NAME : RETHINAGIRI G

ROLL NO : 225229130

COURSE TITLE : Natural Language Pre-processing Lab

Lab : 03 Computing Document Similarity using VSM

Ex-1 : Print TFIDF values

```
In [1]: from sklearn.feature_extraction.text import TfidfVectorizer as tfv
        import pandas as pan
```

```
In [2]: docs=["good movie","not a good movie","did not like","i like it","good one"]
```

```
In [3]: tfidf=tfv(min_df=2,max_df=0.5,ngram_range=(1,2))
        features=tfidf.fit_transform(docs)
        print(features)
```

```
  (0, 0)	0.7071067811865476
  (0, 2)	0.7071067811865476
  (1, 3)	0.5773502691896257
  (1, 0)	0.5773502691896257
  (1, 2)	0.5773502691896257
  (2, 1)	0.7071067811865476
  (2, 3)	0.7071067811865476
  (3, 1)	1.0
```

```
In [4]: df=pan.DataFrame(features.todense(),columns=tfidf.get_feature_names())
        print(df)
```

```
C:\Users\user\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:87: FutureWarning: Function get_feature_names is deprecated; get_featur
e_names is deprecated in 1.0 and will be removed in 1.2. Please use get_feature_names_out instead.
  warnings.warn(msg, category=FutureWarning)

   good movie      like     movie       not
0    0.707107  0.000000  0.707107  0.000000
1    0.577350  0.000000  0.577350  0.577350
2    0.000000  0.707107  0.000000  0.707107
3    0.000000  1.000000  0.000000  0.000000
4    0.000000  0.000000  0.000000  0.000000
```

Ex-2:

```
In [5]: tfidf=tfv(min_df=1,max_df=0.75,ngram_range=(1,2))
        features=tfidf.fit_transform(docs)
        print(features)
```

```
  (0, 3)	0.6098184563533858
  (0, 8)	0.6098184563533858
  (0, 2)	0.5062044059286201
  (1, 10)	0.5422255279709232
  (1, 9)	0.4374641418373903
  (1, 3)	0.4374641418373903
  (1, 8)	0.4374641418373903
  (1, 2)	0.36313475547801904
  (2, 11)	0.4821401170833009
  (2, 1)	0.4821401170833009
  (2, 6)	0.3889876106617681
  (2, 0)	0.4821401170833009
  (2, 9)	0.3889876106617681
  (3, 7)	0.6141889663426562
  (3, 5)	0.6141889663426562
  (3, 6)	0.49552379079705033
  (4, 4)	0.6390704413963749
  (4, 12)	0.6390704413963749
  (4, 2)	0.42799292268317357
```

In [6]:
```
df=pan.DataFrame(features.todense(),columns=tfidf.get_feature_names())
print(df)
```

```
       did  did not      good  good movie  good one        it      like  \
0  0.00000  0.00000  0.506204    0.609818   0.00000  0.000000  0.000000
1  0.00000  0.00000  0.363135    0.437464   0.00000  0.000000  0.000000
2  0.48214  0.48214  0.000000    0.000000   0.00000  0.000000  0.388988
3  0.00000  0.00000  0.000000    0.000000   0.00000  0.614189  0.495524
4  0.00000  0.00000  0.427993    0.000000   0.63907  0.000000  0.000000

    like it     movie       not  not good  not like      one
0  0.000000  0.609818  0.000000  0.000000   0.00000  0.00000
1  0.000000  0.437464  0.437464  0.542226   0.00000  0.00000
2  0.000000  0.000000  0.388988  0.000000   0.48214  0.00000
3  0.614189  0.000000  0.000000  0.000000   0.00000  0.00000
4  0.000000  0.000000  0.000000  0.000000   0.00000  0.63907

C:\Users\user\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:87: FutureWarning: Function get_feature_names is deprecated; get_featur
e_names is deprecated in 1.0 and will be removed in 1.2. Please use get_feature_names_out instead.
  warnings.warn(msg, category=FutureWarning)
```

In [7]:
```
tfidfi=tfv(min_df=2,max_df=0.5,ngram_range=(2,1))
features=tfidf.fit_transform(docs)
print(features)
```

```
  (0, 3)	0.6098184563533858
  (0, 8)	0.6098184563533858
  (0, 2)	0.5062044059286201
  (1, 10)	0.5422255279709232
  (1, 9)	0.4374641418373903
  (1, 3)	0.4374641418373903
  (1, 8)	0.4374641418373903
  (1, 2)	0.36313475547801904
  (2, 11)	0.4821401170833009
  (2, 1)	0.4821401170833009
  (2, 6)	0.3889876106617681
  (2, 0)	0.4821401170833009
  (2, 9)	0.3889876106617681
  (3, 7)	0.6141889663426562
  (3, 5)	0.6141889663426562
  (3, 6)	0.49552379079705033
  (4, 4)	0.6390704413963749
  (4, 12)	0.6390704413963749
  (4, 2)	0.42799292268317357
```

