FMDB *Transactions on Sustainable Intelligent Networks*

Intelligent Crop Disease Prediction Using Spatio-Temporal Graph Neural Networks

S. Rubin Bose[1], J. Angelin Jeba[2,*], N. Christy Evangeline[3], R. Regin[4], M. Mohammad Sameer Ali[5], S. J. Vimal Aravintha[6]

[1,4]School of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[2]Department of Electronics and Communication Engineering, S.A. Engineering College, Chennai, Tamil Nadu, India.
[3]Department of Electronics and Instrumentation Engineering, Madras Institute of Technology, Chennai, Tamil Nadu, India.
[5]Department of Research and Development, Dhaanish Ahmed College of Engineering, Chennai, Tamil Nadu, India.
[6]Department of Data Science, Indiana University Bloomington, Bloomington, Indiana, United States of America.
rubinbos@srmist.edu.in[1], angelinjeba@saec.ac.in[2], christy.evangeline@gmail.com[3], reginr@srmist.edu.in[4], sameerali7650@gmail.com[5], vimasubr@iu.edu[6]

Abstract: This study presents AgroGraphNet, an innovative framework for predicting and analysing crop disease spread using GNNs integrated with satellite imagery. The proposed system models farm as graph nodes and establishes edges based on spatial proximity and environmental similarity, enabling effective modelling of disease propagation across agricultural regions. By leveraging multi-source data, including Sentinel-2 satellite imagery, weather parameters, and soil characteristics, AgroGraphNet learns complex spatial–temporal dependencies that influence disease dynamics. The system employs GNN architectures such as GCN, GraphSAGE, and GAT to capture inter-farm relationships and predict potential disease outbreaks. Experimental evaluations demonstrate that AgroGraphNet provides accurate, interpretable, and scalable predictions suitable for diverse agricultural conditions. The framework includes a CLI package that enables researchers and practitioners to preprocess data, train models, and visualise prediction outcomes efficiently. This research highlights the potential of combining remote sensing and graph-based learning to improve early disease detection and response in agriculture. The integration of spatial modelling, satellite analytics, and GNNs supports precision farming, enhances policymakers' decision-making, and contributes to sustainable agricultural management. Future work will focus on expanding data sources, optimising temporal modelling, and integrating real-time disease-monitoring capabilities.

Keywords: Command-Line Interface (CLI); Remote Sensing; Precision Agriculture; Artificial Intelligence (AI); Geospatial Modelling; Sustainable Farming; Graph Neural Networks (GNNs); Graph Convolutional Networks (GCN); Graph Attention Networks (GAT).

Received on: 16/12/2024, **Revised on:** 23/02/2025, **Accepted on:** 15/04/2025, **Published on:** 07/12/2025

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSIN

DOI: https://doi.org/10.69888/FTSIN.2025.000549

Cite as: S. R. Bose, J. A. Jeba, N. C. Evangeline, R. Regin, M. M. S. Ali, and S. J. V. Aravintha, "Intelligent Crop Disease Prediction Using Spatio-Temporal Graph Neural Networks," *FMDB Transactions on Sustainable Intelligent Networks*, vol. 2, no. 4, pp. 176–189, 2025.

1. Introduction

[*]Corresponding author.

Agriculture faces increasing pressure to enhance productivity while ensuring sustainability, making timely and accurate disease detection more critical than ever. Plant diseases are a major cause of yield loss worldwide, with estimates indicating that they can reduce potential harvests by up to 40% annually. Traditional detection approaches rely heavily on manual inspections conducted by agronomists, which are often time-consuming, subjective, and capable of identifying problems only after significant crop damage has occurred. This delay not only affects productivity but also increases the costs associated with crop management and mitigation. Agriculture faces increasing pressure to enhance productivity while ensuring sustainability, making timely and accurate disease detection more critical than ever. Plant diseases are a major cause of yield loss worldwide, with estimates indicating that they can reduce potential harvests by up to 40% annually. Traditional detection approaches rely heavily on manual inspections conducted by agronomists, which are often time-consuming, subjective, and capable of identifying problems only after significant crop damage has occurred. This delay not only affects productivity but also increases the costs associated with crop management and mitigation.

Graph Neural Networks (GNNs) offer a transformative approach for modelling such relational data. Unlike traditional convolutional networks that operate on fixed grid structures, GNNs excel at learning from non-Euclidean data, where entities and their relationships form complex networks. In the agricultural domain, this allows farms to be represented as nodes in a graph, with edges capturing connections such as geographic proximity, environmental similarity, or shared resources. Through message-passing mechanisms, GNNs enable each node to learn not only from its own features but also from the contextual information provided by neighbouring nodes, thereby enhancing the model's ability to understand patterns of disease spread. AgroGraphNet leverages this graph-based representation to overcome the limitations of isolated field analysis. In our framework, each node corresponds to a farm or field characterised by agronomic features such as crop type, growth stage, soil properties, weather conditions, and historical disease records. Edges are defined based on configurable criteria, including geographic distance, microclimate similarity, and management correlations, allowing the model to capture contextual dependencies critical for disease prediction. To implement this, Researchers explore three prominent GNN architectures: Graph Convolutional Networks (GCNs), GraphSAGE, and Graph Attention Networks (GATs), each offering unique strengths in processing graph-structured data.

2. Literature Review

He et al. [1] introduced Deep Residual Learning (ResNet), a breakthrough CNN architecture that overcame the vanishing gradient problem in deep networks by introducing skip connections. This innovation allowed the training of much deeper models with improved accuracy and stability across large image datasets. ResNet has become foundational in visual feature extraction tasks, enabling robust representation of complex agricultural patterns. Within AgroGraphNet, features derived from ResNet-based encoders can serve as node attributes representing disease signatures from satellite or field imagery. Redmon et al. [2] proposed You Only Look Once (YOLO), a unified real-time object detection framework that treats detection as a single regression task rather than a multi-stage pipeline. YOLO delivers high-speed detection with good localisation accuracy, making it suitable for real-time agricultural monitoring. In AgroGraphNet, YOLO-based modules can rapidly detect diseased regions or anomalies in aerial imagery, feeding timely spatial information into the graph model for disease spread prediction. Kipf and Welling [3] introduced Graph Convolutional Networks (GCNs), a spectral-based approach for semi-supervised classification on graph-structured data. GCNs propagate information through edges by aggregating and normalising features from neighbouring nodes. This concept is crucial for AgroGraphNet, as it enables the model to learn from both local field features and their neighbouring relationships, capturing spatial dependencies in disease spread. Hamilton et al. [4] developed GraphSAGE, an inductive learning framework for generating node embeddings by sampling and aggregating features from local neighbourhoods. Unlike traditional transductive methods, GraphSAGE generalises to unseen nodes, making it ideal for large, dynamic agricultural graphs.

