Crop Yield Recommendation System Using Multi-Source Data Integration

This project focuses on building a comprehensive crop recommendation system by integrating and analyzing diverse datasets to provide actionable insights for farmers and agricultural stakeholders. The system leverages advanced data preprocessing, distributed computing, and machine learning techniques to suggest the most suitable crops based on regional and seasonal parameters.

Objective

To develop a data-driven solution that recommends optimal crop choices for specific regions and seasons, enhancing agricultural productivity and resource efficiency.

Data Collection and Sources

- **Primary Data**: Seven datasets collected from Kaggle, university websites, and agricultural portals, including:
    - **Crop production data** (2000–2014): Crop yields, sown area, and production across six agricultural seasons.
    - **Temperature data** (1995–2020): Daily averages for key Indian cities.
    - **Rainfall data** (1901–2015): Monthly averages across subdivisions.
- **Secondary Data**: Soil type, seed variety, humidity, and wind speed, manually gathered from trusted websites like agricoop and weather portals.

Data Preprocessing

- Uploaded data to Python's Colab environment for cleaning and transformation.
- Used **linear interpolation** to handle missing values and applied **IQR** and **Z-score methods** to remove outliers.
- Employed Python libraries like Pandas, NumPy, and SciPy for data manipulation and computation.

MapReduce Implementation

- **Temperature Data**: Processed monthly averages for key regions using map and reduce functions.
- **Rainfall Data**: Calculated cumulative seasonal rainfall.
- **Crop Production Data**: Derived "produce per area" metrics for crops by season, region, and year.

Recommendation Algorithm

The recommendation engine combines cleaned data to suggest:

1. Top three crops for a user-specified region, season, and month.
2. Top three crops with the best annual yield.

3. Recommended seed types and soil varieties for optimal crop yield.
4. Historical weather conditions (temperature, rainfall, wind speed, and humidity) for successful crop growth.

Technology and Tools

- **Data Processing**: Python libraries (Pandas, NumPy, SciPy), MapReduce framework.
- **Recommendation Engine**: A Python-based algorithm integrating multiple datasets for tailored outputs.
- **Interface**: User-friendly Flask web application for seamless interaction with the system.

Outcome

The system delivers region-specific and season-specific crop recommendations backed by historical data. It empowers farmers to make informed decisions, optimizing agricultural practices and ensuring sustainability.

Future Scope

The project can be extended to include real-time weather data, pest prediction models, and dynamic pricing analytics for agricultural markets. Integration with mobile platforms could further enhance accessibility.
