```
import pandas as pd
import numpy as np
import re
from sklearn.feature_extraction.text import TfidfVectorizer
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud
df = pd.read_csv("arxiv_data.csv.zip")
print("Dataset Shape:", df.shape)
print("DataFrame Columns:", df.columns)
df.head()
category = "cs.AI"
df_cs = df[df['terms'].str.contains(category, na=False)] # Changed 'summaries' to 'terms'
abstracts = df_cs['summaries'].dropna().head(100).tolist()
print("Number of abstracts selected:", len(abstracts))
def clean_text(text):
    text = text.lower()
    text = re.sub(r'\d+', '', text)          # remove numbers
    text = re.sub(r'[^\w\s]', '', text)      # remove punctuation
    text = re.sub(r'\s+', ' ', text).strip()
    return text

cleaned_abstracts = [clean_text(abs) for abs in abstracts]
tokenized_abstracts = [text.split() for text in cleaned_abstracts]

print("Sample Tokenized Abstract:")
print(tokenized_abstracts[0][:20]) # Uncommented
vectorizer = TfidfVectorizer(
    stop_words='english',
    max_features=500
)

tfidf_matrix = vectorizer.fit_transform(cleaned_abstracts) # Uncommented
feature_names = vectorizer.get_feature_names_out() # Uncommented
tfidf_df = pd.DataFrame(
    tfidf_matrix.toarray(),
    columns=feature_names
) # Uncommented

print("\nTF-IDF Feature List:")
tfidf_df.head() # Uncommented
term_scores = tfidf_df.sum(axis=0).sort_values(ascending=False) # Uncommented
top_20_terms = term_scores.head(20) # Uncommented

print("\nTop 20 TF-IDF Terms:")
print(top_20_terms) # Uncommented
wordcloud = WordCloud(
    width=800,
    height=400,
    background_color='white'
).generate_from_frequencies(top_20_terms) # Uncommented

plt.figure(figsize=(12,6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title("Word Cloud of Top TF-IDF Terms (arXiv AI Abstracts)")
plt.show() # Uncommented
top_10_terms = term_scores.head(10).index # Uncommented
heatmap_data = tfidf_df.loc[:4, top_10_terms] # Uncommented

plt.figure(figsize=(12,6))
sns.heatmap(
    heatmap_data,
    annot=True,
    cmap='YlGnBu',
    fmt=".2f"
) # Uncommented
plt.title("TF-IDF Heatmap (Top 10 Terms Across 5 Documents)") # Uncommented
plt.xlabel("Terms") # Uncommented
plt.ylabel("Documents") # Uncommented
plt.show() # Uncommented
print("""
Discussion:
This experiment applies TF-IDF analysis to a subset of arXiv research abstracts
from the Artificial Intelligence domain. Text preprocessing and tokenization
helped standardize the data before feature extraction
helped standardize the data before feature extraction.

The top TF-IDF terms reveal dominant research themes in AI literature.
The word cloud provides an intuitive visualization of important keywords,
while the heatmap highlights how term importance varies across individual documents.

TF-IDF is effective for identifying significant terms in large text corpora
and is widely used in information retrieval, topic modeling, and document analysis.
""")
```

```
Dataset Shape: (51774, 3)
DataFrame Columns: Index(['titles', 'summaries', 'terms'], dtype='object')
Number of abstracts selected: 100
Sample Tokenized Abstract:
['the', 'recent', 'advancements', 'in', 'artificial', 'intelligence', 'ai', 'combined', 'with', 'the', 'extensive', 'amount', 'o

TF-IDF Feature List:

Top 20 TF-IDF Terms:
segmentation    8.706737
image           6.945956
learning        5.246278
images          4.662291
method          4.369352
network         4.319956
approach        4.127425
model           3.946440
results         3.877965
data            3.871289
deep            3.514421
proposed        3.474315
medical         3.392881
training        3.373665
based           3.297902
semantic        3.089234
using           2.968089
methods         2.944290
performance     2.905243
graph           2.890783
dtype: float64
```

Word Cloud of Top TF-IDF Terms (arXiv AI Abstracts)

network
learning
semantic
methods
using
model
performance
segmentation
training
results
proposed
deep
image
images
medical
based
graph
approach