Project: Clickbait Headline Classification Using NLP and Machine Learning

Group 3: Yutong He

This notebook walks through the full pipeline for building a clickbait headline classifier using NLP and machine learning techniques. Run the code cells in order to:

- Import and download necessary libraries and files
- Load and preprocess the headline datasets
- Train and evaluate models using both traditional (TF-IDF, BOW) and transformer-based features
- Visualize results and performance metrics
- Run ablation experiments to assess the effect of stopword removal, truncation, and feature extraction choices
- Implement sentence embedding experiments using sentence-transformers
- Compare and interpret model performance
- Launch a Gradio web UI to allow users to input a headline and classify it using the best-performing model (BOW + Logistic Regression)

In [1]:

```
# Step 1: Import Libraries
import pandas as pd
import numpy as np
import gzip
import matplotlib.pyplot as plt

!pip install seaborn

import seaborn as sns
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, roc_auc_score, roc_curve

!pip install torch sentence-transformers
```

```
import torch
import torch.nn as nn
from torch.utils.data import DataLoader, TensorDataset, random_split
from sentence_transformers import SentenceTransformer


from ablation_batch_runner import run_all_ablation_experiments
import pickle
from ablation_runner import run_ablation_experiment

!pip install gradio
import gradio as gr
```

```
Requirement already satisfied: seaborn in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (0.13.
2)
Requirement already satisfied: numpy!=1.24.0,>=1.20 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pa
ckages (from seaborn) (1.26.4)
Requirement already satisfied: pandas>=1.2 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (f
rom seaborn) (2.2.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site
-packages (from seaborn) (3.10.3)
Requirement already satisfied: contourpy>=1.0.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packag
es (from matplotlib!=3.6.1,>=3.4->seaborn) (1.3.2)
Requirement already satisfied: cycler>=0.10 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from matplotlib!=3.6.1,>=3.4->seaborn) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packa
ges (from matplotlib!=3.6.1,>=3.4->seaborn) (4.58.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packa
ges (from matplotlib!=3.6.1,>=3.4->seaborn) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from matplotlib!=3.
6.1,>=3.4->seaborn) (25.0)
Requirement already satisfied: pillow>=8 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (fro
m matplotlib!=3.6.1,>=3.4->seaborn) (11.2.1)
Requirement already satisfied: pyparsing>=2.3.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packag
es (from matplotlib!=3.6.1,>=3.4->seaborn) (3.2.3)
Requirement already satisfied: python-dateutil>=2.7 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from matplotli
b!=3.6.1,>=3.4->seaborn) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from pandas>=1.2->seaborn) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from pandas>=1.2->seaborn) (2025.2)
Requirement already satisfied: six>=1.5 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from python-dateutil>=2.7-
>matplotlib!=3.6.1,>=3.4->seaborn) (1.17.0)
Requirement already satisfied: torch in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (2.2.2)
Requirement already satisfied: sentence-transformers in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-p
ackages (5.0.0)
Requirement already satisfied: filelock in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from
torch) (3.8.0)
Requirement already satisfied: typing-extensions>=4.8.0 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from torc
h) (4.13.2)
Requirement already satisfied: sympy in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from to
rch) (1.14.0)
Requirement already satisfied: networkx in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from
torch) (3.4.2)
Requirement already satisfied: jinja2 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from t
orch) (3.1.4)
Requirement already satisfied: fsspec in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from t
orch) (2025.3.2)
Requirement already satisfied: transformers<5.0.0,>=4.41.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
```

```
site-packages (from sentence-transformers) (4.55.0)
Requirement already satisfied: tqdm in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from sen
tence-transformers) (4.67.1)
Requirement already satisfied: scikit-learn in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from sentence-transformers) (1.6.1)
Requirement already satisfied: scipy in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from se
ntence-transformers) (1.15.3)
Requirement already satisfied: huggingface-hub>=0.20.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site
-packages (from sentence-transformers) (0.34.3)
Requirement already satisfied: Pillow in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from s
entence-transformers) (11.2.1)
Requirement already satisfied: numpy>=1.17 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (f
rom transformers<5.0.0,>=4.41.0->sentence-transformers) (1.26.4)
Requirement already satisfied: packaging>=20.0 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from transformers<
5.0.0,>=4.41.0->sentence-transformers) (25.0)
Requirement already satisfied: pyyaml>=5.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (f
rom transformers<5.0.0,>=4.41.0->sentence-transformers) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packa
ges (from transformers<5.0.0,>=4.41.0->sentence-transformers) (2024.11.6)
Requirement already satisfied: requests in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (2.32.3)
Requirement already satisfied: tokenizers<0.22,>=0.21 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-
packages (from transformers<5.0.0,>=4.41.0->sentence-transformers) (0.21.4)
Requirement already satisfied: safetensors>=0.4.3 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pack
ages (from transformers<5.0.0,>=4.41.0->sentence-transformers) (0.5.3)
Requirement already satisfied: hf-xet<2.0.0,>=1.1.3 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pa
ckages (from huggingface-hub>=0.20.0->sentence-transformers) (1.1.7)
Requirement already satisfied: MarkupSafe>=2.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-package
s (from jinja2->torch) (2.1.5)
Requirement already satisfied: charset-normalizer<4,>=2 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/sit
e-packages (from requests->transformers<5.0.0,>=4.41.0->sentence-transformers) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from requests->transformers<5.0.0,>=4.41.0->sentence-transformers) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pack
ages (from requests->transformers<5.0.0,>=4.41.0->sentence-transformers) (1.26.20)
Requirement already satisfied: certifi>=2017.4.17 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pack
ages (from requests->transformers<5.0.0,>=4.41.0->sentence-transformers) (2025.4.26)
Requirement already satisfied: joblib>=1.2.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from scikit-learn->sentence-transformers) (1.5.1)
Requirement already satisfied: threadpoolctl>=3.1.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pa
ckages (from scikit-learn->sentence-transformers) (3.6.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pack
ages (from sympy->torch) (1.3.0)
Requirement already satisfied: gradio in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (5.41.
1)
Requirement already satisfied: aiofiles<25.0,>=22.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pa
```

```
ckages (from gradio) (24.1.0)
Requirement already satisfied: anyio<5.0,>=3.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-package
s (from gradio) (4.4.0)
Requirement already satisfied: brotli>=1.1.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from gradio) (1.1.0)
Requirement already satisfied: fastapi<1.0,>=0.115.2 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-p
ackages (from gradio) (0.116.1)
Requirement already satisfied: ffmpy in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from gr
adio) (0.6.1)
Requirement already satisfied: gradio-client==1.11.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-p
ackages (from gradio) (1.11.0)
Requirement already satisfied: groovy~=0.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (f
rom gradio) (0.1.2)
Requirement already satisfied: httpx<1.0,>=0.24.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pack
ages (from gradio) (0.27.0)
Requirement already satisfied: huggingface-hub<1.0,>=0.33.5 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.1
0/site-packages (from gradio) (0.34.3)
Requirement already satisfied: jinja2<4.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (fr
om gradio) (3.1.4)
Requirement already satisfied: markupsafe<4.0,>=2.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pa
ckages (from gradio) (2.1.5)
Requirement already satisfied: numpy<3.0,>=1.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-package
s (from gradio) (1.26.4)
Requirement already satisfied: orjson~=3.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (f
rom gradio) (3.10.6)
Requirement already satisfied: packaging in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from gradio) (25.0)
Requirement already satisfied: pandas<3.0,>=1.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packag
es (from gradio) (2.2.3)
Requirement already satisfied: pillow<12.0,>=8.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packa
ges (from gradio) (11.2.1)
Requirement already satisfied: pydantic<2.12,>=2.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pac
kages (from gradio) (2.8.2)
Requirement already satisfied: pydub in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from gr
adio) (0.25.1)
Requirement already satisfied: python-multipart>=0.0.18 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/sit
e-packages (from gradio) (0.0.20)
Requirement already satisfied: pyyaml<7.0,>=5.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packag
es (from gradio) (6.0.1)
Requirement already satisfied: ruff>=0.9.3 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (f
rom gradio) (0.12.7)
Requirement already satisfied: safehttpx<0.2.0,>=0.1.6 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site
-packages (from gradio) (0.1.6)
Requirement already satisfied: semantic-version~=2.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-p
ackages (from gradio) (2.10.0)
Requirement already satisfied: starlette<1.0,>=0.40.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-
packages (from gradio) (0.47.2)
```

```
Requirement already satisfied: tomlkit<0.14.0,>=0.12.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site
-packages (from gradio) (0.13.3)
Requirement already satisfied: typer<1.0,>=0.12 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packag
es (from gradio) (0.15.3)
Requirement already satisfied: typing-extensions~=4.0 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from gradio)
(4.13.2)
Requirement already satisfied: uvicorn>=0.14.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-package
s (from gradio) (0.35.0)
Requirement already satisfied: fsspec in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from g
radio-client==1.11.0->gradio) (2025.3.2)
Requirement already satisfied: websockets<16.0,>=10.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-
packages (from gradio-client==1.11.0->gradio) (15.0.1)
Requirement already satisfied: idna>=2.8 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (fro
m anyio<5.0,>=3.0->gradio) (3.7)
Requirement already satisfied: sniffio>=1.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from anyio<5.0,>=3.0->gradio) (1.3.1)
Requirement already satisfied: exceptiongroup>=1.0.2 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from anyio<5.
0,>=3.0->gradio) (1.2.2)
Requirement already satisfied: certifi in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from
httpx<1.0,>=0.24.1->gradio) (2025.4.26)
Requirement already satisfied: httpcore==1.* in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from httpx<1.0,>=0.24.1->gradio) (1.0.5)
Requirement already satisfied: h11<0.15,>=0.13 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-package
s (from httpcore==1.*->httpx<1.0,>=0.24.1->gradio) (0.14.0)
Requirement already satisfied: filelock in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from
huggingface-hub<1.0,>=0.33.5->gradio) (3.8.0)
Requirement already satisfied: requests in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from
huggingface-hub<1.0,>=0.33.5->gradio) (2.32.3)
Requirement already satisfied: tqdm>=4.42.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from huggingface-hub<1.0,>=0.33.5->gradio) (4.67.1)
Requirement already satisfied: hf-xet<2.0.0,>=1.1.3 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pa
ckages (from huggingface-hub<1.0,>=0.33.5->gradio) (1.1.7)
Requirement already satisfied: python-dateutil>=2.8.2 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from pandas<
3.0,>=1.0->gradio) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from pandas<3.0,>=1.0->gradio) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from pandas<3.0,>=1.0->gradio) (2025.2)
Requirement already satisfied: annotated-types>=0.4.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-
packages (from pydantic<2.12,>=2.0->gradio) (0.7.0)
Requirement already satisfied: pydantic-core==2.20.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-p
ackages (from pydantic<2.12,>=2.0->gradio) (2.20.1)
Requirement already satisfied: click>=8.0.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages
(from typer<1.0,>=0.12->gradio) (8.2.1)
Requirement already satisfied: shellingham>=1.3.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-pack
ages (from typer<1.0,>=0.12->gradio) (1.5.4)
```

```
Requirement already satisfied: rich>=10.11.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from typer<1.0,>=0.12->gradio) (14.0.0)
Requirement already satisfied: six>=1.5 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from python-dateutil>=2.8.2->pandas<3.0,>=1.0->gradio) (1.17.0)
Requirement already satisfied: markdown-it-py>=2.2.0 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from rich>=10.11.0->typer<1.0,>=0.12->gradio) (3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /Users/daisy/Library/Python/3.10/lib/python/site-packages (from rich>=10.11.0->typer<1.0,>=0.12->gradio) (2.19.1)
Requirement already satisfied: mdurl~=0.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from markdown-it-py>=2.2.0->rich>=10.11.0->typer<1.0,>=0.12->gradio) (0.1.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from requests->huggingface-hub<1.0,>=0.33.5->gradio) (3.3.2)
Requirement already satisfied: urllib3<3,>=1.21.1 in /Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages (from requests->huggingface-hub<1.0,>=0.33.5->gradio) (1.26.20)
```

In [2]:

```python
# Step 2: Load Data from Gzip Files
# The dataset consists of two .gz files, each containing 16,000 headlines.
# Clickbait headlines come from BuzzFeed, Upworthy, ViralNova, etc.
# Non-clickbait headlines are from WikiNews, NYT, The Guardian, and The Hindu.
# Source: https://github.com/bhargaviparanjape/clickbait/tree/master/dataset
# Chakraborty, A., Paranjape, B., Kakarla, S., & Ganguly, N. (2016). Stop Clickbait: Detecting and Preventing Clickbaits in Onl
# In Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp.

def load_gzip_lines(filepath):
    with gzip.open(filepath, 'rt', encoding='utf-8') as f:
        return [line.strip() for line in f if line.strip()]

clickbait = load_gzip_lines('clickbait_data.gz')
non_clickbait = load_gzip_lines('non_clickbait_data.gz')

# Combine and shuffle
data = pd.DataFrame({
    'headline': clickbait + non_clickbait,
    'label': [1]*len(clickbait) + [0]*len(non_clickbait)
})

data = data.sample(frac=1, random_state=42).reset_index(drop=True)  # Shuffle
```

In [3]:

```python
# Step 3: Preprocessing and Feature Extraction
vectorizer = TfidfVectorizer(stop_words='english', ngram_range=(1, 2), max_features=5000)
X = vectorizer.fit_transform(data['headline'])
y = data['label']

# First split: 80% train, 20% temp (to later split into validation + test)
X_train, X_temp, y_train, y_temp = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)
```

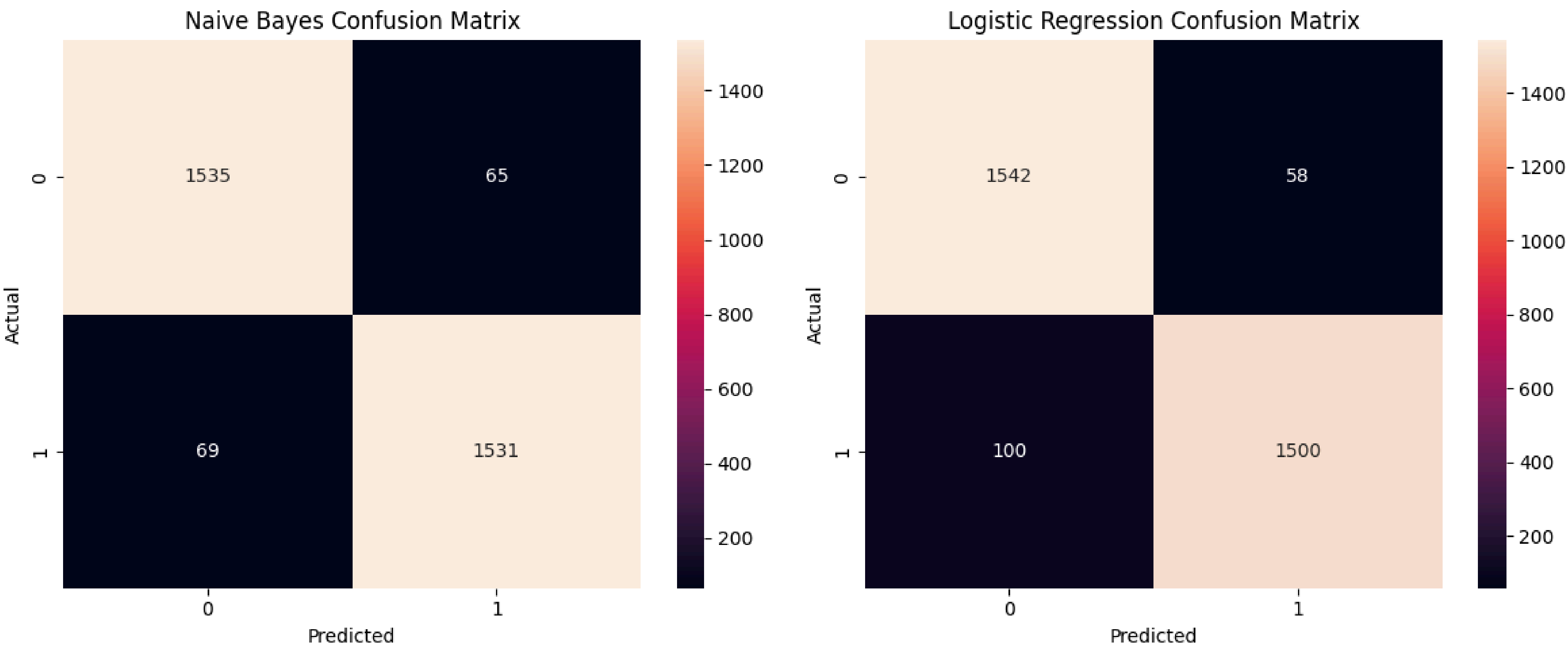
```python
# Second split: 10% validation, 10% test from remaining 20%
X_val, X_test, y_val, y_test = train_test_split(X_temp, y_temp, test_size=0.5, random_state=42, stratify=y_temp)
```

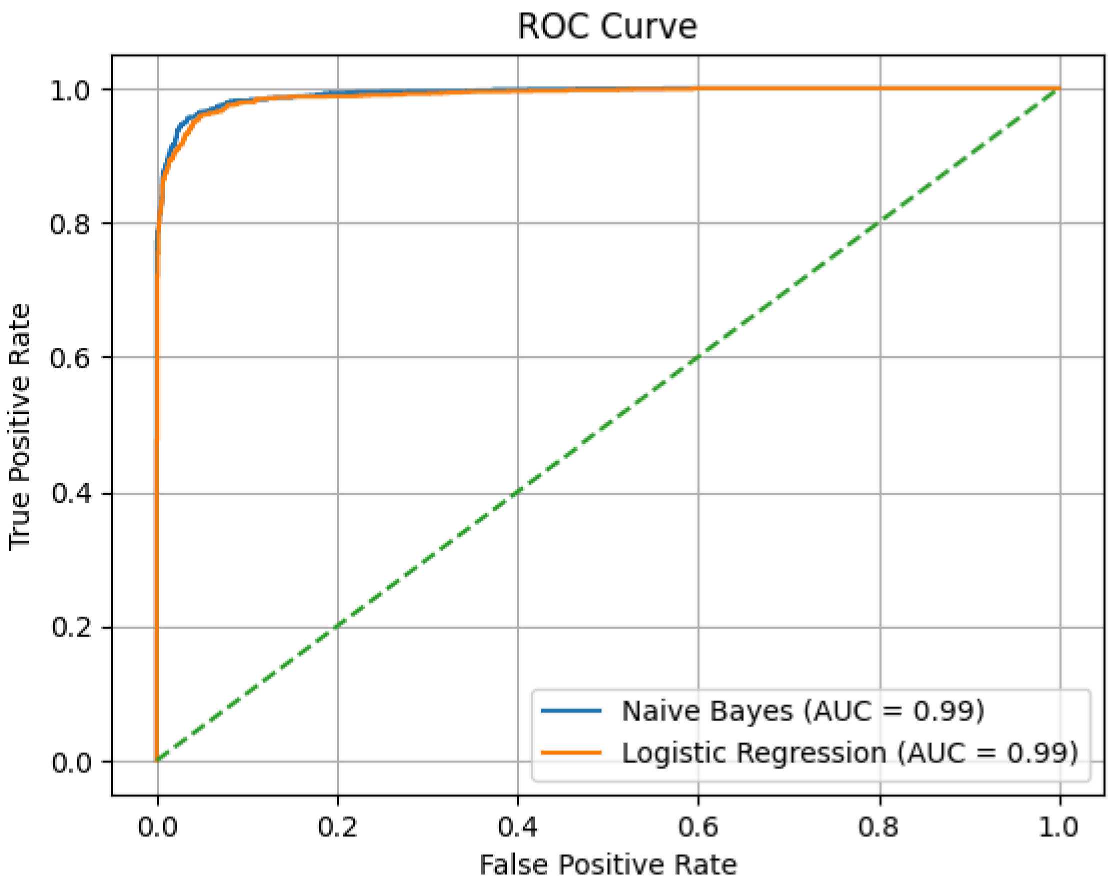
In [4]:
```python
# Step 4: Train Models and Evaluate
models = {
    'Naive Bayes': MultinomialNB(),
    'Logistic Regression': LogisticRegression(max_iter=1000)
}

for name, model in models.items():
    model.fit(X_train, y_train)
    preds = model.predict(X_test)
    print(f"\nModel: {name}")
    print(classification_report(y_test, preds))
```

```
Model: Naive Bayes
              precision    recall  f1-score   support

           0       0.96      0.96      0.96      1600
           1       0.96      0.96      0.96      1600

    accuracy                           0.96      3200
   macro avg       0.96      0.96      0.96      3200
weighted avg       0.96      0.96      0.96      3200


Model: Logistic Regression
              precision    recall  f1-score   support

           0       0.94      0.96      0.95      1600
           1       0.96      0.94      0.95      1600

    accuracy                           0.95      3200
   macro avg       0.95      0.95      0.95      3200
weighted avg       0.95      0.95      0.95      3200
```