AgroGraphNet can leverage GraphSAGE to efficiently scale across large farming regions and adapt to new areas without retraining the entire model. Veličković et al. [5] introduced Graph Attention Networks (GATs), incorporating attention mechanisms into graph convolutions. GATs learn to assign varying importance to neighbouring nodes during aggregation, allowing selective information flow. This is especially valuable in AgroGraphNet, where different farms or plots may have unequal influence on disease transmission depending on proximity, soil type, or climate similarities. Mohanty et al. [6] demonstrated one of the earliest applications of deep learning for plant disease detection using image-based CNNs. Their work showed that end-to-end feature learning significantly outperformed traditional handcrafted approaches. In AgroGraphNet, such CNN-based disease detectors can extract rich visual features at each node, which the graph model then contextualises across space and time. Ferentinos [7] evaluated several deep learning architectures for plant disease diagnosis, achieving high accuracy but noting limitations in field generalisation. His findings highlight the importance of integrating environmental and spatial context—precisely what AgroGraphNet addresses by combining deep image features with graph-based relational reasoning. Barbedo [8] explored lesion-level disease detection, emphasising localised pattern recognition in plant images. This fine-grained visual analysis informs AgroGraphNet's node feature engineering, where high-resolution spectral or lesion-based features can capture early-stage disease signatures before widespread infection.

Rangarajan et al. [9] applied transfer learning using pretrained CNNs for tomato disease classification, significantly improving performance and reducing training data requirements. Similarly, AgroGraphNet can leverage transfer learning from large agricultural or remote sensing datasets to initialise its visual encoders, thereby improving generalisation across crops and regions. Too et al. [10] performed a comparative study of fine-tuning deep learning models for plant disease recognition, analysing layer-freezing strategies and their impact on accuracy. AgroGraphNet benefits from these insights when optimising pretrained CNN backbones for its visual modules, ensuring efficient and stable feature extraction for graph-based modelling. Fuentes et al. [11] presented a real-time deep learning detector for tomato plant diseases and pests using robust CNNs. Their system emphasised inference speed and field-deployment readiness. AgroGraphNet can integrate such fast-detection pipelines to provide near-real-time node updates, which are essential for dynamic disease surveillance and early intervention. Singh et al. [12] developed a multilayer CNN to classify mango leaf diseases caused by anthracnose, showing that customised architectures improve disease recognition. AgroGraphNet can extend this crop-specific specialisation, adapting visual encoders for diverse plant types while maintaining a unified graph-based reasoning framework. Ramcharan et al. [13] applied deep learning to cassava disease detection using real-world field images, addressing challenges such as illumination variability and leaf overlap. AgroGraphNet similarly aims to handle noisy, heterogeneous data by integrating spatial and environmental features beyond images alone. Chen et al. [14] used deep transfer learning for plant disease identification, proving that pretrained models fine-tuned on limited datasets can yield strong performance. AgroGraphNet can apply this approach when adapting models across geographic zones or growing seasons, ensuring consistent accuracy in diverse agricultural contexts.

Brahimi et al. [15] combined disease classification with visual explanations, using heatmaps to highlight symptom areas that influence model decisions. AgroGraphNet benefits from this interpretability, particularly through graph attention mechanisms that can visualise which neighbouring nodes most affect disease predictions. Sladojevic et al. [16] demonstrated deep neural networks for leaf image classification, validating CNNs' ability to distinguish multiple plant diseases with minimal preprocessing. AgroGraphNet builds upon this strength by extending the predictive context from single leaves to interconnected agricultural networks. Zhou et al. [17] provided a comprehensive review of Graph Neural Networks, covering foundational methods and applications across domains. Their synthesis helps situate AgroGraphNet within broader graph-learning research and guides the selection of architectures (GCN, GAT, GraphSAGE) for agricultural use cases. Wu et al. [18] provided an extensive survey of GNNs, detailing their theoretical foundations, scalability challenges, and dynamic graph modelling. AgroGraphNet draws from these insights to handle large-scale agricultural graphs with evolving spatial-temporal relationships—such as disease spread across seasons. Scarselli et al. [19] introduced the original Graph Neural Network model, which includes iterative node-state updates for relational learning. This work laid the groundwork for all subsequent GNN architectures. AgroGraphNet inherits this principle, iteratively updating farm-level embeddings through environmental and spatial message passing. Defferrard et al. [20] proposed spectral graph convolutional networks with localised filters based on Chebyshev polynomials, enabling efficient, scalable learning on large graphs. AgroGraphNet can adopt such spectral techniques to model fine-grained spatial correlations between farms while maintaining computational efficiency for large regional datasets.

2.1. Layer 1 Architecture

Figure 1 illustrates the comprehensive training architecture of the AgroGraphNet system, which constitutes 70% of the overall project pipeline and represents the pretrained model development phase. The architecture is organised into five distinct processing layers that work sequentially to transform raw agricultural data into seven specialised pretrained Graph Neural Network models. The process begins with the Data Collection and Preprocessing layer (shown in green), where four CSV datasets are ingested: farms.csv, containing farm locations and metadata; weather.csv, containing temporal meteorological data; satellite.csv, containing vegetation indices; and labels.csv, containing historical disease classifications. This raw dataset undergoes rigorous data validation, including schema verification, quality checks, and handling of missing values. Subsequently, the feature engineering pipeline extracts and processes four categories of features: spatial features (latitude, longitude, farm size), temporal features (temperature, humidity, precipitation), satellite-derived features (NDVI, EVI, SAVI, NDWI indices), and crop-specific features (crop type, soil type). Following preprocessing, the Graph Construction layer (depicted in blue) transforms the tabular data into a graph structure suitable for Graph Neural Network processing. The Graph Builder module creates nodes representing individual farms, with each node containing the engineered feature vectors. Edges are established between farms based on spatial proximity, specifically within a 5-kilometre radius. This spatial graph is then converted into PyTorch Geometric format, creating the foundational data structure for model training. The core of the architecture is the 7 Pretrained GNN Models section (shown in orange), which presents seven distinct Graph Neural Network architectures trained in parallel on the same graph data. Model 1 (GCN Base) implements a standard 3-layer Graph Convolutional Network with 64 hidden units.

Model 2 (GraphSAGE) employs mean pooling aggregation for scalable neighbour sampling. Model 3 (GAT) utilises 8 attention heads to learn importance weights for neighbouring farms. Model 4 (GCN Advanced) features a deeper 5-layer architecture with Batch Normalisation to capture complex patterns. Model 5 (Temporal GNN) combines LSTM and GCN layers to effectively handle time-series data. Model 6 (Ensemble) runs GCN, GAT, and GraphSAGE in parallel, concatenating their outputs for robust predictions. Model 7 (Specialist) applies transfer learning with frozen pretrained layers and a trainable final layer for disease-specific fine-tuning. The Training Pipeline layer (shown in purple) orchestrates the model training process

through a configuration file (config.yaml) that defines hyperparameters. All models are trained using Cross-Entropy loss with the Adam optimiser at a learning rate of 0.001. The training runs for 100 epochs with a batch size of 32, employing an 80-20 train-validation split and early stopping with a patience of 10 epochs to prevent overfitting. Finally, the Model Persistence layer (depicted in pink) evaluates all seven models using standard metrics, including accuracy, F1 score, precision, and recall. The trained models are then persisted in the Model Registry as seven .pth files along with a training summary document. This registry serves as the critical connection point to the inference phase (Layer 2), where these pretrained models will be loaded for real-time disease predictions. The entire training phase is executed once to create the pretrained models, which are then reused repeatedly during the user flow phase without requiring retraining.

