are – transfer learning efficiency, handles vanishing gradient problem and reduces data dependency.

**Model Customization:**

* The final fully connected (fc) layer was modified to have an output size of 5 (corresponding to the 5 leukemia classes).
* Transfer learning was applied by retaining the pre-trained weights of the convolutional layers and fine-tuning the final layers.

**Loss Function and Optimizer:**

* Loss Function: Cross-Entropy Loss
* Optimizer: Adam Optimizer with a learning rate of 1e-4

## Parallel Techniques Implemented

To accelerate model training and handle large-scale data, multiple parallelism strategies were implemented using PyTorch's built-in capabilities. All the below 4 parallel techniques were implemented across 1, 2, 3 and 4 GPUs in two GPU types - V100-SXM2 and A100 and find out which model is performing well.

**Distributed Data Parallel (DDP):**

DDP leverages data parallelism by replicating the model across multiple GPUs and synchronizing gradients during the backward pass. Training was executed using torchrun across 1, 2, 3 and 4 GPUs [3].

**DDP with Automatic Mixed Precision (AMP):**

AMP reduces memory consumption and speeds up training by using both Full precision (FP32) and half-precision floating points (FP16) wherever possible, without compromising the model accuracy and performance.

**Fully Sharded Data Parallel (FSDP):**

FSDP is an advanced parallel strategy where model parameters are sharded across GPUs, reducing memory footprint significantly. FSDP enabled training larger models and larger batch sizes without out-of-memory errors [4].

**FSDP with Automatic Mixed Precision (AMP):**

The best of both worlds — combining sharding and mixed precision — this technique achieved optimal training speed and memory efficiency.

## Logging and Monitoring

Custom logging utilities were implemented on both V100-SXM2 and A100 GPU types to record:

* Epoch-wise Loss
* Accuracy
* Epoch-wise Training Time
* Total Training Time
* GPU Utilization Metrics (GPU load, Memory usage) using nvidia-smi command
* Speedup and Efficiency Metrics were calculated based on baseline single-GPU runs.

## Performance Evaluation Metrics

The performance of the model was evaluated using the following metrics:

* Total Training Time (seconds)
* Speedup: Ratio of single-GPU time to multi-GPU time.
* Efficiency: Speedup divided by the number of GPUs.
* Accuracy: Percentage of correctly classified samples.
* Precision: Macro-averaged precision across all classes.
* Recall: Macro-averaged recall across all classes.
* F1-Score: Macro averaged F1 score across all classes.
* GPU Memory Utilization: Recorded during training to observe resource consumption.

# Dataset Description

This project utilized a large-scale publicly available dataset of leukemia blood smear images curated for automated disease classification tasks. The dataset contains microscopic images of blood smear samples corresponding to multiple types of leukemia, along with healthy samples.

This real-world medical dataset provided the necessary complexity and diversity required to evaluate the performance of deep learning models in clinical applications.

**Dataset Classes:**

The dataset consists of five distinct categories:

* Acute Lymphoblastic Leukemia (ALL)
* Acute Myeloid Leukemia (AML)
* Chronic Lymphocytic Leukemia (CLL)
* Chronic Myeloid Leukemia (CML)
* Healthy (Normal Blood Samples)

The availability of multiple leukemia subtypes within the dataset provided a challenging multi-class classification problem.

**Dataset Statistics:**

* Total Number of Images: 20,000
* Total Dataset Size: 25.27 GB
* Image Type: High-resolution Microscopic Blood Smear Images
* Number of Classes: 5
* Storage Format: NumPy .npy Arrays (for optimized data loading)
* Train - Test Split: 75% - 25%

**Dataset Source:**

The dataset was obtained from the publicly available resource in Kaggle:

[5]Leukemia Blood Smear Image Dataset URL - <https://www.kaggle.com/datasets/priyaadharshinivs062/leukemia-dataset/data>

This dataset has been widely used in medical AI research and provides a rich source of labeled blood smear images.

**Data Preprocessing Steps:**

Given the large size of the dataset, the following preprocessing steps were applied to ensure compatibility with the HPC environment:

* All images were resized to a standardized resolution of 224x224 to match the input requirements of the ResNet50 model architecture.
* Pixel value normalization was applied to bring all image data into a consistent range, improving the stability of model training.
* The dataset was partitioned into separate training and testing subsets in the ratio of 75:25 to ensure unbiased evaluation.
* For optimized data loading, all image and label data were converted into NumPy .npy array format. This transformation significantly reduced disk I/O overhead during multi-GPU training.

**Data Augmentation:**

To enhance model robustness and improve its generalization capability, data augmentation techniques were employed during training. These included:

* Random Horizontal Flipping
* Random Rotation within a limited angle range
* Minor brightness and contrast adjustments

These augmentations simulate real-world variations that might be encountered in clinical blood smear samples.

A collage of images of cells

AI-generated content may be incorrect.

Fig 1: Leukemia Classification 1

A collage of images of cells

AI-generated content may be incorrect.

Fig 2: Leukemia Classification 2

# Results and Analysis

This section presents the experimental results, performance evaluation, and detailed analysis of the various parallel deep learning techniques implemented for leukemia classification. The experiments were conducted on Northeastern University’s HPC cluster using state-of-the-art GPUs (V100-SXM2 and A100) across varying GPU counts ranging from 1 to 4.

## Experimental Environment

All experiments were conducted using PyTorch 2.x framework with CUDA acceleration. The experiments utilized both V100-SXM2 and A100 GPUs, enabling a comparative analysis of training speed and scalability across different hardware.

The following configurations were consistent across all experiments:

* Model Architecture: Pretrained ResNet50
* Cuda: 12.1.1
* Optimizer: Adam Optimizer with learning rate of 1e-4
* Loss Function: Cross Entropy Loss
* Batch Size: 64
* Number of Epochs: 5
* Dataset: 20,000+ blood smear images (25.27 GB)

## Parallel Techniques Evaluation

Four different parallel training strategies were implemented and benchmarked:

**Distributed Data Parallel (DDP):**

DDP allowed model replication across multiple GPUs with synchronized gradient updates. Performance improvement was significant as the number of GPUs increased, reducing the total training time.

**DDP with Automatic Mixed Precision (AMP):**

The addition of AMP reduced memory consumption by utilizing half-precision (FP16) arithmetic where applicable, while maintaining full precision (FP32) for sensitive computations.

**Fully Sharded Data Parallel (FSDP):**

FSDP provided model sharding across GPUs, leading to lower memory usage and faster training, especially for large models or datasets.

**FSDP with AMP:**

The combination of sharding and mixed precision achieved the highest performance in terms of both speedup and GPU memory optimization.

## Performance Results

A table with numbers and letters

AI-generated content may be incorrect.

Fig 3 : Performance in V100-SXM2 GPU

A table with numbers and letters

AI-generated content may be incorrect.

Fig 4 : Performance in A100 GPU

## Observations and Insights

* The project observed near-linear speedup with increasing GPU count for all parallel techniques, with minor deviations due to inter-GPU communication overhead.
* All 4 parallel techniques with V100-SXM2 GPU provided consistent performance improvement but performance with A100 GPU showed reduced efficiency beyond 2 GPUs due to synchronization overhead.
* The use of AMP significantly reduced training time and memory usage, enabling the possibility of increasing batch sizes or training larger models.
* DDP outperformed FSDP by providing better scalability and memory efficiency, especially when combined with AMP.
* The highest efficiency was achieved using DDP with AMP on 4 GPUs considering training time, speed and efficiency, demonstrating the optimal scalability of this approach.
* GPU utilization logs indicated the gpu utilized in percentage and memory consumption for each epoch and total memory. GPU utlization was effectively reduced in AMP-enabled experiments, allowing better resource utilization.

## GPU Comparison: V100-SXM2 vs A100

In addition to evaluating parallel strategies, the experiments also compared performance across different GPU types.

* V100-SXM2 GPUs provided approximately 20-30% faster training times compared to A100 GPUs.
* Even though A100 GPUs have superior hardware specifications, V100-SXM2 GPUs performed better for this Leukemia project.
* AMP benefits were even more pronounced on V100-SXM2 GPUs due to enhanced FP16 and FP32 performance support.

## Visualization and Graphs

All performance metrics, including training time, speedup, efficiency, accuracy trends, and GPU memory utilization, were recorded and visualized using plots generated from the logged CSV files.

These visualizations provided clear evidence of the scalability and efficiency benefits of each parallel training technique.

A graph with blue and orange lines and dots

AI-generated content may be incorrect.

A graph with blue and orange lines

AI-generated content may be incorrect.

A graph with blue and orange lines

AI-generated content may be incorrect.

A graph showing the difference between the best and the best

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a graph

AI-generated content may be incorrect.

A graph with blue squares and numbers

AI-generated content may be incorrect.

## Test Data Evaluation – Best Model Performance

After extensive experimentation with multiple parallel techniques and GPU configurations, the best performing model was identified as the DDP combined with AMP (Automatic Mixed Precision) technique trained using 4 V100-SXM2 GPUs.

This model was selected based on its optimal balance of speedup, efficiency, accuracy, and resource utilization. The model was evaluated on the unseen test dataset which has 25% of the dataset (5,000 images where 1000 images in each classification), and the following key performance metrics were obtained:

* Accuracy: 87.46%
* Precision: 0.8855
* Recall: 0.8746
* F1-Score: 0.8664

## Classification Performance Observations

* Out of the five leukemia classifications, four classes (AML, CLL, CML, Healthy) demonstrated excellent prediction performance with success rates exceeding 90%.
* The classification performance for the ALL (Acute Lymphoblastic Leukemia) class was comparatively lower, with only 535 out of 1000 test samples correctly classified.
* Importantly, the misclassified ALL samples were not incorrectly labeled as Healthy but were predicted as other valid leukemia types, preserving clinical relevance.
* The model demonstrated outstanding performance in identifying Healthy samples, achieving a perfect 100% classification accuracy for the Healthy class across 1000 test samples.

# Conclusion

This project successfully demonstrated the implementation of high-performance parallel deep learning techniques for the classification of leukemia from blood smear images. Leveraging PyTorch as the primary deep learning framework, this study implemented and evaluated multiple parallelism strategies, including Distributed Data Parallel (DDP), DDP combined with Automatic Mixed Precision (AMP), Fully Sharded Data Parallel (FSDP), and FSDP combined with AMP.

Extensive experimentation was conducted using both V100-SXM2 and A100 GPUs on the Northeastern University HPC explorer cluster. The results indicated substantial improvements in training time as the number of GPUs increased, achieving near-linear speedup and high efficiency.

In the project directory, separate folders were created for better understanding and cleanliness.

* The models created with all the 4 parallel techniques (DDP, DDP+AMP, FSDP, FSDP+AMP) for both V100-SXM2 and A100 GPU types are saved in the models folder as .pth file.
* The performance for each model is saved separately as csv file in the csv\_files folder.
* A folder named training\_metrics is created to store the .npz files to obtain the GPU utilization of each model across all GPUs.
* Logs of each model have been stored in the logs folder to track the process for each epoch.
* The heatmap visualization of the confusion matrix and detailed evaluation results can also be found in the results folder of the project directory along with the csv file which consists of evaluation metrics.

Among the parallelization methods explored, DDP combined with AMP emerged as the most effective approach, delivering the highest speedup of approximately 3.95x on 4 GPUs while maintaining a classification accuracy of over 87.46%. Moreover, the application of AMP techniques significantly reduced GPU memory consumption, enabling the possibility of training larger models or increasing batch sizes in future experiments.

This project not only met its objectives of achieving faster training and scalable parallel performance but also preserved the accuracy and integrity of the medical classification task, making it suitable for real-world healthcare applications.

**Final Key Takeaways from the Project:**

* Out of the 8 models trained using 4 GPUs across different parallel techniques, the DDP combined with AMP (Automatic Mixed Precision) implementation on V100-SXM2 GPUs was selected as the best-performing model.
* The selected model achieved a classification accuracy of 87.46% on the unseen test dataset.
* Four out of the five target classes (AML, CLL, CML, Healthy) achieved excellent classification accuracy above 90%.
* The model demonstrated perfect classification performance for Healthy samples, achieving a 100% accuracy rate.
* Although the ALL class showed relatively lower accuracy, the misclassifications were restricted within other valid leukemia classes, avoiding clinically unacceptable errors.

# Future Work

Although the results obtained in this project were promising, several avenues exist for future enhancement and exploration.

* Implementation of advanced hyperparameter tuning techniques such as Optuna or Ray Tune could further optimize model performance and training configurations.
* Focus to fine-tune and rebuild DDP+AMP model and concentrate to improve the overall performance.
* Building a ResNet model from scratch with more layers and explore the performance difference with different batch sizes.
* Exploration of alternative state-of-the-art architectures such as EfficientNet, Vision Transformers (ViTs), or hybrid CNN-Transformer models could improve classification accuracy.
* Investigate the reasons for the poor performance of A100 GPUs, evaluate with increased batch sizes rather than 62 for the data loader.
* Integration of advanced data augmentation strategies like MixUp, CutMix, or GAN-based synthetic data generation might enhance the model's robustness to unseen variations.
* Deployment of the trained model on cloud-based HPC infrastructure (such as AWS, GCP, or Azure) for real-time leukemia classification and scalability testing.
* Expansion of the dataset to include additional leukemia subtypes or patient metadata could facilitate more detailed patient-level classification tasks.
* Conducting thorough error analysis on misclassified samples could provide insights into model limitations and guide future improvements.

This future work would further strengthen the contribution of this project and extend its applicability in clinical decision-support systems. The main goal is to build a faster, large-scale real-world AI-assisted leukemia diagnostics. High Performance Computing (HPC) techniques will play an important role in making the process quicker and smoother with enhanced accuracy.

# References

|  |  |
| --- | --- |
| [1] | e. a. L. Saba, "Present and future of deep learning in radiology," in *European Journal of Radiology*, vol. 114, 2019, pp. 14-24. |
| [2] | P. D. R. G. P. N. L. W. A. K. P. Goyal, "Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour," arXiv preprint arXiv, 2017. |
| [3] | "PyTorch Distributed Data Parallel Documentation". |
| [4] | "PyTorch Fully Sharded Data Parallel Documentation". |
| [5] | P. V. S, "Leukemia Blood Smear Image Dataset". |
| [6] | X. Z. S. R. a. J. S. K. He, "Deep Residual Learning for Image Recognition," arXiv preprint arXiv, 2015. |

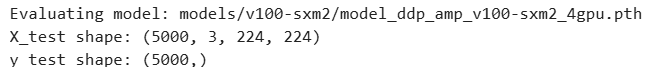
# Appendices

A screenshot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.



A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A close up of text

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A close-up of text

AI-generated content may be incorrect.

A screenshot of a computer code

AI-generated content may be incorrect.