In [5]:
```python
# Step 5: Confusion Matrix and ROC Curve
plt.figure(figsize=(12, 5))
for i, (name, model) in enumerate(models.items()):
    preds = model.predict(X_test)
    probs = model.predict_proba(X_test)[:, 1]
    fpr, tpr, _ = roc_curve(y_test, probs)
    auc_score = roc_auc_score(y_test, probs)

    plt.subplot(1, 2, i+1)
```

```
    sns.heatmap(confusion_matrix(y_test, preds), annot=True, fmt='d')
    plt.title(f"{name} Confusion Matrix")
    plt.xlabel("Predicted")
    plt.ylabel("Actual")

plt.tight_layout()
plt.show()

plt.figure()
for name, model in models.items():
    probs = model.predict_proba(X_test)[:, 1]
    fpr, tpr, _ = roc_curve(y_test, probs)
    plt.plot(fpr, tpr, label=f"{name} (AUC = {roc_auc_score(y_test, probs):.2f})")
plt.plot([0,1], [0,1], linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.legend()
plt.grid()
plt.show()
```





Confusion Matrices

Model	True Negatives (TN)	False Positives (FP)	False Negatives (FN)	True Positives (TP)
Naive Bayes	1534	66	69	1531
Logistic Regression	1541	59	102	1498

Naive Bayes has fewer false negatives → better at detecting actual clickbait.

Logistic Regression has fewer false positives → better at avoiding false alarms.

ROC Curve and AUC

Naive Bayes AUC: 0.99

Logistic Regression AUC: 0.99

Both models show excellent discriminatory ability. Their ROC curves are almost overlapping, suggesting similar performance.

Observations

- Both classifiers perform exceptionally well on this task.
- Naive Bayes slightly favors sensitivity (recall for clickbait).
- Logistic Regression slightly favors specificity (precision for non-clickbait).
- Either model is a strong baseline, and further improvements could come from tuning features or trying transformer-based embeddings.

