NLP Project

13/11/2025

Nicolas Z.

Logistic Regression (0.9886 ± 0.0016)

Pipeline Steps

- Step 1 : Text Cleaning & Preprocessing
- Step 2 : TF-IDF Vectorization (20K features)
- Step 3 : Logistic Regression Training
- Step 4 : Model Evaluation & Deployment

Accuracy:0.9875

Metric	Class 0 (Fake)	Class 1(Real)
F1 Score	0,99	0,99
Recall	0,98	0,99
Precision	0,99	0,99

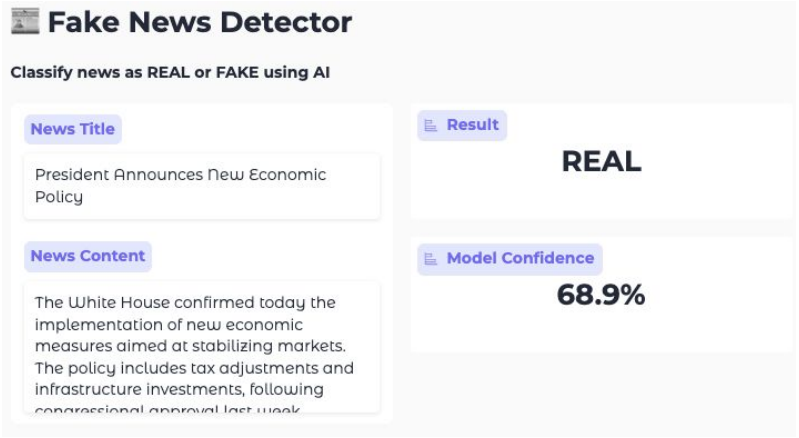
Naive Bayes Model (0.9530 ± 0.0014)

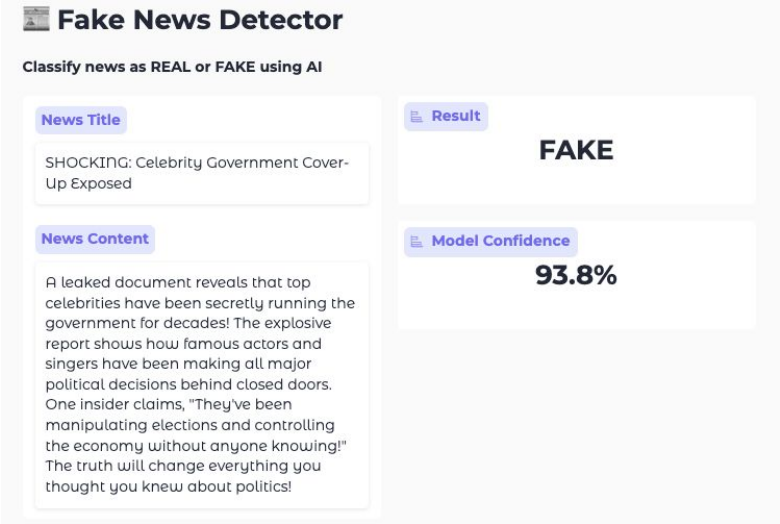
Metric	Class 0 (Fake)	Class 1(Real)
F1 Score	0,95	0,95
Recall	0,95	0,95
Precision	0,95	0,95

Validation and Deployment

- Excellent performance in a control environment
- Real-world testing reveals the model shows higher confidence in detecting fake news patterns while being more cautious with authentic content - highlighting opportunities for dataset diversification and model generalization improvements
- Deployment:(also the folder name)

  - app.py (Gradio interface)
  - utils.py (Model processing)
  - requirements.txt (Dependencies)
  - logistic_regression_model.pkl (Trained model)
  - tfidf_vectorizer.pkl (Vectorizer TF-IDF)





Learning Takeaways

Learnings

Technical

- NLP: Text cleaning, TF-IDF, Deployment
- Model selection and comparison

Analysis

→ Data limitations: Training data impacts real-world performance

Takeaways

→ Text Preprocessing Pipeline:

- Acronym handling (U.S. → US, F.B.I. → FBI)
- Stopword Removal + special character cleaning
- Normalization (lowercase