RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Go, change the world

VI SEMESTER

SEE Presentation on

Crystoper–Prediction of Protein Crystallization Conditions Using Big Data Techniques

Domain: Bioinformatics

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Team Members

Go, change the world

		
K preethi	**Roopa Iranna Bagalkoti**	**Gagan Gowda V S**
1RV23AI402	**1RV23AI404**	**1RV23AI400**

MENTOR

Dr. Vijayalakshmi M N
Associate Professor
Dept. of AIML
RVCE

Contents

1. Introduction
2. Existing system
3. Identification of problem
4. Literature Review
5. Problem definition
6. Objectives
7. Methodology
8. Tools and Techniques used
9. Results and Discussion
10. Conclusion
11. References

Introduction

1. Crystoper is a Big Data-driven system developed to analyze protein crystallization conditions—a crucial step in structural biology and drug discovery.
2. The traditional trial-and-error process of finding suitable crystallization conditions is time-consuming and resource-intensive.
3. With the availability of large-scale experimental datasets, this project leverages MapReduce and Apache Spark to extract insights from over 15,000+ records, providing visual analytics to help researchers determine optimal pH, temperature, and methods for crystallization

Existing System

Traditional Methods:

- Relies on trial-and-error in labs
- Requires testing various combinations of pH, temperature, chemicals
- Low success rate, time-consuming, and costly

Computational Tools

- Some models use sequence-based features for prediction
- Pipelines extract patterns from PDB records
- Mostly use static datasets with limited scope.

Limitations

- No support for large-scale (Big Data) analysis
- Lack of visualization tools for pattern discovery
- No heat/resource monitoring (CPU, memory, time)
- Often complex or costly to set up and use

Identification of problem

Observations from Existing Systems:

1. Manual trial-based crystallization is time-consuming and inefficient.
2. Existing models are not scalable to handle large datasets (10K+ records).
3. Lack of interactive visualizations makes insight extraction difficult.
4. No support for real-time analysis or predictions.
5. Researchers have no visibility into resource usage (CPU, memory, execution time).
6. Many tools are either costly, complex, or require advanced infrastructure.

There is no unified, scalable, and interactive platform that can efficiently analyze, visualize, and predict protein crystallization outcomes using Big Data technologies.

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Literature Review

Go, change the world

Title	Author	Summary
Large-scale analysis of protein crystallization conditions	Newman et al	Created a crystallization condition database and identified frequent successful conditions using clustering.
Predicting successful crystallization using machine learning	Overton et al	Used decision trees to predict crystallization outcomes from protein sequence and experimental features.
Towards automated protein crystallization: using AI to select conditions	Cumbaa et al	Developed an AI system to suggest likely crystallization conditions based on historical success data.
Machine learning in protein crystallization	Georgiev	Explored use of SVMs and neural networks to predict crystallizability from protein sequence data.
Big Data in Structural Biology	Marx	Described how large-scale structural databases (like PDB) and data mining help in understanding macromolecular crystallization.

Summary of Literature Review

Recent research in protein crystallization since 2020 has increasingly integrated AI, machine learning, and big data approaches to improve prediction and success rates.

- Newman et al. developed a crystallization condition database to identify optimal conditions using clustering techniques.
- Overton et al. applied decision tree models to predict crystallization outcomes based on protein sequences and experimental parameters.
- Cumbaa et al. introduced an AI-based recommendation system for selecting suitable crystallization conditions.
- Georgiev employed SVMs and neural networks to analyze protein sequences for crystallization potential.
- Marx highlighted the role of big data and large-scale biological databases in advancing structural biology through data-driven insights.

Problem Definition

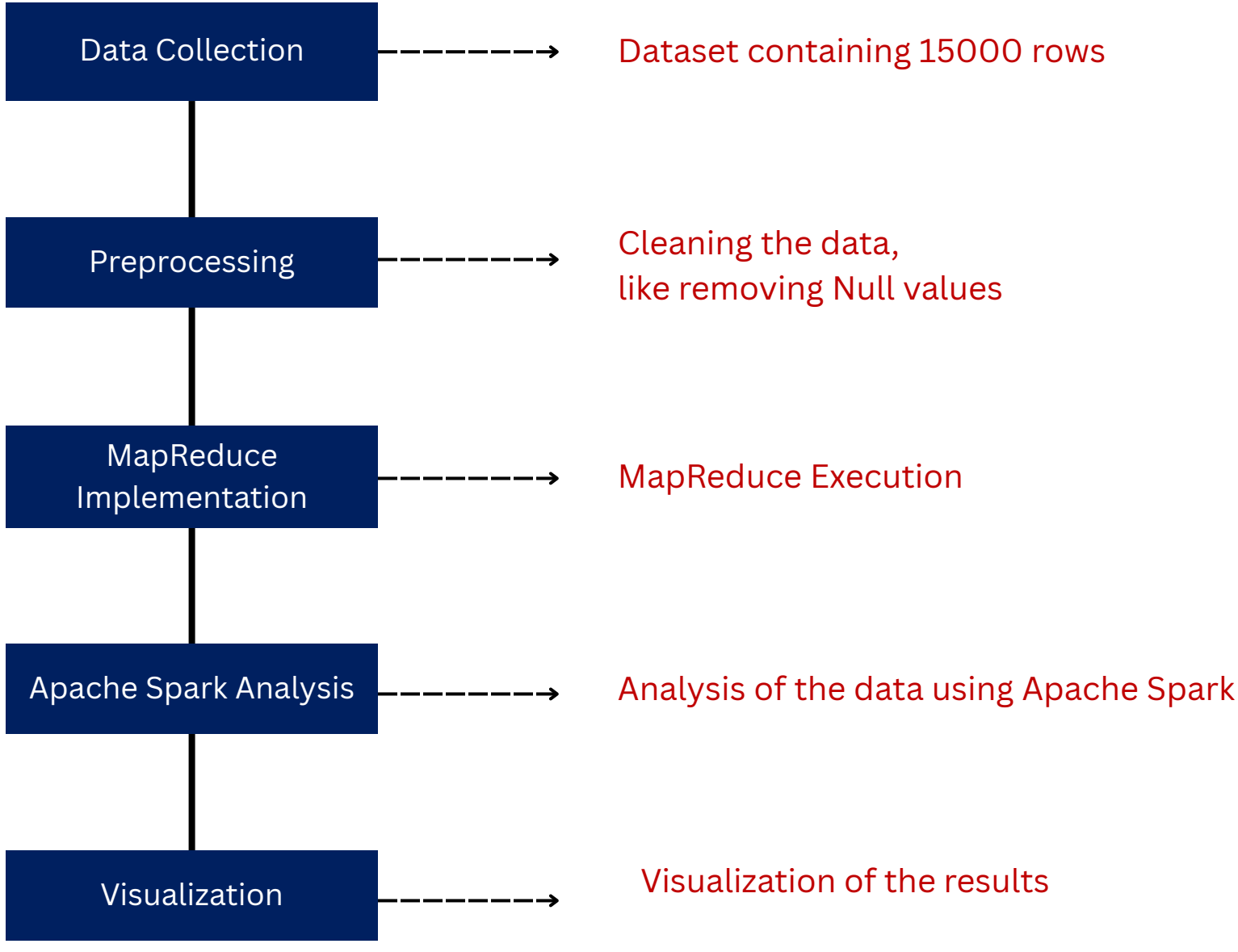
To develop an intelligent Big Data-based system named Crystoper that can efficiently process, analyze, and visualize protein crystallization experiments using MapReduce, Spark simulation, and an interactive dashboard—helping researchers identify optimal crystallization conditions and understand system resource usage at each processing stage.

Objectives

1. Analyze a large-scale dataset (15,000+ records) of protein crystallization experiments.
2. Apply MapReduce to calculateAverage pH, Average temperature, Average sequence length
3. Number of trials per method
4. Use Apache Spark for scalable distributed aggregation and processing.
5. Visualize trends with bar graphs, heatmaps, pie charts, and line plots.
6. Develop an interactive Streamlit dashboard for real-time user interaction.
7. Build a Random Forest ML model for crystallization success prediction

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Methodology



Methodology

Data Collection

- Synthetic dataset with 15,000+ crystallization records
- Includes features like: pH, temperature, sequence length, method, chemicals.

Preprocessing

- Handled missing values, normalized categories
- Converted data types (e.g., float for pH, temperature)
- Balanced the dataset (≈50% crystallized vs non-crystallized)

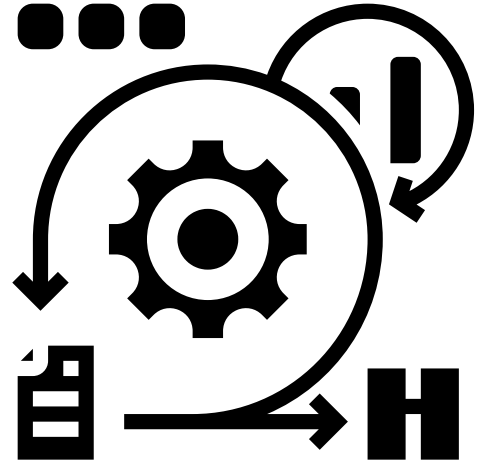
MapReduce Implementation

- Mapper extracts method-wise:
  - Count, pH, temperature, sequence length
- Reducer aggregates to compute averages and totals
- Outputs summarized method-wise statistics

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Methodology

Go, change the world



Apache Spark Analysis

- Spark used for scalable grouped aggregation

Performed:

- Avg pH, temperature, sequence length
- Grouped by crystallization method
- Converted Spark DataFrame → pandas for dashboard rendering

Visualization & Dashboard (Streamlit)

- Plots: bar charts, pie charts, heatmaps
- Interactive UI with upload and result display
- Monitored:
- Execution time
- CPU & memory usage (Heat analysis)

Tools and Techniques Used

Big Data Processing

MapReduce

- Used for distributed summarization (method-wise stats)
- Implemented using Python

Apache Spark

- In-memory processing
- Faster aggregations on large datasets

Visualization

- Matplotlib & Seaborn
  - Bar graphs, pie charts, heatmaps, line plots
- Streamlit
  - Interactive web-based dashboard
  - Real-time display of stats and system metrics

Tools and Techniques Used

Performance Monitoring

psutil (Python)

- CPU usage
- Memory consumption
- Execution time

Machine Learning

- **Random Forest Classifier**
  - For predicting crystallization success
  - Integrated into Streamlit app.

Results and Discussion

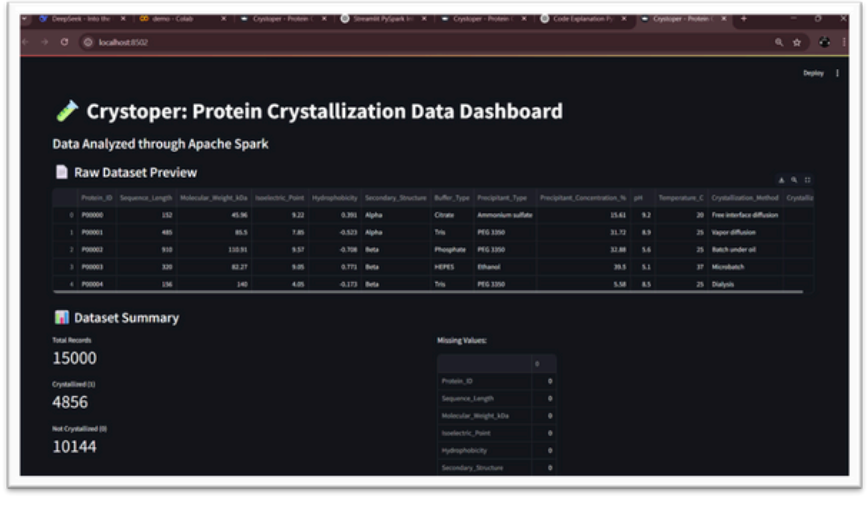
MapReduce Results

1. Analyzed 15,000+ records
2. Identified top methods: Vapor Diffusion, Sitting Drop
3. Computed: Avg pH, Avg Temperature, Avg Sequence Length
4. Execution Time: ~1.23 seconds

