CSE 311 ARTIFICIAL INTELLIGENCE

AquaSense

Rainfall Prediction in India using Artificial Neural Network (ANN)

Name: Rounak Vinod Ukey
Reg No: 2023BCS0152

ABSTRACT:

AquaSense predicts monthly rainfall across Indian regions using an Artificial Neural Network (ANN). It uses historical rainfall data from 1901–2015, obtained from the Indian Meteorological Department (IMD). The data is cleaned, processed, and transformed into a format suitable for machine learning. A feed-forward ANN model with two hidden layers ReLU activation is trained using Mean Squared Error (MSE) as the loss function and the Adam optimizer. The system predicts monthly rainfall and provides evaluation metrics such as Mean Absolute Error (MAE), MSE, and R^2 score. The results show that ANN can learn rainfall patterns across time and geography quite effectively.

The model is also integrated into a Streamlit web application for interactive visualization and predictions. This work demonstrates how AI can support agriculture, water management, and climate research by improving rainfall forecasting accuracy.

INTRODUCTION:

Background and Motivation

Rainfall prediction is crucial for India because it directly affects agriculture, irrigation planning, and disaster management. India's monsoon driven climate leads to major seasonal and regional variations in rainfall. A delay or change in the monsoon can lead to droughts or floods, affecting millions of people.

Using AI, especially neural networks, helps analyze large historical datasets and uncover hidden seasonal patterns that traditional models may miss.

Why This Problem Is Important

Rainfall directly impacts crop production, flood control, and water resource planning. Accurate monthly rainfall prediction can help reduce losses in agriculture. India's long-term rainfall data provides an opportunity to learn seasonal and regional trends over more than 100 years.

Challenges

- Missing values in long historical datasets.
- Capturing both temporal month to month and spatial region to region variations.
- Limited available features in table format.

AI Method Used

This project uses a Feed-forward Artificial Neural Network (ANN) because it is effective for regression problems and can learn non linear patterns. It predicts rainfall amounts based on features like subdivision, month, season, and previous-year rainfall.

Project Focus

The main focus is to design and train a compact ANN for monthly rainfall prediction, cleaning and preparing long term rainfall data, designing a simple but efficient ANN model, evaluating model accuracy using standard metrics, visualizing training and prediction results, creating an interactive web app using Streamlit.

Problem Statement and Objectives:

Develop an AI-based system to predict monthly rainfall in mm for Indian subdivisions using historical data (1901–2015). The system should preprocess the data, train an ANN model, evaluate its accuracy, and give graphical outputs for visualizing.

Objectives

- Preprocess and clean the rainfall dataset, handling missing values.
- Add important features like Month Number, Season, and Previous Year Rainfall.
- Train and test an ANN regression model.
- Evaluate model performance using MAE, MSE, and R^2.
- Give a graph and a web streamlit app for predictions.

Proposed Methodology:

Dataset Description

- **Source:** "Rainfall in India 1901–2015" dataset from Kaggle. https://www.kaggle.com/datasets/rajanand/rainfall-in-india
- **Contents:** Columns for YEAR, SUBDIVISION, and monthly rainfall January to December.
- **Size:** Covers 115 years of data across many subdivisions approx 40,000 to 50,000 records after processing.

Preprocessing Steps:

1. Load dataset using pandas.

2. Fill missing values using subdivision wise monthly averages.

3. Convert wide format (JAN–DEC columns) into long format (one row per month).

4. Add Month Number (1–12) and Season columns.

5. Create a Lag Feature, rainfall of the same month in the previous year.

6. Convert categorical features (Subdivision, Season) using one-hot encoding.

7. Split data into 80% training and 20% testing sets.

8. Standardize numerical features using StandardScaler.

Algorithm / Model Description (Artificial Neural Network)

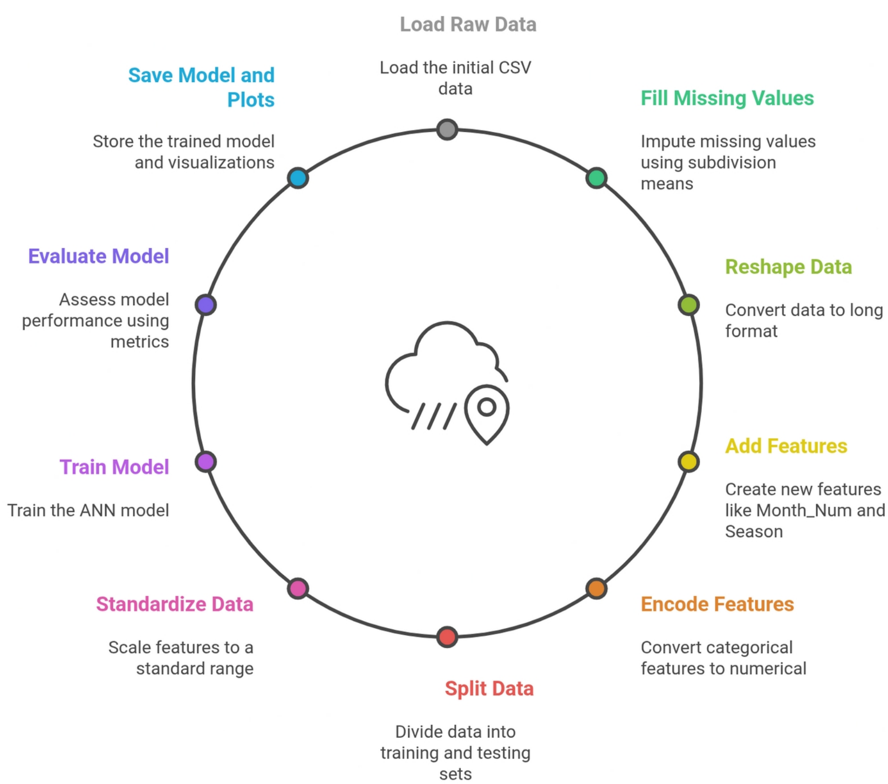
- **Model Type:** Feed-forward Neural Network
- **Architecture:**
    - Input layer
    - Hidden Layer 1: 128 neurons, ReLU activation
    - Hidden Layer 2: 64 neurons, ReLU activation
    - Output Layer: 1 neuron i.e. linear activation for rainfall prediction.

