AI Content Authenticity Challenge

Group 9 :

1. Mukund Tiwari
2. Atharva Vichare

Understanding Problem Statement

Artificial intelligence systems can carry out tasks and gradually improve their capabilities. They can accomplish this because of the data they have collected from tasks. A vast variety of materials may be produced using AI. Articles, reports, posts on social media, and even photos fall under this category. This content frequently gets mixed up with human-generated stuff.

AI Writing Tool Evolution

AI writing tools have evolved and become increasingly complex over time. AI systems use natural language processing to produce a variety of materials. AI writing programmes create stories that mimic human-like language.

Objective

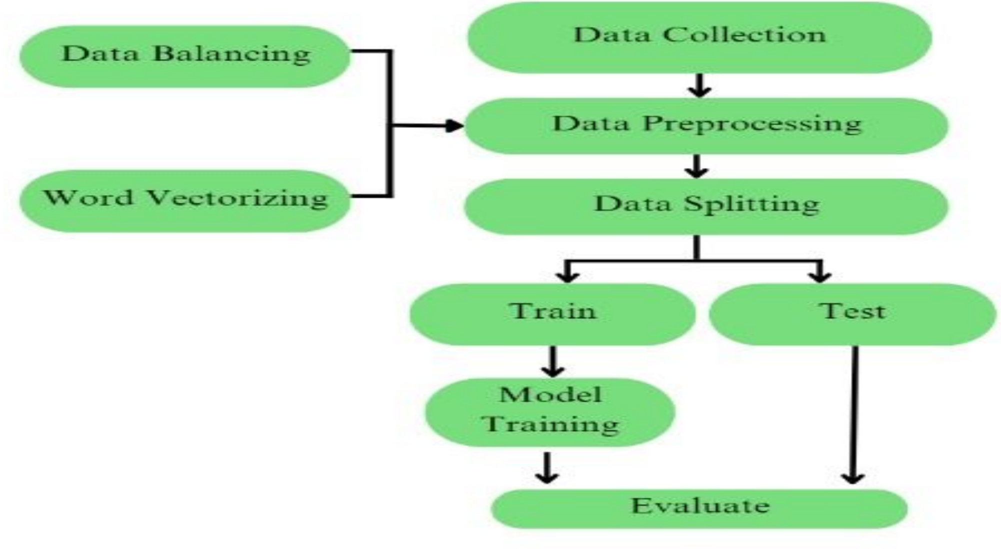
- The main objective is to create a program using Python to detect the AI-generated text
- And also using machine learning to detect the drawbacks of AI-generated Languages increasing the accuracy of our model.

Challenges

- As mentioned earlier Ai is great at mimicking human language and text, hence model would sometimes face difficulty in detecting the text.

- For this, the model needs to be trained with huge data in a constrained time.

Proposed Idea



overview of the process

Technology Used

- Python: The primary programming language used for developing the Streamlit app.
- Streamlit: A Python library used for building interactive web applications for data science and machine learning tasks. Streamlit simplifies the process of creating web interfaces by allowing developers to write Python scripts that generate UI components dynamically.
- Pandas: A Python library used for data manipulation and analysis. In the app, Pandas is used to load and manipulate the dataset containing text samples and their corresponding labels.
- scikit-learn (sklearn): A machine learning library in Python that provides tools for data preprocessing, model selection, and evaluation. In the app, scikit-learn is used for text vectorization (TF-IDF) and training the classification model (SVM).
- NumPy: A Python library used for numerical computing. While not explicitly mentioned in the code, NumPy is often used in conjunction with Pandas and scikit-learn for handling numerical data efficiently.
- HTML/CSS: Streamlit allows developers to customize the appearance of the web application using HTML and CSS. While the provided code doesn't include custom HTML/CSS, you can extend the app's appearance by incorporating these technologies.

Uniqueness

- Streamlit UI: The app uses Streamlit to create a user interface (UI) for the text classification task. Streamlit provides simple commands to create interactive elements such as text inputs, buttons, and data displays.

- Text Input Area: Users can input text into a text area provided by Streamlit. This text serves as the input for the text classification task.

- Classification Button: The app includes a button labeled "Classify." When users input text and click this button, the app triggers the classification process.

- Model Prediction: After clicking the "Classify" button, the app preprocesses the input text, vectorizes it using TF-IDF, and feeds it into the trained SVM model. The model predicts whether the text is written by a human or AI.

- Classification Result: The app displays the classification result (whether the text is written by a human or AI) to the user.