

2021  
contriever

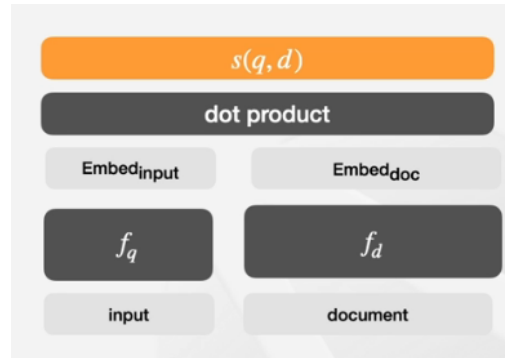
다큐먼트 리트리버에 초점을 맞춘 논문  
리트리버 잘 찾아내는데에 초점 :: BM25, TF IDF  
빠르고 간단하지만 어휘가 조금만 달라져도 문맥이해 잘 못함  
임베딩 유사성으로 문서를 찾기도 함  
레이블링된 문서쌍 없이도 리트리버를 좋게 할 수 있는가?

Contriever

목표 :: 쿼리가 주어졌을 때 많은 문서 중에서 관련 문서를 찾아낼 수 있는 모델 만들기

bi encoder 구조

- Relevance score : 쿼리 임베딩과 문서 임베딩의 내적



$f_q$   $f_d$  잘 학습하기 -> 관련있는건 가깝게 무관한 임베딩과는 거리가 멀도록

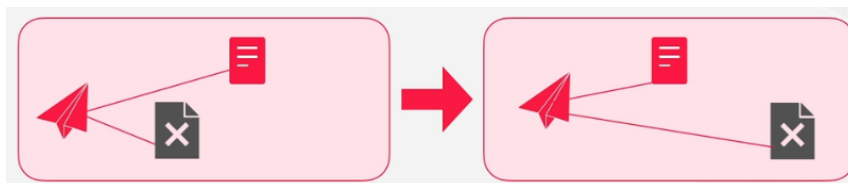
$$\mathcal{L}(q, k_+) = - \frac{\exp(s(q, k_+)/\tau)}{\exp(s(q, k_+)/\tau) + \sum_{i=1}^K \exp(s(q, k_i)/\tau)}$$

\*  $q$ : query

\*  $k_+$ : positive document

\*  $(k_i)_{i=1 \dots K}$ : negative documents

\*  $\tau$ : temperature



=> align

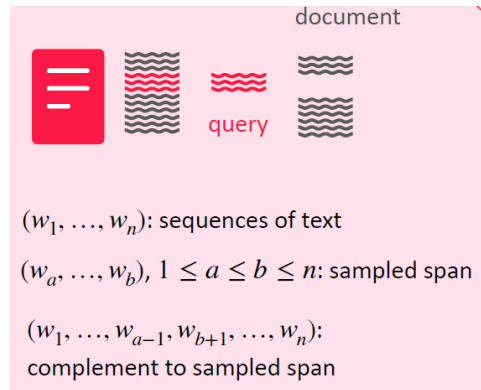
레이블링 전에 어떻게 학습할 것인가?

쿼리가 없는 많은 다큐먼트에서 출발.

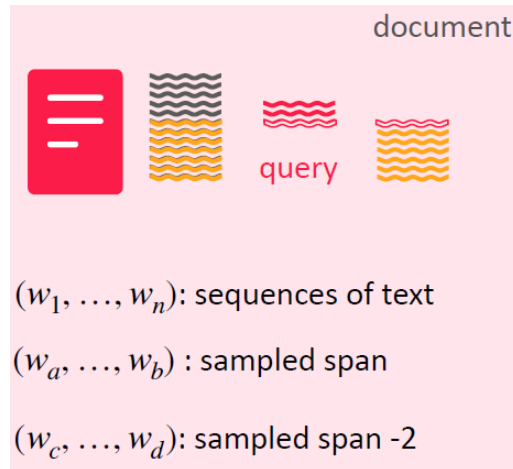
어떻게 positive negative 페어를 구분?

> positive paris

- inverse close task :: 쿼리가 어디서 나왔는지 찾을.

