A Crop Recommendation System Project using Machine Learning

-Chaithranjali S

Date:11/08/2024

1. Abstract

This project focuses on developing a machine learning-based system for crop classification and recommendation, aimed at optimizing agricultural decision-making and maximizing financial returns for farmers. The model integrates domain-specific knowledge with advanced machine learning techniques to classify crops based on environmental factors such as soil type, climate conditions, and geographic location. In addition to classification, the system incorporates a financial modeling component to evaluate the economic viability of different crops, taking into account variables like production costs, market prices, and potential risks.

The resulting system is designed to be accessible via a user-friendly interface, enabling farmers and agribusiness stakeholders to input their specific conditions and receive tailored crop recommendations. This approach ensures that recommendations are not only agriculturally sound but also financially advantageous, thereby supporting sustainable and profitable farming practices.

2. Problem Statement

In agricultural practices, the selection of appropriate crops plays a vital role in maximizing yield and profitability. However, farmers often face challenges in determining the most suitable crops to cultivate based on factors such as soil type, climate conditions, market demand, and their own resources.

The objective of this project is to develop an intelligent Crop Recommendation System (CRS) that assists farmers in making informed decisions regarding crop selection. The CRS will analyse various environmental and socio-economic factors to suggest the most suitable crops for cultivation in a given region or farm.

3. Market/Customer/Business Need Assessment

The agricultural sector is witnessing a shift towards data-driven decision-making and precision agriculture techniques. Farmers are increasingly seeking technological solutions to optimize crop selection, minimize risks, and improve productivity.

The market/customer/business need assessment underscores the significant demand and business opportunities for Crop Recommendation Machine Learning projects in the agricultural sector. Addressing the identified market needs and customer requirements through the development and deployment of effective ML-driven crop recommendation systems can create value for farmers, agribusinesses, and other stakeholders while advancing sustainable agriculture and food security objectives**.**

4. Target Specifications and Characterization

The crop recommendation system is useful for farmers who want to maximize their crop yield and minimize crop failure by selecting the best crop based on the conditions of their land. The system uses machine learning algorithms such as decision trees, random forests, and support vector machines to make predictions and recommend the best crop for the given input data.

The project showcases the power of machine learning and how it can be used to solve real-world problems in the agricultural industry. The web application created using Flask allows farmers to easily input their data and get crop recommendations quickly and efficiently.

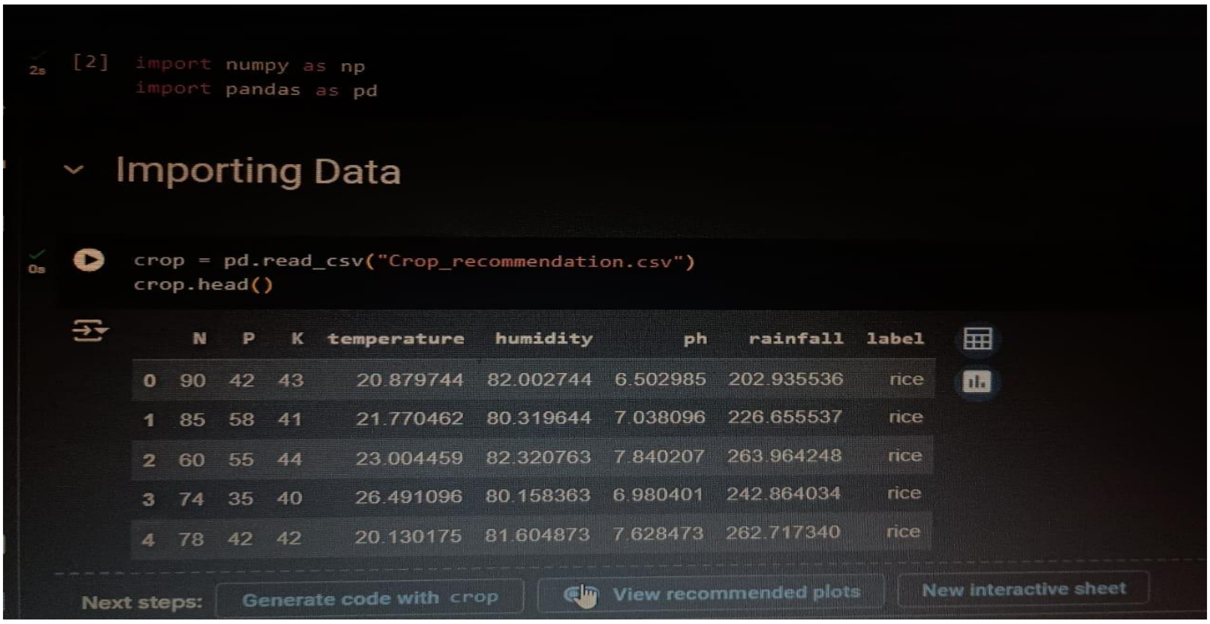
5. External Search

I use the dataset for this project Dataset can be found here:

https://www.kaggle.com/datasets/atharvaingle/crop-recommendation-dataset

Precision agriculture is in trend nowadays. It helps the farmers to get informed decision about the farming strategy. Here, I present you a dataset which would allow the users to build a predictive model to recommend the most suitable crops to grow in a particular farm based on various parameters.

6. CODE IMPLEMENTATION (SMALL SCALE):

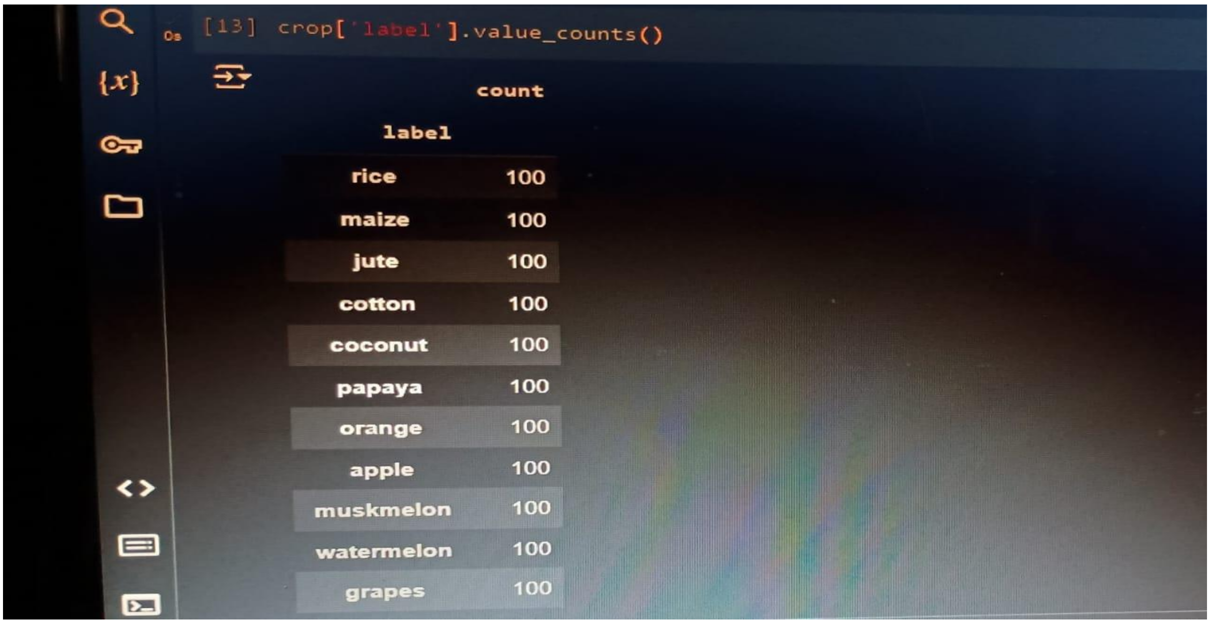


```
[2] import numpy as np
    import pandas as pd
```

