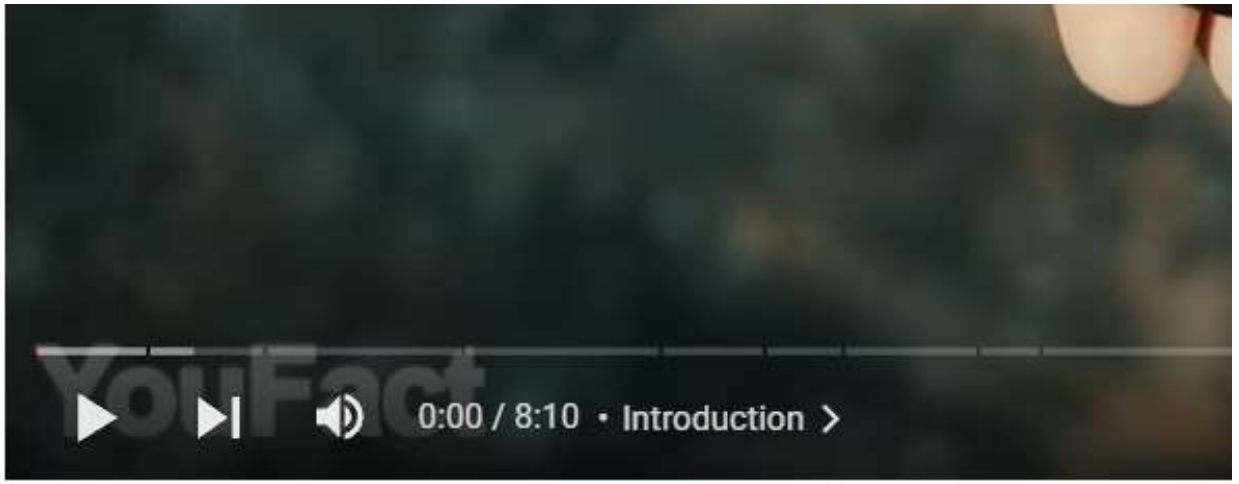
유튜브 태그 생성기

발표자 허성원

2022.11.14

INDEX

01 개요

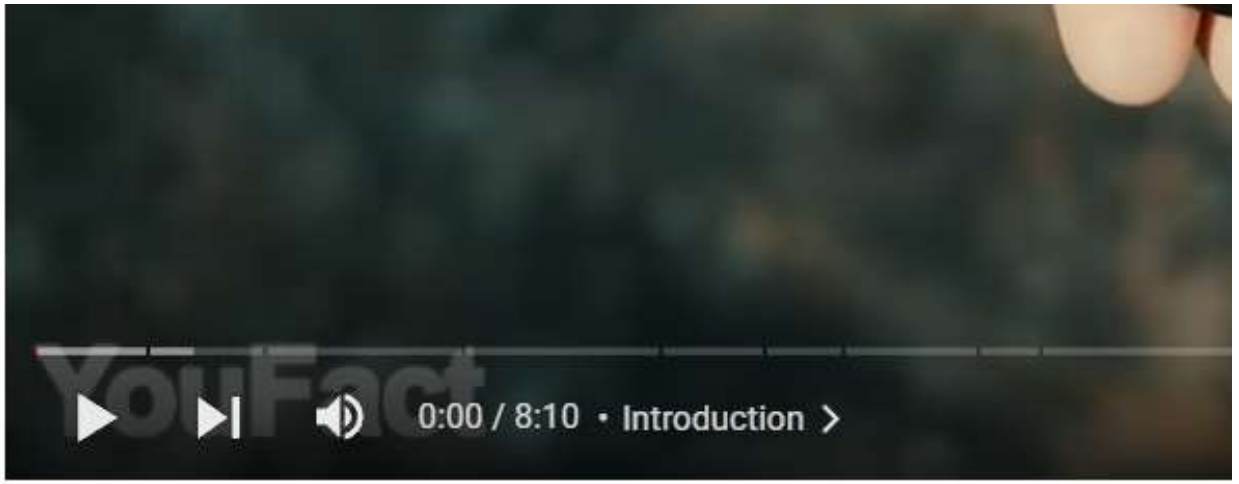


#YouFact #Tech #gadgets

16 Coolest Gadgets That Are Worth Seeing

YouFact Tech
구독자 49.1만명
구독

01 개요

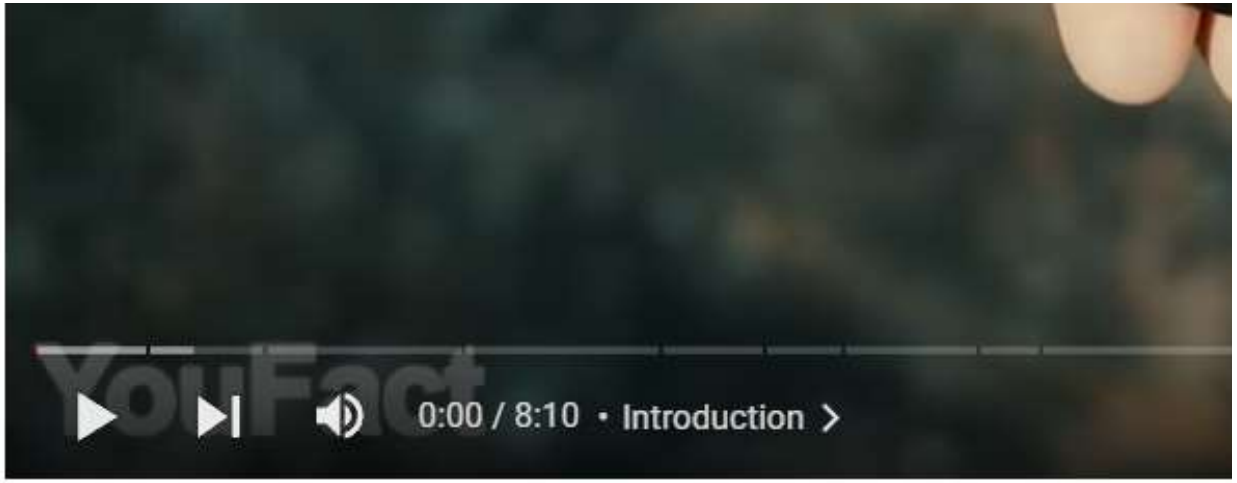


#YouFact #Tech #gadgets

16 Coolest Gadgets That Are Worth Seeing

YouFact Tech
구독자 49.1만명
구독

01 개요



#YouFact #Tech #gadgets

16 Coolest Gadgets That Are Worth Seeing

YouFact Tech
구독자 49.1만명
구독

01 개요

1. 세계 최대 규모의 비디오 플랫폼

2. 제목에 의한 태그 추가

02 관련내용 - LabelEncoder

id	data
1	사과
2	배
3	오렌지
4	오렌지
5	수박
6	사과

data	data
사과	0
배	2
오렌지	1
수박	3

문자열 데이터를
수치형으로 변환

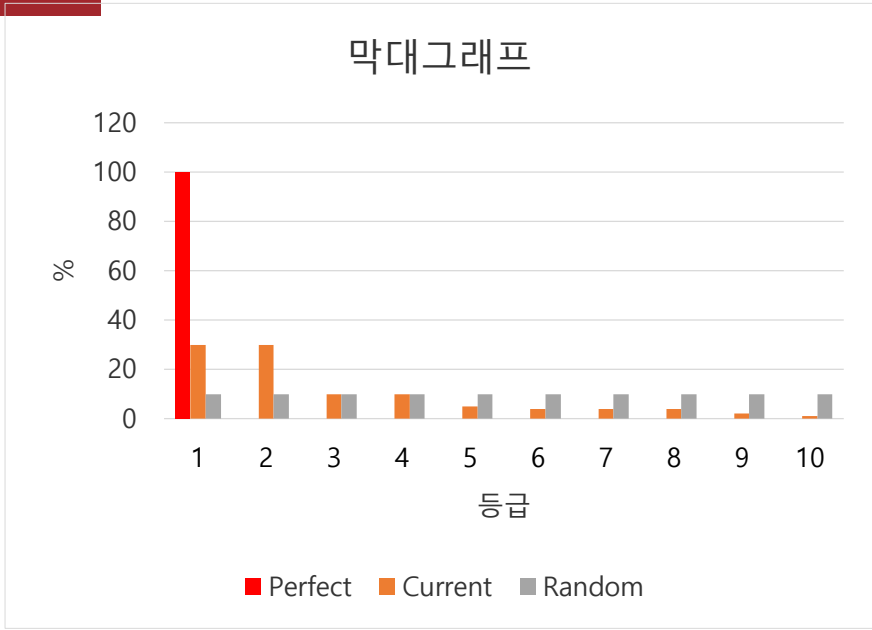
id	data
1	0
2	2
3	1
4	1
5	3
6	0

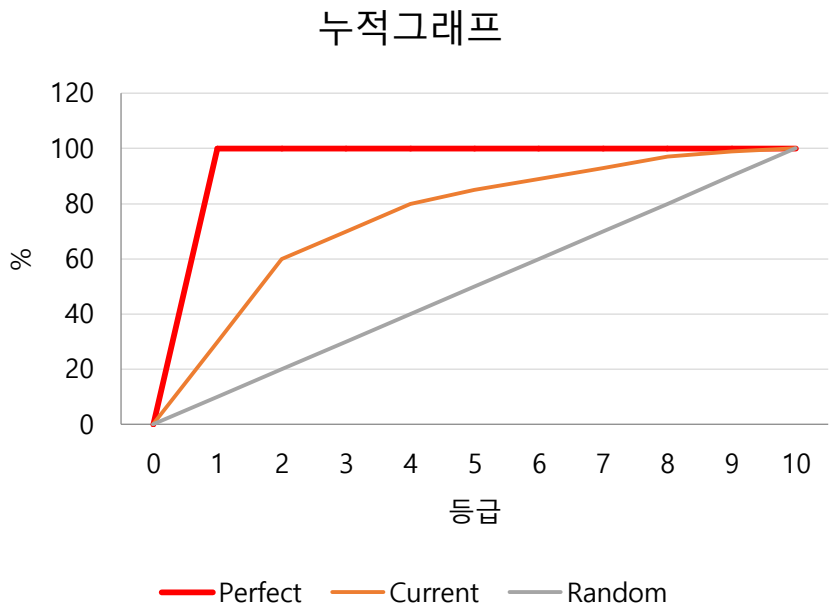
02 관련내용 - PorterStemmer (어간추출)

추출 전	어간 분리	추출 후
formalize	formal + ize	formal
allowance	allow + ance	allow
electricical	electric + ical	electric

정해진 규칙만 보고 단어의 어미를 자르는 작업
추출 후 단어가 사전에 존재하지 않을 수도 있음

02 관련내용 - CAP (누적 정확도 프로파일)





모델의 판별력을 시각화

데이터를 구별해낸 모델이 얼마나
정확하게 구분되었는지 시각적으로 볼 수 있음

03 데이터 전처리 - input data

```
video_stats_data = pd.read_csv("/kaggle/input/youtube-statistics/videos-stats.csv")
