In [*]:
```
# Step 1: Import Libraries
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report

# Step 2: Load Dataset
# Choose categories that represent real news and non-news
news_categories = ['talk.politics.misc', 'rec.sport.baseball', 'sci.space']
non_news_categories = ['alt.atheism', 'misc.forsale', 'soc.religion.christian']

categories = news_categories + non_news_categories

# Load data
data = fetch_20newsgroups(subset='all', categories=categories, remove=('headers',

# Create labels: 1 for news, 0 for non-news
labels = [1 if data.target_names[target] in news_categories else 0 for target in

# Step 3: Preprocessing
vectorizer = TfidfVectorizer(stop_words='english', max_features=5000)
X = vectorizer.fit_transform(data.data)
y = labels

# Step 4: Train/Test Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_s

# Step 5: Train Model
model = LogisticRegression()
model.fit(X_train, y_train)

# Step 6: Evaluate
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred, target_names=["Non-News", "News"]))

# Step 7: Predict on Custom Input
def predict_news(text):
    vec = vectorizer.transform([text])
    prediction = model.predict(vec)[0]
    return "News" if prediction == 1 else "Not News"

# Example
test_text = "NASA launched a new satellite to explore the solar system."
print(f"Input: {test_text}")
print("Prediction:", predict_news(test_text))
```

In [ ]: