Fake News Detector

Group 3

Jacob Farley, Trenton Paxton, Megan Neal

What is Fake News?

Fake News or information disorder is false or misleading information claiming the aesthetics and legitimacy of news. Fake news is often written with the objective of damaging a person or entity's reputation. It can also be used generate advertisement revenue with "clickbait" style marketing.

Why It Matters:
How Headlines Change the Way We Think

SUPPORTIVE POINTS

YADA-YADA-YADA

- https://www.newyorker.com/science/maria-konnikova/headlines-change-way-think

How Machine Learning Can Help

- While Fake News is mass produced and deliberately misleading, human fact-checking is a slow and painstaking process. Machine Learning can help automate and accelerate that process by detecting patterns in text, sources, and writing style.

Our Datasets

Kaggle Fake and Real News Dataset

We utilized two datasets hosted on Kaggle. The first contains two CSV files; one with headlines, content, subject and publication for both true and fake news articles. The other dataset contained four CSVs categorized by true and fake political and gossip articles. Both datasets were created using articles processed by PolitiFact.

Dataset was merged and contains over 20k articles for each classification of true and false.

PolitiFact

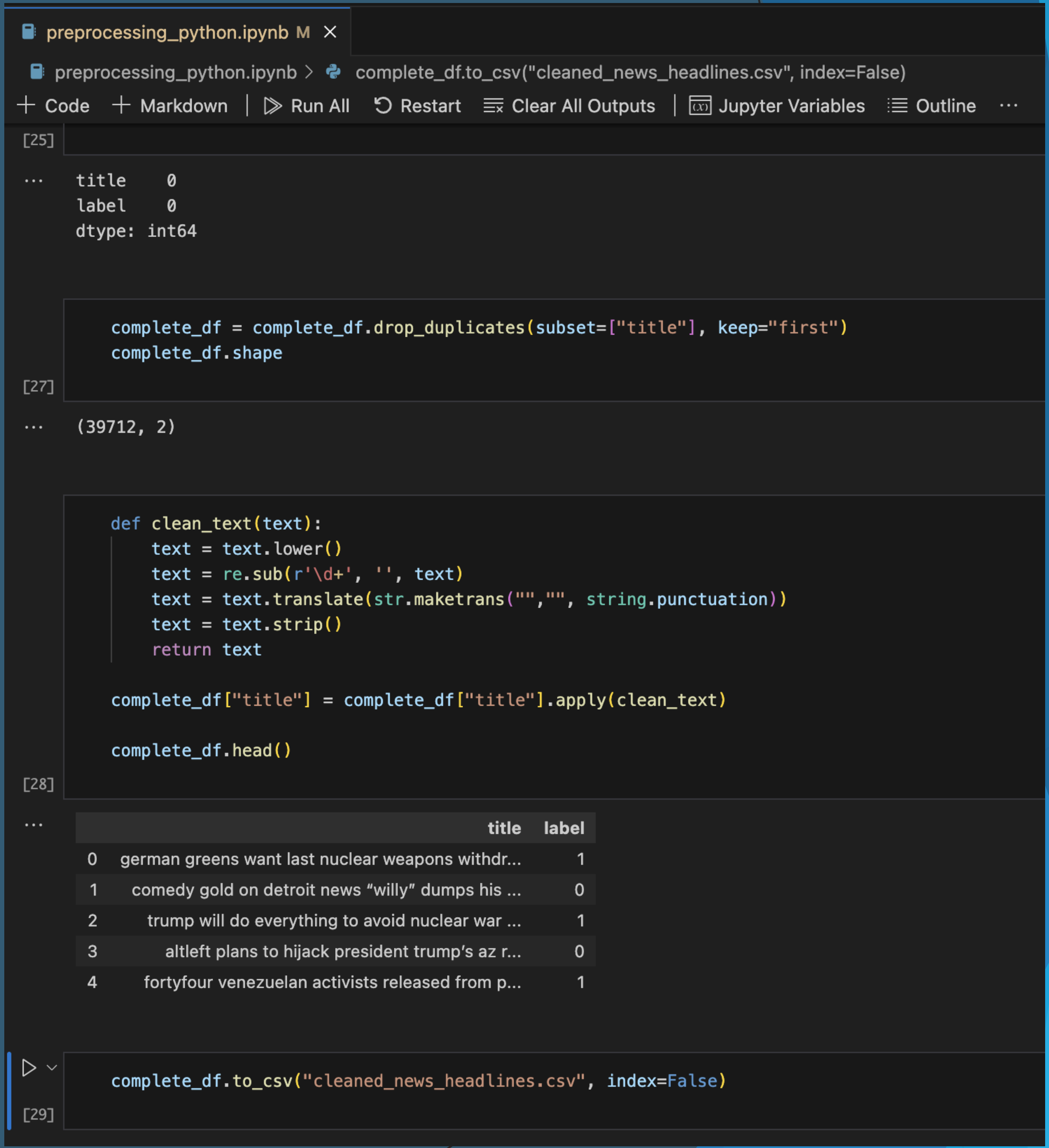
PolitiFact is a fact-checking organization that evaluates the accuracy of statements made by politicians, public figures, and media sources.

- Uses a fact-checking scale that rates things from *True* to *Pants on Fire*.
- Operates independently to verify facts without political bias.
- Uses verified data, expert analysis, and public records.
- Detects misinformation that debunks viral fake news and misleading statements.
- Founded in 2007 and won a Pulitzer Prize for fact-checking.

Data Preprocessing

- Reading in the two CSV's.
- Standardize the data (removed capitalization, typos, and punctuation).
- Assigned Target Variables:
  - 1 = True
  - 0 = False
- Drop unnecessary columns.
- Merge data into a single DataFrame.
- Write the data into a new CSV that is used to train our model.

- A second model was trained using a DataFrame that kept the article content as well as the headlines and Target Variables


preprocessing_python.ipynb M ×

preprocessing_python.ipynb > complete_df.to_csv("cleaned_news_headlines.csv", index=False)

+ Code + Markdown | Run All Restart Clear All Outputs | Jupyter Variables Outline ···

[25]

```
title    0
label    0
dtype: int64
```

```python
complete_df = complete_df.drop_duplicates(subset=["title"], keep="first")
complete_df.shape
```

[27]

```
(39712, 2)
```

```python
def clean_text(text):
    text = text.lower()
    text = re.sub(r'\d+', '', text)
    text = text.translate(str.maketrans("","", string.punctuation))
    text = text.strip()
    return text

complete_df["title"] = complete_df["title"].apply(clean_text)

complete_df.head()
```

[28]

	title	label
0	german greens want last nuclear weapons withdr...	1
1	comedy gold on detroit news “willy” dumps his ...	0
2	trump will do everything to avoid nuclear war ...	1
3	altleft plans to hijack president trump’s az r...	0
4	fortyfour venezuelan activists released from p...	1

```python
complete_df.to_csv("cleaned_news_headlines.csv", index=False)
```

[29]

TF-IDF for Headlines

- Coverts headlines to numerical features
- Term Frequency (TF) and Inverse Document Frequency (IDF)
- IDF provides a lower weight for commonly used connective words (For, the, and that)
- Advantages: Simple, interpretable, efficient for short texts
- Quickly able to verify headlines

```
# Create a pipeline that vectorizes the text and then applies Logistic Regression
model_pipeline = Pipeline([
    ('tfidf', TfidfVectorizer(stop_words="english")),
    ('logreg', LogisticRegression())
])

# Train the model
model_pipeline.fit(X_train, y_train)
print("Model training completed.")
```

TF-IDF Vectorization for Headline Analysis

Scientists discover breakthrough treatment for cancer

1. Term Frequency (TF)

Count how often each word appears in the headline

"scientists": 1 "discover": 1 "breakthrough": 1
"treatment": 1 "for": 1 "cancer": 1

2. Inverse Document Frequency (IDF)

Measures how unique a word is across all headlines

IDF(t) = log(N/df(t))

