Understanding Content-Based

What is Content-Based Filtering?

Content-Based Filtering is a recommendation system technique that suggests items to users based on the characteristics of items and the preferences of the user. Rather than relying on user ratings or interactions (as in collaborative filtering), content-based systems analyze item features such as keywords, descriptions, and metadata. This method matches the features of items a user has liked in the past to new items that share similar attributes. It is particularly effective when there is little user interaction data.

Code Cell 1

Code:

```
!pip install sentence_transformers
```

Explanation:

- - !pip install sentence_transformers

Code Cell 2

Code:

```
import pandas
data =
pandas.read_csv("https://coding.co.ke/datasets/shared_articles.csv
")
data.head(50)
```

Explanation:

- - Imports the pandas library, which is essential for data analysis and manipulation.
- - Loads a CSV file from the provided URL into a pandas DataFrame named `data`.
- - Displays the first 50 rows of the DataFrame to get an overview of the dataset.

Code Cell 3

Code:

```
# We will only work with text colm
data = data.head(50)
X = data['text'].values
#X
```

Explanation:

- - # We will only work with text colm
- - Displays the first 50 rows of the DataFrame to get an overview of the dataset.
- - X = data['text'].values
- - #X

Code Cell 4

Code:

```
from sentence_transformers import SentenceTransformer
model = SentenceTransformer('distilbert-base-nli-stsb-mean-tokens')
embeddings = model.encode(X, show_progress_bar = True)
embeddings
```

Explanation:

- - from sentence_transformers import SentenceTransformer
- - model = SentenceTransformer('distilbert-base-nli-stsb-mean-tokens')
- - embeddings = model.encode(X, show_progress_bar = True)
- - embeddings

Code Cell 5

Code:

```
df = pandas.DataFrame(embeddings)
df
# machinelearningplus.com/nlp/cosine-similarity
```

Explanation:

- - df = pandas.DataFrame(embeddings)
- - df
- - # machinelearningplus.com/nlp/cosine-similarity

Code Cell 6

Code:

```
from sklearn.metrics.pairwise import cosine_similarity
cosine_sim_data = cosine_similarity(df)

# pass above to be like a dataframe
cosine_data = pandas.DataFrame(cosine_sim_data)
cosine_data
```

Explanation:

- - from sklearn.metrics.pairwise import cosine_similarity
- - cosine_sim_data = cosine_similarity(df)
- -
- - # pass above to be like a dataframe
- - cosine_data = pandas.DataFrame(cosine_sim_data)
- - cosine_data

Code Cell 7

Code:

```
# Find similarity
def recommendations(newsid):
   index_recomendations =
cosine_data.loc[newsid].sort_values(ascending=False).index.tolist()[
1:10]
   print("Similar Indexes ", index_recomendations)
   news_titles = data['title'].loc[index_recomendations].values
   print("Similar Articles", news_titles)


recommendations(3)
```

Explanation:

- - # Find similarity
- - def recommendations(newsid):
- - index_recomendations = cosine_data.loc[newsid].sort_values(ascending=False).index.tolist()[1:10]
- - print("Similar Indexes ", index_recomendations)
- - news_titles = data['title'].loc[index_recomendations].values
- - print("Similar Articles", news_titles)
- -
- -
- - recommendations(3)