In [8]:
```
df3=pan.DataFrame(features.todense(),columns=tfidf.get_feature_names())
print(df3)
```

```
       did  did not      good  good movie  good one        it      like  \
0  0.00000  0.00000  0.506204    0.609818   0.00000  0.000000  0.000000
1  0.00000  0.00000  0.363135    0.437464   0.00000  0.000000  0.000000
2  0.48214  0.48214  0.000000    0.000000   0.00000  0.000000  0.388988
3  0.00000  0.00000  0.000000    0.000000   0.00000  0.614189  0.495524
4  0.00000  0.00000  0.427993    0.000000   0.63907  0.000000  0.000000

    like it     movie       not  not good  not like      one
0  0.000000  0.609818  0.000000  0.000000   0.00000  0.00000
1  0.000000  0.437464  0.437464  0.542226   0.00000  0.00000
2  0.000000  0.000000  0.388988  0.000000   0.48214  0.00000
3  0.614189  0.000000  0.000000  0.000000   0.00000  0.00000
4  0.000000  0.000000  0.000000  0.000000   0.00000  0.63907

C:\Users\user\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:87: FutureWarning: Function get_feature_names is deprecated; get_featur
e_names is deprecated in 1.0 and will be removed in 1.2. Please use get_feature_names_out instead.
  warnings.warn(msg, category=FutureWarning)
```

Ex-3 : Compute Cosine Similarity Between 2 Documents

In [9]:
```
from sklearn.metrics.pairwise import linear_kernel as lk
```

In [10]:
```
doc1=features[0:1]
```

In [11]:
```
doc2=features[1:2]
```

In [12]:
```
score=lk(doc1,doc2)
print(score)
```

```
[[0.71736783]]
```

In [13]:
```
scores=lk(doc1,features)
print(scores)
```

```
[[1.         0.71736783 0.         0.         0.2166519 ]]
```

In [14]:
```
query="I like this good movie"
qfeature=tfidf.transform([query])
scores2=lk(doc1,features)
print(scores2)
```

```
[[1.         0.71736783 0.         0.         0.2166519 ]]
```

Ex-4 : Find Top-N similar documents

Q-1

In [15]:
```
docs2=["the house had a tiny little mouse",
       "the cat saw the mouse",
      "the mouse ran away from the house",
      "the cat finally ate the mouse",
      "the end of the mouse story"]
```

In [16]:
```
tfidfi=tfv(min_df=2,max_df=0.5,ngram_range=(1,2))
f2=tfidf.fit_transform(docs2)
print(f2)
```

```
  (0, 18)	0.34706676322953556
  (0, 32)	0.34706676322953556
  (0, 14)	0.34706676322953556
  (0, 16)	0.34706676322953556
  (0, 30)	0.28001127926354535
  (0, 17)	0.34706676322953556
  (0, 31)	0.34706676322953556
  (0, 13)	0.34706676322953556
  (0, 15)	0.28001127926354535
  (1, 26)	0.4821401170833009
  (1, 6)	0.4821401170833009
  (1, 28)	0.3889876106617681
  (1, 25)	0.4821401170833009
  (1, 4)	0.3889876106617681
  (2, 12)	0.34706676322953556
  (2, 3)	0.34706676322953556
  (2, 24)	0.34706676322953556
  (2, 19)	0.34706676322953556
  (2, 11)	0.34706676322953556
  (2, 2)	0.34706676322953556
  (2, 23)	0.34706676322953556
  (2, 30)	0.28001127926354535
  (2, 15)	0.28001127926354535
  (3, 1)	0.3983516165374428
  (3, 10)	0.3983516165374428
  (3, 5)	0.3983516165374428
  (3, 0)	0.3983516165374428
  (3, 9)	0.3983516165374428
  (3, 28)	0.32138757599667
  (3, 4)	0.32138757599667
  (4, 20)	0.3779644730092272
  (4, 22)	0.3779644730092272
  (4, 8)	0.3779644730092272
  (4, 29)	0.3779644730092272
  (4, 27)	0.3779644730092272
  (4, 21)	0.3779644730092272
  (4, 7)	0.3779644730092272
```

Q-2

In [20]:
```
t1=f2[2:3]
print(t1)
```

```
  (0, 12)	0.34706676322953556
  (0, 3)	0.34706676322953556
  (0, 24)	0.34706676322953556
  (0, 19)	0.34706676322953556
  (0, 11)	0.34706676322953556
  (0, 2)	0.34706676322953556
  (0, 23)	0.34706676322953556
  (0, 30)	0.28001127926354535
  (0, 15)	0.28001127926354535
```

In [21]:
```
simi=lk(t1,f2)
print(simi)
```

```
[[0.15681263 0.         1.         0.         0.        ]]
```

Q-3

In [22]:

```
t2=f2[0:2]
simi2=lk(t2,t1)
print(simi2)
```

```
[[0.15681263]
 [0.        ]]
```