video_stats_data.head()
```

	Unnamed: 0	Title	Video ID	Published At	Keyword	Likes	Comments	Views
0	0	Apple Pay Is Killing the Physical Wallet After...	wAZZ-UWGVHI	2022-08-23	tech	3407.0	672.0	135612.0
1	1	The most EXPENSIVE thing I own.	b3x28s61q3c	2022-08-24	tech	76779.0	4306.0	1758063.0
2	2	My New House Gaming Setup is SICK!	4mgePWWCAmA	2022-08-23	tech	63825.0	3338.0	1564007.0
3	3	Petrol Vs Liquid Nitrogen | Freezing Experimen...	kXiYSI7H2b0	2022-08-23	tech	71566.0	1426.0	922918.0
4	4	Best Back to School Tech 2022!	ErMwWXQxHp0	2022-08-08	tech	96513.0	5155.0	1855644.0

Title : 제목
Video ID : 동영상 ID
Published At : 영상을 올린 날짜
Keyword : 키워드

Likes : 좋아요 횟수 - 싫어요 횟수
Comments : 댓글 개수
Views : 조회수

03 데이터 전처리 - 데이터 분리

```
videos_titles = list(video_stats_data["Title"].values)
keywords = list(video_stats_data["Keyword"].values)

for i in range(0,3) :
    print(videos_titles[i])
    print(keywords[i] + "\n")
```

```
Apple Pay Is Killing the Physical Wallet After Only Eight Years | Tech News Briefing Podcast | WSJ
tech

The most EXPENSIVE thing I own.
tech

My New House Gaming Setup is SICK!
tech
```

제목과 키워드를 리스트로 저장

