Assignment 01

20231758
김상환

1. window_size가 1일 때와 2일 때의 비교

window_size가 1일 때는 코사인 유사도를 구하고 이를 내림차 순으로 정리했을 때 'hello'가 'you'와 가장 높은 유사도를 보였다. 반면 window_size를 2로 설정한 뒤 다시 comatrix를 계산하고 코사인 유사도를 계산한 결과 이전엔 가장 높은 유사도를 보였던 'hello'는 순위가 내려가고 'and'가 가장 높은 유사도를 보였다. 이러한 결과가 나온 이유로는 window_size가 1 이였을 때는 'you'와 'hello' 모두 근접한 단어가 'say' 하나이기 때문에 코사인 유사도를 계산했을 때 1이 나온 것이다. 실제로 ['you','say','goodbye','and','I','hello'] 이런 방식으로 단어 리스트를 설정했을 때 'you'의 벡터와 'hello'의 벡터는 모두 [0, 1, 0, 0, 0, 0] 이다. 이 두 벡터의 코사인 유사도는 $(0*0+0*0+0*0+0*0+0*0+0*0+1*1)/(\sqrt[2]{1^2}*\sqrt[2]{1^2})$ 이므로 1이다. 문맥상 'you'는 동사의 주체인 대명사이고 'hello'는 동사의 목적어이므로 유사도가 낮아야 하지만 window_size가 너무 작고 표본 수가 적어 유사도가 높게 나온 것으로 예상된다.

이제 window_size가 2일 때 유사도가 가장 높았던 'and'와 'you'의 유사도도 직접 계산해보면 다음과 같다. ['you','say','goodbye','and','I','hello']로 단어 리스트를 설정하면 'you'의 벡터는 [0, 1, 1, 0, 0, 0], 'and'의 벡터는 [0, 2, 1, 0, 1, 1] 이므로 둘의 코사인 유사도는 $(1 + 2)/(\sqrt[2]{2} \times \sqrt[2]{7}) = 0.866...$ 이다. 여전히 'you'와 'and' 사이의 실제 유사도는 낮지만 실제 유사도가 높은 'I'의 유사도가 더 올라왔다는 점에선 window_size를 늘림으로써 실제 단어 사이의 유사도를 더 정확히 판단할 수 있다는 점을 알 수 있었다. 물론 더 정확한 유사도를 계산하려면 충분한 데이터와 많은 시도를 통해 최적의 window_size를 찾아야 할 것 같다.

2. 불용어 제거 후 분석

불용어를 제거한 후 동일한 방식으로 코사인 유사도를 계산하고 유사 단어를 비교해본 결과, 단어 간 유사도와 그 순위에서 변화가 발생하였다.

불용어 리스트에 'and'를 추가하여 preprocess() 함수에 적용한 결과, window_size 1일 때의 유사도 순위 상위권은 변화가 없었지만 window_size가 2일 때 가장 유사도가 높았던 'and'가 사라짐으로써 가장 유사도가 높은 단어가 'I'가 되었다.

사실 문자열이 너무 작아 이런 불용어를 제거하여 의미 있는 변화를 보기 어렵지만 문자가 많아지고 복잡해지면 불용어 제거를 하는 것이 의미있는 역할을 수행할 것이라는 생각이 들었다.