

데이터사이언스개론

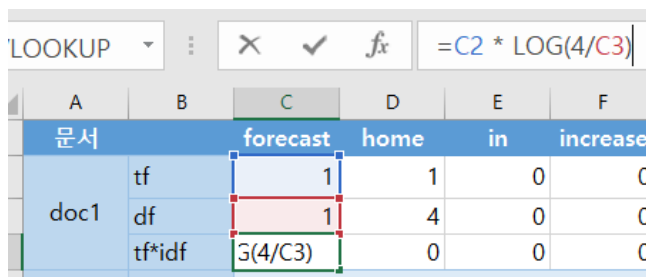
Similarity between documents 과제

컴퓨터공학과 201811259 배수빈

0. tf/ df/ tf* idf 계산

문서		forecast	home	in	increase	july	new	November	rise	sales	top
doc1	tf	1	1	0	0	0	1	0	0	1	1
	df	1	4	0	0	0	2	0	0	4	1
	tf*idf	0.60206	0	0	0	0	0.30103	0	0	0	0.60206
doc2	tf	0	1	1	0	1	0	0	1	1	0
	df	0	4	3	0	2	0	0	2	4	0
	tf*idf	0	0	0.12494	0	0.30103	0	0	0.30103	0	0
doc3	tf	0	1	2	1	1	0	0	0	1	0
	df	0	4	3	1	2	0	0	0	4	0
	tf*idf	0	0	0.24988	0.60206	0.30103	0	0	0	0	0
doc4	tf	0	1	1	0	0	1	1	1	1	0
	df	0	4	3	0	0	2	1	2	4	0
	tf*idf	0	0	0.12494	0	0	0.30103	0.60206	0.30103	0	0

위의 표는 doc1, 2, 3, 4 에 포함되어 있는 term 들에 대하여 tf 와 df 직접 세어 각각 입력하고, tf * idf 는 엑셀의 함수를 활용하여 계산한 결과이다.



문서	forecast	home	in	increase
doc1	tf: 1	1	0	0
	df: 1	4	0	0
	tf*idf: 0.60206	0	0	0

이렇게 $tf * \log(4/df)$ 로 각 term 에 대한 tf* idf 를 계산했다.

1. tf-idf 에 기반한 벡터로 표현하기

(벡터의 차원의 순서는 term 의 알파벳 순서)

tf-idf기반 벡터											
문서	forecast	home	in	increase	july	new	November	rise	sales	top	
doc1	0.60206	0	0	0	0	0.30103	0	0	0	0.60206	
doc2	0	0	0.12494	0	0.30103	0	0	0.30103	0	0	
doc3	0	0	0.24988	0.60206	0.30103	0	0	0	0	0	
doc4	0	0	0.12494	0	0	0.30103	0.60206	0.30103	0	0	

위에서 계산한 tf*idf 계산 결과만 활용해서 벡터를 표현하였다.

2. Similarity 계산 (높은쌍 -> 낮은 쌍 순서)

Similarity 또한 엑셀의 함수를 활용하여 계산하였다.

Similarity			similarity가 높은 순서쌍		
v1	v2	cos similarity	v1	v2	cos similarity
doc1	doc1	1	doc1	doc1	1
	doc2	0	doc2	doc2	1
	doc3	0	doc3	doc3	1
	doc4	0.134170419	doc4	doc4	1
doc2	doc2	1	doc2	doc3	0.38246381
	doc3	0.38246381	doc2	doc4	0.320143749
	doc4	0.320143749	doc1	doc4	0.134170419
doc3	doc3	1	doc3	doc4	0.058138497
	doc4	0.058138497	doc1	doc2	0
doc4	doc4	1	doc1	doc3	0

각 similarity 는 v1 과 v2 에 대해서 $\frac{\text{내적값}}{|\text{v1 벡터의 크기}| * |\text{v2 벡터의 크기}|}$ 를 활용하여 계산하였다.

우측의 표는 similarity 가 높은 순서쌍에서 낮은 순서쌍 순으로 정렬하여 나타낸 표이다.

◆ 엑셀에서의 similarity 계산 식

DOOKUP $= (H4*H10 + D4*D10 + K4*K10 + L4*L10 + C4*C10 + J4*J10 + E4*E10 + G4*G10 + F4*F10 + I4*I10) / (L17 * L19)$

문서	forecast	home	in	increase	july	new	November	rise	sales	top								
doc1	tf	1	1	0	0	0	1	0	0	1	1							
	df	1	4	0	0	0	2	0	0	4	1							
	tf*idf	0.60206	0	0	0	0	0.30103	0	0	0.60206								
doc2	tf	0	1	1	0	1	0	0	1	1	0							
	df	0	4	3	0	2	0	0	2	4	0							
	tf*idf	0	0	0.12494	0	0.30103	0	0	0.30103	0	0							
doc3	tf	0	1	2	1	1	0	0	0	1	0							
	df	0	4	3	1	2	0	0	0	4	0							
	tf*idf	0	0	0.24988	0.60206	0.30103	0	0	0	0	0							
doc4	tf	0	1	1	0	0	1	1	1	1	0							
	df	0	4	3	0	0	2	1	2	4	0							
	tf*idf	0	0	0.12494	0	0	0.30103	0.60206	0.30103	0	0							

tf-idf기반 벡터

문서	forecast	home	in	increase	july	new	November	rise	sales	top	크기
doc1	0.60206	0	0	0	0	0.30103	0	0	0	0.60206	0.90309
doc2	0	0	0.12494	0	0.30103	0	0	0.30103	0	0	0.44368
doc3	0	0	0.24988	0.60206	0.30103	0	0	0	0	0	0.71801
doc4	0	0	0.12494	0	0	0.30103	0.60206	0.30103	0	0	0.74788

위 그림의 상단에는 doc1 과 doc3 의 similarity 를 계산한 식을 볼 수 있다.

$(H4*H10 + D4*D10 + K4*K10 + L4*L10 + C4*C10 + J4*J10 + E4*E10 + G4*G10 + F4*F10 + I4*I10)$ 는 각 벡터의 내적을 계산하는 식이고, $(L17 * L19)$ 는 각 벡터의 크기를 곱한 것이다.

◆ 엑셀에서의 벡터크기 계산 식

[illegible]

위 그림의 상단에는 doc1 의 tf-idf 기반 벡터의 크기를 계산 한 식을 볼 수 있다.

각 tf-idf 값에 대해서 제공한 값들을 모두 더해서 루트를 계산하였다.