N = total headlines, df(t) = headlines with term t

3. TF-IDF Score = TF × IDF

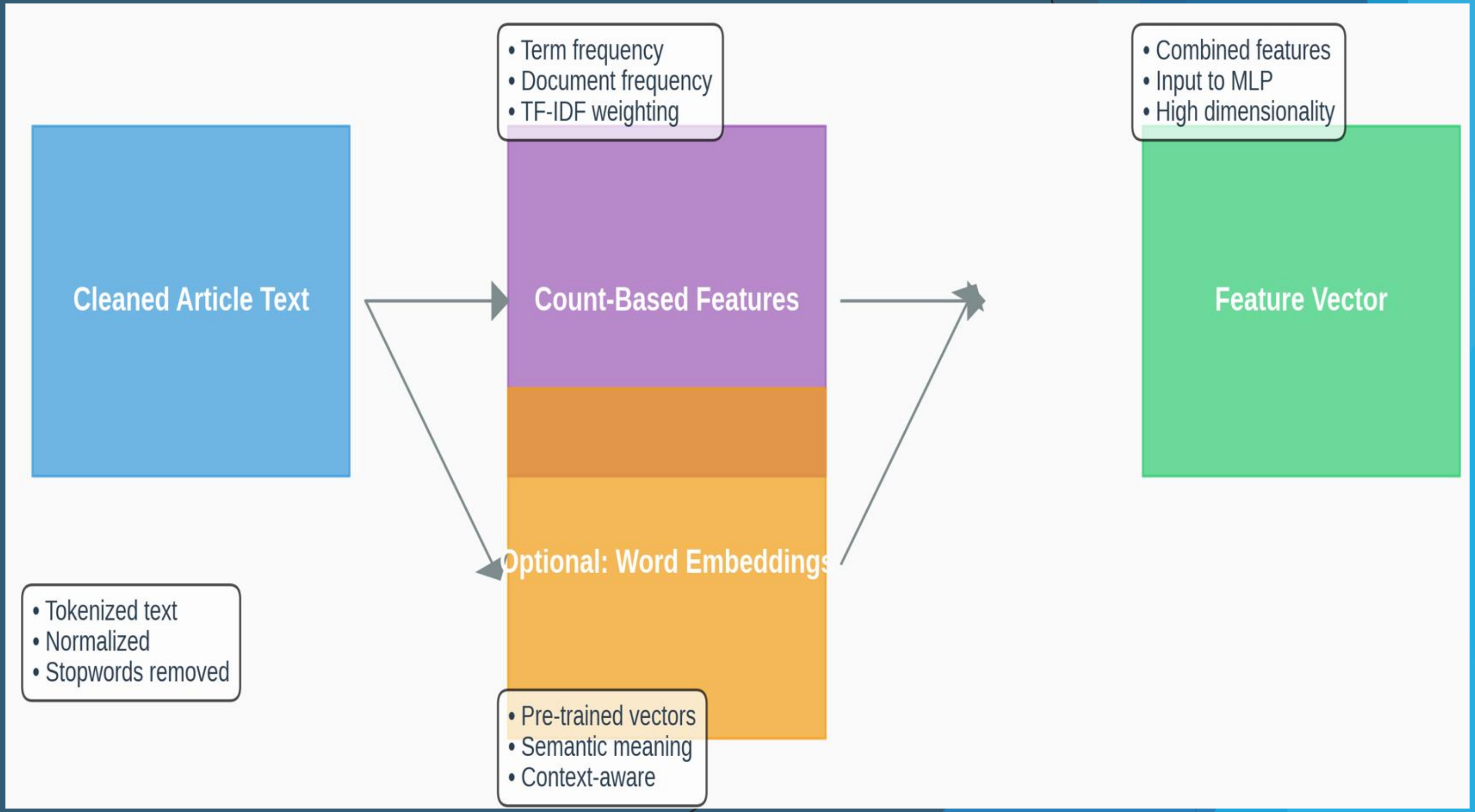
Term	TF	IDF	TF-IDF
breakthrough	1	3.1	3.1
scientists	1	2.3	2.3
discover	1	1.8	1.8
treatment	1	1.5	1.5

4. Final Vector Representation

[0, 0, 3.1, 0, 0.2, 1.2, 2.3, 1.8, 1.5, 0, ...]

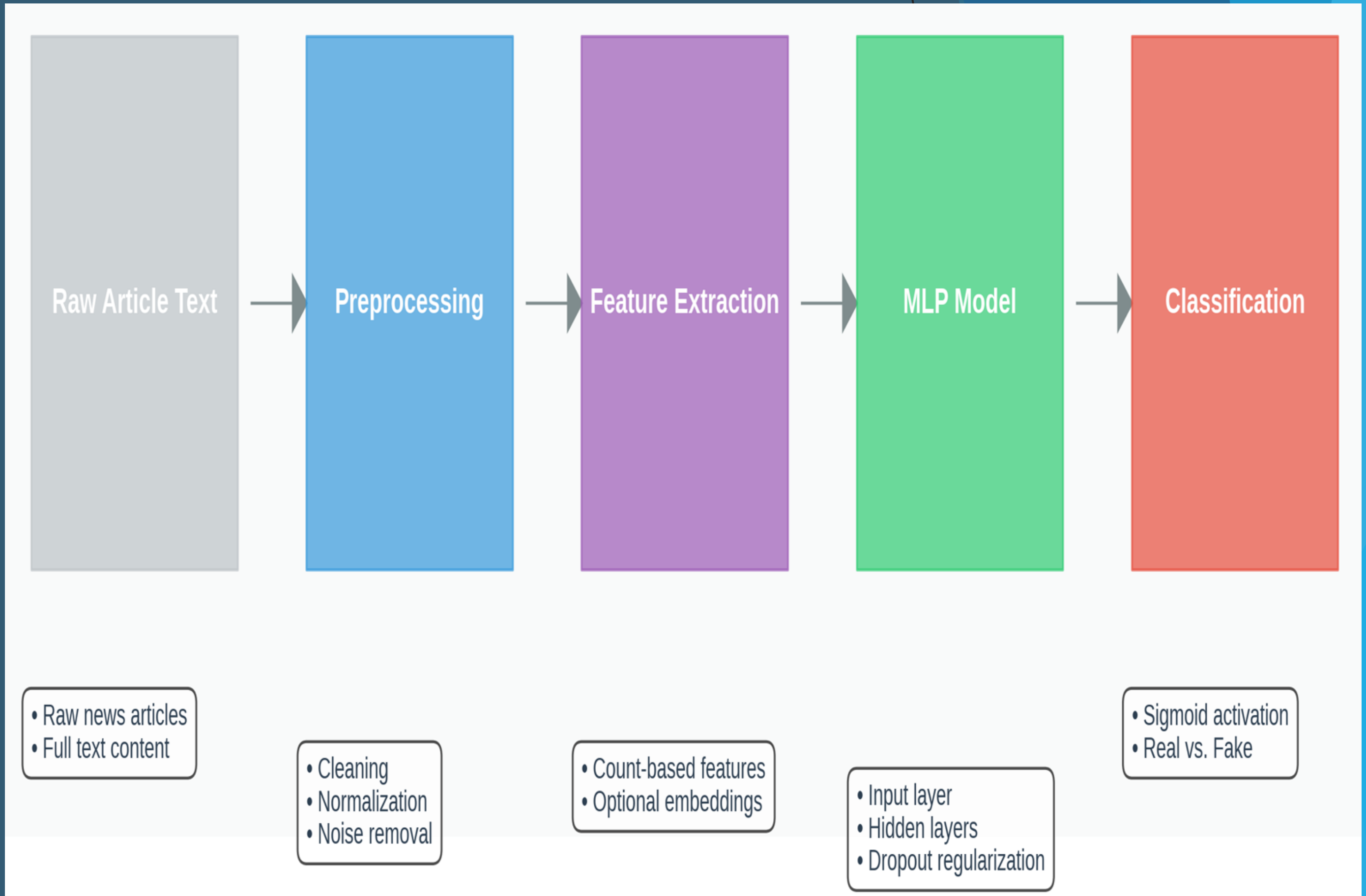
Article processing

- Advanced Preprocessing: cleaning, normalization, removing noise.
- Feature Extraction: count-based features and optionally embedding aggregation.
- MLP Architecture: Input layer matching feature size, hidden ReLU layers, dropout regularization, output layer with sigmoid activation.
- Hyperparameter Tuning via grid search/cross-validation.