03 데이터 전처리 - 제목 어간추출

```
print(videos_titles[0])

titles = [v_t.split('|')[0] for v_t in videos_titles]
print(titles[0])

eng_stopwords = stopwords.words('english')
corpus = []

for vt in titles:
    cleaned_title = re.sub('[^a-zA-Z]', ' ', vt)
    cleaned_title = cleaned_title.lower()
    cleaned_title = cleaned_title.split()
    stemmer = PorterStemmer()
    cleaned_title = [stemmer.stem(token) for token in cleaned_title if not token in set(eng_stopwords)]
    corpus.append(' '.join(cleaned_title))
print(corpus[0])
```

```
Apple Pay Is Killing the Physical Wallet After Only Eight Years | Tech News Briefing Podcast | WSJ
Apple Pay Is Killing the Physical Wallet After Only Eight Years
appl pay kill physic wallet eight year
```

제목의 어간 추출

03 데이터 전처리 - 키워드 LabelEncoder

```
cv = CountVectorizer(max_features=1500)
X = cv.fit_transform(corpus).toarray()
lbe = LabelEncoder()
y = lbe.fit_transform(keywords)

print(y)
```

```
[37 37 37 ... 23 23 23]
```

키워드 LabelEncoder

03 데이터 전처리 - 학습 데이터와 테스트 데이터 분리

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
print(len(X_train), len(X_test))
print(len(y_train), len(y_test))
```