- **Loss Function:** Mean Squared Error (MSE)
- **Optimizer:** Adam
- **Epochs:** 100
- **Batch Size:** 32

Implementation Tools

- **Language:** Python 3.12
- **Libraries:** pandas, numpy, scikit-learn, TensorFlow/Keras, matplotlib, seaborn, Streamlit.

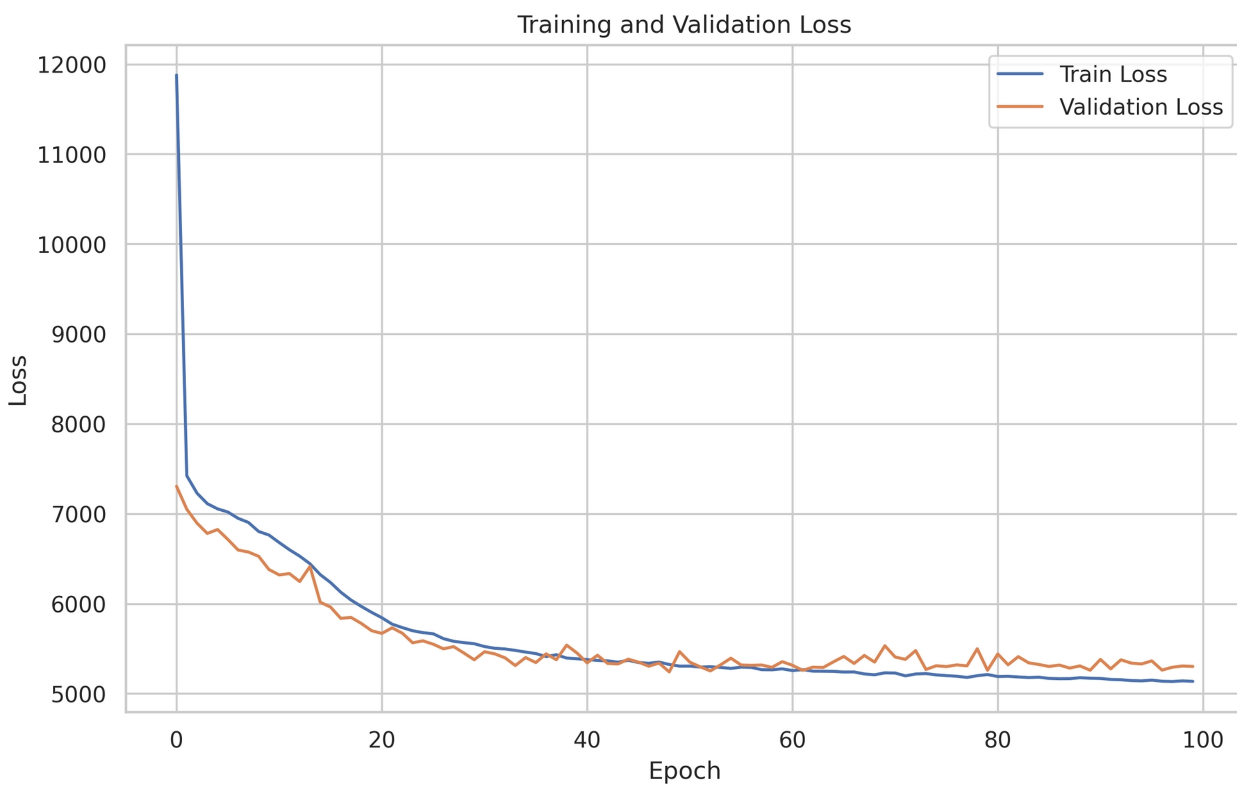
Workflow Diagram:

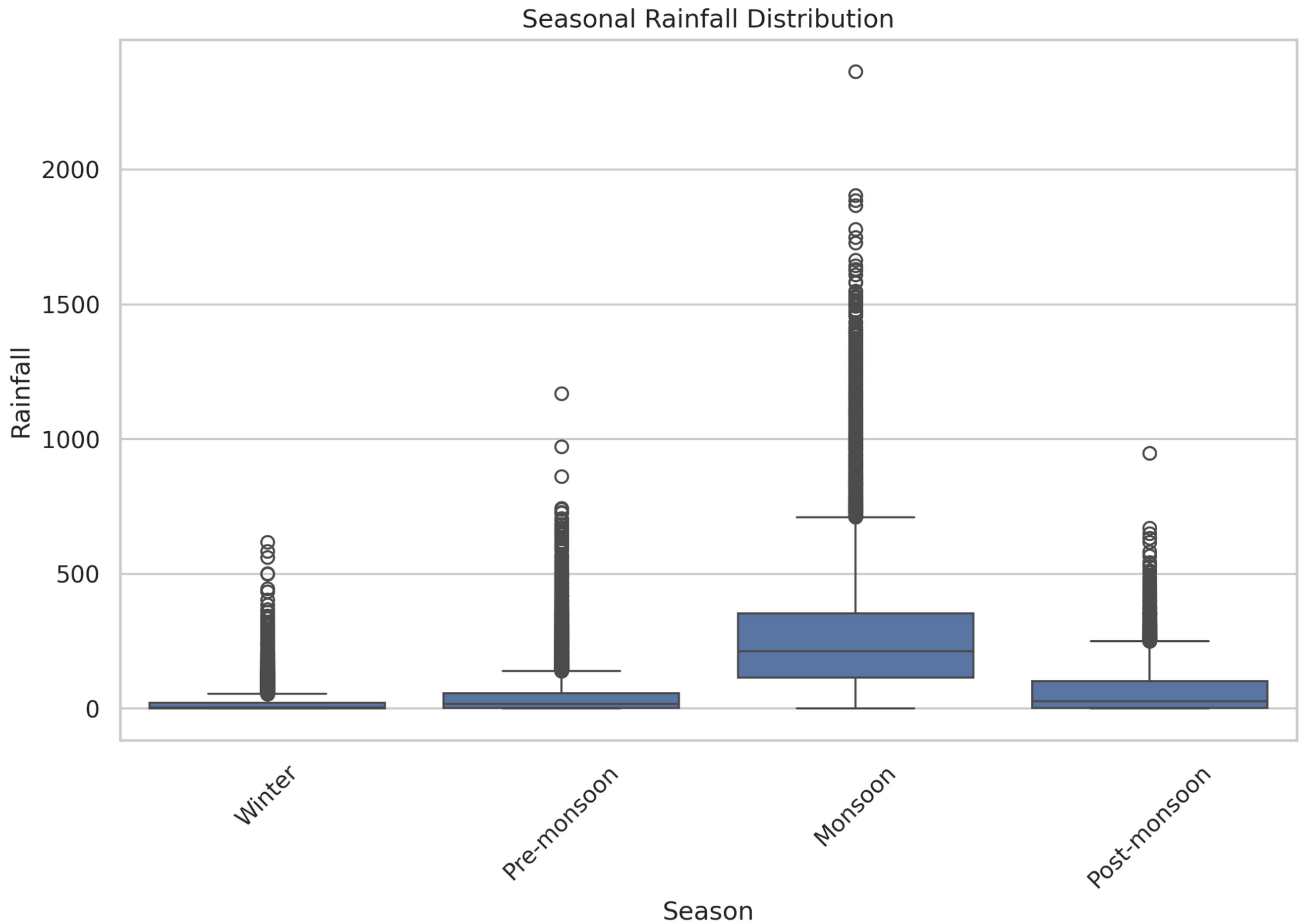


Experimental Setup and Results:

Training and Testing

- Training: 80% of data
- Testing: 20% of data
- Validation: Performed during training
- Optimizer: Adam
- Epochs: 100
- Batch size: 32





