cwy_ 3

May 28, 2023

[ ]:
```
import os
%env OPENBLAS_NUM_THREADS=1
import nltk
from sklearn.feature_extraction.text import TfidfVectorizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score, calinski_harabasz_score
import matplotlib.pyplot as plt
from sklearn.manifold import TSNE

nltk.download('punkt')
nltk.download('stopwords')
```

```
env: OPENBLAS_NUM_THREADS=1
```

```
[nltk_data] Downloading package punkt to /home/cwy/nltk_data…
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /home/cwy/nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
```

[ ]:
```
True
```

[ ]:
```
def preprocess(raw_text):
    raw_text = raw_text.lower()  #
    words = word_tokenize(raw_text)  #
    words = [word for word in words if word.isalpha()]  #
    stop_words = set(stopwords.words('english'))  #
    words = [word for word in words if word not in stop_words]  #
    clean_text = ' '.join(words)  #
    return clean_text

def read_data():
    texts=[]
    for f1 in os.listdir("../data/20news"):
        for f2 in os.listdir("../data/20news/"+f1):
            fp="../data/20news/"+f1+"/"+f2
```

```
            try:
                with open(fp, mode="r", encoding="utf-8") as f:
                    text="".join(f.readlines())
                    texts.append(text)
            except:
                pass
    return texts
```

[ ]:
```
#
raw_texts=read_data()
preprocessed_texts = [preprocess(raw_text) for raw_text in raw_texts]
print(f"count of texts: {len(preprocessed_texts)}")
```

```
count of texts: 18756
```

[ ]:
```
#
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(preprocessed_texts)

#
k = 20
kmeans = KMeans(n_clusters=k, random_state=42)
kmeans.fit(X)
```

```
/home/cwy/miniconda3/envs/cwy_py39/lib/python3.9/site-
packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(
```

[ ]:
```
KMeans(n_clusters=20, random_state=42)
```

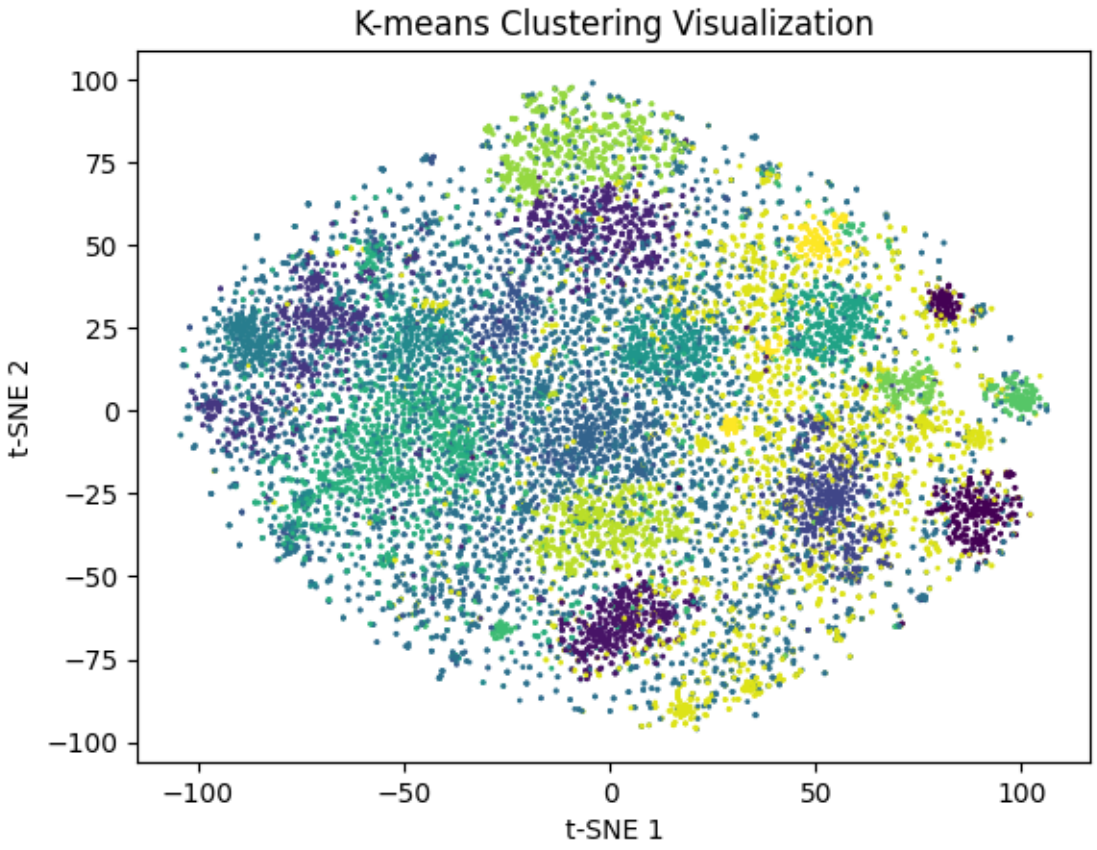
[ ]:
```
#
silhouette_coefficient = silhouette_score(X, kmeans.labels_)
ch_index = calinski_harabasz_score(X.toarray(), kmeans.labels_)
print("Silhouette Coefficient:", silhouette_coefficient)
print("Calinski-Harabasz Index:", ch_index)
```

```
Silhouette Coefficient: 0.005702419785734724
Calinski-Harabasz Index: 23.382172419798053
```

[ ]:
```
#
X_embedded = TSNE(n_components=2, random_state=42).fit_transform(X.toarray())
# plt.savefig("result.png")
```

[ ]:
```
plt.scatter(X_embedded[:, 0], X_embedded[:, 1], c=kmeans.labels_,␣
 ↪cmap='viridis', s=1)
```

```
plt.title('K-means Clustering Visualization')
plt.xlabel('t-SNE 1')
plt.ylabel('t-SNE 2')
plt.show()
```



[ ]: