

Graph Neural Networks for Predicting Mechanical Response in Defective Graphene

Project Implementation Plan

Project Objective

To develop a **Graph Neural Network (GNN)**–based predictive framework that accurately estimates **mechanical properties** — such as **Young’s modulus** and **fracture strain** — of **defective graphene** structures, based on atomic-level information like **void fraction**, **atomic coordinates**, and **bond connectivity**.

Problem Statement

Graphene’s exceptional mechanical strength degrades in the presence of defects (voids, vacancies). Traditional simulations like **Molecular Dynamics (MD)** are computationally expensive.

Hence, this project aims to use a **machine learning–driven approach (GNN)** to learn the relationship between **graph structure** and **mechanical response**, significantly reducing computation time while maintaining high accuracy.

Project Scope

Scope Area	Description
Domain	Materials Science + Machine Learning
Data Source	MD simulation dataset containing atomic structures and mechanical properties
Input	Void fraction, atomic positions, bonding network
Output	Predicted mechanical responses (Young’s modulus, fracture strain)
Technology Stack	Python, PyTorch Geometric, NumPy, Pandas, NetworkX, Matplotlib
Expected Deliverables	① GNN model notebook ② Visual results ③ Technical report ④ Presentation slides

Project Phases & Tasks

Phase 1: Literature Review & Problem Understanding (Week 1)

Goal: Gain a deep understanding of GNNs and their applications in materials modeling.

Phase 2: Dataset Collection & Preparation (Week 2-3)

Goal: Prepare the molecular dynamics data for GNN input.

Phase 3: Graph Construction (Week 4)

Goal: Convert each atomic structure into a **graph representation**.

Phase 4: Model Design & Implementation (Week 5–6)

Goal: Build and train a **Graph Neural Network model** for property prediction.

Phase 5: Training, Testing & Evaluation (Week 7)

Goal: Evaluate the performance of the GNN model.

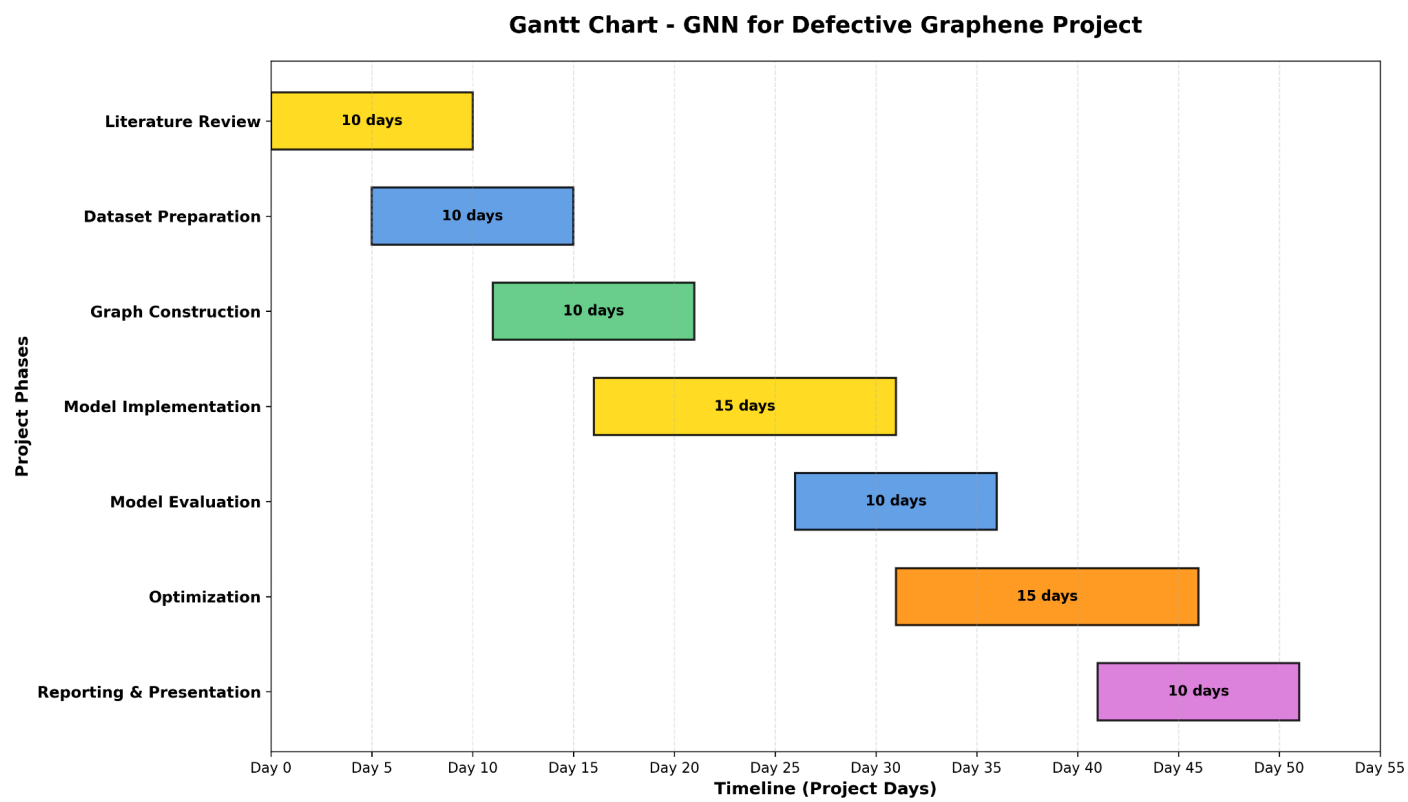
Phase 6: Optimization & Advanced Experiments (Week 8–9)

Goal: Improve model accuracy and generalization.

Phase 7: Visualization, Results & Reporting (Week 10)

Goal: Present findings with clarity.

Project Timeline (10 Weeks)



Tools & Technologies

Category	Tools Used
Programming Language	Python
Frameworks	PyTorch, PyTorch Geometric
Data Handling	NumPy, Pandas
Visualization	Matplotlib, Seaborn
Graph Processing	NetworkX
Evaluation Metrics	Scikit-learn

Expected Outcomes

Type	Description
Technical	GNN model capable of predicting mechanical response of defective graphene
Research	Demonstrates the link between atomic defects and material behavior
Practical	Faster prediction compared to MD simulations
Deliverables	Source code, trained model, performance graphs, final report

Risks & Mitigation

Potential Issue	Impact	Mitigation
Limited dataset size	Model underfits	Data augmentation, cross-validation
Overfitting	Poor generalization	Regularization, dropout
High GPU demand	Slow training	Use smaller graphs or cloud GPU
Unstable convergence	Training failure	Learning rate scheduling, normalization

Future Scope

- Extend model to **other 2D materials** (MoS₂, h-BN)
- Add **geometric GNNs** (e.g., SchNet, DimeNet)
- Implement **multi-task learning** for simultaneous property prediction
- Deploy trained model using **FastAPI/Streamlit** for real-time prediction

Summary

This project plan provides a complete roadmap for developing a **Graph Neural Network–based materials prediction system**. By integrating **graph theory, machine learning, and materials science**, this project will enable accurate, data-driven predictions of graphene’s mechanical response, reducing computational costs and paving the way for AI-assisted materials discovery.