Dense Passage Retrieval for Open-Domain Question Answering

EMNLP 2020

# of citations 2886

Catholic University of Korea / Biomedical Signal Processing Lab
박수형 pshpulip22@catholic.ac.kr

Contents

Background

Background

open-domain question answering

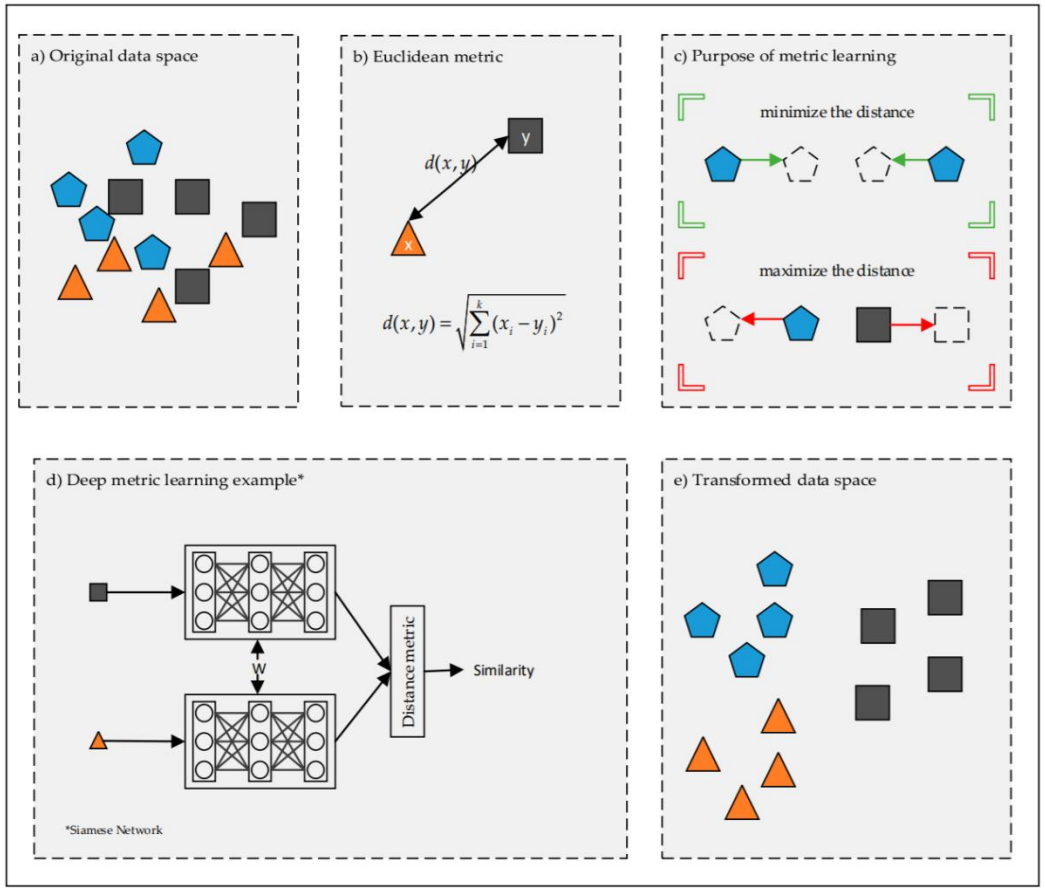
- Open-Domain Question Answering: 사실 적인 질문(e.g (G)-IDLE의 데뷔일은 언제인가?)이 주어졌을 때 시스템은 다양한 주제를 포함한 방대한 corpus를 사용하여 질문에 답해야함
  - 말뭉치에 D개의 문서 $d_1, d_2, \cdots, d_D$가 포함되어있다고 가정
  - 각 문서를 기본 검색 단위로서 동일한 길이의 텍스트 문단으로 분할(chunking) → M개의 문단 $C = \{p_1, p_2 \cdots, p_M\}$
  - p_i는 token sequence $w_1^{(i)}, w_2^{(i)}, \cdots, w_{|p_i|}^{(i)}$로 볼 수 있으며, q에 대해 답변할 수 있는 p_i중 하나에서 질문에 대한 답변을 포함한 token sequence를 찾는 것이 과제