Apache Spark Insights

1. Performed grouped aggregation at scale
2. Avg pH, Temp, and Sequence Length per method

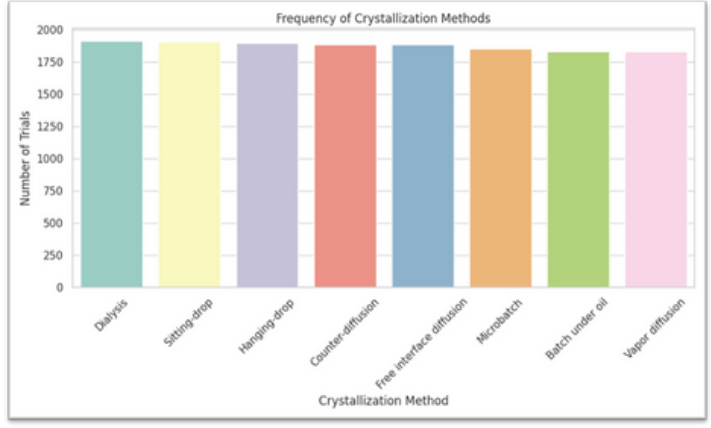
1. Execution Time: ~3.14 seconds
2. Memory Usage: ~250 MB
3. Better scalability for larger datasets



Results and Discussion

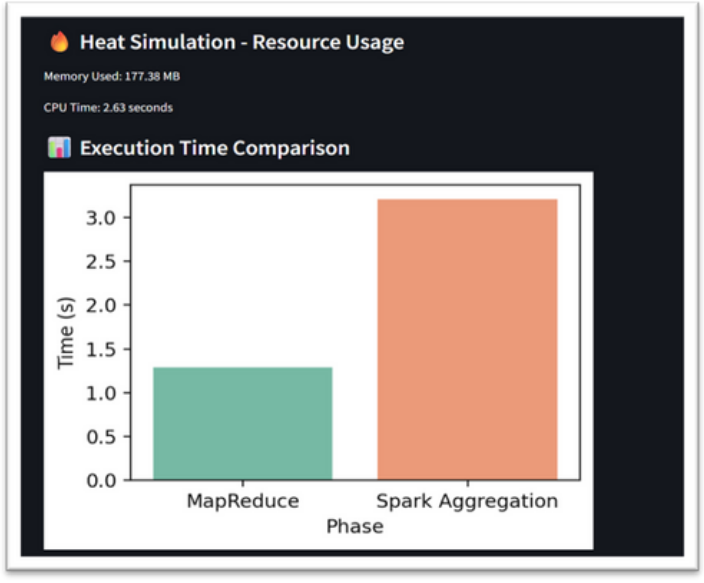
Visual Insights

- Pie Chart: ~50–60% crystallization success rate
- Bar Graph: Vapor Diffusion most commonly used method
- Line Chart: Different methods favor specific pH ranges
- Heatmap: Moderate positive correlation between pH and crystallization success.



System Load Analysis (Heat Monitoring)

- Spark consumed more CPU & memory, ideal for large-scale future use
- MapReduce better suited for medium-sized datasets



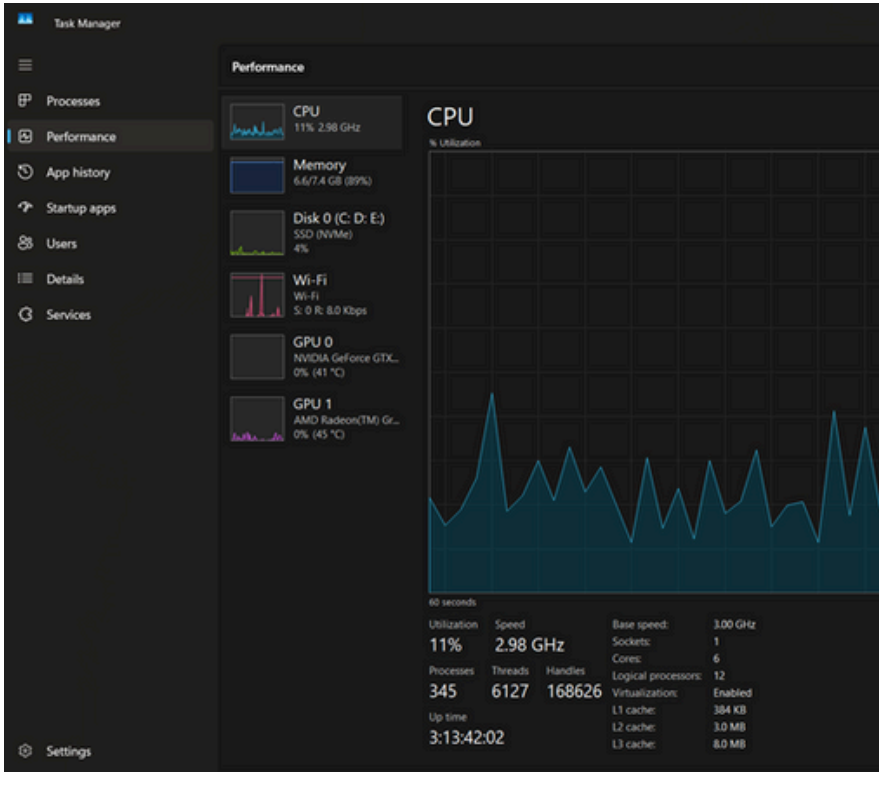
Results and Discussion

1. Data-driven approach helps prioritize optimal crystallization conditions
2. Visualization improves understanding for biologists
3. System performance metrics support real-world feasibility
4. Platform is modular and integrated with ML models
5. The model achieved an accuracy of 90.96%, indicating strong predictive capability.