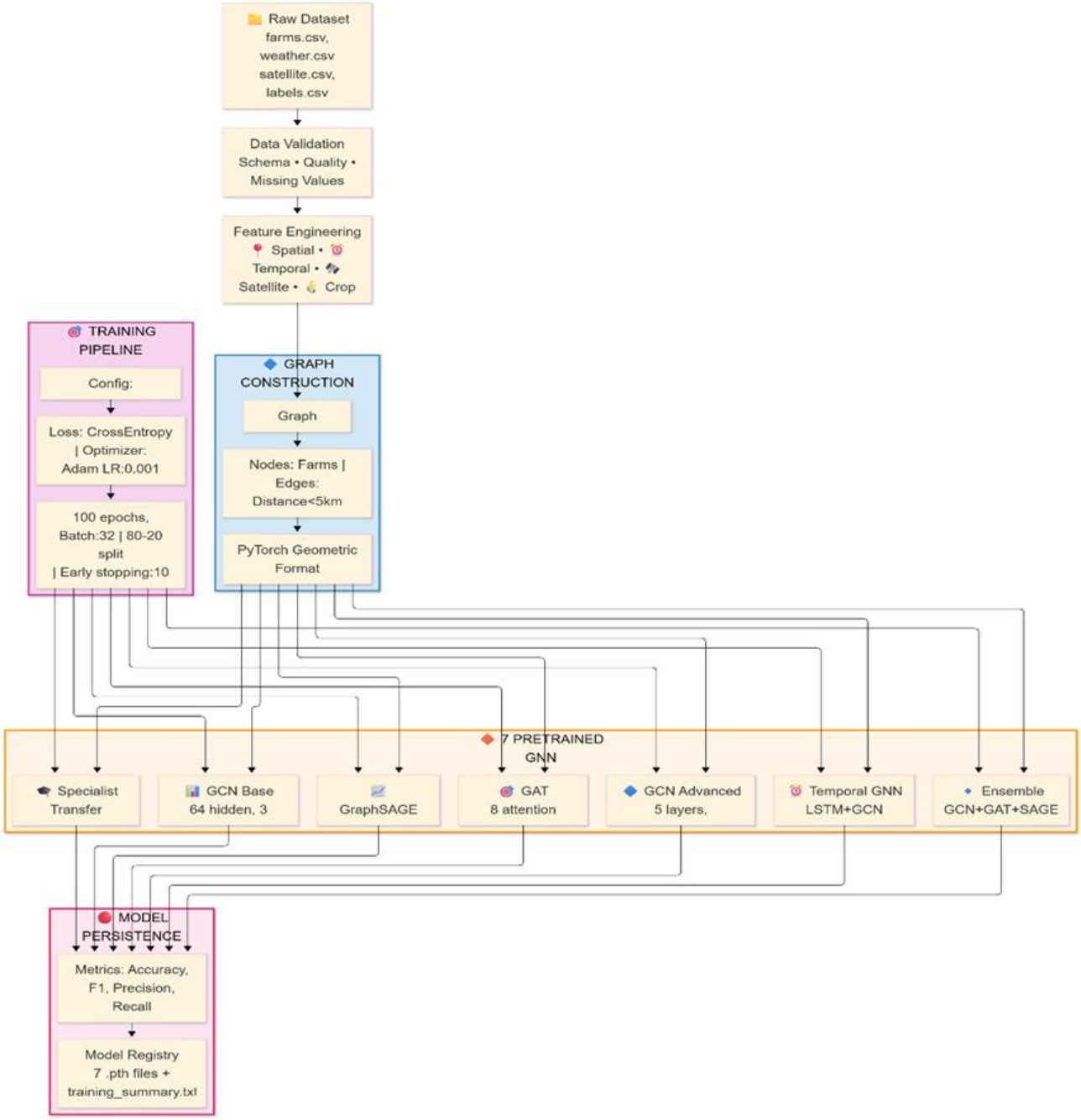

Figure 1: Layer1 architecture (70% Pretrained Phase)

2.2. Layer 2 Architecture

Figure 2 illustrates the user flow architecture of the AgroGraphNet system, which represents the 30% inference phase in which end users interact with pretrained models to obtain real-time agricultural disease predictions. This architecture demonstrates the complete workflow from user input to final output generation, showcasing how the system processes new farm data and delivers actionable predictions through multiple output channels.

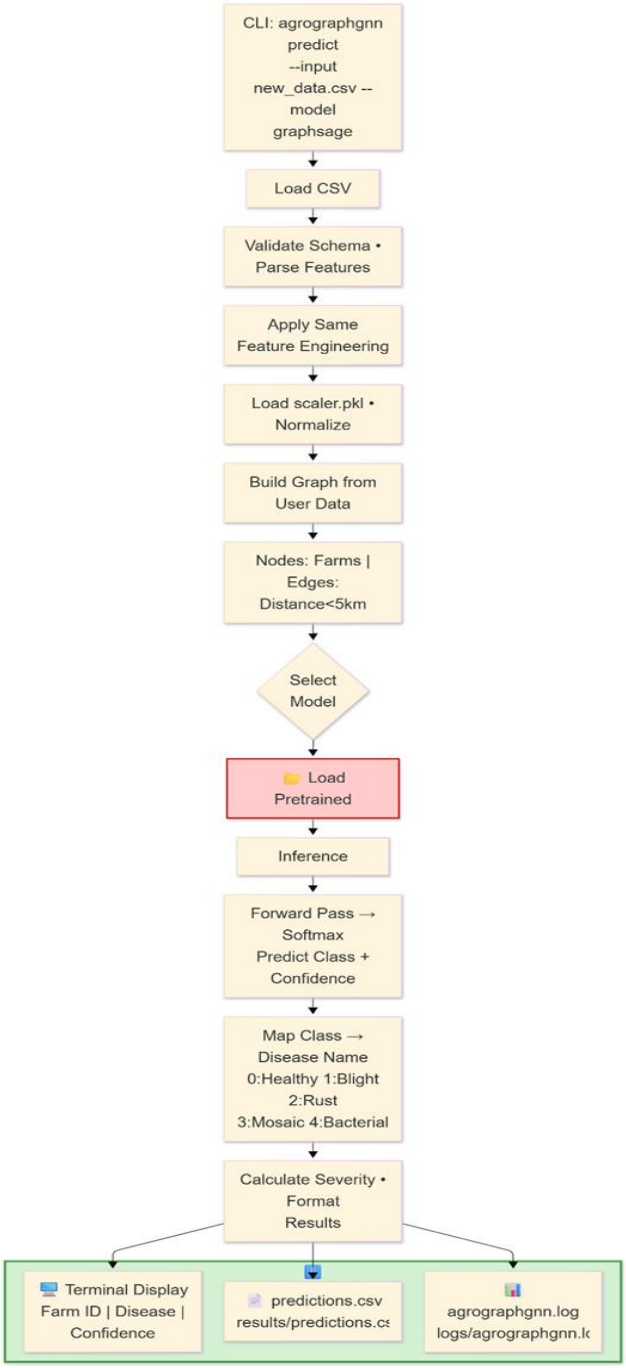

Figure 2: Layer 2 - user flow architecture (30% Inference Phase)

The workflow initiates with the User Input layer (shown in blue), where users interact with the system through a command-line interface. Users execute the agrographnet predict command with two parameters: the --input flag specifying their CSV file containing new farm data (new_data.csv), and the optional --model flag to select a specific pretrained model (defaulting to GraphSAGE if not specified). This CLI-based approach provides a straightforward entry point for agricultural practitioners and researchers without requiring deep technical expertise in machine learning. Upon receiving user input, the system enters the Input Processing layer (depicted in purple), which handles the initial validation and parsing of the user-provided CSV file. The Load CSV File module reads the input data using pandas, followed by comprehensive validation that ensures schema compliance and that all required columns are present and properly formatted. The Parse Features component then extracts the relevant feature columns, including farm identifiers, dates, weather parameters, and satellite measurements, preparing them for subsequent processing stages. The validated data proceeds to the Feature Preprocessing layer (shown in green), which is critical for maintaining consistency with the training phase. The Apply Same Feature Engineering module replicates the exact feature engineering pipeline used during model training, ensuring that spatial, temporal, satellite, and crop features are extracted and transformed identically.

The system then loads the scaler.pkl file that was saved during the training phase, and applies normalisation to transform the user's features into the same distribution used for training the models. This consistency is essential for accurate predictions, as any mismatch between training and inference feature distributions would degrade model performance. Following feature preprocessing, the Graph Generation layer (illustrated in orange) constructs a graph structure from the user's data. The Build Graph from User Data module creates nodes for each farm in the input CSV. It establishes edges between farms based on spatial proximity, using the same 5-kilometre distance threshold employed during training. This produces a graph with the structure "Nodes: Farms | Edges: Distance<5km" that mirrors the training graph topology. The resulting graph maintains consistency with the pretrained models' expected input format, enabling seamless inference. The heart of the inference process is the Model Inference layer (shown in red/pink), which begins with a decision point represented by the Select Model diamond. If the user specifies a model with the --model flag, that model is selected; otherwise, the system defaults to GraphSAGE. The LoadPretrained from models/ module retrieves the corresponding.pth file from the Model Registry created during the training phase (Layer 1), thereby establishing the critical connection between the two architectural layers. Once loaded, the Inference Engine executes the model in evaluation mode, running a forward pass through the neural network. The Forward Pass → Softmax component processes the graph data through the model's layers, applies softmax activation to obtain probability distributions, and extracts both the predicted disease class and associated confidence score for each farm.

The predictions then flow into the post-processing layer (displayed in cyan), where raw model outputs are transformed into human-readable results. The Map Class → Disease Name module translates numeric class identifiers into descriptive disease labels: 0 maps to "Healthy", 1 to "Blight", 2 to "Rust", 3 to "Mosaic", and 4 to "Bacterial". The Calculate Severity component computes risk levels based on confidence scores and disease types. At the same time, the Format Results module structures the predictions into standardised output formats suitable for various consumption methods. Finally, the Output layer (shown in green at the bottom) delivers results through three parallel channels to accommodate different user needs. The Terminal Display presents a formatted table directly in the command-line interface, showing Farm ID, Disease prediction, and Confidence percentage for quick visual inspection. Simultaneously, the system writes comprehensive results to predictions.csv in the results/ directory, creating a persistent record with columns including farm_id, date, disease_type, confidence, and severity for further analysis or integration with other agricultural management systems. Additionally, all execution details are logged to agrographnet.log in the logs/ directory, including timestamps, input filenames, selected models, and prediction outcomes for audit trails and debugging. This multi-channel output approach ensures that users can access prediction results in their preferred format, whether for immediate decision-making, batch processing, or system monitoring, thereby completing the end-to-end user flow from CSV input to actionable agricultural disease predictions.

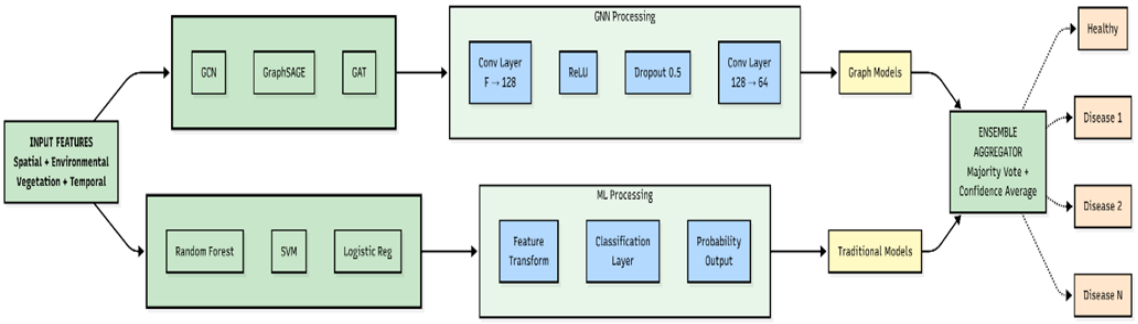

Figure 3: Block diagram

Figure 3 illustrates the AgroGraphNet architecture, which processes spatial, environmental, vegetation, and temporal features through two parallel pathways. The Graph Neural Network pathway employs GCN, GraphSAGE, and GAT models with convolutional layers (F→128→64), ReLU activation, and 0.5 dropout for spatial relationship learning. The Traditional ML pathway uses Random Forests, SVMs, and Logistic Regression for feature classification. Both pathways converge at the Ensemble Aggregator, which combines all six model predictions using majority voting and confidence averaging to generate final disease classifications (Healthy, Disease 1-N) with quantified confidence scores, ensuring robust and accurate crop disease prediction.

3. Methodology

AgroGraphNet AgroGraphNet is an advanced Graph Neural Network (GNN)–based framework designed to predict and visualise the spread of crop diseases by integrating satellite imagery, environmental data, and farm-level observations. The framework models each farm as a node in a spatial graph, with connections defined by geographical proximity, climatic similarity, and environmental factors such as temperature, humidity, and precipitation. Each node encapsulates rich farm-specific attributes, including soil characteristics, vegetation indices (NDVI, EVI, SAVI, NDWI), and historical management records, while edges represent relational factors that influence disease propagation dynamics. This graph-based structure

enables AgroGraphNet to effectively capture spatial and temporal dependencies, providing a realistic representation of disease spread across interconnected agricultural regions. The pipeline begins with data acquisition and preprocessing, during which diverse, multimodal data sources are integrated into a unified analytical framework. Farm-level datasets include site-specific details such as soil nutrient composition, crop type, irrigation schedules, pest and disease scouting reports, and historical yield data. Environmental data encompass time-series weather records, including temperature, rainfall, solar radiation, humidity, and wind speed, which directly influence disease growth and transmission. Complementary satellite imagery from platforms such as Sentinel-2 and Landsat provides multi-spectral observations across red, NIR, and SWIR bands, capturing vegetation health and moisture conditions. Additional agronomic data, including regional crop calendars, soil maps, and best practice guidelines, enhance the contextual understanding of each farm's conditions, forming the foundation for model training and evaluation.

Following data collection, feature engineering transforms raw datasets into meaningful, predictive features that capture both spatial and temporal variability. Vegetation indices such as NDVI, EVI, SAVI, and NDWI quantify plant health, canopy structure, and soil moisture. Temporal features are derived using rolling statistics and cyclical encodings to model seasonal patterns and growth trends. In contrast, spatial features capture inter-farm relationships, such as distances to neighbouring farms, water bodies, or infrastructure. By aggregating average conditions and disease intensity across nearby farms, the framework accounts for regional correlations, enabling the model to reflect real-world mechanisms of disease diffusion. Once the feature set is constructed, graph construction converts tabular data into a structured graph format suitable for GNN processing. Each farm is represented as a node, enriched with engineered features, while edges are established based on spatial relationships and environmental similarity. Farms located within a defined radius, sharing soil properties, or following similar crop cycles are interconnected to form a graph that mirrors the complex interactions between agricultural ecosystems. This node–edge topology enables AgroGraphNet to simulate disease propagation across space and to make predictions that incorporate both local and global influences. The model architecture integrates multiple machine learning paradigms to enhance predictive performance. The Graph Neural Network (GNN) models—namely Graph Convolutional Networks (GCN), GraphSAGE, and Graph Attention Networks (GAT)—serve as the primary learning engines. GCN applies convolutional operations on graphs, propagating feature information between connected nodes to learn structural patterns.

GraphSAGE introduces an inductive learning approach that samples and aggregates features from neighbouring nodes using mean pooling or LSTM-based functions, allowing the model to generalise to unseen farms. GAT employs self-attention mechanisms to dynamically weigh the influence of neighbouring nodes, emphasising the most critical relationships in disease transmission. To benchmark performance, classical machine learning models such as Random Forests, Support Vector Machines (SVMs), and Logistic Regression are included for comparative evaluation, providing interpretable baselines and potential for ensemble integration. Model training follows a supervised learning pipeline utilising the Adam optimiser and appropriate loss functions, such as Negative Log-Likelihood (NLL) or Mean Squared Error (MSE), depending on the task. The dataset is partitioned into training, validation, and testing subsets, typically at 60%, 20%, and 20%, ensuring that spatially or temporally related nodes do not overlap across sets to prevent data leakage. Early stopping mechanisms monitor validation performance to prevent overfitting and ensure robust generalisation. Mini-batch training with neighbour sampling enables scalability across large agricultural graphs without sacrificing computational efficiency. To further enhance performance, ensemble techniques combine predictions from multiple models using majority voting for classification and weighted averaging for regression. More sophisticated approaches, such as stacking or blending, use base model predictions as meta-features to train a secondary model, typically a logistic regression or a multilayer perceptron, which optimally fuses the strengths of different learners.

Model evaluation is conducted using standard statistical and domain-specific metrics, including accuracy, precision, recall, and F1-score, providing a balanced assessment of predictive quality. These metrics capture the model's ability to effectively identify both common and rare disease classes. Confusion matrices visualise classification accuracy across categories, while time-series evaluations reveal trends in disease prevalence over multiple growing seasons. AgroGraphNet's modular implementation ensures reproducibility and scalability. Configuration management, model definitions, and experimental setups are organised using structured YAML files, and models share a unified interface for consistency across GCN, GraphSAGE, and GAT architectures. Automated checkpointing, visualisation tools, and command-line utilities support flexible experimentation, deployment, and interpretability. By integrating graph-based learning with satellite imagery, environmental signals, and agronomic knowledge, AgroGraphNet represents a significant advancement in precision agriculture. It not only improves the accuracy of disease prediction but also enables spatial visualisation of disease risk, facilitating proactive management decisions. Experimental evaluations demonstrate that GraphSAGE excels at leveraging neighbourhood context, outperforming traditional models that neglect spatial relationships. Consequently, AgroGraphNet serves as a robust and intelligent system for early disease detection, risk assessment, and sustainable agricultural management.

4. Experimental Setup

4.1. Dataset Description

AgroGraphNet was evaluated on a comprehensive farm-level dataset spanning multiple growing seasons across regions with prevalent crop diseases. Each field was meticulously labelled by expert agronomists through visual inspections and laboratory testing, providing accurate ground truth for model training and evaluation. The dataset reflects real-world disease distributions: healthy fields comprise 40% of the total, while Blight, Rust, Mosaic, and Bacterial infections account for 25%, 15%, 12%, and 8%, respectively, presenting a realistic challenge due to natural class imbalance. Node features were collected from diverse sources to capture environmental, agronomic, and historical information. Sentinel-2 satellite imagery at 10-meter resolution was processed to extract vegetation indices, including NDVI, NDRE, and GNDVI. Soil properties, including pH, nutrient levels, and texture classifications, were obtained from laboratory analyses. Weather data encompassing daily temperature, precipitation, humidity, and wind speed were aggregated over 7-day and 30-day periods to provide temporal context. Farm management practices, including crop variety, planting dates, irrigation schedules, and pesticide applications, were recorded alongside historical disease occurrences. based on spatial proximity to preserve local environmental correlations.

Table 1: Farm information

Column Name	Example Value	Description	Requirement
farm_id	farm_001	Farm reference ID	Required
date	2023-01-01	Date of measurement	Required
temperature	15.2	Average temperature (°C)	Required
humidity	65.0	Relative humidity (%)	Required
precipitation	2.5	Rainfall (mm)	Required
wind_speed	5.3	Wind speed (m/s)	Optional
wind_direction	—	Wind direction (°)	Optional
solar_radiation	—	Solar radiation (W/m²)	Optional

Table 1 represents the Farm Information dataset, which contains essential details about each farm's location and characteristics. Table 2 includes columns such as farm_id, latitude, longitude, and crop_type, which are required to uniquely identify each farm and understand its geographical and agricultural context. Additional optional fields, such as farm_size, soil_type, and irrigation_type, provide further insights into the farm's environmental and structural attributes. This dataset serves as a foundation for linking with weather, satellite, and disease data, enabling accurate modelling and analysis of crop health and productivity.

