Infectious Risk Prediction and Analysis System

Final Project Report

CS-351: Artificial Intelligence

Fall 2025

Instructor: Ahmed Nawaz

Submitted By:

Zain Jamshaid	2023772
Hamid Quyyum	2023418
Hiba Zulfiqar	2023246

Faculty of Computer Science and Engineering

GIK Institute of Engineering Sciences and Technology

Semester: Fall 2025

Abstract

This project presents an Artificial Intelligence framework designed to predict infectious disease outbreaks and simulate their spread across geographic regions. The system integrates graph-based search algorithms with advanced machine learning architectures. The work progresses through three distinct phases: (1) the development of graph traversal algorithms (BFS and A*) to simulate disease transmission vectors and risk assessment, (2) the establishment of a robust feature engineering pipeline to handle temporal climate and disease data, and (3) the implementation of a Deep Artificial Neural Network (ANN) to capture non-linear dependencies in outbreak data. Evaluated on a global dataset combining climate variables and disease metrics, the final ANN model achieved a prediction accuracy of $R^2 = 0.95$ for Dengue cases and $R^2 = 0.94$ for Malaria cases, significantly outperforming traditional Random Forest baselines. The framework demonstrates that combining path-finding algorithms with deep learning creates a powerful tool for both proactive outbreak prediction and reactive spread simulation.

1 Introduction & Motivation

1.1 Context and Significance

Infectious diseases continue to threaten public health worldwide, with transmission rates accelerated by human mobility and shifting environmental conditions. Traditional monitoring often relies on reactive data, lacking the capability to forecast outbreaks before they occur. This project addresses this gap by developing an intelligent system that not only predicts case counts based on environmental factors but also models the physical spread of disease across borders.

1.2 Project Evolution

This work evolved through three progressive phases, identifying limitations in early approaches to drive subsequent improvements:

- **Phase 1: Simulation & Risk Assessment:** Implementation of Breadth-First Search (BFS) to model wave-like disease propagation and A* Search to calculate transmission paths based on regional connectivity.
- **Phase 2: Baseline Modeling & Feature Engineering:** Development of a Random Forest Regressor. Initial results showed poor performance ($R^2 \approx 0.11$), prompting the creation of a complex feature engineering pipeline (lag features, rolling windows) which raised accuracy significantly.
- **Phase 3: Deep Learning Integration:** Transitioning to an Artificial Neural Network (ANN) to address the residual variance in Dengue predictions. This phase achieved the highest system performance, with significant error reduction in multi-output regression tasks.

1.3 Motivation

While traditional models often treat disease data as static, this project combines temporal dynamics (time-series forecasting) with spatial dynamics (graph-based spread). This dual approach allows stakeholders to answer two critical questions: "How many cases will occur?" (ML Prediction) and "Where will the disease go next?" (Search Simulation).

2 Problem Definition & Objectives

2.1 Core Research Questions

1. Can graph search algorithms effectively model the propagation of disease across connected regions?
2. How significant is the impact of temporal feature engineering (lags, rolling averages) on model performance?

3. Can Deep Neural Networks capture complex environmental-disease relationships better than ensemble tree methods?

2.2 Project Objectives

- **Primary:** Develop an end-to-end framework integrating data collection, data processing, spread simulation, and outbreak prediction.
- **Technical:** Achieve $R^2 > 0.90$ for both Malaria and Dengue case predictions.
- **Simulation:** Visualize disease transmission paths using weighted connectivity graphs.

3 System Architecture

The system architecture is divided into two primary modules: the Graph-Based Simulation Module and the Predictive Analysis Module.