Importing Data

```
crop = pd.read_csv("Crop_recommendation.csv")
crop.head()
```

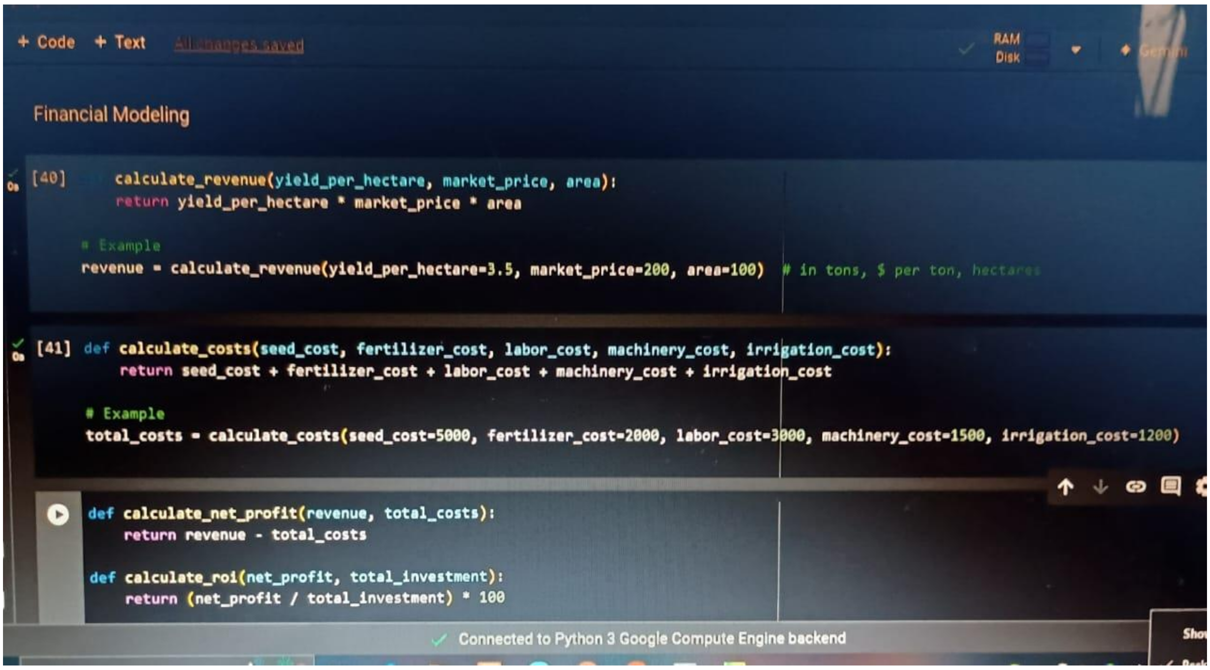
```
    N   P   K  temperature   humidity        ph    rainfall  label
0  90  42  43    20.879744  82.002744  6.502985  202.935536   rice
1  85  58  41    21.770462  80.319644  7.038096  226.655537   rice
2  60  55  44    23.004459  82.320763  7.840207  263.964248   rice
3  74  35  40    26.491096  80.158363  6.980401  242.864034   rice
4  78  42  42    20.130175  81.604873  7.628473  262.717340   rice
```



```
[13] crop['label'].value_counts()
```

```
              count
      label
       rice     100
      maize     100
       jute     100
     cotton     100
    coconut     100
     papaya     100
     orange     100
      apple     100
  muskmelon     100
 watermelon     100
     grapes     100
```

FINANCIAL MODELLING



Financial Modeling

```
[40]    calculate_revenue(yield_per_hectare, market_price, area):
        return yield_per_hectare * market_price * area

     # Example
     revenue = calculate_revenue(yield_per_hectare=3.5, market_price=200, area=100)  # in tons, $ per ton, hectares
```

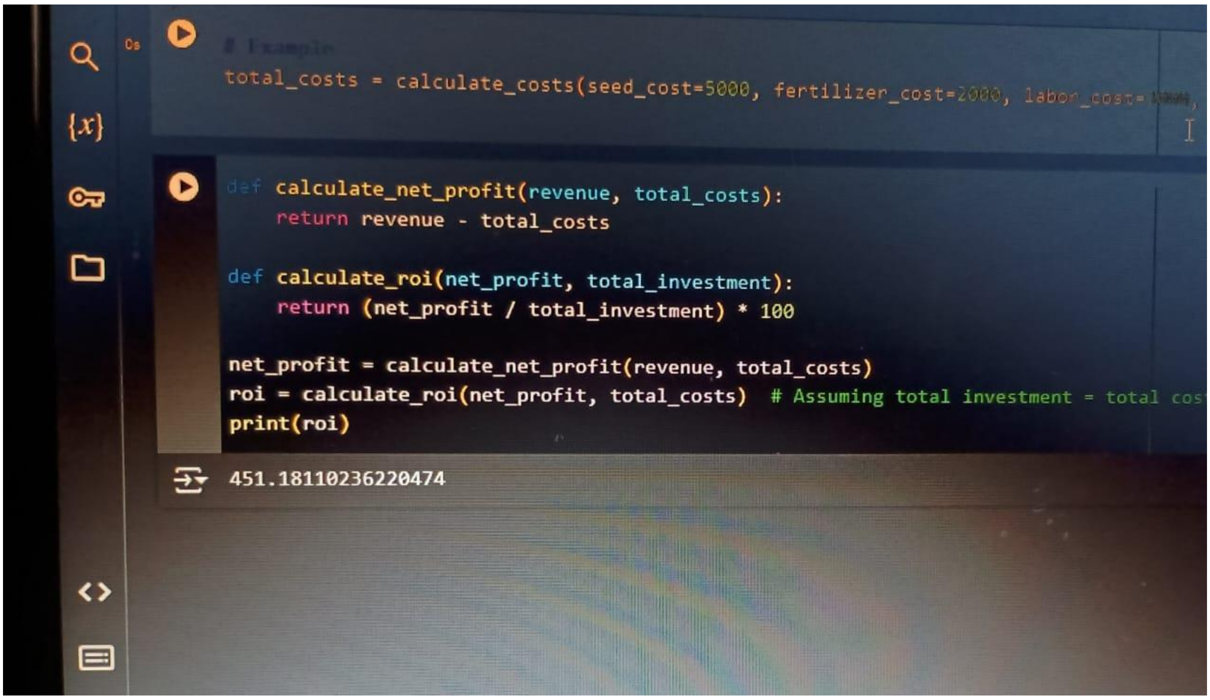
```
[41] def calculate_costs(seed_cost, fertilizer_cost, labor_cost, machinery_cost, irrigation_cost):
         return seed_cost + fertilizer_cost + labor_cost + machinery_cost + irrigation_cost

     # Example
     total_costs = calculate_costs(seed_cost=5000, fertilizer_cost=2000, labor_cost=3000, machinery_cost=1500, irrigation_cost=1200)
```

```
def calculate_net_profit(revenue, total_costs):
    return revenue - total_costs

def calculate_roi(net_profit, total_investment):
    return (net_profit / total_investment) * 100
```

Connected to Python 3 Google Compute Engine backend



```
# Example
total_costs = calculate_costs(seed_cost=5000, fertilizer_cost=2000, labor_cost=3000,
```

```
def calculate_net_profit(revenue, total_costs):
    return revenue - total_costs

def calculate_roi(net_profit, total_investment):
    return (net_profit / total_investment) * 100

net_profit = calculate_net_profit(revenue, total_costs)
roi = calculate_roi(net_profit, total_costs)  # Assuming total investment = total cos
print(roi)
```

```
451.18110236220474
```