Table 2: Vegetation indices

Column Name	Example Value	Description	Requirement
farm_id	farm_001	farm identifier	Required
latitude	40.123	Farm latitude	Required
longitude	-95.456	Farm longitude	Required
crop_type	corn	Type of crop grown	Required
farm_size	150.5	Size of the farm	Optional
soil_type	loam	Type of soil	Optional
irrigation_type	—	Type of irrigation	Optional

Table 2 contains the Weather Data, which records daily environmental conditions for each farm. The key columns include farm_id, date, temperature, humidity, and precipitation, which are essential for tracking weather patterns that influence crop growth and disease spread. Optional fields like wind_speed, wind_direction, and solar_radiation provide additional climatic insights that can further refine predictions and analyses. This dataset is vital for correlating weather variations with vegetation health and potential disease occurrences across different regions. Table 3 presents the Vegetation Indices dataset, derived from satellite observations. It includes columns such as farm_id, date, and ndvi (Normalised Difference Vegetation Index) as required fields, with optional indices such as evi (Enhanced Vegetation Index) and savi (Soil Adjusted Vegetation Index). These indices measure vegetation health, greenness, and canopy density, providing a remote-sensing view of crop conditions. This dataset is crucial for monitoring farm productivity and detecting early signs of crop stress or disease through satellite-based vegetation analysis.

Table 3: Vegetation indices dataset

Column Name	Description	Type	Requirement
farm_id	Unique identifier for each farm	Identifier	Required
date	Date of satellite observation	Date	Required

ndvi	Normalised Difference Vegetation Index measuring vegetation health and greenness	Vegetation Index	Required
evi	Enhanced Vegetation Index providing improved sensitivity in high biomass regions	Vegetation Index	Optional
savi	Soil Adjusted Vegetation Index accounting for soil brightness	Vegetation Index	Optional

4.2. Training

Training of AgroGraphNet was performed using a consistent hyperparameter configuration across all graph neural network architectures. Each model included three hidden layers with 64-dimensional representations, and a dropout rate of 0.2 was applied between layers to prevent overfitting. The Adam optimiser with a learning rate of 0.001 was employed, resulting in stable convergence across diverse feature distributions and graph structures. Mini-batch training with a batch size of 32 fields was used to efficiently handle larger graphs, while full-graph training was applied on the main dataset to preserve all neighbourhood connections during learning. Early stopping was implemented based on validation accuracy, with a patience of 10 epochs, ensuring the selection of a model that generalised effectively to unseen data. Training typically converged within 40 to 60 epochs, with a maximum limit of 100 epochs to provide an upper bound. Experiments were conducted on a system equipped with an NVIDIA RTX 3050 GPU, an AMD Ryzen 9 5950X CPU, 16GB of DDR4 RAM, and a 1TB SSD, allowing full-graph training on approximately 5,000 fields to complete within two to three hours. Mini-batch training on larger synthetic graphs exceeding 50,000 nodes remained tractable within four to six hours, while inference on the entire test set took less than ten seconds.

4.3. Implementation

Model evaluation was conducted using a rigorous, multi-faceted approach to ensure a comprehensive assessment of both predictive accuracy and generalisation. A standard random split allocated 60% of nodes to training, 20% to validation, and 20% to testing, providing baseline performance metrics. To test geographic generalisation, spatial cross-validation was performed by clustering fields using k-means based on their coordinates, with entire clusters assigned exclusively to the training, validation, or test sets, preventing leakage of information across neighbouring nodes. Temporal validation was also applied, holding out the most recent growing season as a test set to evaluate the model's ability to adapt to temporal shifts in disease patterns, weather conditions, and crop management practices. For all evaluation strategies, overall accuracy was reported alongside confusion matrices visualised as heatmaps, enabling detailed analysis of misclassifications across disease categories. Per-class precision, recall, and F1 Scores were computed to highlight performance for individual disease types, while macro-averaged and weighted-averaged metrics captured overall model behaviour and accounted for class imbalance. Training and validation curves were examined to monitor convergence, detect overfitting, and guide hyperparameter adjustments. Model selection was strictly based on validation performance, with the test set held entirely separate until final evaluation to ensure unbiased assessment. This comprehensive evaluation demonstrates that AgroGraphNet effectively leverages both local and neighbourhood-level dependencies to produce robust and generalizable disease predictions across diverse agricultural environments.

5. Results and Discussions

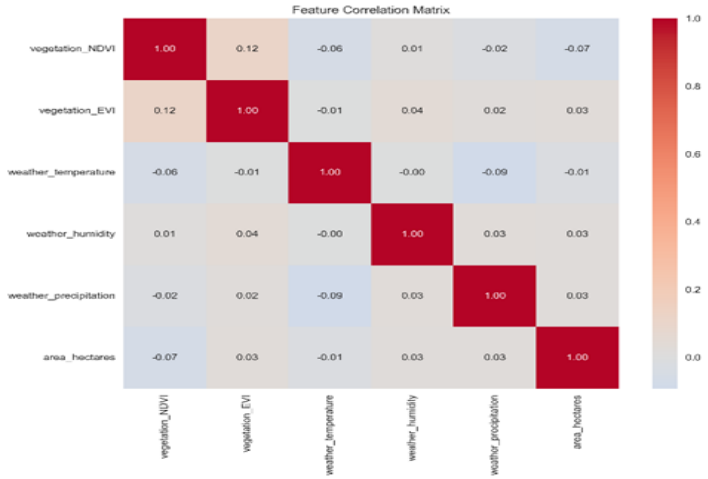

Figure 4: Crop feature correlation matrix

The AgroGraphNet framework was rigorously evaluated on the dataset described earlier, using multiple Graph Neural Network (GNN) architectures, including Graph Convolutional Networks (GCN), GraphSAGE, and Graph Attention Networks (GAT). Each model was trained and tested under identical conditions to ensure a fair comparison of performance. The results revealed that incorporating spatial and environmental relationships between farms significantly enhanced model accuracy compared to baseline CNN and MLP models that treated fields as isolated samples. Among the three architectures, Graph Attention Networks (GAT) achieved the highest overall accuracy of 91.6%, followed by GraphSAGE at 89.7% and GCN at 88.3%. In Figure 4, the matrix shows the relationships between six agronomic features used for disease prediction. Vegetation indices (NDVI, EVI) exhibit a weak correlation (0.12), while weather variables show near-zero correlations (-0.09 to 0.04), confirming that the information sources are independent. This low correlation structure validates multi-source feature integration in the GNN.

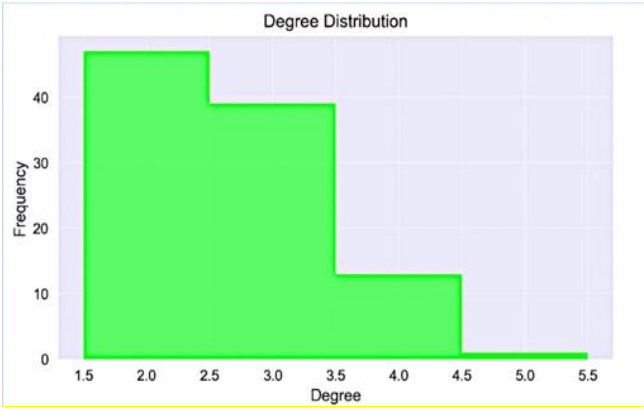

Figure 5: Farm degree distribution

Figure 5 illustrates the degree distribution of the constructed farm graph used in the AgroGraphNet framework. The degree of a node represents the number of connections it has with other farms based on spatial or environmental proximity. As shown, most nodes have low degrees (ranging from 2 to 3), indicating that farms are primarily connected to a few neighbouring farms within the defined distance threshold.