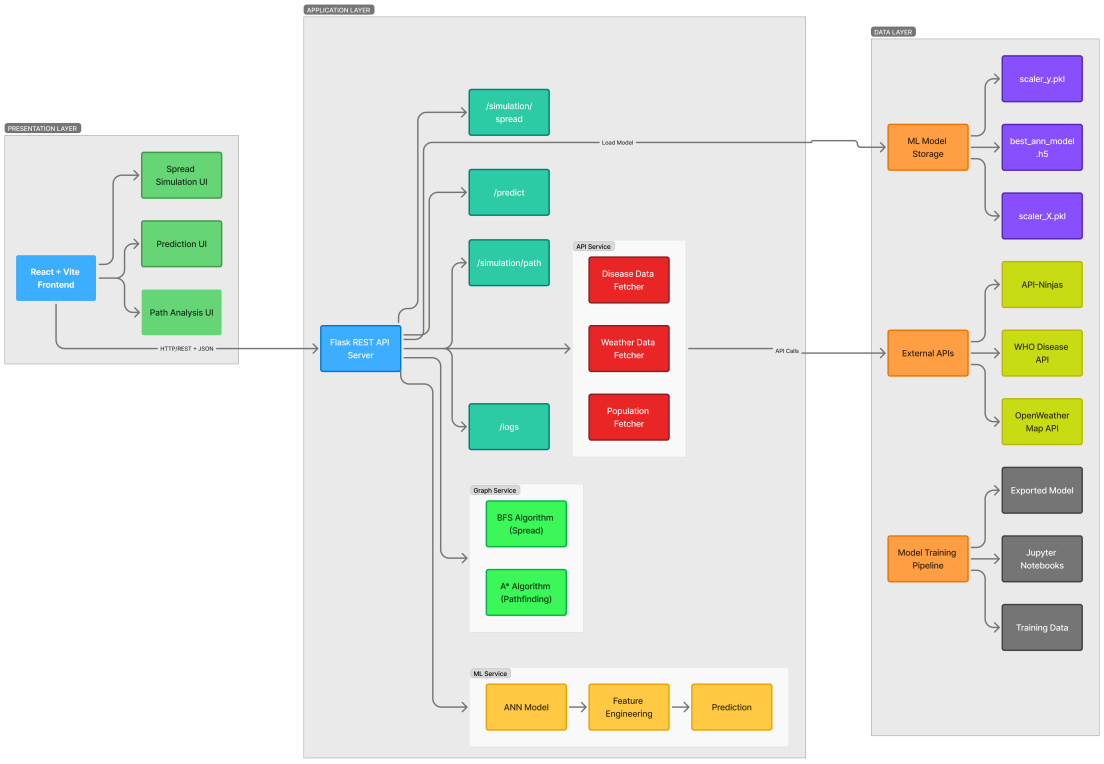

Figure 1: Overall System Architecture of the Infectious Risk Prediction and Analysis System. The framework integrates a Data Processing Pipeline, Graph-Based Simulation Module (using BFS and A* for spread and risk assessment), and Predictive Analysis Module (ANN for outbreak forecasting).

3.1 Component Description

- **Data Processing Pipeline:** Ingests raw climate and health data, applying cleaning, normalization, and feature extraction.
- **Graph Engine:** Represents countries/regions as nodes and connectivity (direct flights/borders) as edges. Uses BFS for spread simulation and A* for risk pathfinding.

- **Predictive Core:** A multi-output Deep Learning model (ANN) trained on historical data to forecast future case counts.

4 Data Description & Feature Engineering

4.1 Dataset Overview

The dataset spans the years 2000–2023, covering 120 countries. It includes meteorological data (temperature, precipitation, UV index) and health metrics (Malaria/Dengue cases).

year	month	country	region	avg_temp_c	precipitation_mm	air_quality_index	uv_index	malaria_cases	dengue_cases	population_density	healthcare_budget
2000	1	Palestinian Territory	Central	28.13	152.08	110.49	12.00	53	145	113	1068
2000	2	Palestinian Territory	Central	30.89	119.59	83.47	12.00	132	48	113	1068
2000	3	Palestinian Territory	Central	31.37	95.88	93.10	12.00	34	80	113	1068
2000	4	Palestinian Territory	Central	28.48	175.32	105.53	9.40	23	133	113	1068
2000	5	Palestinian Territory	Central	26.89	191.45	60.21	9.94	39	74	113	1068

Table 1: Sample rows from the dataset

4.2 Advanced Feature Engineering

Phase 2 analysis revealed that raw data was insufficient for accurate prediction ($R^2 = 0.11$). A robust engineering pipeline was implemented:

- **Lag Features:** Added case numbers from 1, 2, 3, 6, and 12 months prior to capture infection momentum.
- **Rolling Windows:** Calculated 3-month and 6-month rolling averages and standard deviations to smooth noise.
- **Cyclical Encoding:** Transformed 'Month' into sin and cos components to preserve seasonal continuity (e.g., Dec to Jan transition).
- **Scaling:** Applied StandardScaler to inputs for ANN stability.

5 Algorithmic Implementation: Search & Simulation

5.1 BFS: Disease Spread Simulation

To model the "wave-like" spread of a contagion, Breadth-First Search (BFS) was implemented.

- **Logic:** The algorithm creates transmission waves. Level 0 is the outbreak source. Level 1 comprises direct neighbors, and Level 2 comprises neighbors of neighbors.
- **Result:** In a simulation starting in Belgium, BFS successfully identified a spread to 12 countries over 3 levels, moving from the Central Region to the West.

5.2 A*: Risk Assessment Model

To determine the safest transmission routes (lowest disease risk), A* Search was utilized.

- **Cost Function:** Disease risk is measured by predicted malaria cases—lower is safer. Each country's risk is normalized (malaria cases $\div$ 100) to calculate the cumulative path cost.
- **Heuristic:** Geographic distance using the Manhattan distance formula based on latitude/longitude coordinates between the current node and destination.
- **Algorithm:** A* formula $f(n) = g(n) + h(n)$, where:
  - $g(n)$ = cumulative disease risk cost from start to current node.

- $h(n)$ = geographic distance heuristic to destination.

- **Result:** The algorithm calculates a "Total Risk Cost" for the safest path. For example, a path from Pakistan to Bangladesh yielded a Risk Score of 0.0523 (representing low cumulative disease risk along that route), while an alternative path through high-risk countries showed a higher score, indicating a less safe transmission route.

6 Machine Learning Implementation: Prediction

6.1 Baseline: Random Forest Regressor

The initial approach used a Random Forest ($n\_estimators = 100$). While robust, it struggled with the complex non-linearity of Dengue cases without feature engineering, resulting in an RMSE of 43.38 for Malaria initially.

6.2 Advanced: Artificial Neural Network (ANN)

To maximize predictive accuracy, a Deep Feedforward Neural Network was designed using TensorFlow/Keras.

- **Input Layer:** Matching feature space dimension.
- **Hidden Layers:** Four dense layers with decreasing neuron counts ($256 \rightarrow 128 \rightarrow 64 \rightarrow 32$).
- **Regularization:** Batch Normalization and Dropout (0.2–0.3) to prevent overfitting.
- **Activation:** ReLU for hidden layers.
- **Optimization:** Adam optimizer ($lr = 0.001$) with Mean Squared Error (MSE) loss.
- **Training:** Implemented EarlyStopping and ReduceLROnPlateau.

7 Results & Discussion

7.1 Quantitative Analysis

The progression from Phase 2 (Random Forest) to Phase 3 (ANN) yielded clear performance gains. The inclusion of temporal features was the single most impactful change, followed by the architectural shift to Neural Networks.

Model	**Feature Engineering?**	**Malaria R^2**	**Dengue R^2**	**Notes**
Random Forest	No	0.11	0.26	Baseline failure
Random Forest	Yes	0.94	0.83	Strong Malaria prediction
ANN (20 Epochs)	Yes	0.94	0.95	Best Overall Performance

Table 2: Comparative Performance Metrics (R^2 Score)

7.2 Visual Performance Comparison

- **Random Forest:** The scatter plot for Dengue showed a dispersed cloud ($R^2 = 0.83$), indicating high variance.
- **ANN:** The scatter plot for Dengue tightened significantly around the "Perfect Prediction" line ($R^2 = 0.95$), demonstrating the network's ability to capture complex disease dynamics.

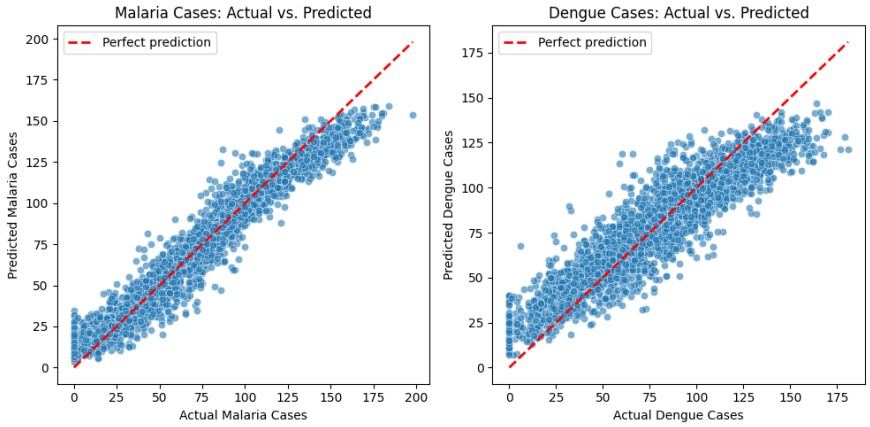

Figure 2: Random Forest predictions for Dengue cases (dispersed scatter, $R^2 = 0.83$).

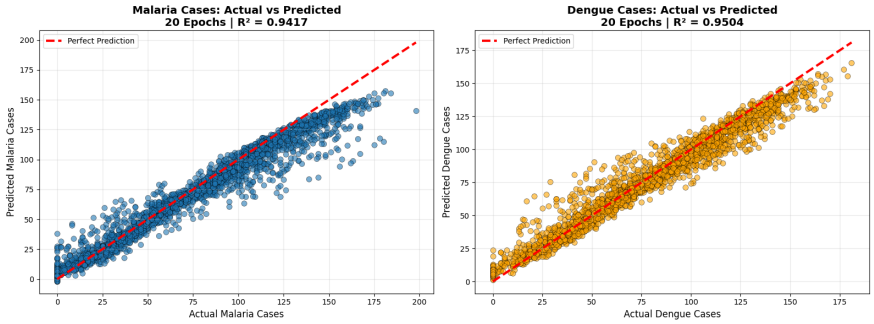

Figure 3: ANN predictions for Dengue cases (tight alignment around perfect prediction line, $R^2 = 0.95$).

7.3 Training Dynamics

The ANN showed rapid convergence. While 50 and 100 epoch runs were attempted, the model achieved optimal generalization at 20 epochs, preventing overfitting while maintaining the highest validation scores (R^2 Dengue: 0.9504).

8 Software Implementation: Web Application

To make the system accessible and interactive, a complete full-stack web application was developed.

- **Backend (Flask):** The Python Flask framework serves the trained ANN model and search algorithms as API endpoints. It handles requests for risk prediction and runs live BFS/A* simulations based on user-selected regions.
- **Frontend (React):** A responsive React.js interface allows users to:
  - **Predict:** Input values for any country to check climatic features and understand why a specific prediction was made.
  - **Simulate:** Select a "Patient Zero" country to visualize the BFS spread simulation on an interactive map.
  - **Assess Risk:** View risk assessment paths generated by the A* algorithm.

9 Conclusion & Future Work

9.1 Summary of Contributions

This project successfully delivered a robust infectious risk system:

- **Simulation:** Validated BFS and A* for modeling disease propagation paths.
- **Data-Centric Improvement:** Proved that temporal feature engineering reduced RMSE by ∼75% compared to raw data.
- **Predictive Power:** Achieved state-of-the-art accuracy ($R^2 \approx 0.95$) using Deep Neural Networks, specifically solving the challenge of Dengue prediction where previous models lagged.
- **Deployment:** The integration of a Flask backend and React UI transforms these complex models into an accessible application.

9.2 Limitations

Despite the system's success, the following limitations define the scope for future work:

1. **Granularity:** Currently, the model operates at the country level. This is a coarse abstraction; for more realistic spread simulations, the system needs to predict and simulate at the city or region level.
2. **Disease Scope:** The predictive model is currently limited to Malaria and Dengue due to the specific constraints of the dataset used. Expanding to novel viruses (e.g., COVID-19 variants) would require retraining with new datasets.
3. **Data Dependency:** The system relies on external APIs for connectivity and climate data. Any latency or downtime in these third-party services can impact real-time simulation capabilities.

9.3 Conclusion

By achieving high accuracy ($R^2 \approx 0.95$) and enabling visual spread simulation, this framework provides a solid foundation for future advancements in AI-driven public health monitoring.