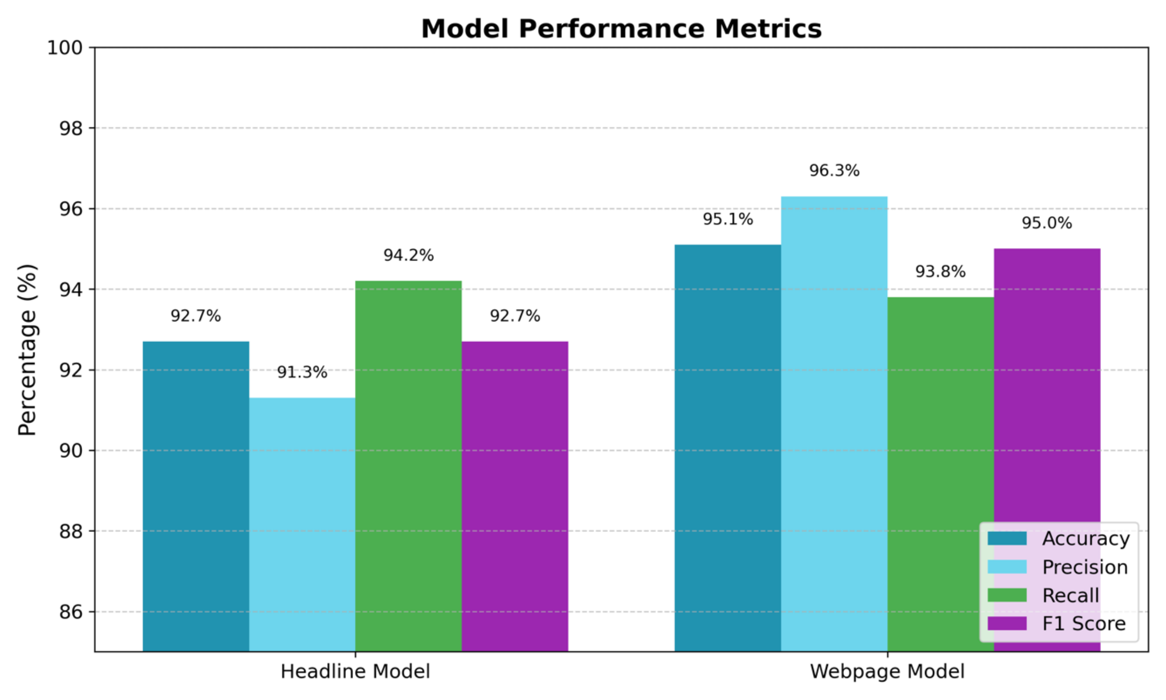


Article processing 2

- Input Layer: Dense vector from extracted features.
- Hidden Layers: One or two layers with ReLU activations capturing non-linear relationships.
- Regularization: Dropout layers to mitigate overfitting due to high-dimensional inputs.
- Output Layer: Sigmoid function for binary classification (real vs. fake).



Brain dump pics



"Our Model Name Here" Demo

Challenges

Formal vs. Emotionally Driven Language

Real news is typically formal, structured, and neutral, whiel fake news is often sensational, emotionally charged, or exaggerated.

Hard to quantify: ML struggles to objectively measure emotional intensity and writing style.

Fake News Can Look Real

Some fake articles are purely satirical (The Onion) and mimic real news formatting.

Misclassification Risk: A well-written fake news article might get classified as real.

Computational Limitations

Large datasets slow down training on local machines.

Processing millions of words requires substantially more memory and GPU power.

The Subjectivity of Truth

Truth is often context-dependent, which ML models often struggle to capture.

Articles can contain a mix of facts and misinformation, making strict classification difficult.

User Input in Demo Introduces Bias

Allowing users to input their own determination is interesting but opens doors for manipulation.

Considerations

Machine Learning Doesn’t "Verify" Facts

ML helps classify questionable vs. Reliable content but doesn’t fact-check like a human would.

- Example: A headline might be plausible but misleading, making automated verification difficult.

Balancing Accuracy and Explainability

Advanced models perform better but are less interpretable

We aimed for a balance between performance and interpretability so the results are understandable.

Bias and Ethics in Misinformation Detection

Who decides what's real or fake? If our dataset is biased, the model will learn biased patterns.

Potential Risks: The model could lead to unfair censorship or false positives for new or controversial stores

Improvements

Implement a Web Scraping Tool

- A tool that lets users input a website URL to analyze the news credibility.

Granular Scoring Instead of Binary Classification

- Instead of just "Real (1)" or "Fake (0)", we could introduce a credibility score (0-100).
  - Example: "Likely Fake: 20% Credibility" instead of a strict 0/1 label

Sentiment & Contextual Analysis

- Enhance ML by understanding text context (sarcasm, satire, misleading wording)
- Use Deep Learning (BERT, GPT) for better nuance detection.

Improve Dataset Quality & Bias Reduction

- Expand datasets to include more verified fact-checking sources.
- Adjust for political, cultural, and regional biases in misinformation

Summary

- Fake News detection is complex and some articles are misleading but not outright false.
- ML helps categorize reliability but doesn’t replace human verification.
- Future improvements could include website scraping, credibility scoring, and contextual analysis to make detection more accurate and user-friendly.
- While our fight against misinformation is ongoing, AI is just one tool in the solution