- 다양한 도메인을 다루기 위해 corpus는 수백만개의 문서에서 reader를 통해 답변 추출 전 retriever를 사용하여 검색 진행
  - $R: (q, C) \rightarrow C_{\mathcal{F}}$
  - $C_{\mathcal{F}} \subseteq C$를 반환
  - $|C_{\mathcal{F}}| = k \ll |C|$
  - top-k retrieving accuracy로 평가

Background

Metric Learning

- input data 간의 거리를 학습하는 것
- 즉, input data가 존재하고, 이 둘 간의 거리/유사도를 알고 있다면 이를 맞추어 나가는 과정
- 이를 통해, embedding을 학습

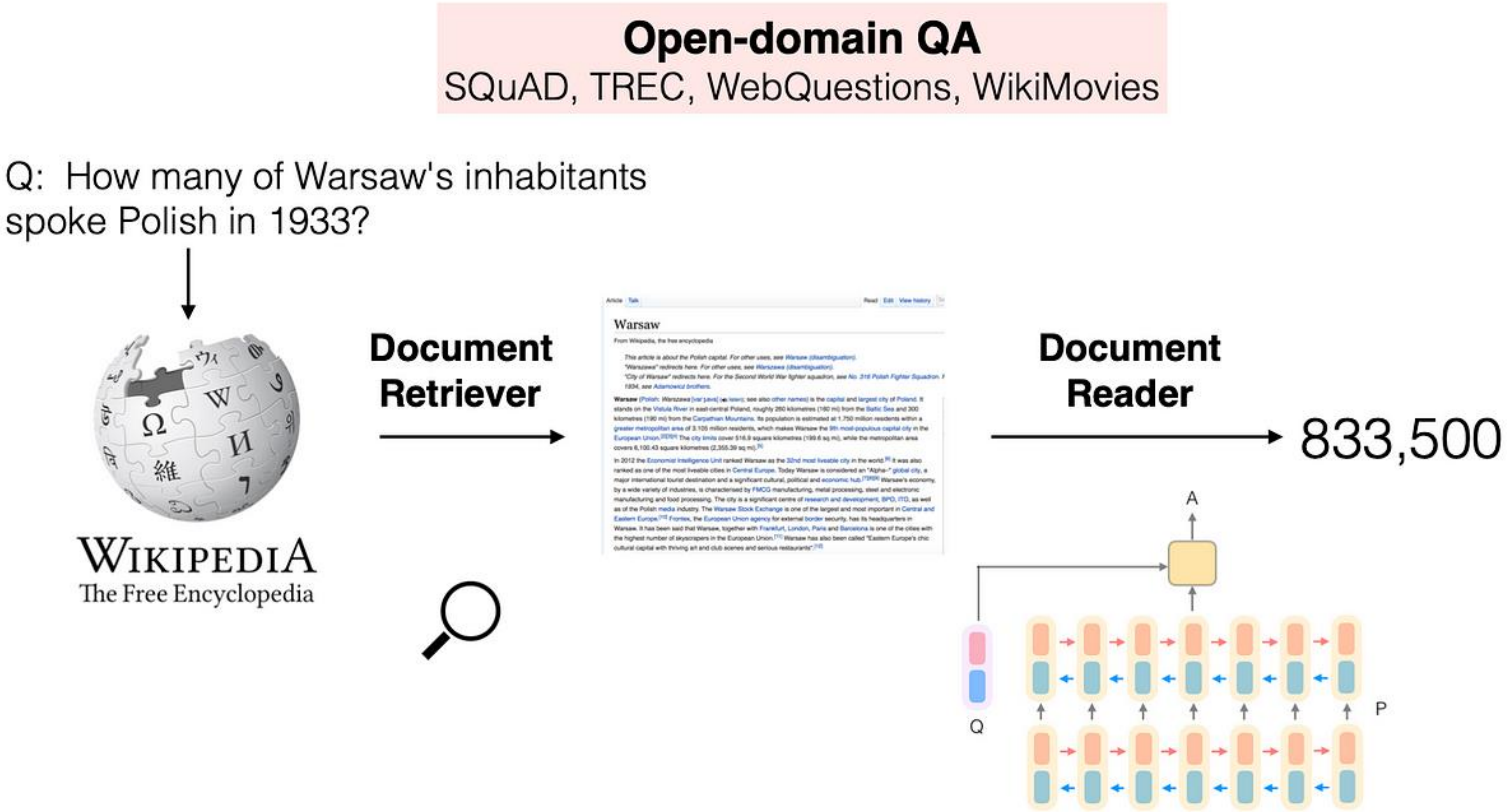


Introduction

Introduction

- 초기 QA system: 여러 구성 요소들로 이루어진 복잡한 구조 → NLU 모델의 발전으로 두단계의 framework로 분류가능
  - passage retriever: 질문에 대한 답이 포함된 작은 부분집합 선택
  - Machine reader가 검색된 passage를 철저히 검토하여 답 식별

  → 성능 저하 발생



Introduction

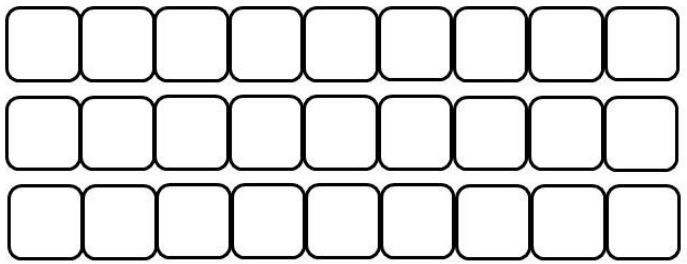
- 일반적인 open-domain question answering 검색방법으로 TF-IDF 또는 BM25 사용