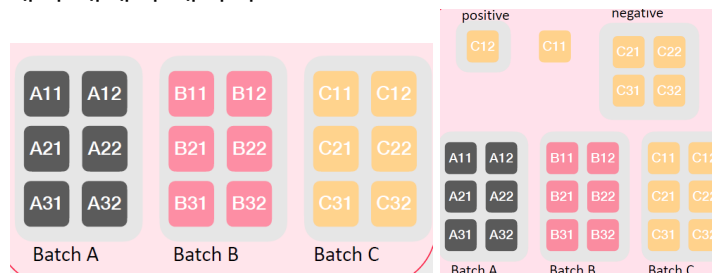


- independent Cropping :: 여러 조각들로 이루어진 다큐먼트들을 특정 부분으로 샘플링 → ICT에서는 쿼리를 뽑아내고 남은걸 샘플링. (쿼리와 다큐먼트 독립) independent cropping은 원본 문서에서 다시 도큐먼트 선택. 두 번의 샘플링이 독립적이라 쿼리랑 다큐먼트 사이에 겹치는 부분이 존재.
- 또다른 차이점은 ict 쿼리의 여집합을 다큐먼트로 생성해서 중간이 잘려있음. RC에서는 모두 연속되어있음

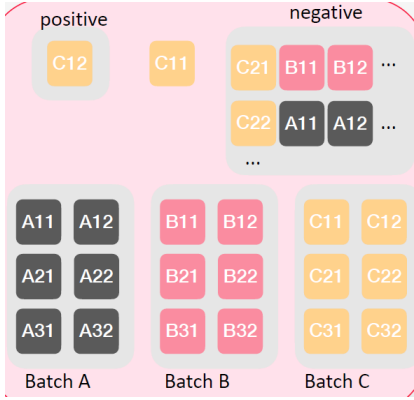
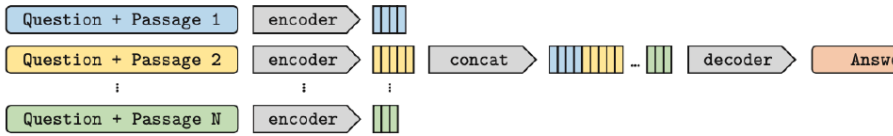


#### Negative pairs

- 배치 내에서 네거티브.



- C11에 대한 네거티브 페어
- 이런 방식으로 만들면 사용가능한 네거티브 페어가 배치 사이즈에 국한된다는 단점이 있음. 그래서 학습이 되려면 배치 사이즈가 커야함
- Negative across batch
- 네거티브 샘플 구성 시 이전 배치 샘플도 고려

	 <ul style="list-style-type: none"> <li>- 문제점 ) 쿼리와 다큐먼트는 각각의 인코더가 있고 그 사이에 유사도를 구했었음.. 근데 여러 배치에서 하려면 그때마다 인코더를 재학습 시키고 불일치가 발생할 수 있음</li> <li>- 그래서 MoCo가 제안됨</li> <li>-</li> </ul>
	<p>Contriever - 비지도 학습이 목표 최종 선택</p> <ul style="list-style-type: none"> <li>- Building training pairs <ul style="list-style-type: none"> <li>- positive : Independent Cropping</li> <li>- Negative : across batch <ul style="list-style-type: none"> <li>- 배치 샘플을 큐에 저장해 사용</li> </ul> </li> </ul> </li> <li>- 구조 : MoCo <ul style="list-style-type: none"> <li>- Momentum encoder</li> </ul> </li> <li>- Training data <ul style="list-style-type: none"> <li>- Wikipedia CCNet</li> </ul> </li> </ul>
ATLAS	<p>컨트리버를 리트리버로 삼고 제너레이터를 T5로 실행</p> <ul style="list-style-type: none"> <li>- Retrieval-augmented language model pretrained on unlabeled data that exhibits few shot abilities on knowledge intensive tasks</li> <li>- 좋은 성능을 얻기 위해서는 많은 labeling 데이터가 필요</li> </ul> <p>Contriever</p> <ul style="list-style-type: none"> <li>- 미리 학습된 문서 없이도 비지도학습</li> <li>- pre trained =&gt; contrastive learning</li> <li>- 랜덤 Cropping. 하나는 쿼리로 하나를 다큐먼트로 사용</li> <li>- 랜덤 토큰 삭제</li> <li>- 다큐먼트 인코더의 업데이트 제한을 위해 MoCo 도입</li> <li>- momentum encoder -&gt; query encoder로 조금씩 document 업데이트</li> </ul> <p>LM</p> <ul style="list-style-type: none"> <li>- T5 + FID</li> </ul>  <p>Figure 2: Architecture of the Fusion-in-Decoder method.</p> <ul style="list-style-type: none"> <li>- 여러개 encoder를 사용하고 decoder로 통합</li> <li>- 문서를 더 잘 종합</li> </ul> <p>Training object - Retriever</p> <ol style="list-style-type: none"> <li>1. Attention Distillation - 동저자의 타논문. 각 쿼리와 document를 합친 input representation: X. self attention에서 나온 H.</li> </ol>

cross attention : query key value

$$\mathbf{Q} = \mathbf{W}_Q \mathbf{H}$$

$$\mathbf{K} = \mathbf{W}_K \mathbf{X}$$

$$\mathbf{V} = \mathbf{W}_V \mathbf{X}$$

한 문서의 토큰과 시퀀스의 어텐션 값으로 중요도 측정

