Allen Hoskins: DS7337 - Natural Language Processing

In [ ]:

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.feature_extraction.text import TfidfVectorizer
from scipy import spatial
from sentence_transformers import SentenceTransformer, util
import warnings
warnings.filterwarnings('ignore')
```

```
/Users/allen/virtualenvs/NLP/lib/python3.9/site-packages/tqdm/auto.py:22: TqdmWarning: IProgress not found. Please update jupyter and ipywidgets. See https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
```

In [ ]:

```
p1_c1 = '50 Inch Class H6570G 4K Ultra HD Android Smart TV with Alexa Compatibility 2.5" 2020 Model Black Silver White HDR LED'
p1_c2 = 'Hisense H6570G'
p2_c1 = 'QN75Q90TAFXZA crystal 2.5" Quantum LCD'
p2_c2 = 'Samsung crystal UN55TU8000FXZA QLED'
p3_c1 = 'EGLF2 50 Ultra Full Motion Articulating TV Wall Mount Bracket swivel full'
p3_c2 = 'VIZIO EGLF2'
```

In [ ]:

```
tfidf_data = {
            'Site 1': [p1_c1,p2_c1,p3_c1],
            'Site 2': [p1_c2,p2_c2,p3_c2]
            }

tfidf_data = pd.DataFrame(tfidf_data)
```

In [ ]:

```
#initialize Sentence Transformer
model = SentenceTransformer('sentence-transformers/all-MiniLM-L6-v2')

#create list of sentences
sentences1 = tfidf_data['Site 1'].tolist()
sentences2 = tfidf_data['Site 2'].tolist()

#create new combined column
tfidf_data['ab'] = tfidf_data.apply(lambda x : x['Site 1'] + ' ' + x['Site 2'], axis=1)

#init vectorizers
clf_tfidf = TfidfVectorizer()
clf_cvec = CountVectorizer()

#fit vectorizers
clf_tfidf.fit(tfidf_data['ab'])
clf_cvec.fit(tfidf_data['ab'])

#transform fitted vectorizers
cntvec_a = clf_cvec.transform(tfidf_data['Site 1']).todense()
cntvec_b = clf_cvec.transform(tfidf_data['Site 2']).todense()

tfidf_a = clf_tfidf.transform(tfidf_data['Site 1']).todense()
tfidf_b = clf_tfidf.transform(tfidf_data['Site 2']).todense()

#Compute embedding for both lists
embeddings1 = model.encode(sentences1, convert_to_tensor=True)
embeddings2 = model.encode(sentences2, convert_to_tensor=True)

#Compute cosine-similarities
cosine_scores = util.cos_sim(embeddings1, embeddings2)

output =[]
for i in range(len(tfidf_a)):
    output.append(
        {
            'Site 1': sentences1[i],
            'Site 2': sentences2[i],
            'CountVectorizer Cosine Score': cosine_similarity(cntvec_a[i],cntvec_b[i])[0][0],
            'TFIDF Cosine Score': cosine_similarity(tfidf_a[i],tfidf_b[i])[0][0],
            'Sentence Transformer (sBERT) Cosine Score': cosine_scores[i][i].numpy()
        }
    )
fin_cosine= pd.DataFrame(output)
fin_cosine.head()
```

Out[ ]:

	Site 1	Site 2	CountVectorizer Cosine Score	TFIDF Cosine Score	Sentence Transformer (sBERT) Cosine Score
0	50 Inch Class H6570G 4K Ultra HD Android Smart...	Hisense H6570G	0.158114	0.163364	0.41582918
1	QN75Q90TAFXZA crystal 2.5" Quantum LCD	Samsung crystal UN55TU8000FXZA QLED	0.250000	0.250000	0.583774
2	EGLF2 50 Ultra Full Motion Articulating TV Wal...	VIZIO EGLF2	0.188982	0.198145	0.3856305

Jaccard

In [ ]:

```
p1_c1 = '50 Inch Class H6570G 4K Ultra HD Android Smart TV with Alexa Compatibility 2.5" 2020 Model Black Silver White HDR LED'
p1_c2 = 'Hisense H6570G'
p2_c1 = 'QN75Q90TAFXZA crystal 2.5" Quantum LCD'
p2_c2 = 'Samsung crystal UN55TU8000FXZA QLED'
p3_c1 = 'EGLF2 50 Ultra Full Motion Articulating TV Wall Mount Bracket swivel full'
p3_c2 = 'VIZIO EGLF2'
```

In [ ]:

```
p1_c1 = set(p1_c1.split())
p1_c2 = set(p1_c2.split())
p2_c1 = set(p2_c1.split())
p2_c2 = set(p2_c2.split())
p3_c1 = set(p3_c1.split())
p3_c2 = set(p3_c2.split())
```

In [ ]:

```
def jac(x:set,y:set):
    shared = x.intersection(y)
    return len(shared)/len(x.union(y))
```

In [ ]:

```
jac_data = {
            'Site 1': [p1_c1,p2_c1,p3_c1],
            'Site 2': [p1_c2,p2_c2,p3_c2],
            'Jaccard Score': [jac(p1_c1,p1_c2),jac(p2_c1,p2_c2),jac(p3_c1,p3_c2)]
            }

pd.DataFrame(jac_data)
```

Out[ ]:

	Site 1	Site 2	Jaccard Score
0	{Android, with, Smart, HD, 50, 2.5", LED, Blac...	{Hisense, H6570G}	0.045455
1	{crystal, Quantum, LCD, 2.5", QN75Q90TAFXZA}	{UN55TU8000FXZA, QLED, crystal, Samsung}	0.125000
2	{EGLF2, Motion, Mount, full, TV, Bracket, Full...	{VIZIO, EGLF2}	0.076923

Combine DataFrames

In [ ]:

```
final_data = {
    'Product Title 1 (Site 1)': fin_cosine['Site 1'],
    'Product Title 2 (Site 2)': fin_cosine['Site 2'],
    'CountVectorizer Cosine Score': fin_cosine['CountVectorizer Cosine Score'],
    'TFIDF Cosine Score': fin_cosine['TFIDF Cosine Score'],
    'Sentence Transformer (sBERT) Cosine Score': fin_cosine['Sentence Transformer (sBERT) Cosine Score'],
    'Jaccard Score': jac_data['Jaccard Score']
}

final_data = pd.DataFrame(final_data)
final_data.head()
```

Out[ ]:

	Product Title 1 (Site 1)	Product Title 2 (Site 2)	CountVectorizer Cosine Score	TFIDF Cosine Score	Sentence Transformer (sBERT) Cosine Score	Jaccard Score
0	50 Inch Class H6570G 4K Ultra HD Android Smart...	Hisense H6570G	0.158114	0.163364	0.41582918	0.045455
1	QN75Q90TAFXZA crystal 2.5" Quantum LCD	Samsung crystal UN55TU8000FXZA QLED	0.250000	0.250000	0.583774	0.125000
2	EGLF2 50 Ultra Full Motion Articulating TV Wal...	VIZIO EGLF2	0.188982	0.198145	0.3856305	0.076923