7. Bench marking alternate products

Feature	Crop Recommendation without AI	Crop Recommendation with AI
Approach	Rule-based, heuristic	Data-driven, machine learning
Data Analysis	Manual	Automated
Scalability	Limited	High
Accuracy	Moderate	High
Personalization	Limited	High
Adaptability	Limited	High
Dependency on Expertise	High	Moderate
Real-time Decision Support	Limited	Available
Feedback Integration	Limited	Possible
Learning and Improvement	Limited	Continuous
Complexity Handling	Limited	High
Cost Efficiency	Moderate	Variable
User Interface	Manual	Automated/Interactive
Example Systems	Traditional agricultural advisory services	AI-driven agricultural decision support systems

8. Applicable Patents

The project showcases the power of machine learning and how it can be used to solve real-world problems in the agricultural industry. The web application created using Flask allows farmers to easily input their data and get crop recommendations quickly and efficiently.

Flask provides configuration and conventions, with sensible defaults, to get started. This section of the documentation explains the different parts of the Flask framework and how they can be used, customized, and extended. Beyond Flask itself, look for community-maintained extensions to add even more functionality.

Dataset: Precision agriculture is in trend nowadays. It helps the farmers to get informed decision about the farming strategy. Here, I present you a dataset which would allow the users to build a predictive model to recommend the most suitable crops to grow in a particular farm based on various parameters.

9. Applicable Regulations

Data Encryption: Implement encryption and other security measures to protect sensitive data collected or processed by the system.

Access Controls: Restrict access to data and ensure that only authorized individuals have permission to view or modify data.

10. Applicable Constraints

- Data Availability and Quality: Availability of comprehensive and high-quality datasets on soil characteristics, climate conditions, crop yields, and market trends may be limited**.**
- Data Quality: Incomplete, inaccurate, or outdated data can impact the effectiveness and reliability of machine learning models.
- User Acceptance: User acceptance and adoption of the system may be influenced by factors such as usability, accessibility, language barriers, and trust in the recommendations provided.
- Technical Proficiency: End-users' technical proficiency and familiarity with technology may vary, requiring the system to be intuitive and user-friendly.

11. Business Model

- Subscription-based Service: Offer a subscription-based model where farmers or agricultural businesses pay a recurring fee to access the crop recommendation system. Different subscription tiers can offer varying levels of features, support, and customization options.
- Pay-per-Use Model: Implement a pay-per-use or pay-per-recommendation model where users are charged based on the number of recommendations generated or the

volume of data processed. This model can be particularly suitable for users with occasional or seasonal needs.

- Premium Features or Insight: Offer premium features or advanced insights as add-on services for users willing to pay extra. This could include access to advanced analytics, personalized recommendations, or integration with proprietary datasets or algorithms.

12. Concept Generation

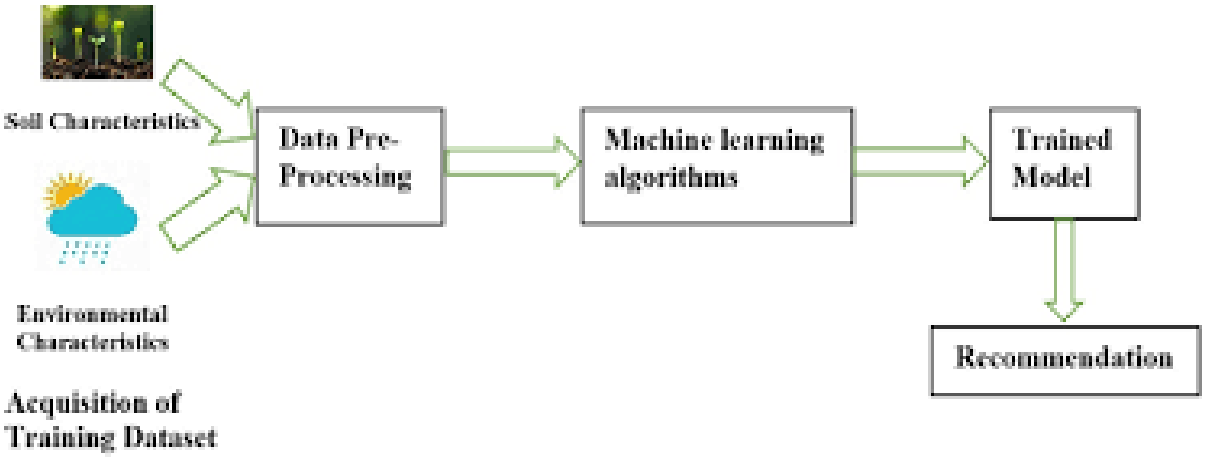
To develop a crop recommendation systems using machine learning has the potential to improve the productivity and sustainability of agriculture. By helping farmers to choose.

13. Concept Development

Python: Programming language used for model development, data preprocessing, and web application development. Scikit-learn: Machine learning library used for model training, evaluation, and prediction. Pandas: Data manipulation library used for data preprocessing and analysis. NumPy: Library for numerical computing used for handling arrays and mathematical operations. Flask: Web framework used for building the user interface and handling HTTP requests. HTML/CSS: Markup and styling languages used for designing the web interface. JavaScript: Scripting language used for client-side interactions and enhancing the user interface.

14. Final Product Prototype

Crop analysis and prediction is a rapidly growing field that plays a vital role in optimizing agricultural practices. Crop recommendation plays a pivotal role in agriculture, empowering farmers to make informed decisions about the most suitable crops for their specific land and climate conditions. Traditionally, this process heavily relied on expert knowledge, which proved time-consuming and labor-intensive. Moreover, considering the projected global population of 9.7 billion by 2050, the need to produce more food sustainably becomes imperative. Machine learning techniques can play a crucial role in effectively automating crop recommendations to enable farmers to optimize their yield from the land while simultaneously maintaining soil fertility and replenishing essential nutrients. This project analyses the performance of crop recommendation across seven distinct machine-learning algorithms. The proposed system leverages various features, including soil composition and climate data, to accurately predict the most suitable crops for specific locations. This system has the potential to revolutionize crop recommendation, benefiting farmers of all scales by enhancing crop yields, sustainability, and overall profitability. Through extensive evaluation of a comprehensive historical data set, we have achieved near-perfect accuracy by training and testing models the machine learning algorithms with various configurations.


Soil Characteristics
Environmental Characteristics
Acquisition of Training Dataset
Data Pre-Processing
Machine learning algorithms
Trained Model
Recommendation

14.1 Schematic Diagram

15. Product detail

15.1 How does it work?

The Crop Recommendation System is a machine learning-based application that provides recommendations for suitable crops based on various environmental and soil conditions. It aims to assist farmers and agricultural professionals in making informed decisions about crop selection, optimizing yields, and maximizing profitability.

The system takes into account several factors such as soil type, climate, rainfall, temperature, humidity, and pH levels to determine the most suitable crops for a given region. By analyzing historical data and using predictive models, the system provides personalized recommendations tailored to the specific conditions of a farm or agricultural area.

13.2 Data Sources

The models are trained on historical data and evaluated using appropriate performance metrics to ensure accuracy and reliability.

13.3Algorithms, frameworks, software etc. needed

Python: Programming language used for model development, data preprocessing, and web application development. Scikit-learn: Machine learning library used for model training, evaluation, and prediction. Pandas: Data manipulation library used for data preprocessing and analysis. NumPy: Library for numerical computing used for handling arrays and mathematical operations. Flask: Web framework used for building the user interface and handling HTTP requests. HTML/CSS: Markup and styling languages used for designing the web interface. JavaScript: Scripting language used for client-side interactions and enhancing the user interface.

13.4 Team required to develop.

1. Machine learning engineering
2. Business analyst
3. Software developer

GITHUB LINK: https://github.com/Chaithranjali/CROP-RECOMENDATION-FEYNNLABS-FINAL-PROJECT/upload/main