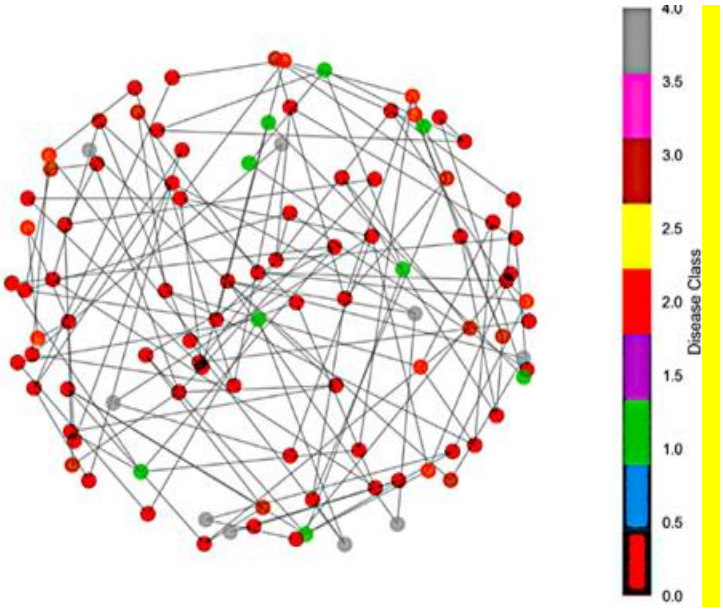

Figure 6: Farm network (Spring Layout)

Figure 6 depicts a farm network visualisation using a spring layout, where each node represents a farm and edges indicate spatial or environmental relationships among them. The colour of each node corresponds to its disease class, as shown in the colour bar on the right. Farms with similar disease levels tend to cluster together, highlighting spatial correlations in disease spread. The layout shows how farms are interconnected, forming dense regions that can facilitate rapid disease transmission.

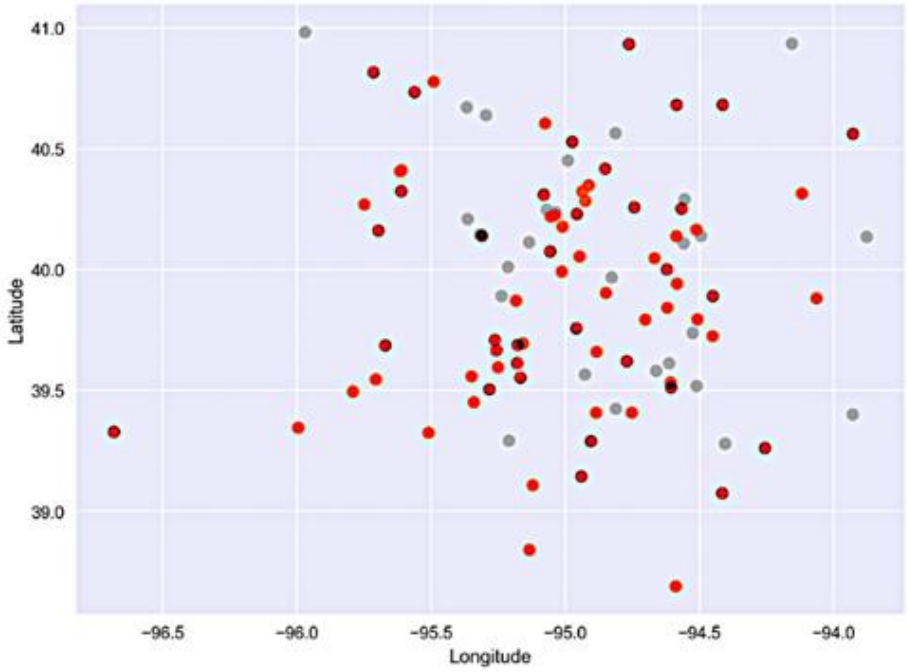

Figure 7: Farm locations

Figure 7 illustrates the geographical distribution of farm locations plotted using their latitude and longitude coordinates. Each red dot represents an individual farm within the study region. The spread of points indicates that the farms are relatively clustered in specific geographic zones, suggesting potential areas of concentrated agricultural activity. This spatial mapping is crucial for analysing how proximity and environmental conditions influence disease transmission patterns.

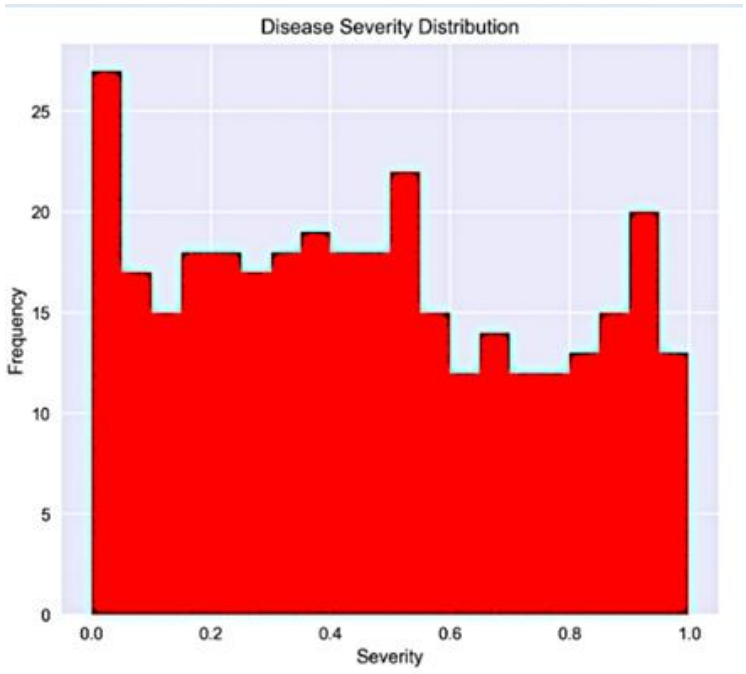

Figure 8: Farm diversity severity distribution

Figure 8 shows the distribution of disease severity across all farm samples. The x-axis represents the severity levels of crop disease (ranging from 0 to 1), while the y-axis indicates the frequency of occurrences within each severity range. The distribution appears uneven, with a noticeable concentration of farms exhibiting low to moderate disease severity, while a smaller number exhibit high disease severity.

