Winter Vacation Seminar & Paper Review

Dense Passage Retrieval for Open-Domain Question Answering

28th, January, 2021

Natural Language Learning Lab
Taegyeong, Eo

Open-Domain Question Answering

- ✓ **Open-Domain : General한 Context**
- ✓ **정답이 있는 Passage가 주어지는 QA**(= Single-Document QA) **task와는 다름**
  - **Ex) SQuAD**(The Stanford Question Answering Dataset)
- ✓ **Knowledge Base에 정답이 있는 질문**(factual question) **을 던지고 받는 것**

Question : 아인슈타인은 무엇으로 노벨상을 받았는가?

The **Nobel Prize** in Physics 1921 was awarded to **Albert Einstein** "for his services to Theoretical Physics, and especially for his discovery of the law of the photoelectric effect." **Albert Einstein** received his **Nobel Prize** one year later, in 1922.

start end

Knowledge Base

WIKIPEDIA
The Free Encyclopedia

Answer : 광전 효과

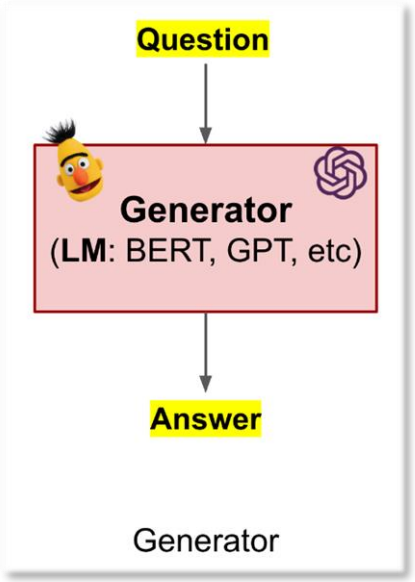
① **Information Retrieval**(IR) : **Context를 좁혀보자**

② **Machine Reading at Scale**(MRS) : **좁혀진 Context에서 Machine Reading Comprehension 해보자**

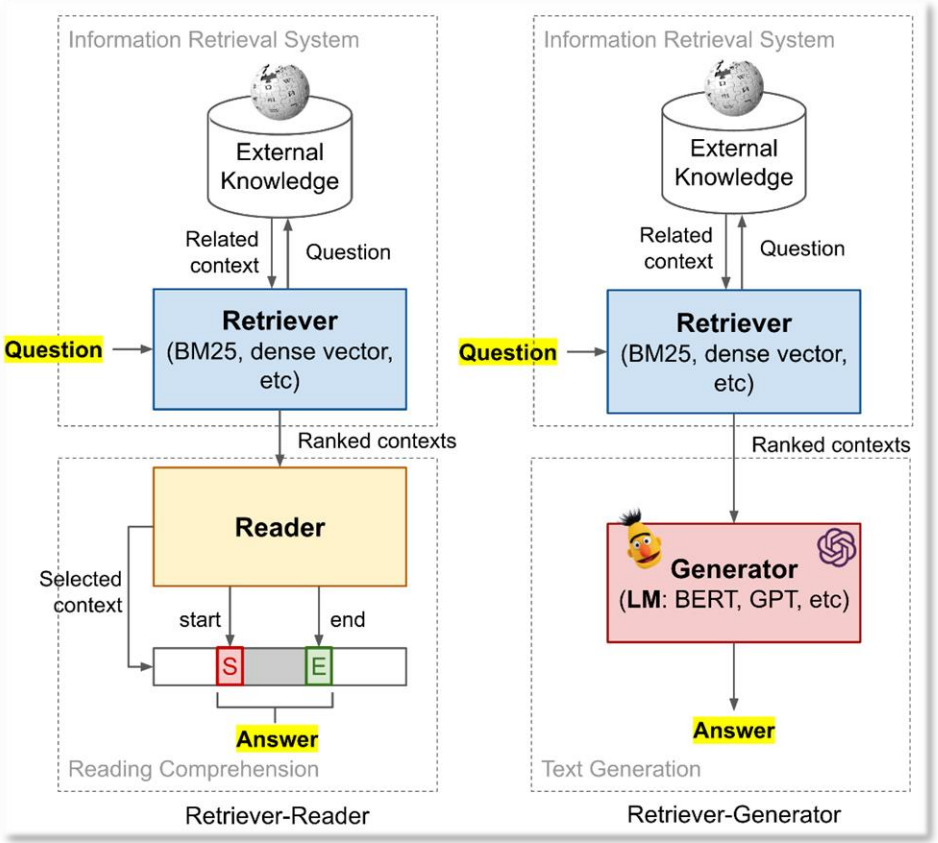
Open-Domain Question Answering

✓ 크게 External source of knowledge(=Knowledge Base)의 참조 유무로 분류

- 일반적으로 Wikipdia 문서를 말함
- External Knowledge O → **Open-book QA**
- External Knowledge X → **Closed-book QA**



Closed-book QA



Open-book QA

Information Retrieval System

✓ Document를 Query에 대해 Scoring

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) * \frac{f(q_i,D) * (k_1+1)}{f(q_i,D) + k_1 * (1 - b + b * \frac{|D|}{avgdl})}$$

$f(q_i, D)$: 문서 D에서 q_i의 term frequency
$|D|$: 문서 D 의 길이
k_1, b: 파라메터
$avgdl$: 문서 집합의 평균 문서 길이

❖ **Sparse Vector : tf-idf를 통해 Scoring** ex) BM25

$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,D)$$

t: 쿼리의 단어
d: 해당 문서
D: 문서 내 단어의 빈도수
$\mathrm{idf}(t,D)$: 전체 문서 내 단어의 희소성

$$IDF(q_i) = \ln(1 + \frac{(docCount - f(q_i) + 0.5)}{f(q_i) + 0.5})$$

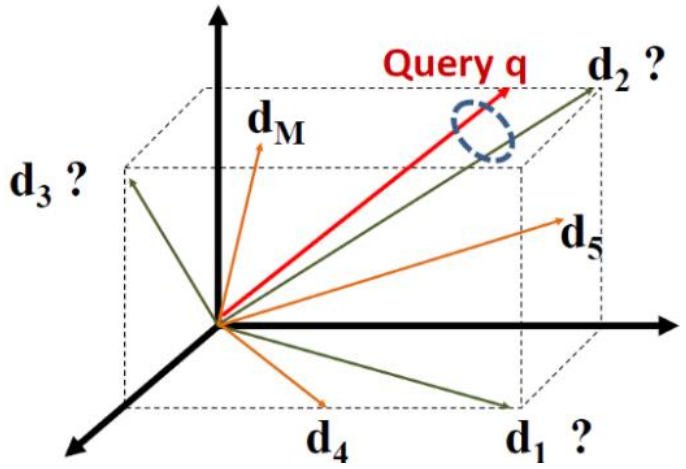
$docCount$: 총 문서의 개수
$f(q_i)$: 해당 단어를 포함하는 문서의 개수

※ BM25 https://littlefoxdiary.tistory.com/12

➡ **Keyword Matching에 유리**

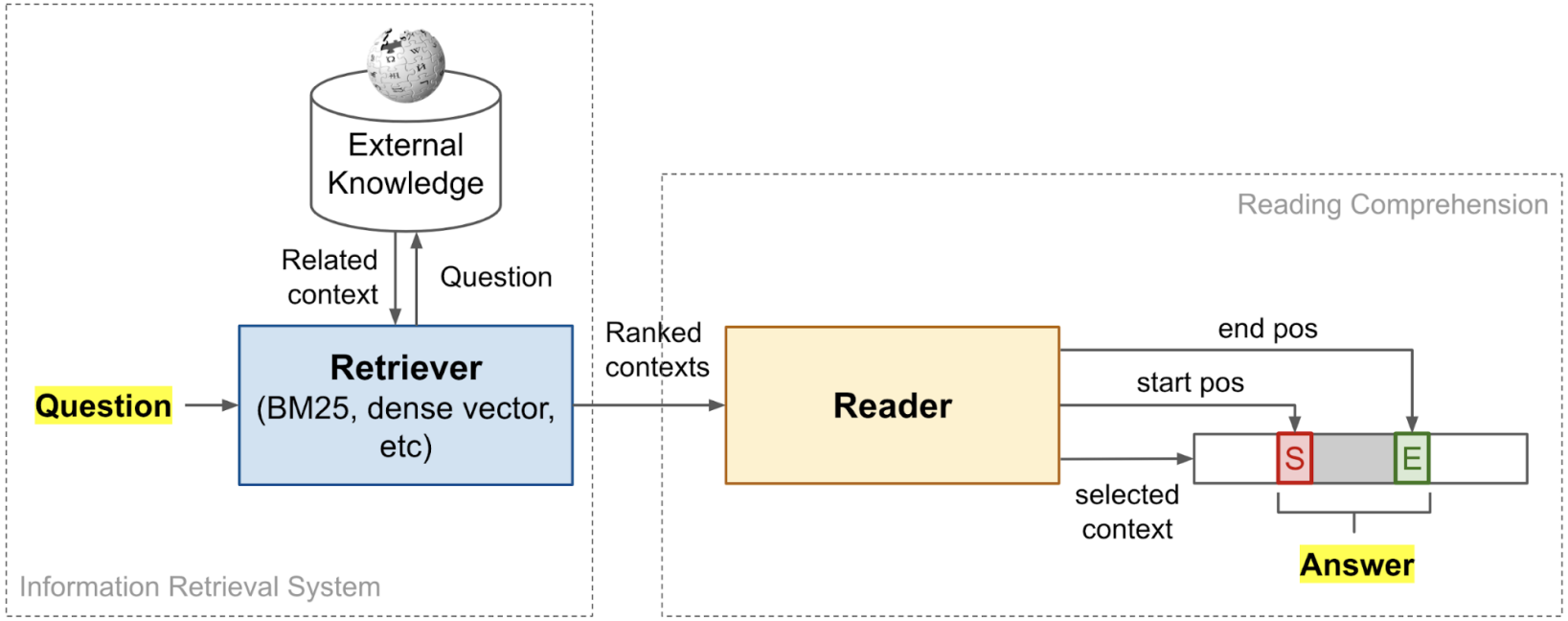
❖ **Dense Vector : Cosine Similarity 계산** ex) LSA, Word2Vec, Doc2Vec …