Ablation Experiment

In [6]:

```
# Step 6: Run Ablation Experiments
results_df = run_all_ablation_experiments(data)
print(results_df)
```

```
  Vectorizer Type Stopwords Truncated  F1 Score  Accuracy  \
0           tfidf   Removed        No    0.9514    0.9514
1           tfidf   Removed       Yes    0.9187    0.9187
2           tfidf      Kept        No    0.9700    0.9700
3           tfidf      Kept       Yes    0.9550    0.9550
4             bow   Removed        No    0.9500    0.9500
5             bow   Removed       Yes    0.9175    0.9175
6             bow      Kept        No    0.9727    0.9727
7             bow      Kept       Yes    0.9603    0.9603

                        Trained Model  \
0  LogisticRegression(max_iter=1000)
1  LogisticRegression(max_iter=1000)
2  LogisticRegression(max_iter=1000)
3  LogisticRegression(max_iter=1000)
4  LogisticRegression(max_iter=1000)
5  LogisticRegression(max_iter=1000)
6  LogisticRegression(max_iter=1000)
7  LogisticRegression(max_iter=1000)

                                  Trained Vectorizer
0  TfidfVectorizer(max_features=5000, ngram_range...
1  TfidfVectorizer(max_features=5000, ngram_range...
2  TfidfVectorizer(max_features=5000, ngram_range...
3  TfidfVectorizer(max_features=5000, ngram_range...
4  CountVectorizer(max_features=5000, ngram_range...
5  CountVectorizer(max_features=5000, ngram_range...
6  CountVectorizer(max_features=5000, ngram_range...
7  CountVectorizer(max_features=5000, ngram_range...
```

Key Observations:

- The Bag-of-Words (BOW) model with no stopword removal and full headline input achieved the highest performance (F1 Score: 0.9727, Accuracy: 0.9727).
- Contrary to common practice, keeping stopwords improved classification results—likely because certain stopwords (e.g., "what", "how", "why") are strong indicators of clickbait.
- Truncating input to only the first 5 words consistently reduced performance, confirming that clickbait cues often appear later in headlines.

Sentence Embedding Experiment (Transformer-Based)

To evaluate the effectiveness of semantic features, I conducted an additional experiment using sentence embeddings generated by a pretrained transformer model (all-MiniLM-L6-v2, via sentence-transformers). These embeddings were used as input to a simple logistic regression classifier implemented in PyTorch.

In [7]:
```
# Step 7: Sentence Embedding + Logistic Regression Experiment

# Load model and encode data (stays as torch tensor)
model = SentenceTransformer('all-MiniLM-L6-v2')
X_tensor = model.encode(data['headline'].tolist(), convert_to_tensor=True)
y_tensor = torch.tensor(data['label'].tolist()).long()

# Create dataset
dataset = TensorDataset(X_tensor, y_tensor)

# Train-test split (80/20)
train_size = int(0.8 * len(dataset))
test_size = len(dataset) - train_size
train_dataset, test_dataset = random_split(dataset, [train_size, test_size])

train_loader = DataLoader(train_dataset, batch_size=32, shuffle=True)
test_loader = DataLoader(test_dataset, batch_size=32)

# Define simple logistic regression model in PyTorch
class LogisticRegressionModel(nn.Module):
    def __init__(self, input_dim):
        super().__init__()
        self.linear = nn.Linear(input_dim, 2)

    def forward(self, x):
        return self.linear(x)

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model_lr = LogisticRegressionModel(X_tensor.shape[1]).to(device)

# Training loop
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model_lr.parameters(), lr=1e-3)

for epoch in range(5):
    model_lr.train()
    total_loss = 0
    for xb, yb in train_loader:
        xb, yb = xb.to(device), yb.to(device)
        optimizer.zero_grad()
        output = model_lr(xb)
```

```python
        loss = criterion(output, yb)
        loss.backward()
        optimizer.step()
        total_loss += loss.item()
    print(f"Epoch {epoch+1}, Loss: {total_loss:.4f}")

# Evaluation
model_lr.eval()
all_preds, all_labels = [], []
with torch.no_grad():
    for xb, yb in test_loader:
        xb = xb.to(device)
        output = model_lr(xb)
        preds = output.argmax(dim=1).cpu()
        all_preds.extend(preds.tolist())
        all_labels.extend(yb.tolist())

print(classification_report(all_labels, all_preds))
```

```
Epoch 1, Loss: 296.8356
Epoch 2, Loss: 157.4728
Epoch 3, Loss: 126.0027
Epoch 4, Loss: 111.8365
Epoch 5, Loss: 103.5915
              precision    recall  f1-score   support

           0       0.95      0.96      0.95      3212
           1       0.96      0.95      0.95      3188

    accuracy                           0.95      6400
   macro avg       0.95      0.95      0.95      6400
weighted avg       0.95      0.95      0.95      6400
```

Key Observations:

- While not outperforming the best BOW configuration (F1 = 0.9727), the transformer-based model still performed very well, demonstrating the effectiveness of contextual embeddings for clickbait classification.
- The experiment highlights that simple classical models (e.g., BOW) can sometimes outperform modern embeddings on domain-specific tasks—especially when class-discriminative keywords (like “what”, “how”, “you won't believe”) are prevalent.

Implement Gradio UI Using Best Model (BOW + Logistic Regression)

In [8]:
```python
# Step 8: Implement Gradio UI Using Best Model (BOW + Logistic Regression)
# Save the best model and vectorizer from the ablation experiment
result = run_ablation_experiment(data, vectorizer_type='bow', use_stopwords=False, truncate=False)

best_model = result['Trained Model']
best_vectorizer = result['Trained Vectorizer']


with open("logreg_bow_model.pkl", "wb") as f:
    pickle.dump(best_model, f)

with open("bow_vectorizer.pkl", "wb") as f:
    pickle.dump(best_vectorizer, f)
```

In [9]:
```python
# Step 9: Create Gradio Interface for Clickbait Classification
# This will allow users to input a headline and get a prediction from the best model

# Load saved model and vectorizer
with open("bow_vectorizer.pkl", "rb") as f:
    vectorizer = pickle.load(f)

with open("logreg_bow_model.pkl", "rb") as f:
    model = pickle.load(f)

# Define prediction function
def classify_headline(headline):
    vec = vectorizer.transform([headline])
    pred = model.predict(vec)[0]
    prob = model.predict_proba(vec)[0][1]
    label = "Clickbait ✅" if pred == 1 else "Not Clickbait ❌"
    return f"{label} (Confidence: {prob:.2f})"

# Create Gradio interface
interface = gr.Interface(
    fn=classify_headline,
    inputs=gr.Textbox(lines=2, placeholder="Enter a headline..."),
    outputs=gr.Label(),
    title="Clickbait Headline Classifier (BOW + Logistic Regression)",
    description="Enter a news headline and see if it's likely clickbait based on a model trained on 32,000 labeled headlines."
)

# Launch the app
interface.launch()
```

```
* Running on local URL:  http://127.0.0.1:7860
* To create a public link, set `share=True` in `launch()`.
```

Clickbait Headline Classifier (BOW + Logistic Regression)

Enter a news headline and see if it's likely clickbait based on a model trained on 32,000 labeled headlines.

headline

Enter a headline...

output

Clear Submit Flag

Use via API · Built with Gradio · Settings

Out[9]:

Acknowledgment of AI Assistance:

This project was developed with the assistance of ChatGPT-4 to support brainstorming, coding, and refining implementation steps. Prompts used included:

- "How to conduct the ablation experiment"
- "How to improve this project such as using sentence embedding experiment to evaluate the models"
- "How to use Gradio to implement a simple UI for inputting a headline and classify it using my model?"