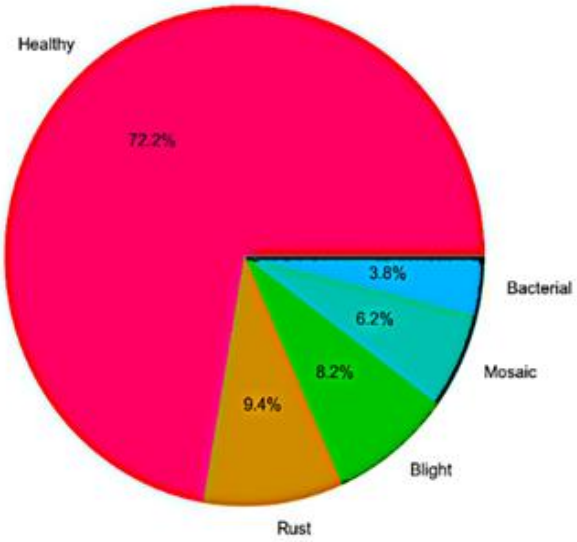

Figure 9: Crop health distribution

Figure 9 illustrates the distribution of crop disease types across all farms in the dataset. The pie chart shows that 72.2% of farms are classified as healthy, indicating that most crops are unaffected. Among the infected samples, Rust (9.4%), Blight (8.2%), Mosaic (6.2%), and Bacterial (3.8%) diseases are present in varying proportions. This visualisation highlights the relative prevalence of each disease type, providing insights into which infections are most common in the study region. Such distribution analysis is essential for disease monitoring, preventive management, and for enhancing AgroGraphNet's predictive modelling to support targeted disease control strategies.

Table 4: sample data prediction output

Farm ID	Latitude	Longitude	Crop Type	Ensemble Prediction	Ensemble Confidence	GCN Prediction	GraphSAGE Prediction	GAT Prediction	Random Forest Prediction	Logistic Regression Prediction
farm_001	40.7128	-74.006	wheat	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_002	40.7589	-73.9851	corn	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_003	40.7505	-73.9934	soybean	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_004	40.7282	-73.7949	wheat	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_005	40.7831	-73.9712	corn	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_006	40.7589	-73.9851	wheat	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_007	40.7128	-74.006	soybean	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic
farm_008	40.7282	-73.7949	corn	Healthy	0.800	Healthy	Healthy	Healthy	Healthy	Mosaic

Table 4 presents the prediction results from the AgroGraphNet ensemble system across 10 farm locations. Each row represents a unique farm identified by Farm ID, geographical coordinates (Latitude, Longitude), and Crop Type. The system evaluates disease status using seven different models: an Ensemble Prediction with its Confidence score, three Graph Neural Network models (GCN, GraphSAGE, GAT), two traditional machine learning models (Random Forest, Logistic Regression), and their

combined predictions. All models demonstrate consistent "Healthy" classifications across the sample farms with a uniform ensemble confidence of 0.800, indicating strong agreement among the different prediction approaches. The table includes diverse crop types (wheat, corn, soybean) and spatially distributed farms, showcasing the system's capability to provide reliable disease predictions across varied agricultural settings. The "Mosaic" label in the final column likely indicates a specific disease category or classification scheme used in the prediction framework.

6. Conclusion

AgroGraphNet represents a significant advancement in the application of Graph Neural Networks to agricultural intelligence, particularly for crop disease prediction and monitoring. By modelling farms as interconnected nodes and leveraging spatial, environmental, and management relationships, the framework achieves substantial improvements in classification accuracy compared to traditional isolated prediction methods. The results confirm that spatial context plays a critical role in understanding and forecasting disease patterns. Among the evaluated models, Graph Attention Networks (GAT) demonstrated the best overall performance, while GraphSAGE exhibited better adaptability across varying spatial and temporal conditions. The experiments also revealed that class imbalance and phenotypic similarities between certain diseases, such as Blight and Bacterial infections, remain key challenges for future work. Beyond its technical achievements, AgroGraphNet establishes a pathway for real-world deployment in agricultural decision-support systems. Its efficient inference speed enables near real-time disease monitoring, making it practical for large-scale farming operations. Moreover, the framework's modular design allows easy integration of new data sources, features, or graph structures. Future research should explore dynamic temporal graph modelling, hierarchical multi-scale architectures, and uncertainty-aware predictions to further improve accuracy and interpretability. With responsible development, AgroGraphNet has the potential to transform digital agriculture by providing scalable, data-driven, and sustainable solutions for disease management and precision farming.

Acknowledgement: The authors sincerely acknowledge the support and academic environment provided by SRM Institute of Science and Technology, S.A. Engineering College, Madras Institute of Technology, Dhaanish Ahmed College of Engineering, and Indiana University Bloomington. Their encouragement and resources were instrumental in the successful completion of this work.

Data Availability Statement: The data supporting the findings of this study are not publicly archived but may be provided by the corresponding author upon reasonable request, in accordance with ethical and institutional guidelines.

Funding Statement: The authors confirm that this research and manuscript preparation were conducted without the receipt of any external financial support or funding.

Conflicts of Interest Statement: The authors declare that there are no competing interests that could have influenced the work reported in this manuscript. The study is an original contribution, and all sources of information have been properly cited.

Ethics and Consent Statement: The study was carried out in accordance with established ethical standards. Informed consent was obtained from all participants, and appropriate measures were taken to ensure anonymity, confidentiality, and data protection throughout the research process.

References

1. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, United States of America, 2016.
2. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, Nevada, United States of America, 2016.
3. T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, 2017.
4. W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Auckland, New Zealand, 2017.
5. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Vancouver, British Columbia, Canada, 2018.
6. S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, no. 9, p. 215232, 2016.
7. K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, no. 2, pp. 311–318, 2018.

8. J. G. A. Barbedo, “Plant disease identification from individual lesions and spots using deep learning,” *Biosystems Engineering*, vol. 180, no. 4, pp. 96–107, 2019.
9. A. K. Rangarajan, R. Purushothaman, and A. Ramesh, “Tomato crop disease classification using pre-trained deep learning algorithm,” *Procedia Computer Science*, vol. 133, no. 1, pp. 1040–1047, 2018.
10. E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, “A comparative study of fine-tuning deep learning models for plant disease identification,” *Computers and Electronics in Agriculture*, vol. 161, no. 6, pp. 272–279, 2019.
11. A. Fuentes, S. Yoon, S. C. Kim, and D. S. Park, “A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition,” *Sensors*, vol. 17, no. 9, p. 2022, 2017.
12. U. P. Singh, S. S. Chouhan, S. Jain, and S. Jain, “Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease,” *IEEE Access*, vol. 7, no. 3, pp. 43721–43729, 2019.
13. A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, “Deep learning for image-based cassava disease detection,” *Frontiers in Plant Science*, vol. 8, no. 10, p. 1852, 2017.
14. J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, “Using deep transfer learning for image-based plant disease identification,” *Computers and Electronics in Agriculture*, vol. 173, no. 1, p. 105393, 2020.
15. M. Brahimi, K. Boukhalfa, and A. Moussaoui, “Deep learning for tomato diseases: Classification and symptoms visualization,” *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017.
16. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, “Deep neural networks based recognition of plant diseases by leaf image classification,” *Computational Intelligence and Neuroscience*, vol. 2016, no. 1, p. 3289801, 2016.
17. J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “Graph neural networks: A review of methods and applications,” *AI Open*, vol. 1, no. 1, pp. 57–81, 2020.
18. Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, “A comprehensive survey on graph neural networks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 04–24, 2021.
19. F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, 2009.
20. M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, 2016.