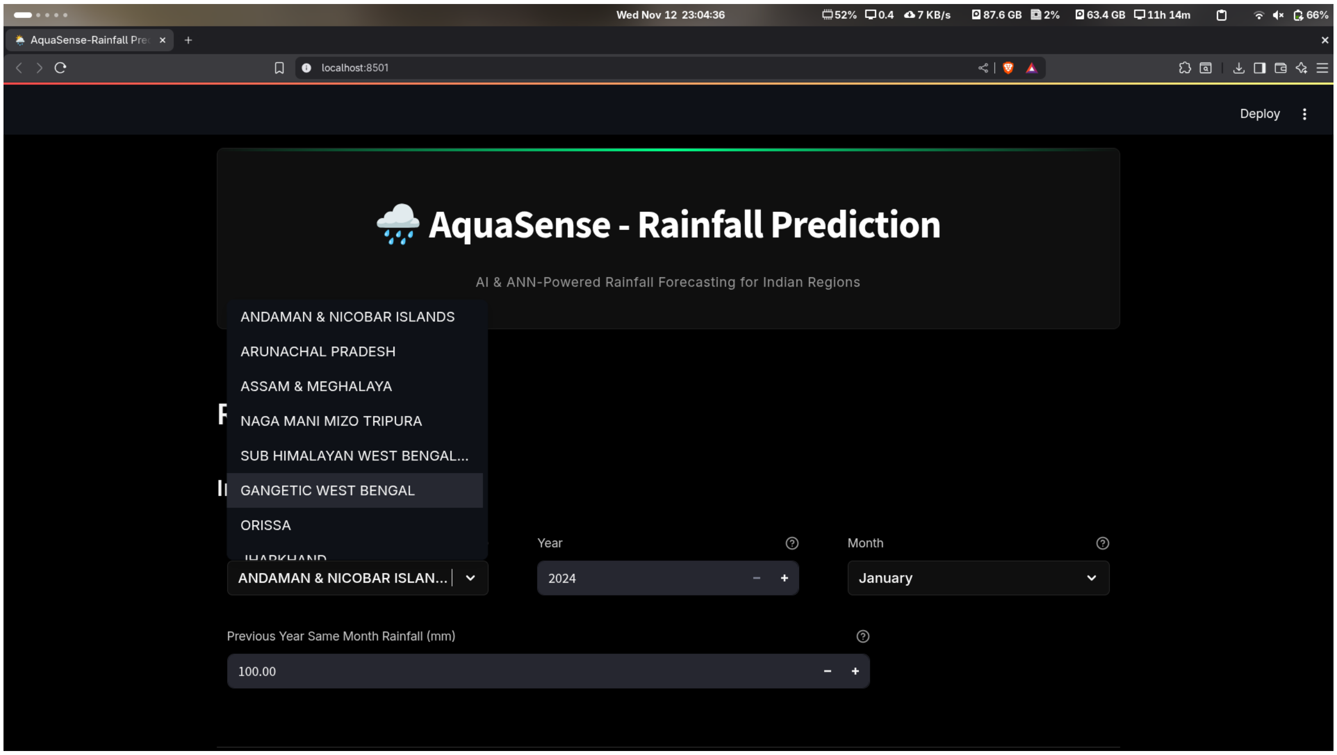
Training and Validation Loss

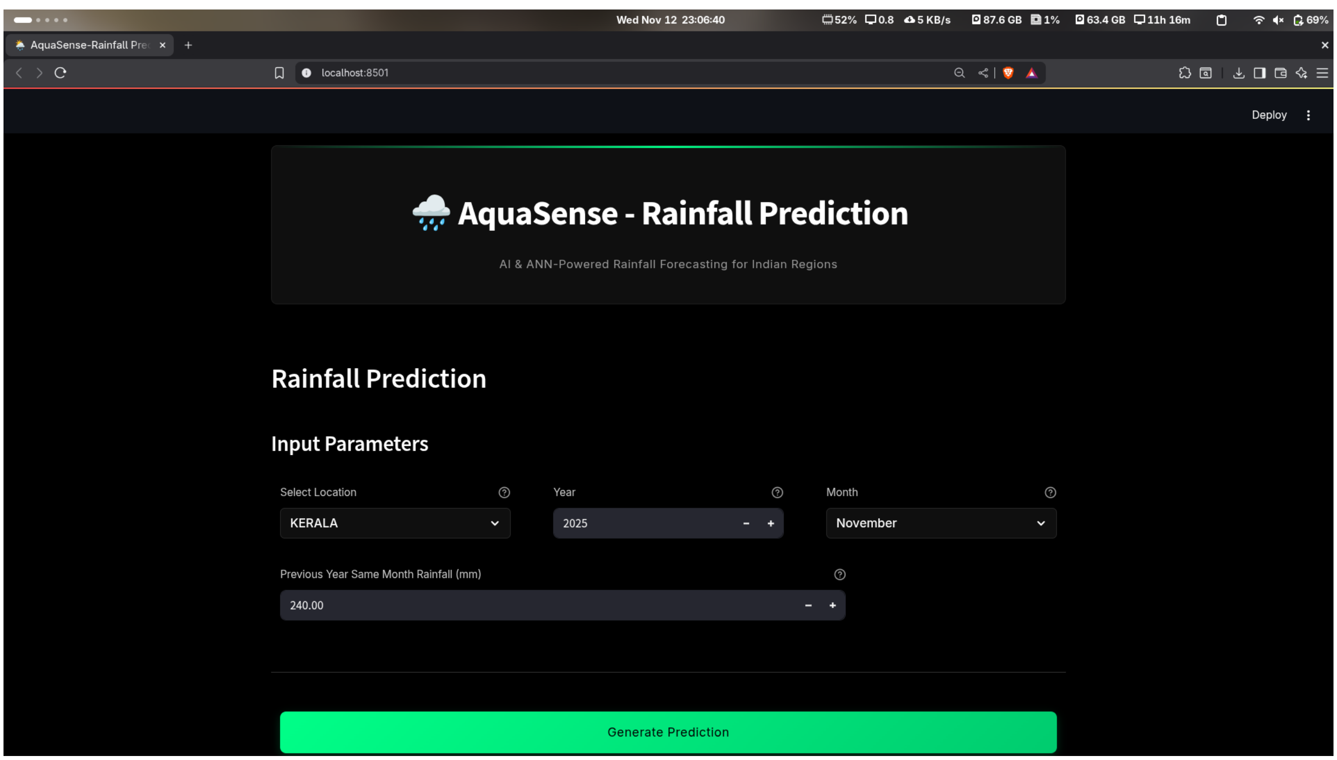
The training and validation loss both decreased rapidly during the first few epochs and then stabilized. This shows that the model learned effectively without over fitting.