➡ **Context를 반영하기 좋음**



Retriever-Reader Model

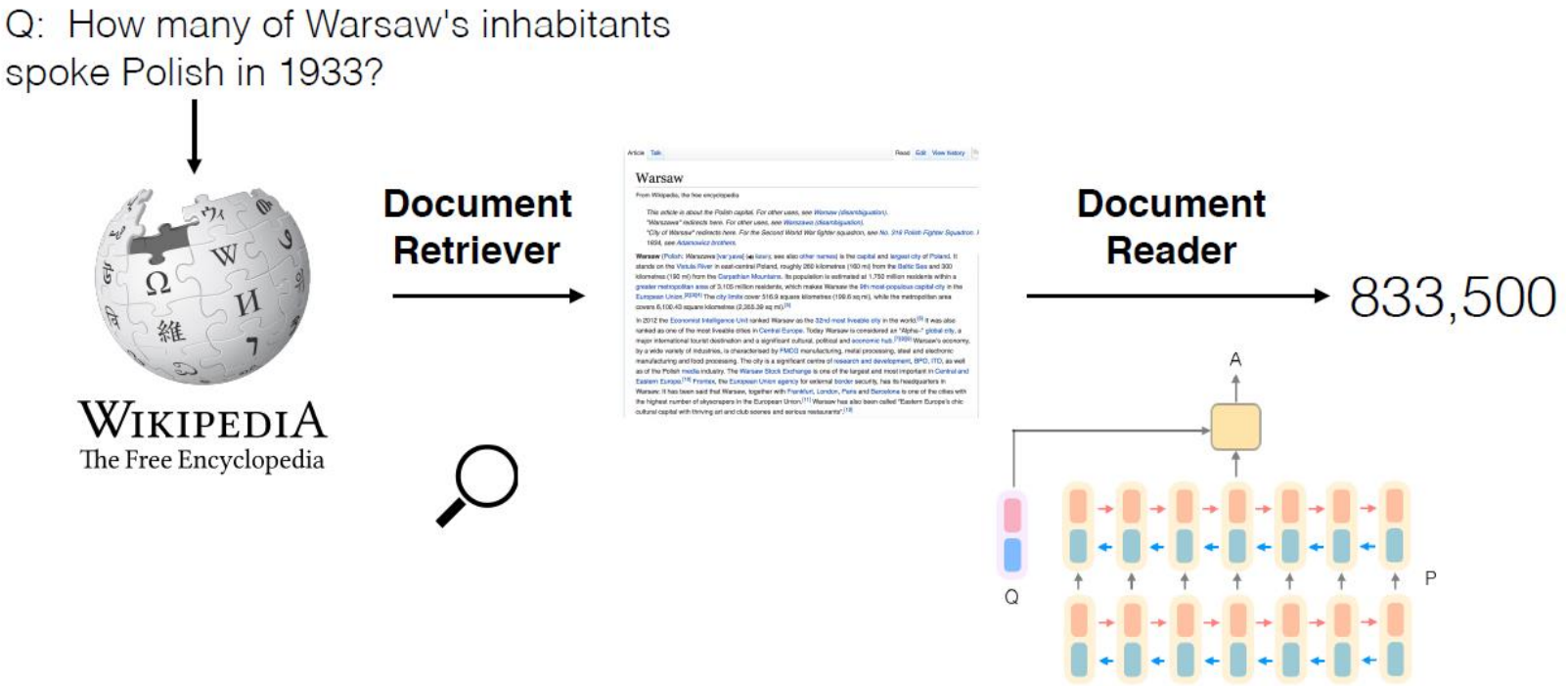
- ✓ IR Model + QA Model
- ✓ 최근 연구 대부분이 이 모델을 채택하고 있음
- ✓ Retriever(IR) : Query와 관련있는 상위 k개의 문서를 리턴
- ✓ Reader(QA) : k개의 문서를 보고 가장 정답일 확률이 높은 Span을 리턴



1 Introduction

DrQA (described in Reading Wikipedia to Answer Open-Domain Questions)

- ✓ 대표적인 Retrival-Reader Model
- ✓ ***Document Retriever*** : tf-idf Vector space model
- ✓ ***Document Reader*** : Glove + Bidirectional LSTM

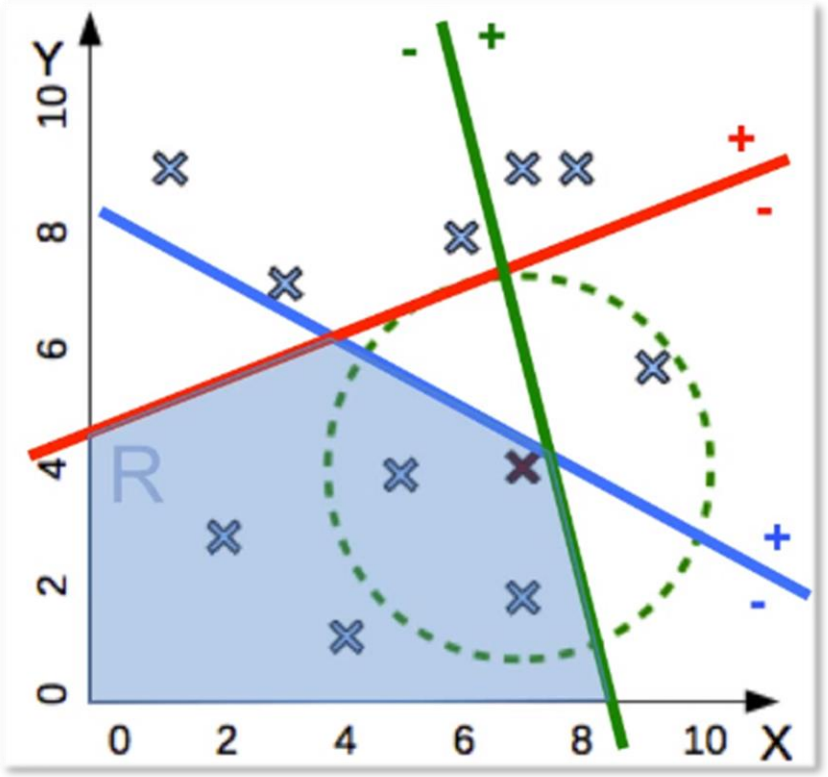


1 Introduction

ORQA (described in Latent Retrieval for Weakly Supervised Open Domain Question Answering)

- ✓ 모든 Component가 BERT기반
  - *Retriever Component* : Question Encoder, Passage Encoder
    - 두 인코더의 Output의 내적을 통해 Similarity 측정
  - *Reader Component* : Reading Comprehension Model

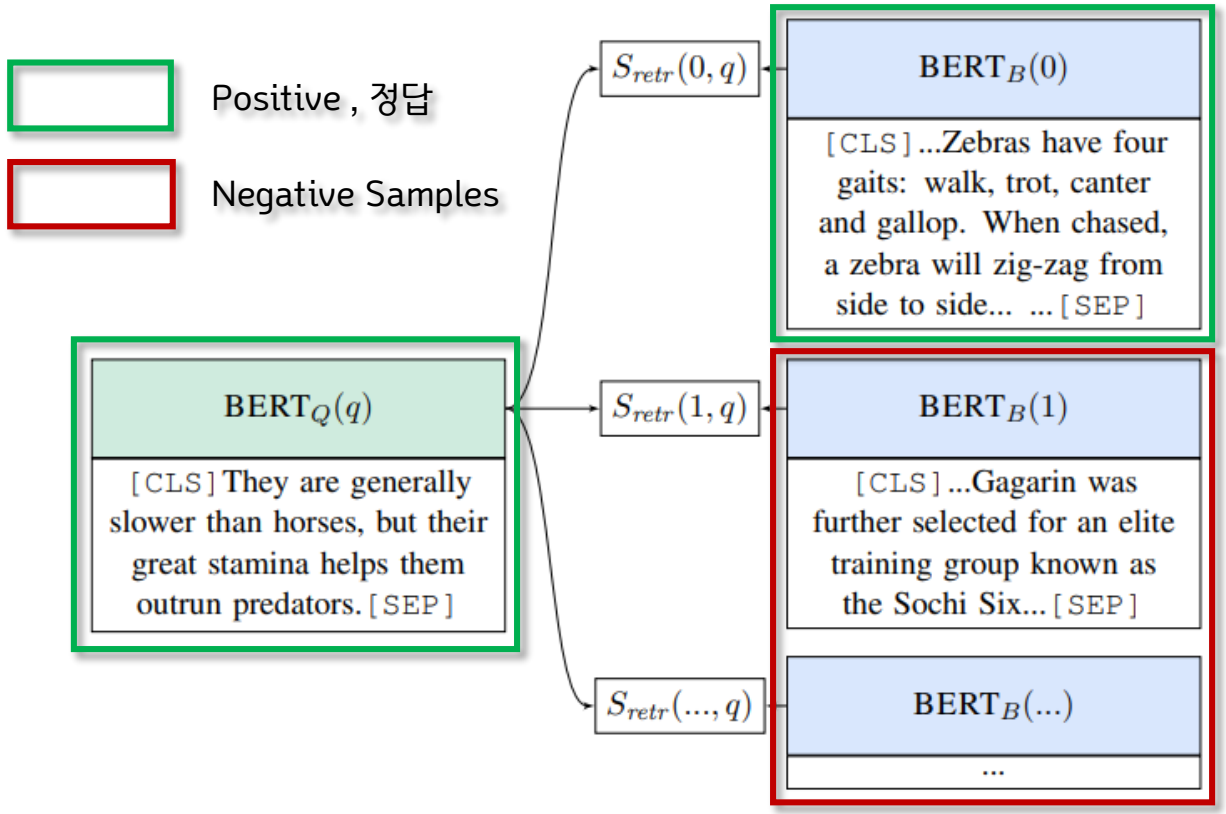
- ✓ DrQA와 달리 제한된 Set에 대해 QA하지 않음
- ✓ 1,300만개의 Passage을 모두 고려
- ✓ Locality-Sensitive Hashing으로 Passage을 미리 인코딩
  - 같은 Bucket의 Passage들은 유사하다고 판단
  - k개의 Passage을 빠르게 탐색가능 (Using Beam-Search)



1 Introduction

ORQA (described in Latent Retrieval for Weakly Supervised Open Domain Question Answering)

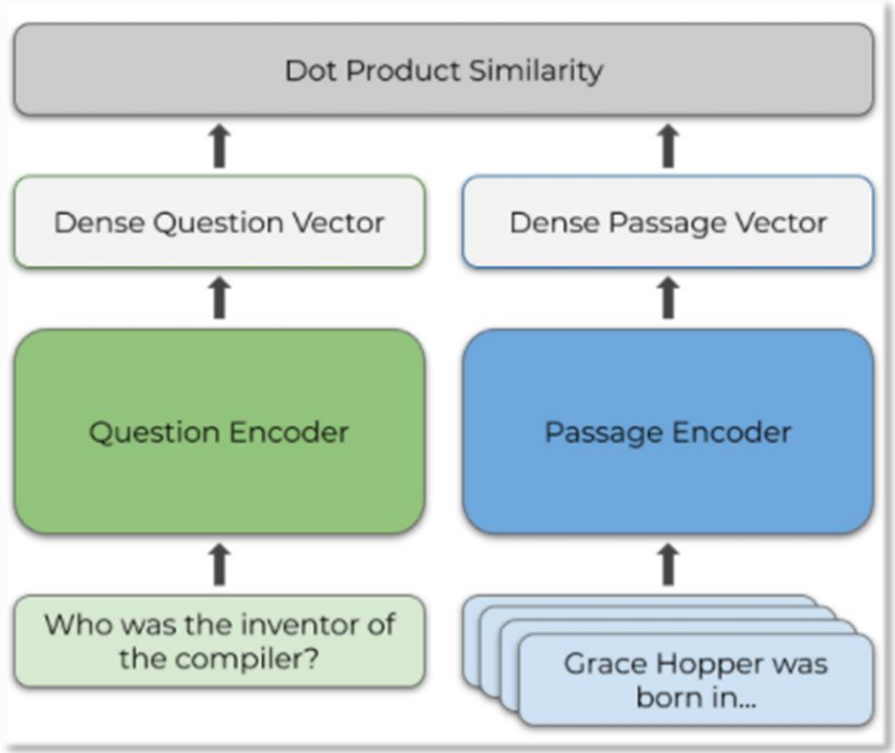
- ✓ Inverse Cloze Task로 Retriever를 Pre-training
  - Negative sampling을 통해 학습

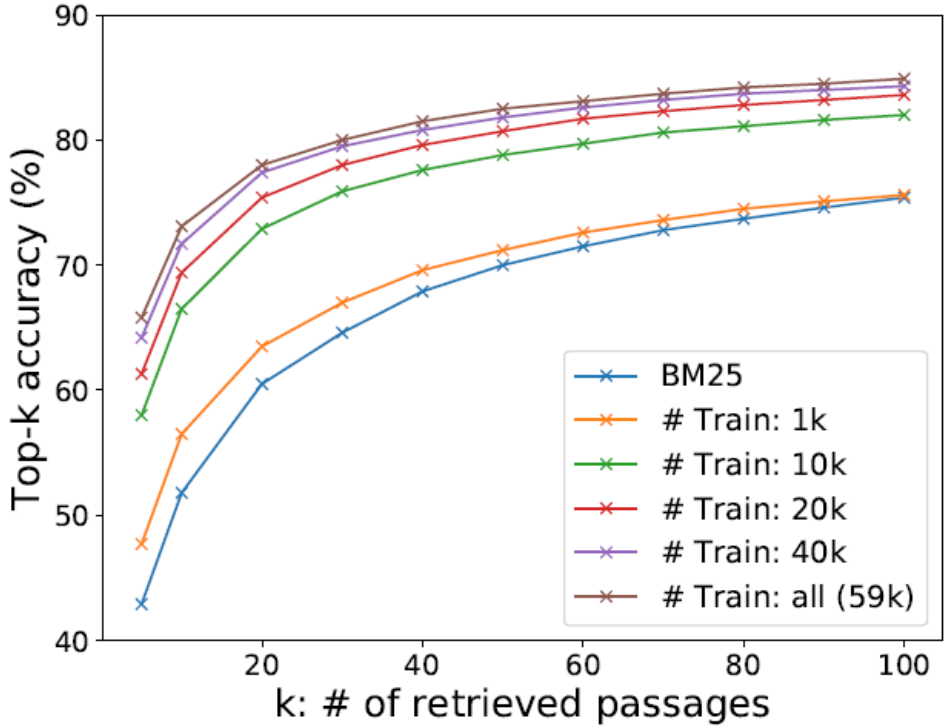


DPR 제안 배경

- ✓ ORQA에서 Inverse Cloze Task로 Retriever를 Pre-training에 연산량 ↑
- ✓ Question-Answer Set이 아닌 Self-supervised Learning하여 Optimal 하지 않음

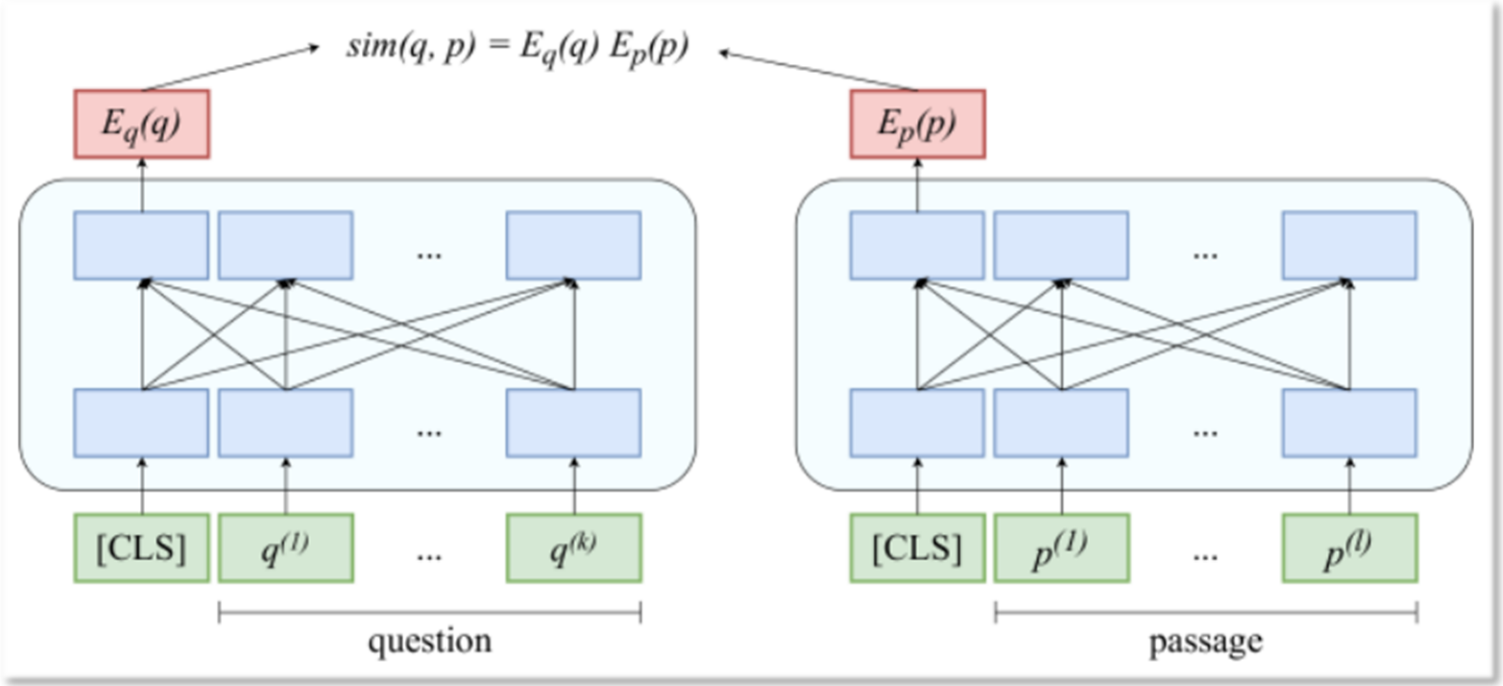
➡ **Pre-training 없이 Dual-encoder를 small QA set으로 학습시켜보자**





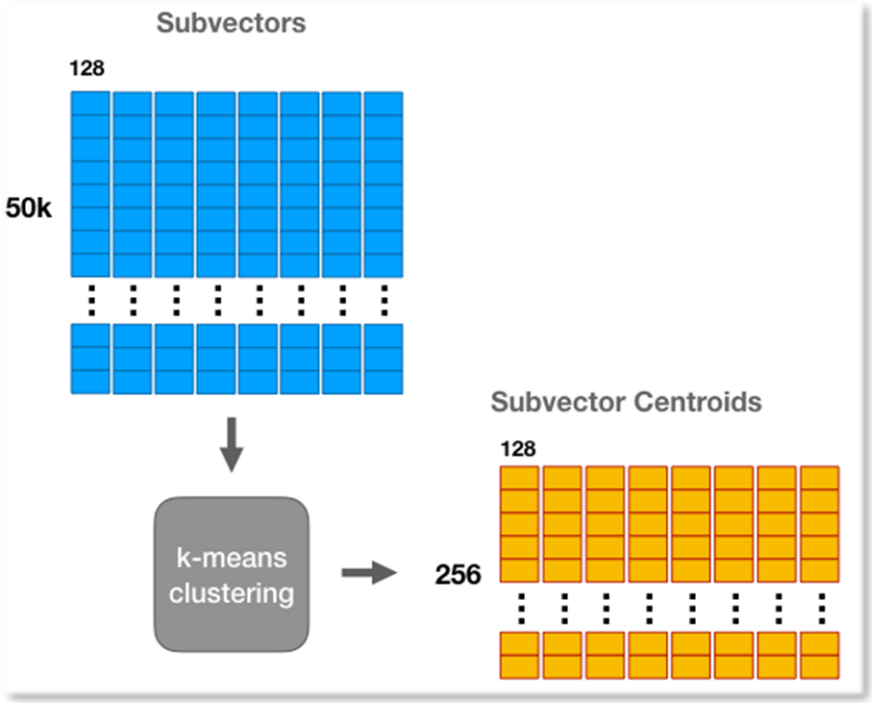
Dual Encoder

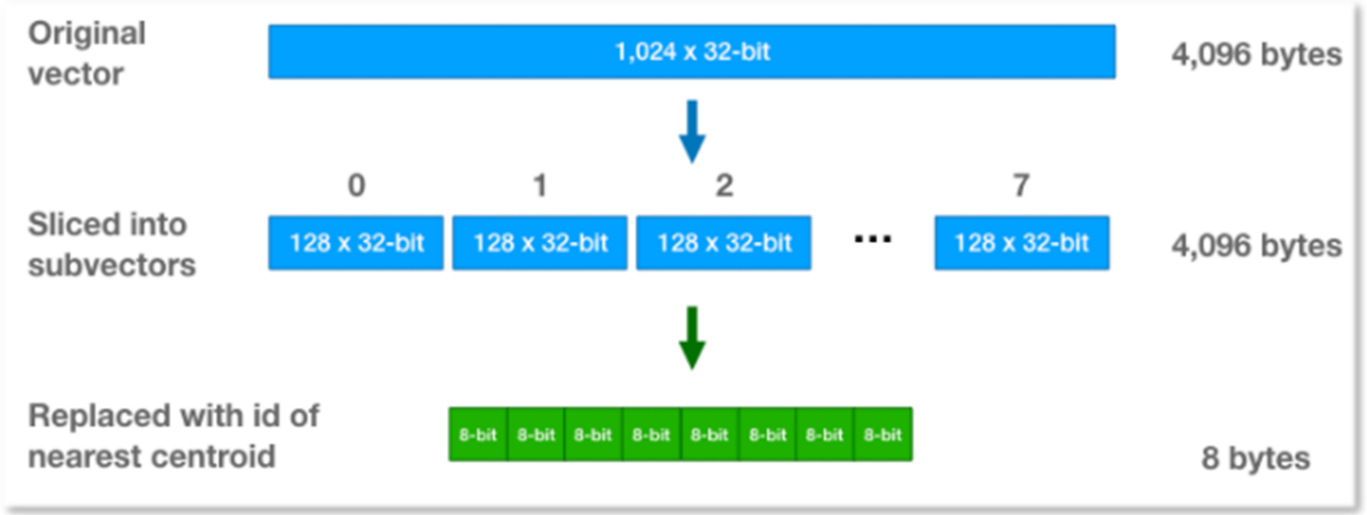
- ✓ ORQA와 동일하게 Question Encoder와 Passage Encoder를 가짐
- ✓ Output은 [CLS] 토큰의 위치의 Vector
- ✓ Question과 Passage의 유사도를 Cosine Similarity로 정의



Indexing Passage

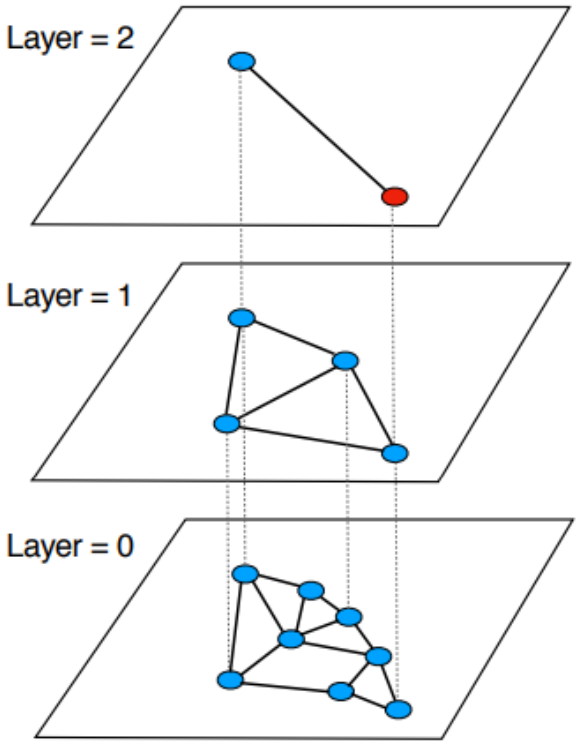
- ✓ FAISS 라이브러리의 HNSW Indexing을 사용하여 빠르게 k개의 문서를 리턴
- ✓ Product Quantization
  - 벡터를 쪼개고 그룹에 대한 k개의 중심점을 구함
  - 그룹 내의 subvector를 가장 근접한 중심점의 인덱스로 매핑



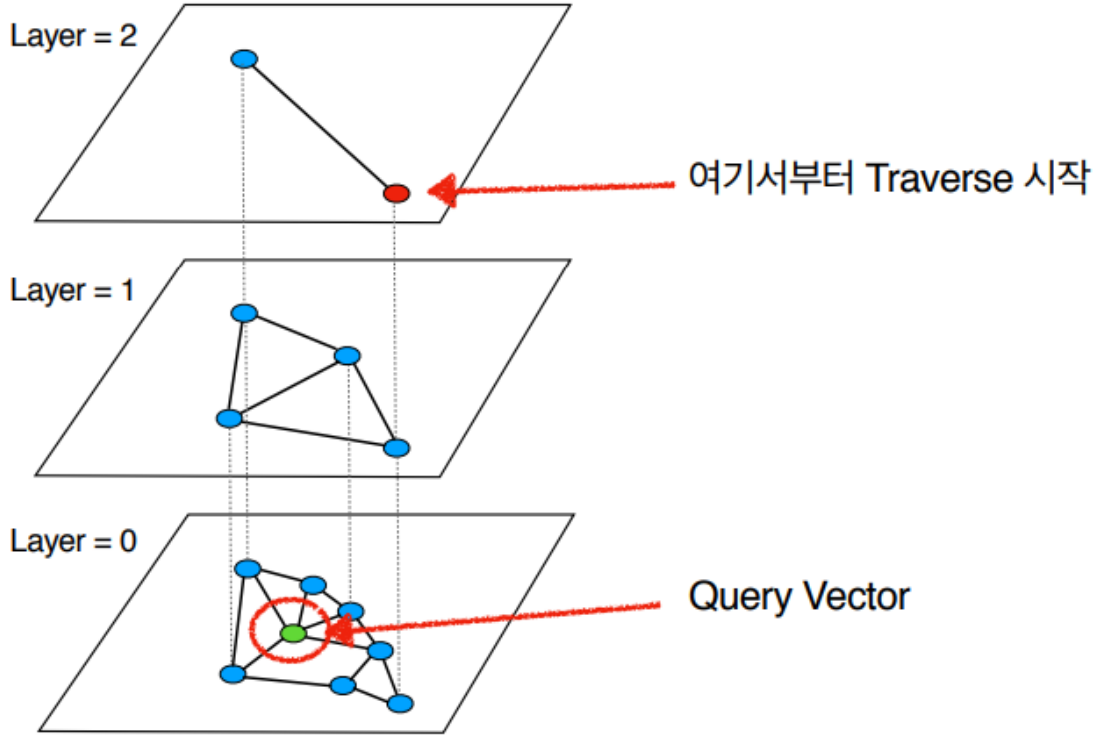


Indexing Passage

✓ HNSW(Hierarchical Navigable Small World graphs) Indexing



- Layer=0에는 모든 노드가 존재
- 그 중 일부가 상위 레벨에도 존재
- 각 노드의 레벨은 Index build 타이밍에 랜덤으로 결정됨



Training

- ✓ 정답 + Negative samples = Batch
- ✓ 정답에 대해 Loss가 감소하도록 학습

Question

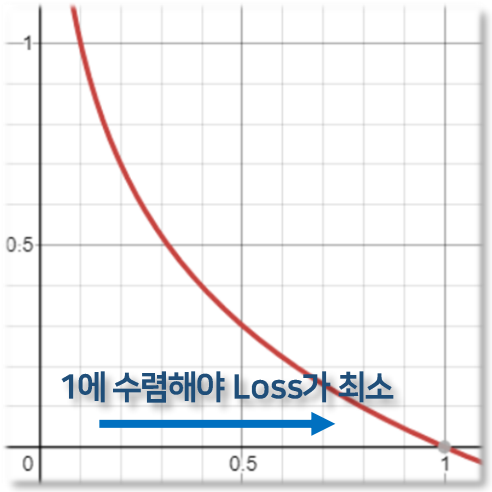
$$L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-)$$

Positive , 정답

Negative Samples

$$= -\log \frac{e^{\mathrm{sim}(q_i, p_i^+)}}{e^{\mathrm{sim}(q_i, p_i^+)} + \sum_{j=1}^{n} e^{\mathrm{sim}(q_i, p_{i,j}^-)}}.$$

-log(x)



