TDM 직접구현

[1]
```
docs = ['동물원 코끼리',
        '동물원 원숭이 바나나',
        '엄마 코끼리 아기 코끼리',
        '원숭이 바나나 코끼리 바나나']
```

1. 토큰화 : 공백으로 토큰화 진행

[2]
```
doc_ls=[]
for doc in docs:
  doc_ls.append(doc.split(' '))
doc_ls

# doc_ls = [doc.split() for doc in docs] # 위의 for문을 한줄로 쓰는 경우
```

```
[['동물원', '코끼리'],
 ['동물원', '원숭이', '바나나'],
 ['엄마', '코끼리', '아기', '코끼리'],
 ['원숭이', '바나나', '코끼리', '바나나']]
```

2. 유니크한 토큰 사전 구하기

2-1. 빈 딕셔너리 생성

[3]
```
from collections import defaultdict

word2id = defaultdict(lambda : len(word2id))
word2id
```

```
defaultdict(<function __main__.<lambda>()>, {})
```

2-2. 위에서 생성한 빈 딕셔너리에 유니크한 토큰 넣기

[4]
```
for doc in doc_ls:
    for token in doc:
        word2id[token]
        print(token)
        print('\t',word2id)
word2id

# [word2id[token] for doc in doc_ls for token in doc ] # 위의 for문을 한줄로 쓰는 경우
```

[4]
```
동물원
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0})
코끼리
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1})
동물원
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1})
원숭이
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2})
바나나
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3})
엄마
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4})
코끼리
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4})
아기
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
코끼리
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
원숭이
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
바나나
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
코끼리
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
바나나
        defaultdict(<function <lambda> at 0x7cc25fe4ca60>, {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
defaultdict(<function __main__.<lambda>()>,
            {'동물원': 0, '코끼리': 1, '원숭이': 2, '바나나': 3, '엄마': 4, '아기': 5})
```

[5]
```python
import numpy as np

TDM = np.zeros((len(word2id), len(doc_ls)), dtype=int)

for i, doc in enumerate(doc_ls):
  print(doc)
  for token in doc:
    TDM[word2id[token], i] += 1 # 해당 토큰의 위치(column)
    print(token)
    print(TDM)
    print('\t')
```

```
['동물원', '코끼리']
동물원
[[1 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]]

코끼리
[[1 0 0 0]
 [1 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]]

['동물원', '원숭이', '바나나']
동물원
[[1 1 0 0]
 [1 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]
```

[5]
```
원숭이
[[1 1 0 0]
 [1 0 0 0]
 [0 1 0 0]
 [0 0 0 0]
 [0 0 0 0]
 [0 0 0 0]]

바나나
[[1 1 0 0]
 [1 0 0 0]
 [0 1 0 0]
 [0 1 0 0]
 [0 0 0 0]
 [0 0 0 0]]

['엄마', '코끼리', '아기', '코끼리']
엄마
[[1 1 0 0]
 [1 0 0 0]
 [0 1 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 0]]

코끼리
[[1 1 0 0]
 [1 0 1 0]
 [0 1 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 0]]

아기
[[1 1 0 0]
 [1 0 1 0]
 [0 1 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 1 0]]

코끼리
[[1 1 0 0]
 [1 0 2 0]
 [0 1 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 1 0]]

['원숭이', '바나나', '코끼리', '바나나']
원숭이
[[1 1 0 0]
 [1 0 2 0]
 [0 1 0 1]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 1 0]]
```

[6]
```
TDM
# 행렬로 표기(BOW와 차이점: BOW는 1차원 배열)
```

```
[6] array([[1, 1, 0, 0],
           [1, 0, 2, 1],
           [0, 1, 0, 1],
           [0, 1, 0, 2],
           [0, 0, 1, 0],
           [0, 0, 1, 0]])
```

```
[7]  1 import pandas as pd
     2
     3 doc_names = ['문서'+ str(i) for i in range(len(doc_ls))]
     4 print('doc_names',doc_names)
     5 sorted_vocab = sorted((value, key) for key, value in word2id.items())
     6 vocab = [v[1] for v in sorted_vocab]
     7 df_TDM = pd.DataFrame(TDM, columns=doc_names)
     8 df_TDM['단어'] = vocab
     9 df_TDM.set_index('단어')
    10
```

```
doc_names ['문서0', '문서1', '문서2', '문서3']
```

단어	문서0	문서1	문서2	문서3
동물원	1	1	0	0
코끼리	1	0	2	1
원숭이	0	1	0	1
바나나	0	1	0	2
엄마	0	0	1	0
아기	0	0	1	0

sklearn

```
[8]  1 docs = ['동물원 코끼리',
     2         '동물원 원숭이 바나나',
     3         '엄마 코끼리 아기 코끼리',
     4         '원숭이 바나나 코끼리 바나나']
```

```
[9]  1 from sklearn.feature_extraction.text import CountVectorizer
     2
     3 count_vect = CountVectorizer() # 참고 sklearn은 DTM으로 만들어지게 설정되어 있음.
     4 DTM = count_vect.fit_transform(docs)
     5 DTM.toarray()
```

```
array([[1, 0, 0, 0, 0, 1],
       [1, 1, 0, 0, 1, 0],
       [0, 0, 1, 1, 0, 2],
       [0, 2, 0, 0, 1, 1]])
```

[10]
```
DTM.toarray().T
```

```
array([[1, 1, 0, 0],
       [0, 1, 0, 2],
       [0, 0, 1, 0],
       [0, 0, 1, 0],
       [0, 1, 0, 1],
       [1, 0, 2, 1]])
```

```
import pandas as pd

doc_names = ['문서'+ str(i) for i in range(len(doc_ls))]
vocab = count_vect.get_feature_names_out()
print(vocab)
df_TDM = pd.DataFrame(DTM.toarray().T, columns=doc_names)
df_TDM['단어'] = vocab
df_TDM.set_index('단어')
```

```
['동물원' '바나나' '아기' '엄마' '원숭이' '코끼리']
```

단어	문서0	문서1	문서2	문서3
동물원	1	1	0	0
바나나	0	1	0	2
아기	0	0	1	0
엄마	0	0	1	0
원숭이	0	1	0	1
코끼리	1	0	2	1