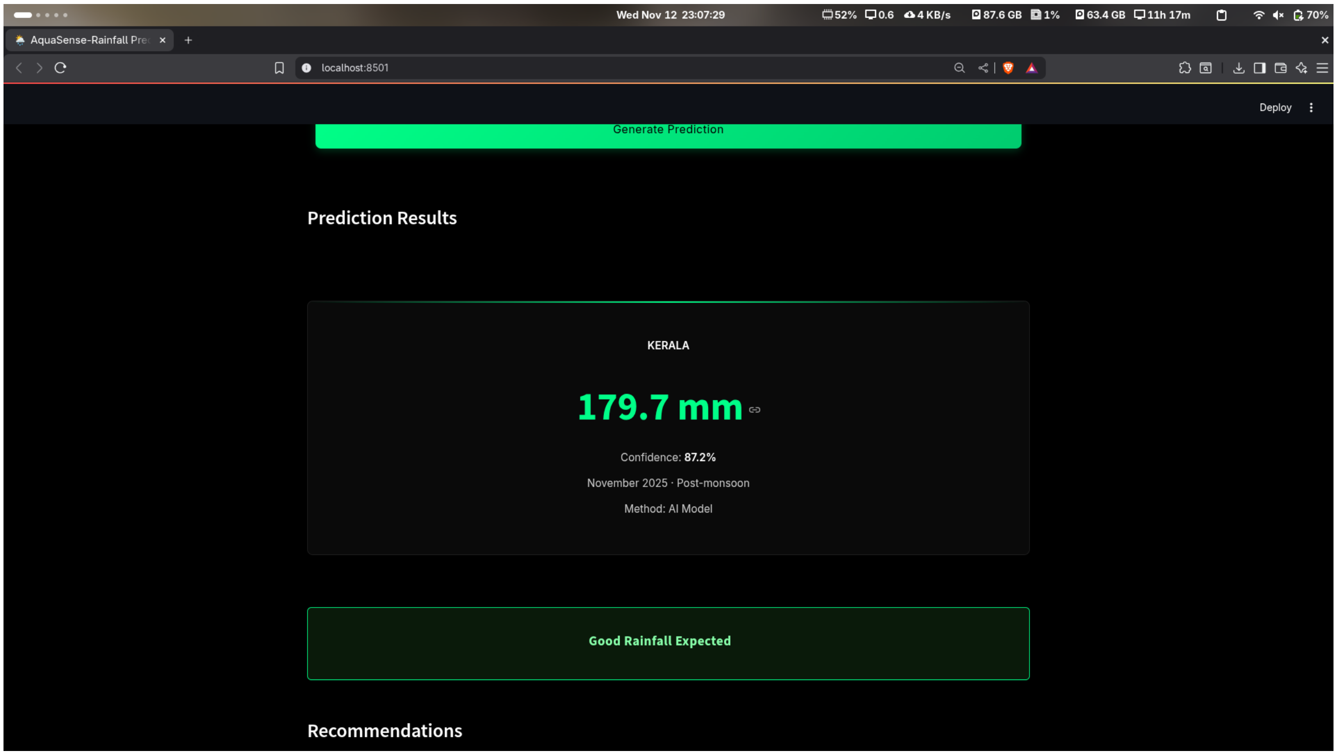
Seasonal Rainfall Distribution

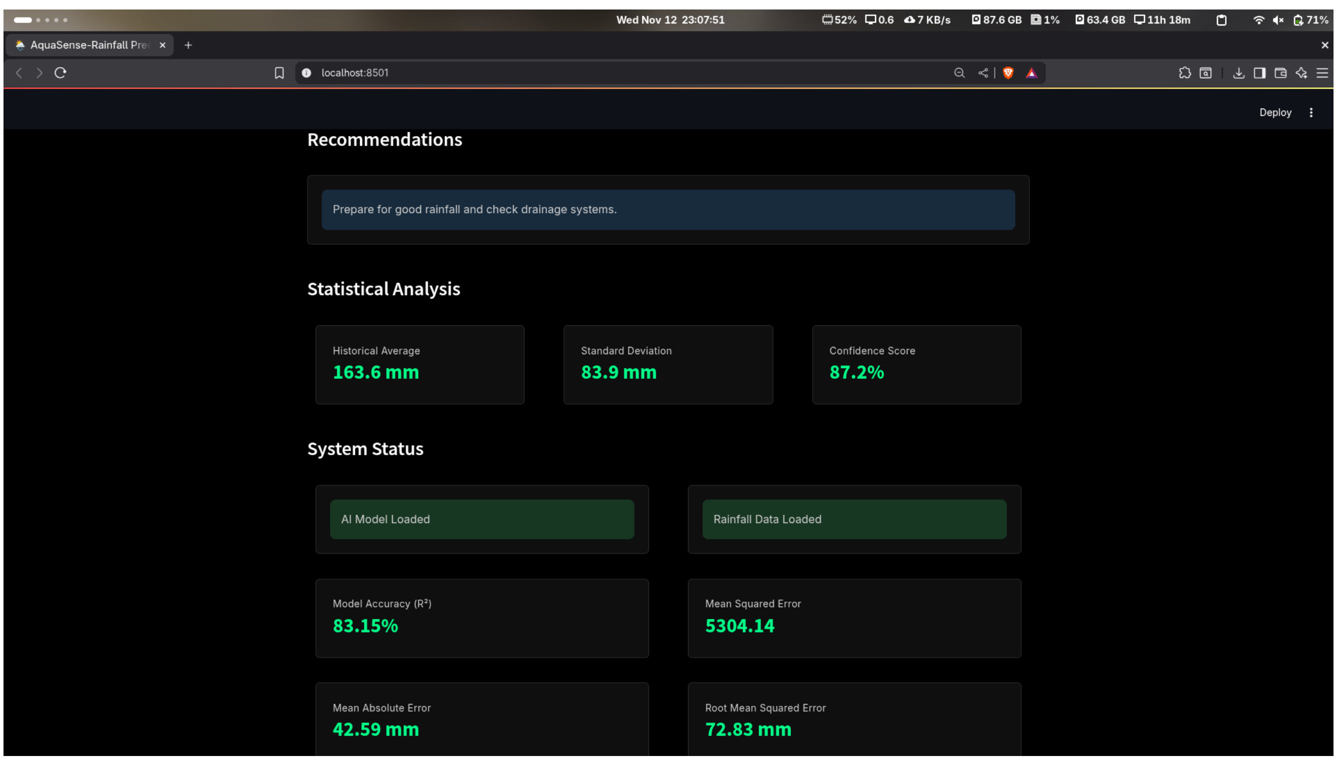
The Monsoon season shows the highest rainfall, while Winter has the lowest. This confirms that the data captures realistic seasonal variation patterns, and the model can learn from these differences.