```
1504 377
1504 377
```

학습 데이터 : 1504개(80%)
테스트 데이터 : 377개(20%)

03 데이터 전처리 - 학습 데이터와 테스트 데이터 분리

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
print(len(X_train), len(X_test))
print(len(y_train), len(y_test))
```

```
1504 377
1504 377
```

학습 데이터 : 1504개(80%)
테스트 데이터 : 377개(20%)

04 데이터 학습 - 로지스틱 회귀분석

```
lr_reg = LogisticRegression(multi_class='ovr', solver='liblinear')
lr_reg.fit(X_train, y_train)
y_pred = lr_reg.predict(X_test)
acc = accuracy_score(y_test, y_pred)

for i in range(0,10) :
    print(lbe.inverse_transform([y_pred[i], y_test[i]]))
```

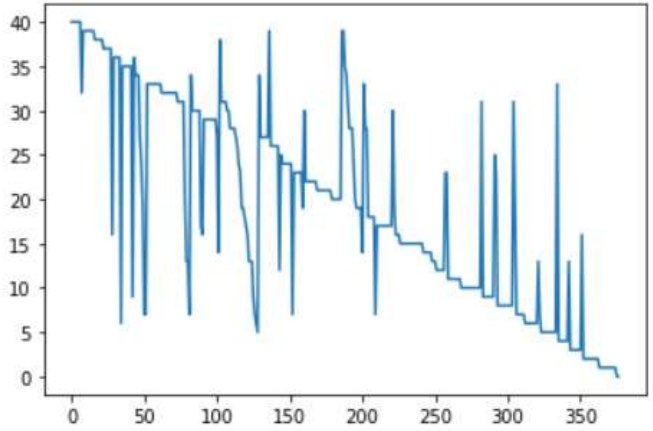
```
['lofi' 'lofi']
['google' 'google']
['interview' 'interview']
['literature' 'literature']
['mukbang' 'mukbang']
['history' 'mrbeast']
['chess' 'chess']
['nintendo' 'nintendo']
['tutorial' 'tutorial']
['physics' 'physics']
```

출력값과 정답을 비교

04 데이터 학습 - CAP(누적 정확도 프로파일) 곡선

```
model_y = [y for _, y in sorted(zip(y_pred, y_test), reverse=True)]
plt.plot(model_y)
```

```
[<matplotlib.lines.Line2D at 0x7fc7c4466550>]
```



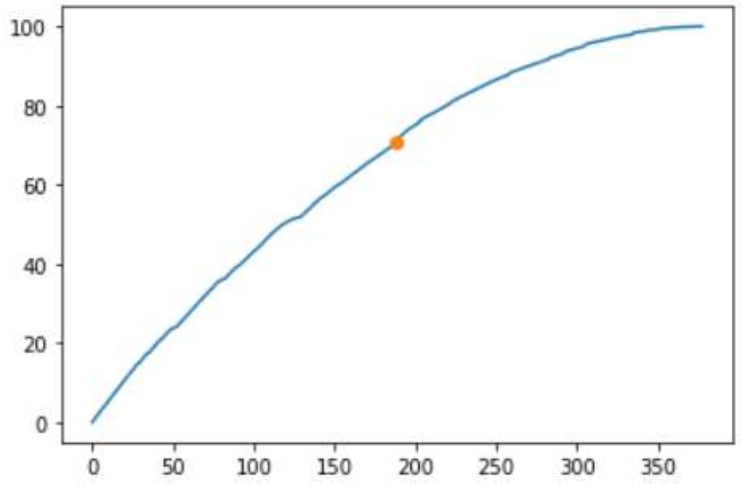
정답값의 크기를 기준으로 출력값을 정렬
-> 출력값과 정답이 같지 않으면 대각선에서 멀어짐

04 데이터 학습 - CAP(누적 정확도 프로파일) 곡선

```
nb_y = np.append([0], np.cumsum(model_y))
half_x = int((50 * len(y_test) / 100))
cap = nb_y[half_x] * 100 / max(nb_y)

plt.plot(100 * nb_y/max(nb_y))
plt.plot(half_x,cap, marker ='o')
```

```
[<matplotlib.lines.Line2D at 0x7fc7c43ebe10>]
```



정렬한 값들을 누적곡선으로 변경하고
중앙값과 최댓값의 비율을 계산
-> 모델의 판별력
70% 이상이면 충분히 판별력 있는 모델

05 테스트 결과

```
print("Accuracy is: {} %".format(round(acc * 100,2)))
print("CAP: {} %".format(round(cap,2)))
```

```
Accuracy is: 79.05 %
CAP: 70.97 %
```

정확도 : 79.05%
테스트 데이터의 판별력 : 70.97%

THANK YOU!