**Title: IMDB Sentiment Analysis & Web Application Deployment**

**1. Introduction**

This project involves building a machine learning model to analyze the sentiment of IMDB movie reviews and deploying it as a web application using Gradio. The goal is to predict whether a given review has a positive or negative sentiment.

**2. Dataset**

* **Source**: IMDB Dataset for Sentiment Analysis (commonly available via Kaggle or TensorFlow Datasets).
* **Size**: 50,000 reviews labeled as either positive or negative.
* **Preprocessing**: Tokenization, stopword removal, padding/truncation (for deep learning models), and vectorization.

**3. Model Development**

**3.1 Data Preprocessing**

1. Loaded the IMDB dataset.
2. Cleaned text (removed punctuation, stopwords, converted to lowercase).
3. Tokenized and converted text into numerical format using TF-IDF or word embeddings (e.g., Word2Vec or GloVe).

**3.2 Model Selection & Training**

Two different approaches were tested:

* **Traditional ML**: Used Logistic Regression and Naive Bayes.
* **Deep Learning**: Implemented an LSTM-based neural network.

**Final Model Chosen: The best-performing model was selected based on accuracy and generalization capability.**

**4. Web Application Development**

**4.1 Setting Up Gradio**

We used the Gradio library to create an interactive interface where users can enter movie reviews and get real-time sentiment predictions.

import gradio as gr

def predictive\_system(review):

# Preprocess input review

processed\_review = preprocess\_text(review)

# Predict sentiment using trained model

prediction = model.predict([processed\_review])

sentiment = "Positive" if prediction == 1 else "Negative"

return sentiment

interface = gr.Interface(

fn=predictive\_system,

inputs=gr.Textbox(label="Enter a Review"),

outputs=gr.Label(label="Sentiment"),

title="IMDB Sentiment Analysis",

description="Enter a movie review to predict its sentiment."

)

if \_\_name\_\_ == "\_\_main\_\_":

interface.launch(share=True)

**4.2 Running the Web Application**

* interface.launch(share=True) generates a public link for external users.
* Users enter a review, and the system predicts its sentiment in real-time.

**5. Challenges & Solutions**

1. **Issue**: McAfee flagged Gradio’s executable as a potential threat.
   * **Solution**: Avoided unnecessary exceptions by running locally instead of using a public link.
2. **Issue**: Model size affected deployment speed.
   * **Solution**: Used a lightweight model for faster predictions.

**6. Future Enhancements**

* Improve the model using transformer-based architectures like BERT.
* Deploy on cloud services like Hugging Face Spaces, AWS, or Render.
* Add a database to store user reviews and feedback.
* Visualize word importance using SHAP or attention mechanisms.

**7. Conclusion**

This project successfully built and deployed a sentiment analysis model for IMDB reviews using machine learning and Gradio. While simple, it serves as a foundational exercise in ML model deployment.