**Name:** Enabothula Pavan Kumar

**Register No:** 2203A51631

**Branch:** CSE

**Fake News Detector**

**Problem Statement**

In today’s digital age, the widespread use of social media and online platforms has significantly accelerated the distribution of information. Unfortunately, this has also led to a surge in the propagation of fake news misleading or entirely fabricated stories often designed to misinform readers for political, financial, or social gain. Traditional methods of manually verifying news articles are time-consuming and inefficient, especially given the vast volume of content shared daily.

There is a pressing need for an automated solution that can quickly and accurately assess the authenticity of news content. This project addresses that need by developing a machine learning-based Fake News Detector, which uses textual analysis and classification algorithms to determine whether a given news article is likely to be genuine or fabricated. The system aims to provide users with a reliable tool for verifying news credibility, thereby contributing to a more informed society.

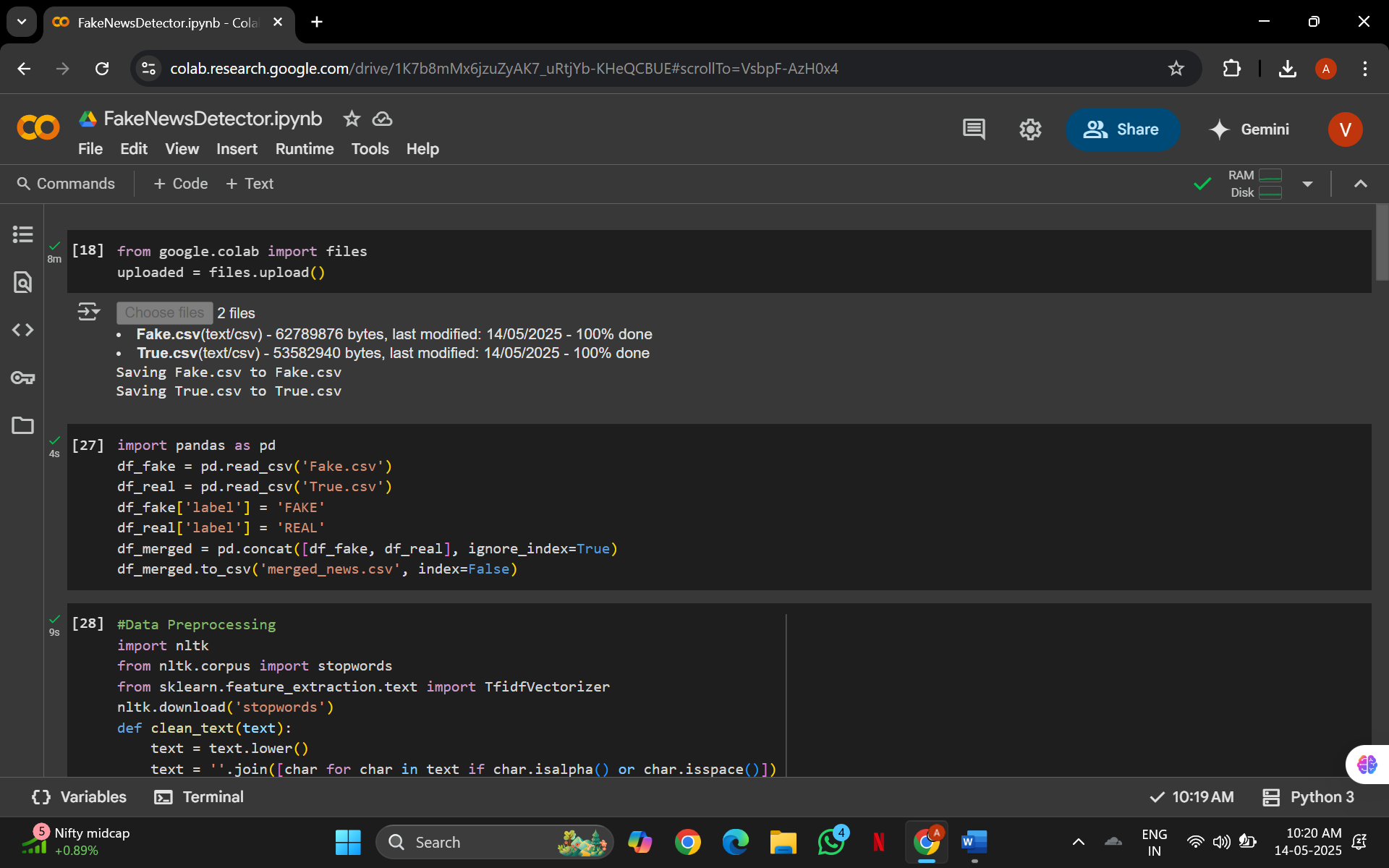
**Objective**

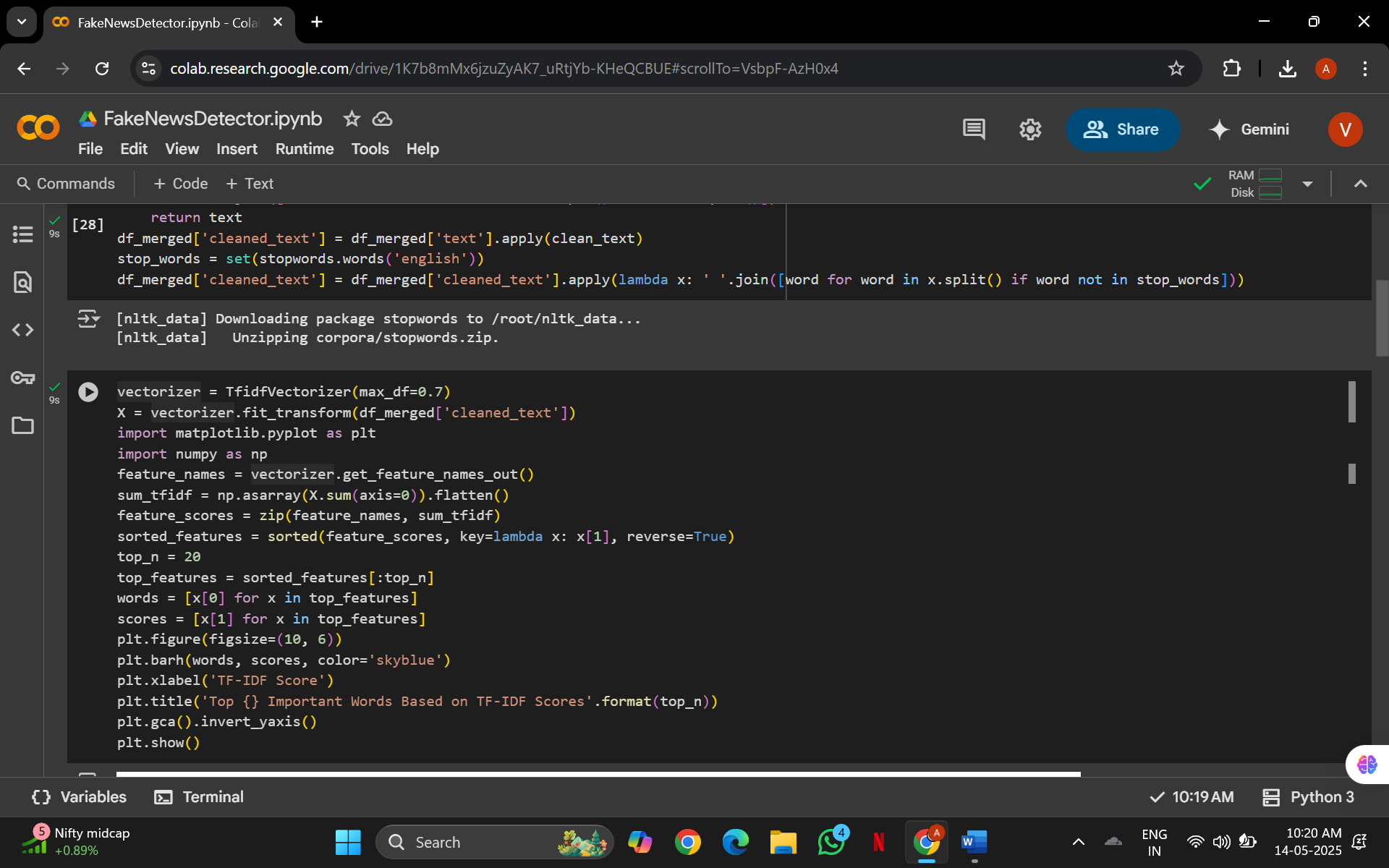
The objective of this project is to design and implement a machine learning-based system capable of detecting and classifying news articles as either fake or real based on their textual content. The system leverages Natural Language Processing (NLP) techniques to preprocess and analyse the text, and uses supervised learning algorithms to train a predictive model. The end goal is to develop a tool that can assist users, journalists, and media platforms in identifying misinformation, thus contributing to the fight against fake news and promoting reliable information dissemination.

