→ HW3

B0928022-杜云驊

Colab-link: https://colab.research.google.com/drive/1cfF-B5GfiacrxDcYcYLQQ0Cke3MPXIUq? usp=sharing

```
1 class Movie:
       def __init__(self, name, intro, label) -> None:
 2
 3
           self_name = name
 4
           self.intro = intro
 5
           self.label = label
 1 import json
 2 path = "hw3.json"
 3 movie data = []
 4 \text{ movies} = []
 5
 6 with open(path) as f:
       movie_data = json.load(f)
 8
 9 for movie in movie_data:
10
       try:
           label = movie['label'][0]
11
12
       except:
           label = 'NA'
13
       m = Movie(movie['cname'], movie['intro'], label)
14
15
       movies.append(m)
16
17 print(movie_data[0])
18 print(movies[0].label)
   {'doc_id': 1, 'cname': '一世狂野', 'ename': 'Blow', 'pagerank': 1.1489996653
   劇情
```

```
1 with open("inverted.json") as f:
2    inverted_index = json.load(f)
3
4 key_map = {}
5 x = 0
6
7 for key in inverted_index.keys():
8    if not key in key_map.keys():
9        key_map[key] = x
10        x += 1
```

```
1 import pandas as pd
 2 import jieba
 3
 4 dataframe = pd.DataFrame()
 5i = 0
 6 for movie in movies:
      seg_list = []
 7
 8
      for seg in jieba.cut_for_search(movie.name):
           seg list.append(key map[seg])
 9
      for seg in jieba.cut_for_search(movie.intro):
10
           seg list.append(key map[seg])
11
      df = pd.DataFrame({'name':[movie.name], 'label':[movie.label]})
12
      df = pd.concat([df, pd.DataFrame([seg list])], axis=1)
13
      dataframe = pd.concat([dataframe, df], ignore_index=True)
14
15
      i += 1
      if i == 6000:
16
17
          break
18
19 dataframe.fillna(-1, inplace=True)
20
```

21 dataframe

Building prefix dict from the default dictionary ...

Loading model from cache /var/folders/g6/4x15mmjd04df29tnj_rj7p1h0000gn/T/j

Prefix dict has been built successfully.

Loading model cost 0.766 seconds.

	name	label	0	1	2	3	4	5	6	
0	一世 狂野	劇情	0	1	2	3.0	4.0	5.0	6.0	7
1	玩命 關頭	動作	169	170	171	172.0	35.0	173.0	174.0	175
2	戰雲 密佈	動作	339	340	341	342.0	343.0	344.0	9.0	345
3	騎士 風雲 錄	動作	448	449	450	451.0	452.0	453.0	9.0	454
4	金法 尤物	喜劇	571	572	6	573.0	574.0	575.0	389.0	573
5995	少年 (2017)	動作	1397	488	89504	840.0	4910.0	693.0	7550.0	396
	兒子									

```
1 from sklearn.model_selection import train_test_split
 2 from sklearn.neighbors import KNeighborsClassifier
 3
 4 dataframe.columns = dataframe.columns.astype(str)
 5
 6 X data = dataframe.drop(columns=["name", "label"])
 7 y data = dataframe["label"]
 9 X_data
10
11 X_train, X_test, y_train, y_test = train_test_split(X_data, y_data,
12 clf = KNeighborsClassifier()
13 clf.fit(X train,y train)
14 clf.score(X_test,y_test)
   0.3592233009708738
 1 from sklearn import tree
 2
 3 clf = tree.DecisionTreeClassifier()
 4 clf.fit(X_train,y_train)
 5 clf.score(X_test,y_test)
   0.34951456310679613
```