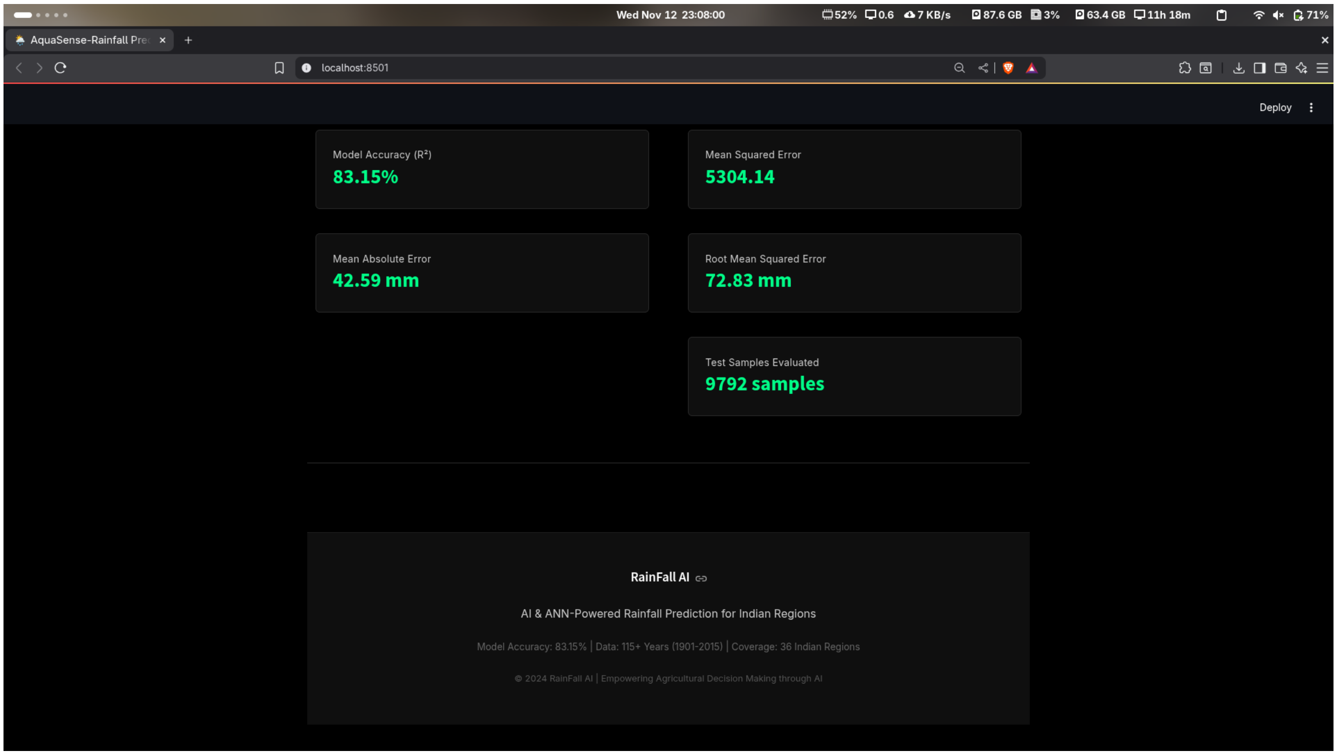
Website Demo Pictures

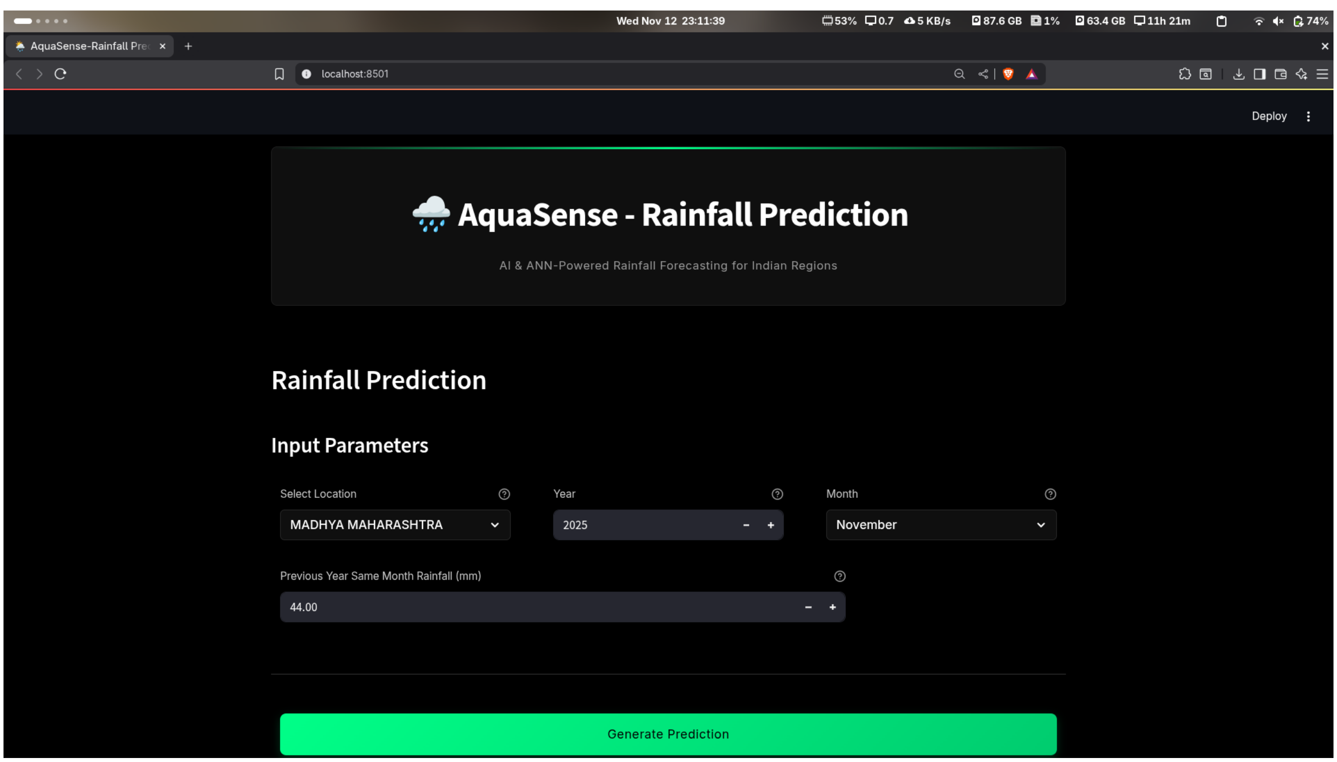


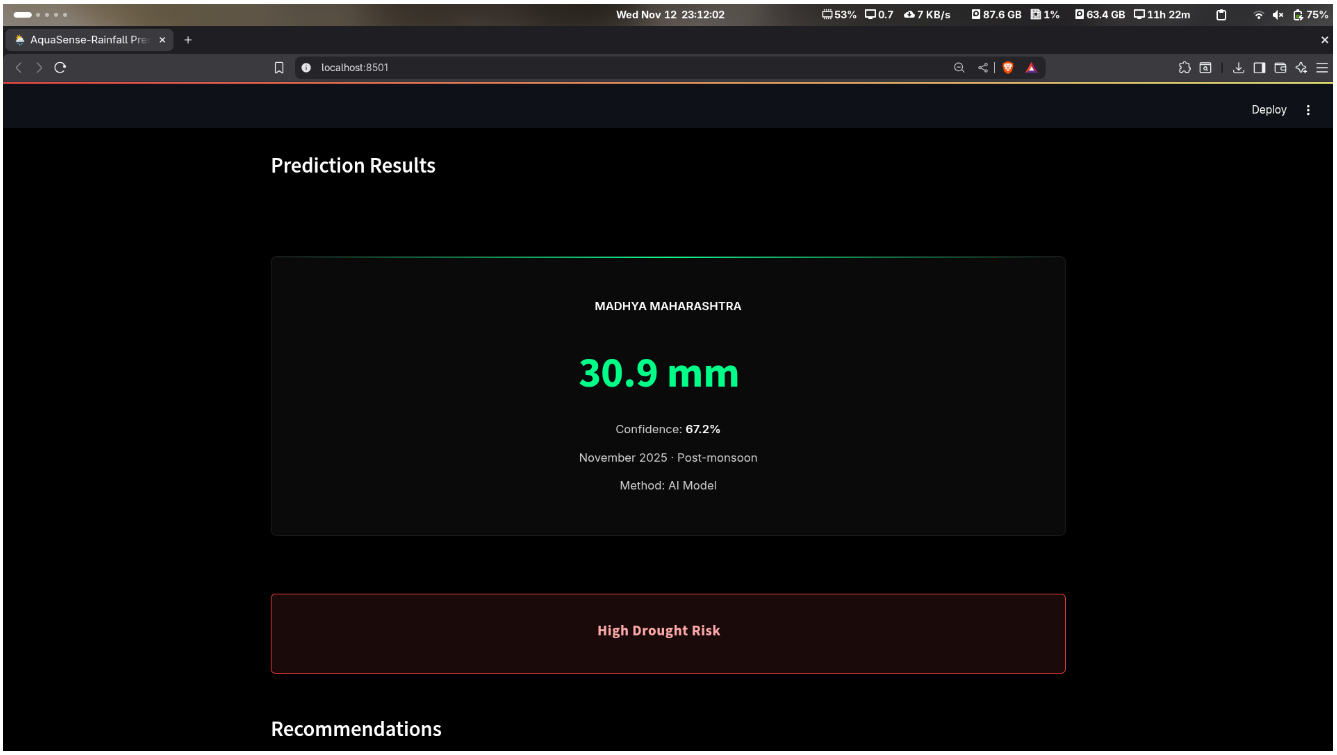


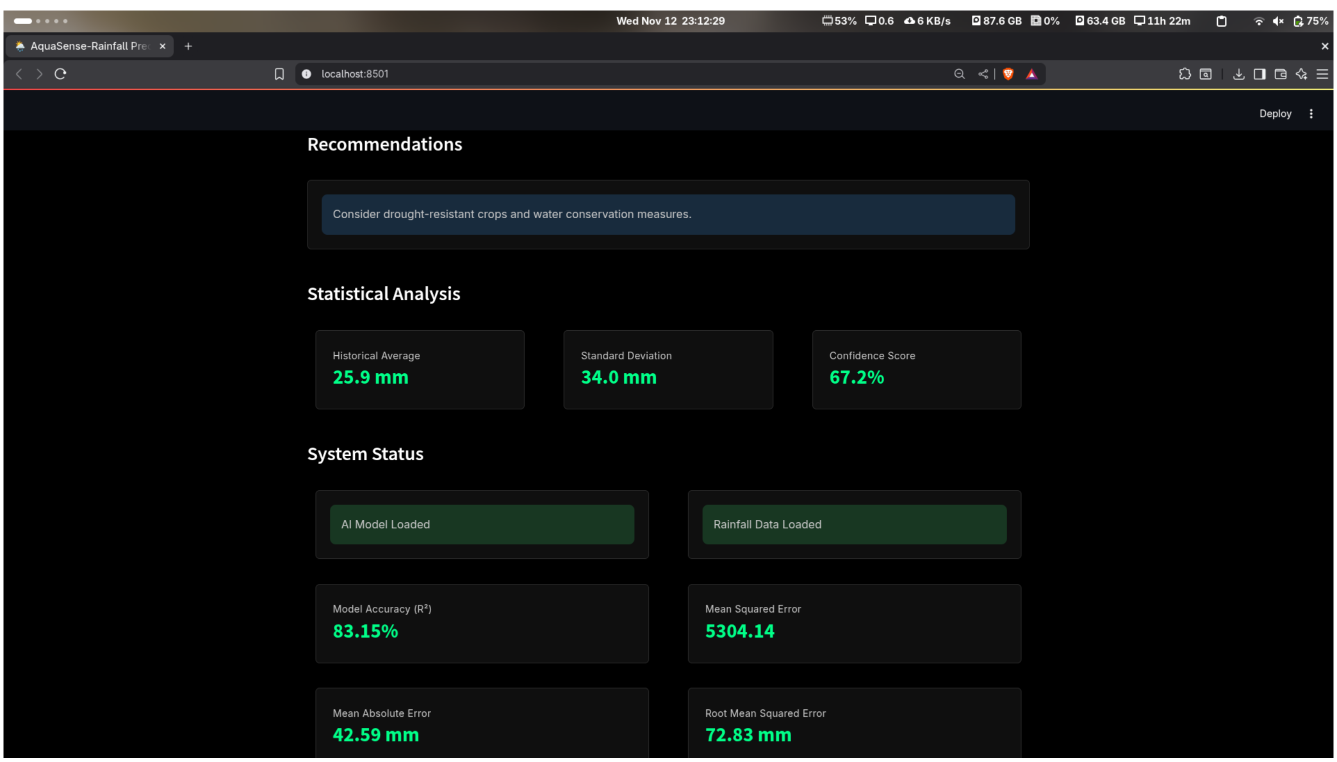

Wed Nov 12 23:07:29
AquaSense-Rainfall Pre
localhost:8501
Deploy
Generate Prediction
Prediction Results
KERALA
179.7 mm
Confidence: 87.2%
November 2025 · Post-monsoon
Method: AI Model
Good Rainfall Expected
Recommendations

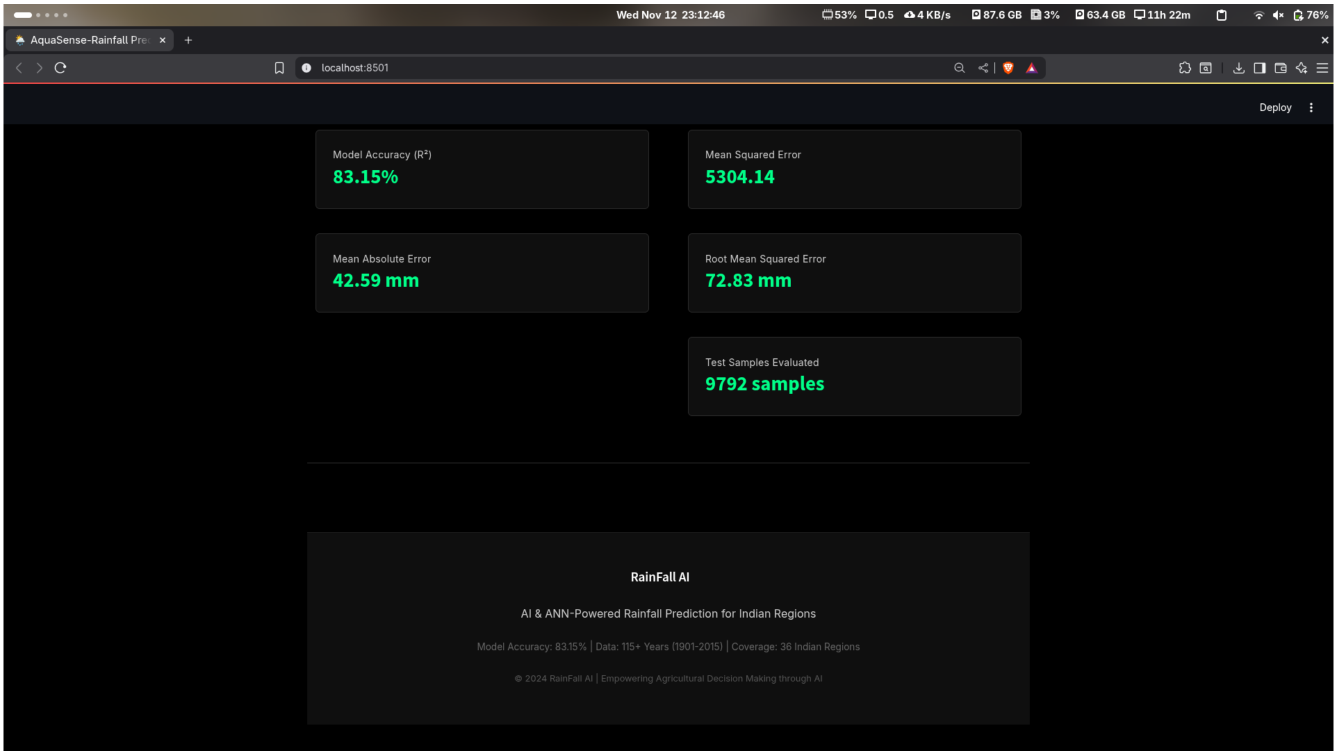

Wed Nov 12 23:07:51
AquaSense-Rainfall Pre
localhost:8501
Deploy
Recommendations
Prepare for good rainfall and check drainage systems.
Statistical Analysis
Historical Average
163.6 mm
Standard Deviation
83.9 mm
Confidence Score
87.2%
System Status
AI Model Loaded
Rainfall Data Loaded
Model Accuracy (R²)
83.15%
Mean Squared Error
5304.14
Mean Absolute Error
42.59 mm
Root Mean Squared Error
72.83 mm


Wed Nov 12 23:08:00
AquaSense-Rainfall Pre
localhost:8501
Deploy
Model Accuracy (R²)
83.15%
Mean Squared Error
5304.14
Mean Absolute Error
42.59 mm
Root Mean Squared Error
72.83 mm
Test Samples Evaluated
9792 samples
RainFall AI
AI & ANN-Powered Rainfall Prediction for Indian Regions
Model Accuracy: 83.15% | Data: 115+ Years (1901-2015) | Coverage: 36 Indian Regions
© 2024 RainFall AI | Empowering Agricultural Decision Making through AI


Wed Nov 12 23:11:39
AquaSense-Rainfall Pre
localhost:8501
Deploy
AquaSense - Rainfall Prediction
AI & ANN-Powered Rainfall Forecasting for Indian Regions
Rainfall Prediction
Input Parameters
Select Location
MADHYA MAHARASHTRA
Year
2025
Month
November
Previous Year Same Month Rainfall (mm)
44.00
Generate Prediction


Wed Nov 12 23:12:02
53% 0.7 7 KB/s 87.6 GB 1% 63.4 GB 11h 22m 75%
AquaSense-Rainfall Pre
localhost:8501
Deploy
Prediction Results
MADHYA MAHARASHTRA
30.9 mm
Confidence: 67.2%
November 2025 · Post-monsoon
Method: AI Model
High Drought Risk
Recommendations


Wed Nov 12 23:12:29
53% 0.6 6 KB/s 87.6 GB 0% 63.4 GB 11h 22m 75%
AquaSense-Rainfall Pre
localhost:8501
Deploy
Recommendations
Consider drought-resistant crops and water conservation measures.
Statistical Analysis
Historical Average
25.9 mm
Standard Deviation
34.0 mm
Confidence Score
67.2%
System Status
AI Model Loaded
Rainfall Data Loaded
Model Accuracy (R²)
83.15%
Mean Squared Error
5304.14
Mean Absolute Error
42.59 mm
Root Mean Squared Error
72.83 mm



Performance Metrics

MAE : 42.59 mm, This is low MAE means it shows good prediction accuracy.

MSE : 5304.14

R^2 : 83.15%, means model understands and explain 83.15% of rainfall variation.

Discussion and Analysis

- The ANN model performed well and successfully captured rainfall trends across regions and seasons.

- The training curve shows good convergence and minimal over fitting.
- The results suggest ANN is a strong baseline for rainfall prediction tasks.

Challenges faced:

- Handling missing or inconsistent data.
- High variance between regions caused noisy predictions.
- Limited input features like no temperature, humidity, or pressure data used.

Possible Improvements:

- Use LSTM or GRU models to handle sequential dependencies.
- Add climate indices as features.
- Fine tune ANN hyperparmeters for better accuracy.

Applications and Future Scope

Real World Applications

- Agriculture: Helps farmers plan crops and irrigation schedules.
- Water Resource Management: Useful for reservoir and dam operations.

- Disaster Preparedness: Early warning for droughts and floods.
- Climate Analysis: Study long-term changes in rainfall patterns.

Future Enhancements

- Use Recurrent Neural Networks LSTM or GRU for time-series forecasting.
- Add satellite or climate variables for richer inputs.
- Deploy as a web or mobile app for public access.
- Include confidence intervals in predictions to show uncertainty.

Conclusion

The AquaSense project demonstrates how Artificial Neural Networks can be used to predict rainfall using over a century of historical data. The model performs well with a simple ANN structure and serves as a strong starting point for more advanced forecasting systems. By improving feature engineering and adopting sequential models, accuracy can be enhanced further.
This project highlights how AI based rainfall prediction can be a valuable tool for agriculture, climate research, and disaster management in India.