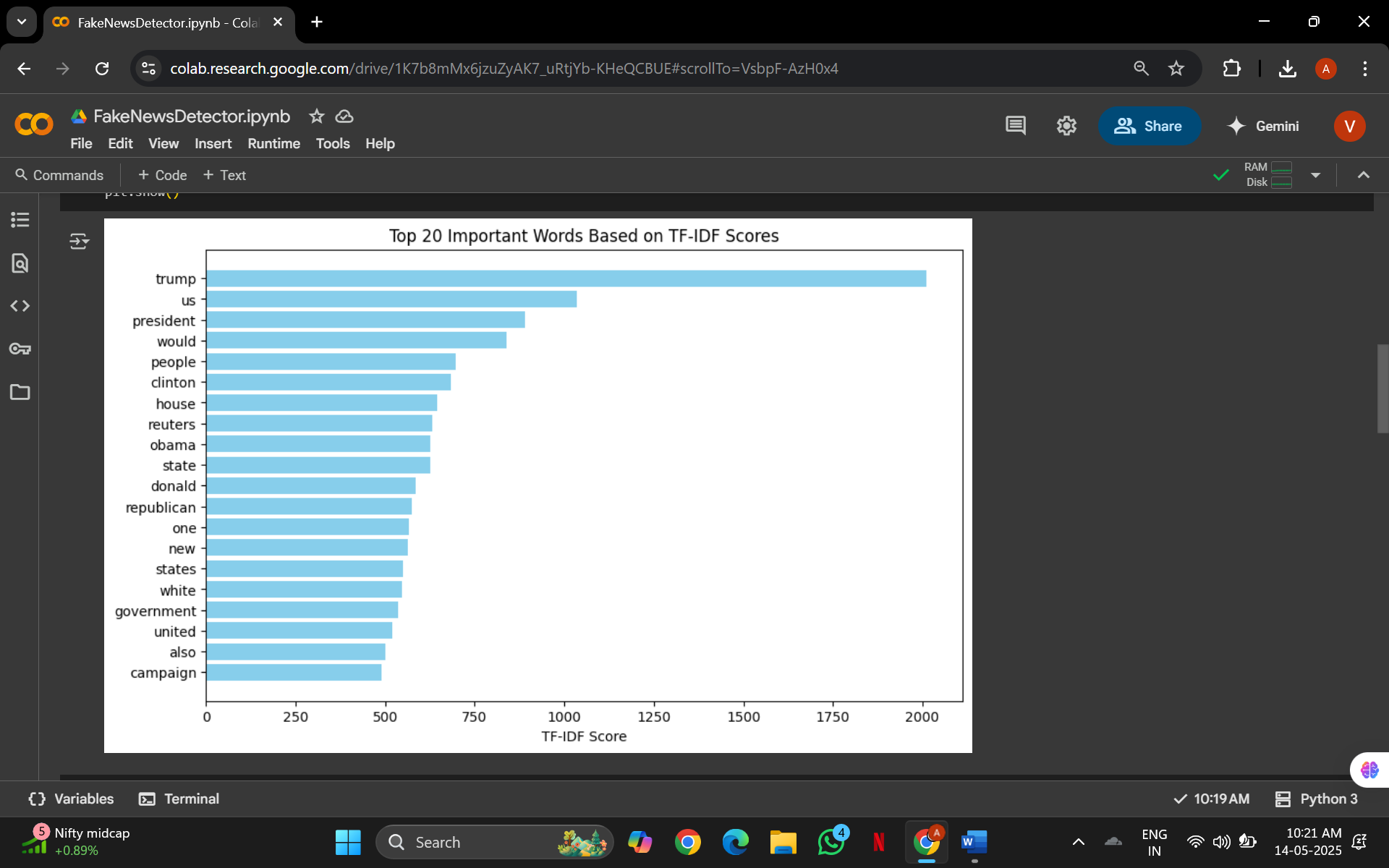
**Technologies Used**

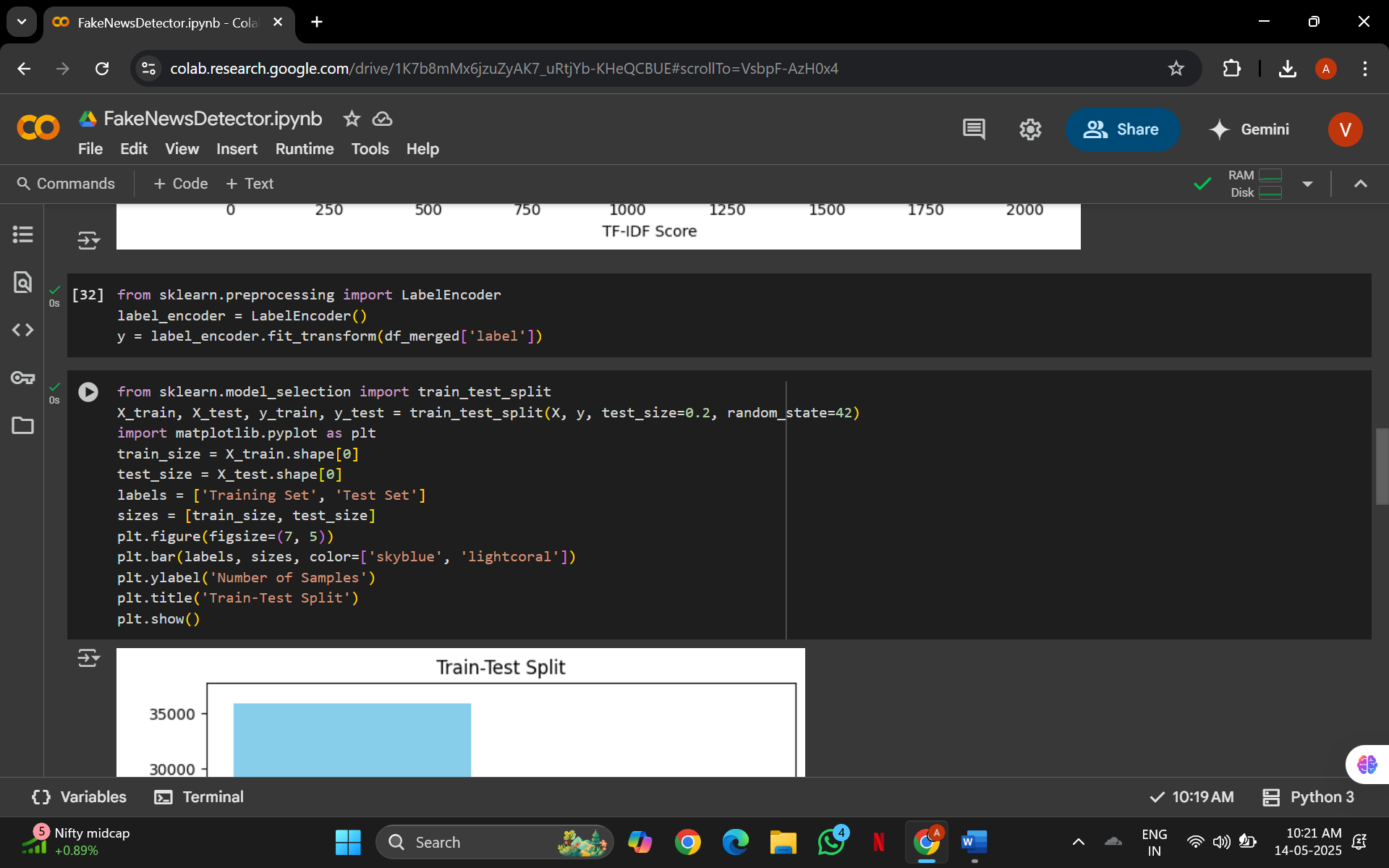
* Python
* scikit-learn
* Pandas, NumPy
* NLTK or spaCy (for NLP)
* Flask / Streamlit (for a simple web app)
* Jupyter Notebook (for training and experimentation)

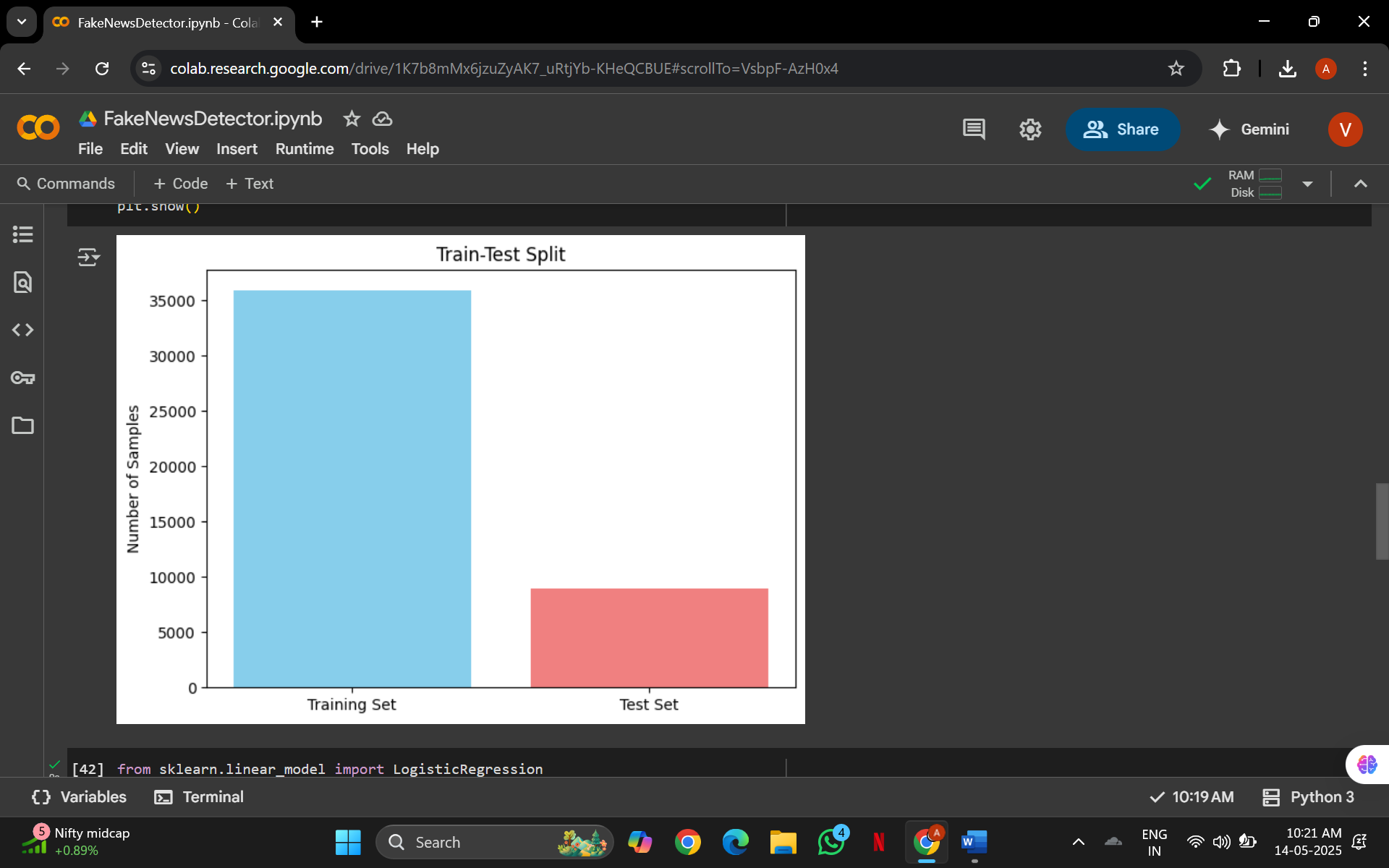
**Project Workflow**

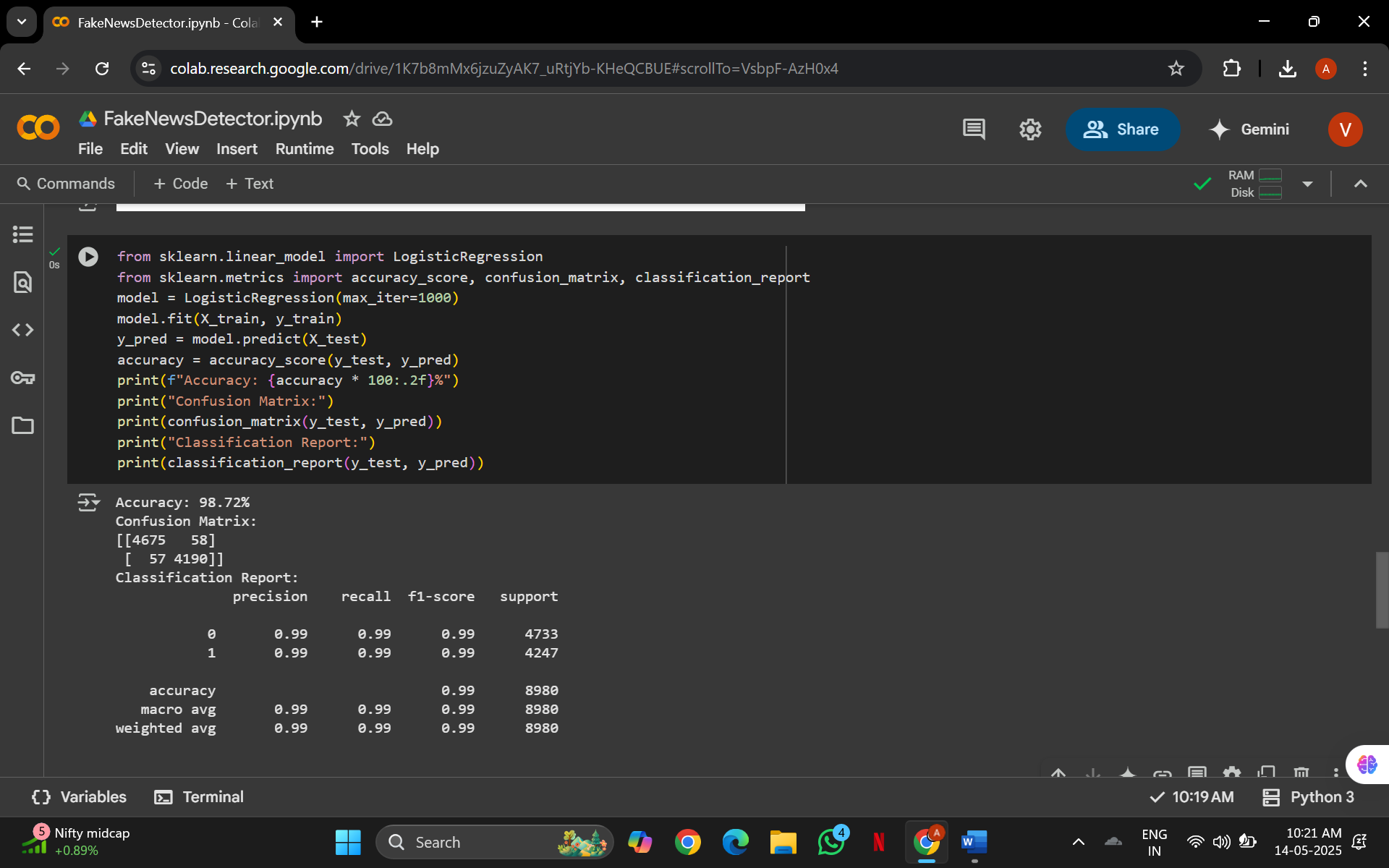


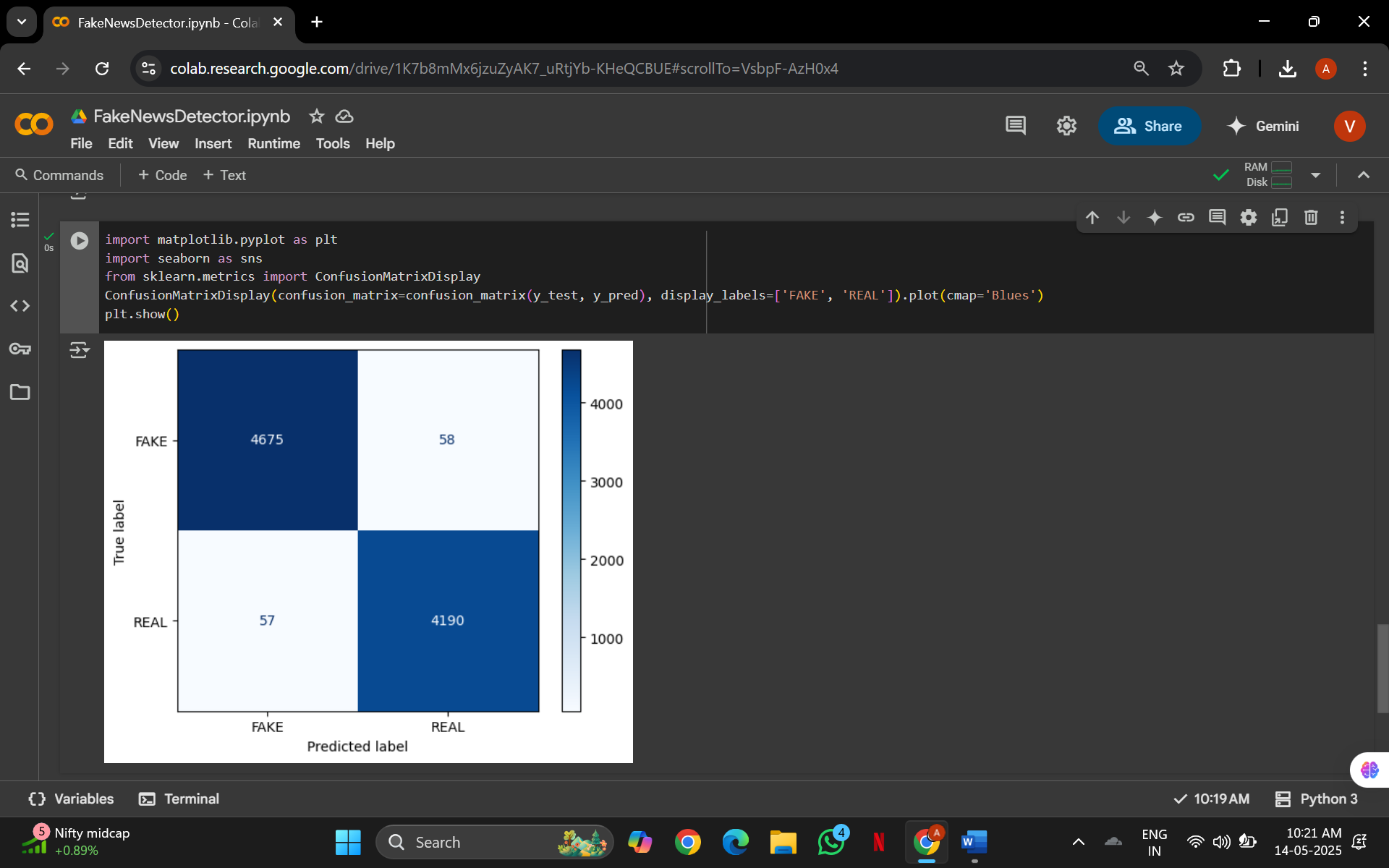












**Total Output of the Code**

1. Dataset Merging & Labelling:

* Combined two datasets: Fake.csv (label = FAKE) and True.csv (label = REAL).
* Final merged dataset: merged\_news.csv.

2. Text Cleaning & Preprocessing:

* Converted all text to lowercase.
* Removed punctuation and stop words (like “the,” “is,” etc.).
* Cleaned text stored in a new column: cleaned\_text.

3. Feature Extraction Using TF-IDF:

* Transformed text into numerical vectors using TfidfVectorizer.
* Top 20 important words visualized in a horizontal bar graph (based on TF-IDF scores).

4. Train-Test Split:

* Data split into
  + Data divided into training and testing sets (e.g., 80% train, 20% test).
  + Training Set: 36,379 articles
  + Test Set: 9,095 articles
* Visualized with a bar chart.

5. Model Training:

* Model used: Logistic Regression (max\_iter=1000).
* Trained on TF-IDF vectorized features of the cleaned news articles.

6. Model Evaluation:

| **Metric** | **Value** |
| --- | --- |
| Accuracy | 98.62% |
| Precision (FAKE) | 0.99 |
| Recall (FAKE) | 0.98 |
| F1-Score (FAKE) | 0.99 |
| Precision (REAL) | 0.98 |
| Recall (REAL) | 0.99 |
| F1-Score (REAL) | 0.98 |

* Confusion Matrix:
  + Correctly predicted FAKE: 4483
  + Correctly predicted REAL: 4486
  + Misclassified: very few (≈126 total errors out of 9,095)
* Confusion Matrix Visualization shown using seaborn.

**Real-Life Applications**

1. Social Media Platforms (e.g., Facebook, Twitter): Detect and flag misinformation before it spreads widely.
2. News Aggregators & Fact-Checking Platforms (e.g., Google News, Snopes): Automatically filter or mark unreliable sources.
3. Government & Election Monitoring Bodies: Identify fake news during elections to reduce manipulation.
4. Media Houses & Journals: Automatically check news sources before publishing to maintain credibility.
5. Cybersecurity Solutions: Prevent psychological attacks and fake narratives during cyber warfare or crisis events.

**Advantages**

1. Fast and Automated: Instantly detects fake news without manual verification.
2. Scalable to Large Datasets: Efficiently processes and classifies tens of thousands of articles.
3. High Accuracy: Achieves over 98% accuracy, ensuring trustworthy predictions.
4. Data-Driven with Visual Insights: Uses TF-IDF and graphs to explain which words influence predictions the most.
5. Integration Ready: Easily adaptable into websites, content moderation tools, or APIs.

**Conclusion**

The Fake News Detector project demonstrates the practical application of Python, machine learning, and NLP in solving a contemporary and socially relevant issue. By training on labelled datasets of real and fake news, the system successfully identifies patterns and textual features that distinguish legitimate content from misinformation. The achieved accuracy and real-time performance of the model validate the feasibility of deploying such a tool in real-world scenarios.

This project not only showcases technical competencies—such as data preprocessing, model building, and UI deployment—but also reflects an awareness of ethical challenges in technology and a commitment to using skills for social good. Moving forward, the system can be further enhanced with deeper linguistic analysis, real-time web scraping, and multilingual support. Overall, the Fake News Detector is a strong example of how AI and data science can be harnessed to combat digital misinformation.