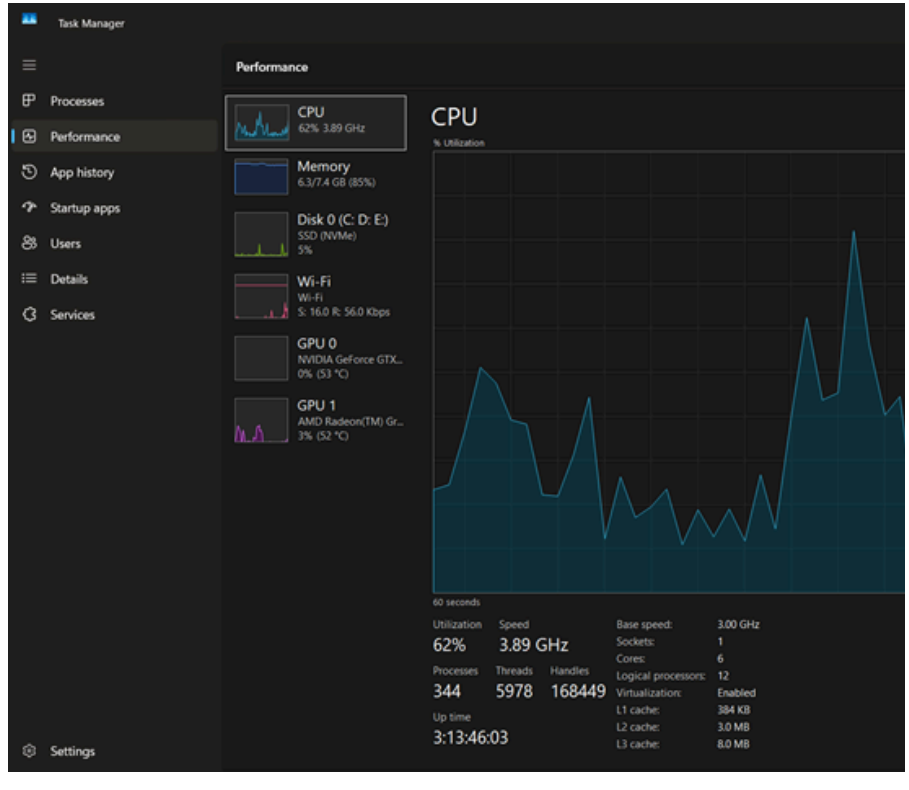
Results and Discussion

Heat Analysis of CPU:

CPU uasge Before Processing



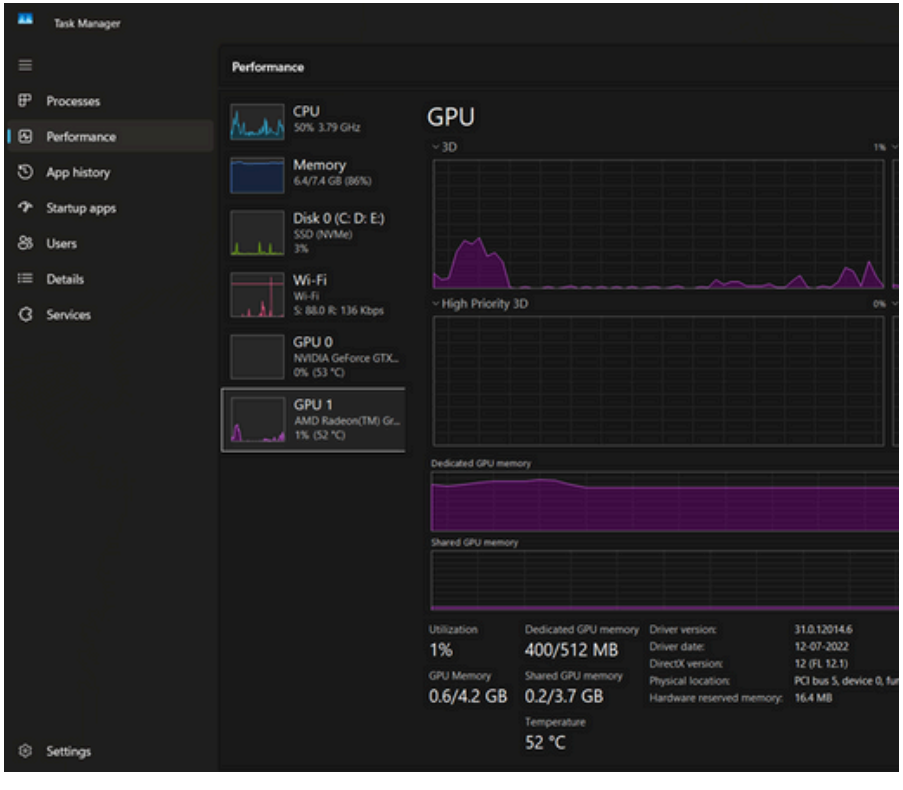
CPU uasge After Processing



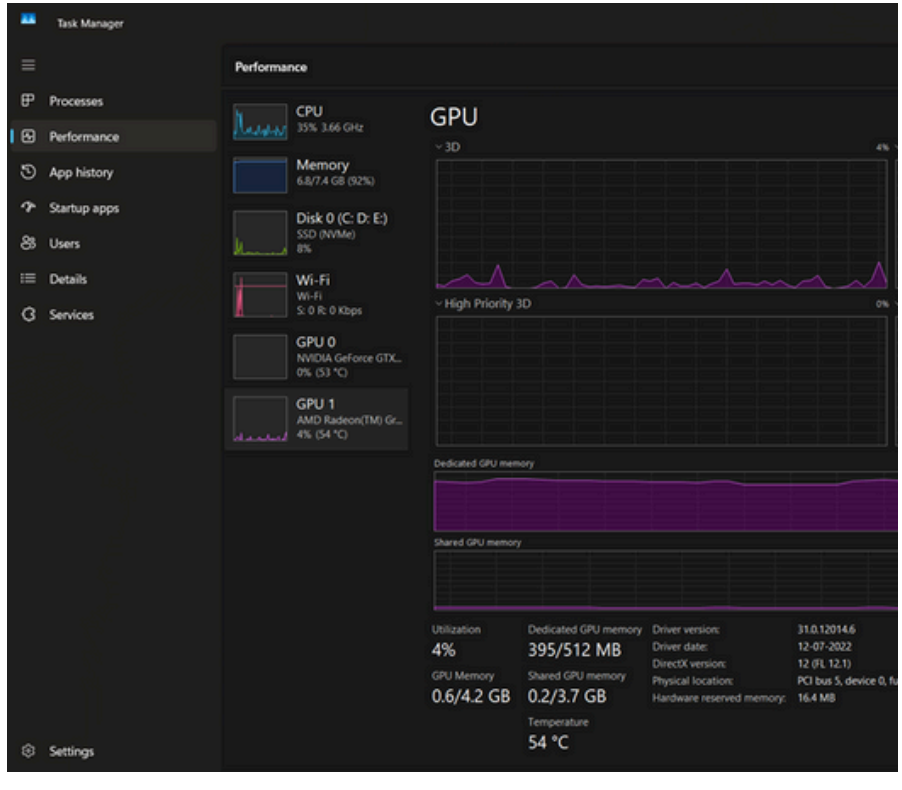
Results and Discussion

Heat Analysis of GPU:

GPU uasge Before Processing

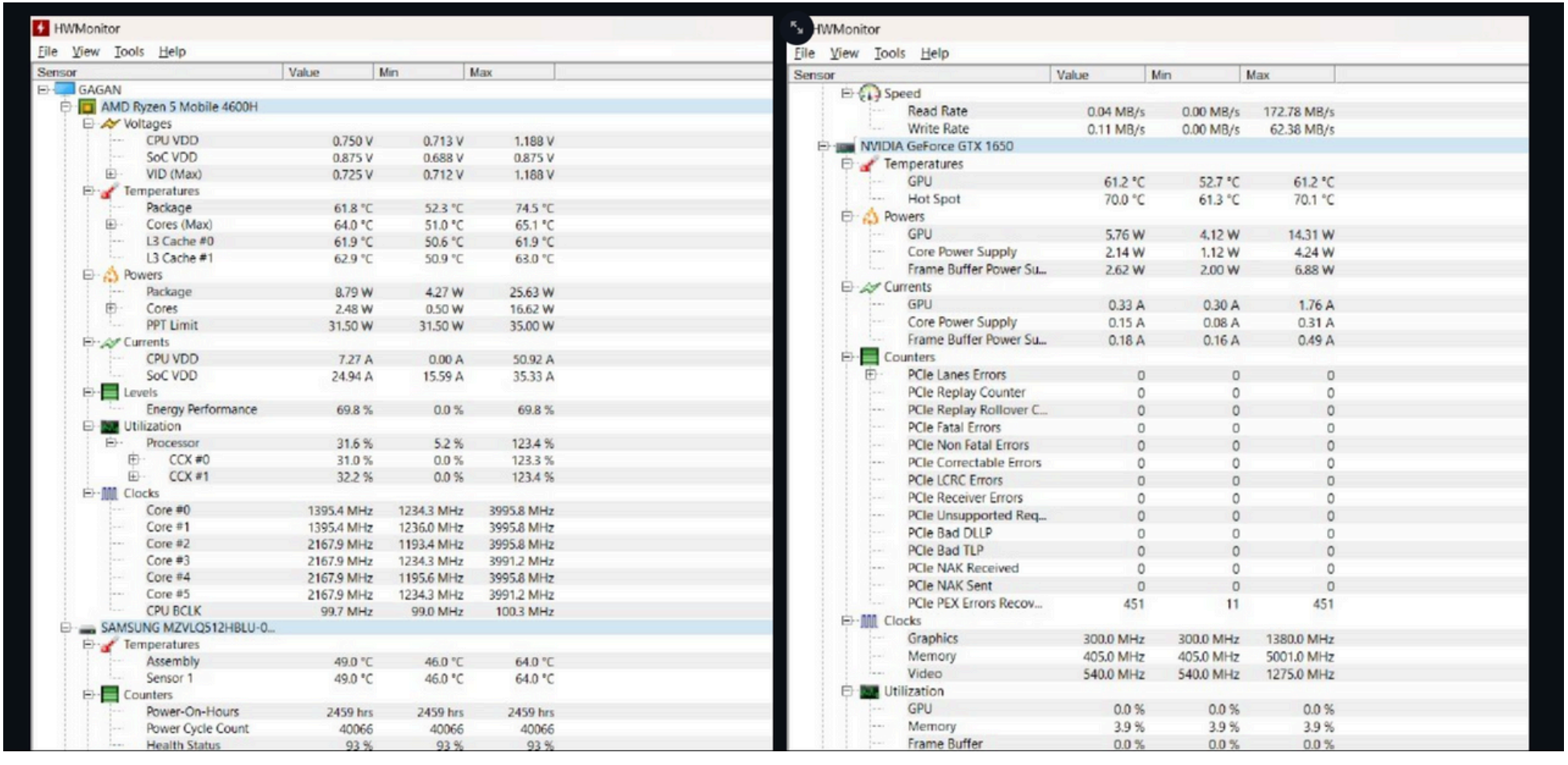


GPU uasge After Processing



Results and Discussion

GPU usage Using HW-Monitor during Processing



Conclusion

1. Built Crystoper – a Big Data-driven system for protein crystallization analysis
2. Processed 15,000+ experimental records
3. Used MapReduce and Apache Spark for distributed data processing
4. Supports data-driven decision-making in structural biology
5. Replaces expensive, inefficient trial-and-error lab processes
6. Platform is scalable and modular for research and academic use

References

[1] A. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster Computing with Working Sets," Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing (HotCloud '10), Boston, MA, USA, June 2010.

[2] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," Communications of the ACM, vol. 51, no. 1, pp. 107–113, Jan. 2008.

[3] M. Zaharia et al., "Apache Spark: A Unified Engine for Big Data Processing," Communications of the ACM, vol. 59, no. 11, pp. 56–65, Nov. 2016.

[4] RCSB Protein Data Bank, "PDB Statistics: Growth of Released Structures per Method," [Online]. Available: https://www.rcsb.org/stats/. [Accessed: June 2025].

[5] P. Bhatotia, A. Wieder, R. Rodrigues, U. A. Acar, and R. Pasquin, "Incoop: MapReduce for Incremental Computations," in Proceedings of the 2nd ACM Symposium on Cloud Computing, 2011, pp. 1–14.

[6] S. Ghosh, "Big Data Analytics Using Spark for Massive Datasets," International Journal of Computer Applications, vol. 179, no. 27, pp. 15–19, Feb. 2019.

[7] D. Loshin, Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph, Morgan Kaufmann, 2013.

[8] M. Armbrust et al., "Scaling Spark in the Real World: Performance and Usability," in Proceedings of the VLDB Endowment, vol. 8, no. 12, pp. 1840–1851, Aug. 2015.

[9] S. Ahmad and J. Lee, "Big Data Analytics in Bioinformatics," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 12, no. 1, pp. 13–24, Jan.–Feb. 2015.

[10] L. Neumann et al., "Protein Crystallization: From Traditional Methods to Microfluidics," Angewandte Chemie International Edition, vol. 47, no. 36, pp. 6832–6850, Sept. 2008.

Thank You