- ✓ Negative Log Likelihood(NLL) Loss 사용

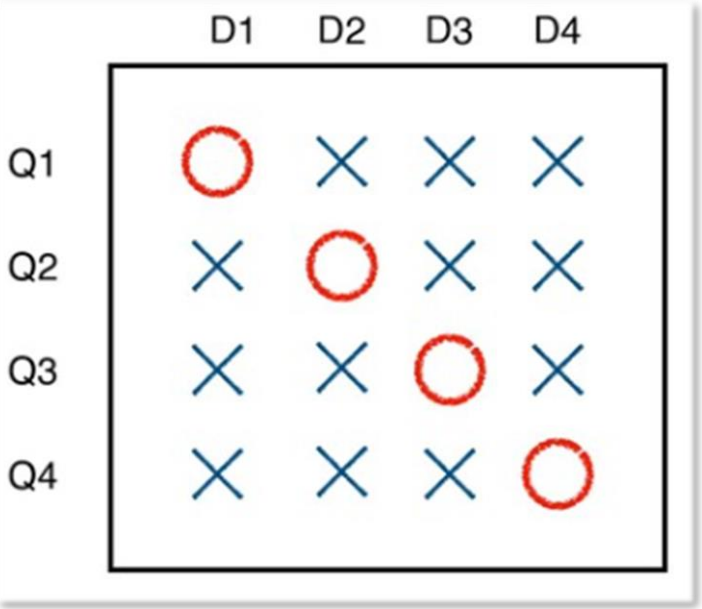
Training : Select Negatives

✓ 총 3가지 방법으로 Negative sample을 선택

- **Random** : 전체 Corpus에서 랜덤으로 샘플링
- **BM25** : 정답을 포함하지 않는 BM25 상위 Passage로 샘플링
- **Gold** (+ In-batch negatives) : Training set의 다른 data의 정답 Passage로 샘플링
  - batch_size x batch_size Metrix로 정리 → Easy and Memory-efficient

Type	#N	IB	Top-5	Top-20	Top-100
Random	7	✗	47.0	64.3	77.8
BM25	7	✗	50.0	63.3	74.8
Gold	7	✗	42.6	63.1	78.3
Gold	7	✓	51.1	69.1	80.8
Gold	31	✓	52.1	70.8	82.1
Gold	127	✓	55.8	73.0	83.1
G.+BM25$^{(1)}$	31+32	✓	65.0	77.3	84.4
G.+BM25$^{(2)}$	31+64	✓	64.5	76.4	84.0
G.+BM25$^{(1)}$	127+128	✓	**65.8**	**78.0**	**84.9**

$$S = QDT$$



Question Answering Datasets

✓ 다양한 QA Dataset으로 Benchmark

- **Natural Questions** : 실제 구글검색어와 위키디피아 문서의 정답 Span으로 구성
- **TriviaQA** : 웹 크롤링한 사소한 퀴즈와 그에 대한 응답으로 구성

✓ Retriever가 리턴한 k개의 Passage에 정답이 없으면 해당 셋은 Skip

Dataset	Train		Dev	Test
Natural Questions	79,168	58,880	8,757	3,610
TriviaQA	78,785	60,413	8,837	11,313
WebQuestions	3,417	2,474	361	2,032
CuratedTREC	1,353	1,125	133	694
SQuAD	78,713	70,096	8,886	10,570

실제 학습한 dataset

Passage Retrieval

✓ 3가지 Retrieval Model을 Benchmark

- **BM25** : BM25로 scoring
- **DPR** : batch_size = 128, In-batch negative, 질문 하나에 BM25 negative 하나 추가
- **BM25 + DPR** : $\mathrm{BM25}(q,p) + \lambda \cdot \mathrm{sim}(q,p)$, λ=1.1

✓ 2가지 Training Type

- **Single** : 단일 Task에 대한 Training set으로 학습
- **Multi** : SQuAD 제외 Training set을 섞어서 학습

① 지문을 보고 질문을 작성한 dataset
② 한정된 주제 → General하지 못함

Training	Retriever	Top-20					Top-100				
		NQ	TriviaQA	WQ	TREC	SQuAD	NQ	TriviaQA	WQ	TREC	SQuAD
None	BM25	59.1	66.9	55.0	70.9	68.8	73.7	76.7	71.1	84.1	80.0
Single	DPR	78.4	79.4	73.2	79.8	63.2	85.4	**85.0**	81.4	89.1	77.2
	BM25 + DPR	76.6	79.8	71.0	85.2	**71.5**	83.8	84.5	80.5	92.7	**81.3**
Multi	DPR	**79.4**	78.8	**75.0**	**89.1**	51.6	**86.0**	84.7	**82.9**	93.9	67.6
	BM25 + DPR	78.0	**79.9**	74.7	88.5	66.2	83.9	84.4	82.3	**94.1**	78.6

Question Answering

- ✓ Small dataset(WQ, TREC)에서 Multiple dataset Training이 성능향상 ↑
- ✓ 가장 최신 모델인 REALM을 넘어 SOTA 달성

Training	Model	NQ	TriviaQA	WQ	TREC	SQuAD
Single	BM25+BERT (Lee et al., 2019)	26.5	47.1	17.7	21.3	33.2
Single	ORQA (Lee et al., 2019)	33.3	45.0	36.4	30.1	20.2
Single	HardEM (Min et al., 2019a)	28.1	50.9	-	-	-
Single	GraphRetriever (Min et al., 2019b)	34.5	56.0	36.4	-	-
Single	PathRetriever (Asai et al., 2020)	32.6	-	-	-	**56.5**
Single	$\text{REALM}_{\text{Wiki}}$ (Guu et al., 2020)	39.2	-	40.2	46.8	-
Single	$\text{REALM}_{\text{News}}$ (Guu et al., 2020)	40.4	-	40.7	42.9	-
Single	BM25	32.6	52.4	29.9	24.9	38.1
	DPR	**41.5**	56.8	34.6	25.9	29.8
	BM25+DPR	39.0	57.0	35.2	28.0	36.7
Multi	DPR	**41.5**	56.8	**42.4**	49.4	24.1
	BM25+DPR	38.8	**57.9**	41.1	**50.6**	35.8

Conclusion

- ✓ Sparse Retrieval < Dense Retrieval
- ✓ 심플한 Dual-encoder 구조로 성능향상
  - ORQA처럼 Pre-training없이 Only fine-tunning
  - 복잡한 구조나 수식 필요없이 내적으로 Similarity만 계산하면 끝
- ✓ SOTA 달성