

AIM: Categorizing news articles into topics like sports, politics or technology.

```
-----CODE-----

from datasets import load_dataset
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score

dataset = load_dataset("ag_news", split='train[:2000]')

news_articles = dataset['text']
categories = dataset['label']

vectorizer = TfidfVectorizer(max_features=5000)
X = vectorizer.fit_transform(news_articles)

X_train, X_test, y_train, y_test = train_test_split(X, categories, test_size=0.2, random_state=42)

classifier = MultinomialNB()
classifier.fit(X_train, y_train)

y_pred = classifier.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy * 100:.2f}%")

new_articles = [
    "The stock market has seen a significant drop today amid fears of inflation.",
    "The soccer team is preparing for the World Cup qualifiers.",
    "NASA has launched a new satellite into orbit.",
]

new_X = vectorizer.transform(new_articles)
predictions = classifier.predict(new_X)

category_names = {0: 'World', 1: 'Sports', 2: 'Business', 3: 'Sci/Tech'}

for article, category in zip(new_articles, predictions):
    print(f"Article: {article}\nPredicted Category: {category_names[category]}\n")
```

```
-----OUTPUT-----

Accuracy: 78.50%
Article: The stock market has seen a significant drop today amid fears of inflation.
Predicted Category: Business

Article: The soccer team is preparing for the World Cup qualifiers.
Predicted Category: Sports

Article: NASA has launched a new satellite into orbit.
Predicted Category: Sci/Tech
```