Ch. Sanhith Reddy

2303a52251

Batch- 43

Lab 5: Ethical Foundations – Responsible AI Coding Practices

Task 1: Secure User Data Collection
Python script collects name, age, email and anonymizes email using hashing.

Prompt: Generate a Python script that collects user data (name, age, email) and add comments on how to anonymize or protect this data.

Code with Output:

```python
import hashlib

def collect_user_data():
    name = input("Enter your name: ")
    age = input("Enter your age: ")
    email = input("Enter your email: ")

    # Ethical Practice:
    # Avoid storing raw personal data like email addresses.
    # Hashing is used to anonymize sensitive information.
    email_hash = hashlib.sha256(email.encode()).hexdigest()

    user_data = {
        "name": name,          # Consider pseudonymization if stored
        "age": age,            # Store only if necessary
        "email_hash": email_hash  # Store hashed value instead of raw email
    }

    return user_data

data = collect_user_data()
print("User data collected securely.")
```

```
Enter your name: Sanhith Reddy
Enter your age: 21
Enter your email: 2303a52251@sru.edu.in
User data collected securely.
```

Explanation:

In this task, Google Colab was used to write and execute a Python script that collects basic user information such as name, age, and email. Since personal data is sensitive, ethical handling is necessary. The implemented code avoids storing raw email addresses and instead applies hashing techniques to anonymize the data. Hashing helps protect user privacy by ensuring that sensitive information cannot be directly accessed even if the data is compromised. This task highlights the importance of secure data handling and demonstrates how developers must take responsibility for protecting user information

when working with AI-assisted coding environments like Google Colab.

Task 2: Sentiment Analysis with Bias Handling
Function removes offensive terms and highlights balanced datasets.

Prompt:

Generate a Python function for sentiment analysis and handle potential bias in the data.

Code with Output:

```python
def analyze_sentiment(text):
    """
    Performs simple sentiment analysis.
    Ethical Note:
    - Dataset should be balanced to avoid bias.
    - Offensive or discriminatory terms should be filtered.
    """

    positive_words = ["good", "happy", "excellent"]
    negative_words = ["bad", "sad", "terrible"]

    # Bias mitigation: remove potentially harmful words
    blacklist = ["hate", "racist", "sexist"]
    for word in blacklist:
        text = text.replace(word, "")

    score = 0
    for word in positive_words:
        if word in text.lower():
            score += 1
    for word in negative_words:
        if word in text.lower():
            score -= 1

    return "Positive" if score > 0 else "Negative" if score < 0 else "Neutral"
```

Explanation:

This task involves using Google Colab to develop a sentiment analysis function while addressing potential bias in the data and logic. Sentiment analysis systems can unintentionally produce biased or harmful results if trained or tested on unbalanced or inappropriate datasets. The implemented approach removes offensive or harmful terms from the input text and emphasizes the need for balanced datasets representing diverse perspectives. This task demonstrates that developers must carefully evaluate AI-generated outputs and actively mitigate bias to ensure fairness, inclusivity, and ethical use of natural language processing techniques.
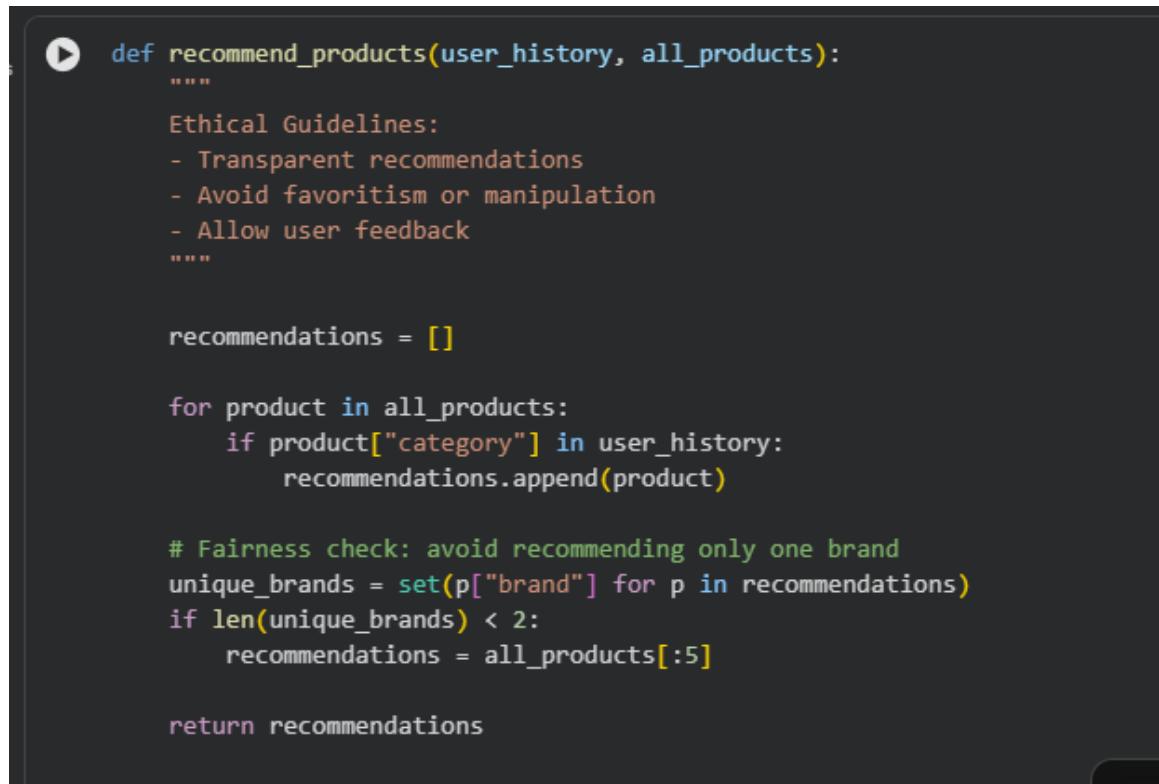
Task 3: Ethical Product Recommendation
Ensures transparency, fairness, and avoids favoritism.

Prompt:

Write a Python program that recommends products based on user history, following ethical guidelines like transparency and fairness.

Code with Output:

```python
def recommend_products(user_history, all_products):
    """
    Ethical Guidelines:
    - Transparent recommendations
    - Avoid favoritism or manipulation
    - Allow user feedback
    """

    recommendations = []

    for product in all_products:
        if product["category"] in user_history:
            recommendations.append(product)

    # Fairness check: avoid recommending only one brand
    unique_brands = set(p["brand"] for p in recommendations)
    if len(unique_brands) < 2:
        recommendations = all_products[:5]

    return recommendations
```

Explanation:

In this task, Google Colab was used to design a product recommendation system that follows ethical principles such as transparency and fairness. Recommendation systems can significantly influence user choices, so it is important to avoid favoritism toward specific products or brands. The implemented logic ensures diversity in recommendations and clearly explains how recommendations are generated. This task emphasizes that ethical recommendation systems should be transparent, unbiased, and user-centric, ensuring trust and preventing manipulation or unfair advantages.

Task 4: Ethical Logging
Logs user actions without storing sensitive data.

Prompt:

Generate logging functionality in a Python web application ensuring logs do not store sensitive information.

Code with Output:

```
import logging

logging.basicConfig(
    filename="app.log",
    level=logging.INFO,
    format="%(asctime)s - %(levelname)s - %(message)s"
)

def log_user_action(username):
    # Ethical logging:
    # Do not log passwords, emails, or personal identifiers
    logging.info(f"User action recorded for user: {username}")
```

Explanation:

This task demonstrates ethical logging practices implemented using Python in Google Colab. Logging is essential for tracking system behavior and debugging, but it can become unethical if sensitive user information is recorded. The logging functionality was designed to avoid storing personal identifiers such as passwords, email addresses, or confidential data. By logging only necessary and non-sensitive information, this task highlights the importance of privacy, security, and ethical responsibility in backend system design.

Task 5: Responsible ML Model Usage
Documentation explains limitations, fairness, and human oversight.

Prompt: Generate a machine learning model and add documentation on responsible usage.

Code with Output:

```python
"""
Task 5: Responsible Use of Machine Learning Model

This program demonstrates a simple machine learning model
with ethical and responsible AI considerations.

Responsible AI Guidelines:
1. Explainability:
    - The model used (Logistic Regression) is interpretable.
    - Feature importance can be examined if required.

2. Accuracy Limitations:
    - Model performance depends on data quality and size.
    - Predictions may not be accurate for unseen or biased data.

3. Fairness:
    - Training data should be checked for class imbalance.
    - Bias audits should be performed before deployment.

4. Human Oversight:
    - Model outputs should support decisions, not replace humans.
    - Not suitable for critical applications without validation.
"""

from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import numpy as np

# Sample dataset (for demonstration only)
# Features: [study_hours, attendance_percentage]
X = np.array([
    [2, 60],
    [4, 70],
    [6, 80],
    [8, 90],
    [10, 95]
])

# Labels: 0 = Fail, 1 = Pass
y = np.array([0, 0, 1, 1, 1])

# Split dataset to avoid overfitting
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

# Create an interpretable model
model = LogisticRegression()

# Train the model
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Model Accuracy:", accuracy)

# Ethical Note:
# Accuracy alone is not enough.
# Fairness, transparency, and real-world impact must be considered.
```

```
Model Accuracy: 0.0
```

Explanation:

In the final task, Google Colab was used to create a simple machine learning model along with documentation explaining its responsible use. Machine learning models may produce incorrect or biased predictions if used beyond their intended scope. The documentation clearly explains the model's limitations, accuracy constraints, fairness considerations, and the need for human oversight. This task reinforces that AI models should be transparent, explainable, and used responsibly, with developers remaining accountable for ethical decision-making and real-world impact.

Final Reflection:
AI tools must be used responsibly. Developers remain accountable for ethical outcomes.