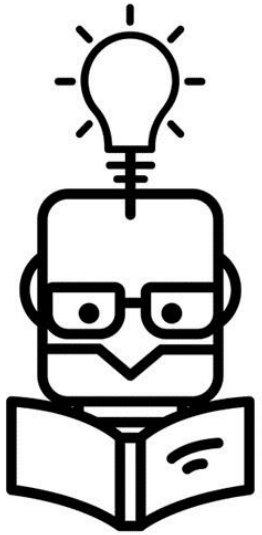
  → keyword를 효율적으로 매칭, 질문과 문맥을 고차원적인 sparse vector로 표현 가능
- dense representation encoding은 디자인상 sparse representation과 상호 보완적
- 키워드 기반의 검색 방법은 같은 의미이나 단어가 다른 경우 검색 불가능 (e.g bad guy & villain)

TF-IDF

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$

$$idf(t,D) = log\frac{N}{|\{d \in D : t \in d\}|}$$





$$\text{BM25}(t,d) = \sum_{t \in q} IDF(t) \cdot \frac{f(t,d) \cdot (k_1+1)}{f(t,d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{\text{avgdl}})}$$

where:

- $f(t,d)$ is the frequency of t in d,
- $|d|$ is the length of document d,
- avgdl is the average document length in the collection,
- k_1 and b are hyperparameters that control the weighting.

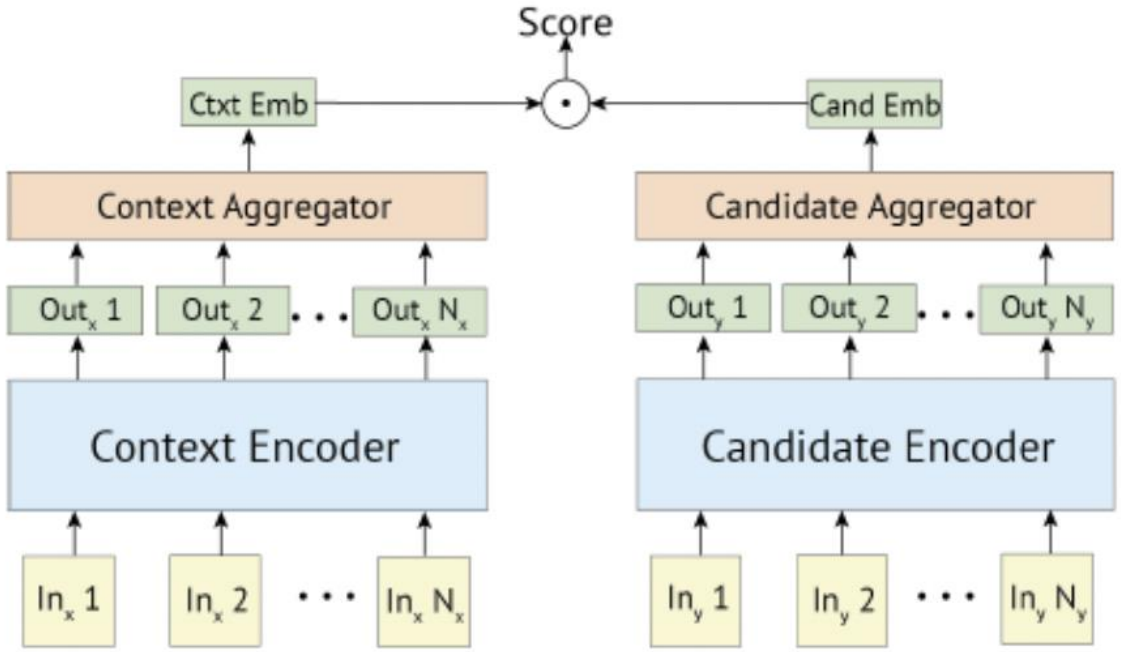
Introduction

- dense representation을 학습하려면 많은 수의 질문과 문맥이 pair로 존재해야함
- ORQA 이전까지는 dense retrieving 방법이 open-domain question answering에서 TF-IDF/BM25를 능가 입증 X

- ORQA에서는 BM25를 능가할 수 있음을 입증하였으나 computational cost가 높으며, context encoder가 학습X
  → 해당 representation이 최적이 아닐 수 있음

Introduction

- pretrain없이, 질문과 문단 pair만을 사용해 더 나은 dense embedding model을 training할 수 있는지 탐구
- BERT 사전학습 모델과 Bi-encoder architecture를 활용하여, 질문과 문단 pair를 사용한 효과적인 훈련 방식을 개발
- DPR은 BM25를 큰 차이로 능가하며, open-domain question answering 정확도에서 SOTA 달성



(a) Bi-encoder

Introduction

Contribution

- 적절한 훈련 설정만으로, 기존의 query-passage pair에서 간단한 fine-tuning을 통해 BM25를 크게 능가할 수 있으며, 추가적인 사전학습이 필요하지 않을 수 있음을 입증

- open-domain question answering에서 높은 정확도가 end-to-end QA 정확도를 향상시키며, 최상위 검색 문단에 SOTA model을 적용하여 여러 복잡한 system보다 뛰어난 성능 달성

Dense Passage Retriever (DPR)

Dense Passage Retriever (DPR)

- open-domain question answering에서 검색 구성 요소를 개선하는 것에 집중
- M개의 텍스트 문단 집합이 주어졌을 때 모든 문단을 low dimension이며, continuous한 space에 indexing
- 입력 query에 대한 top-k개의 문단을 효율적으로 검색할 수 있도록 하며, M은 매우 많을 수 있으며, k는 일반적으로 20~100

Dense Passage Retriever (DPR)

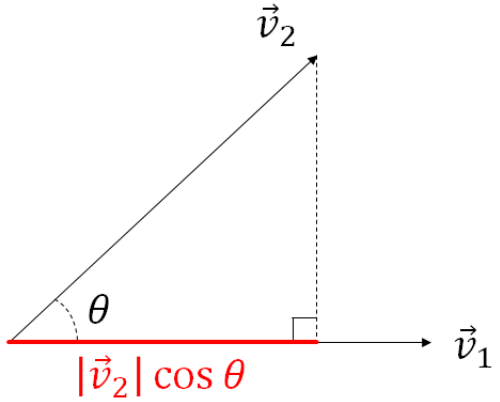
Overview

- DPR은 임의의 텍스트 문단을 d차원의 실수 vector로 mapping하고, 모든 M개의 문단에 대해 index를 구축하는 dense encoder $E_P(\cdot)$
- DPR은 질문을 d차원의 실수 vector로 mapping하는 다른 $E_Q(\cdot)$를 적용하고, query vector에 가장 가까운 k개의 문단 검색
  → Vector Inner Product 사용

$$sim(q,p) = E_Q(q)^T E_P(p)$$

$$\vec{v}_1 \cdot \vec{v}_2 = [a \quad b]\begin{bmatrix} x \\ y \end{bmatrix} = ax + by$$

why?

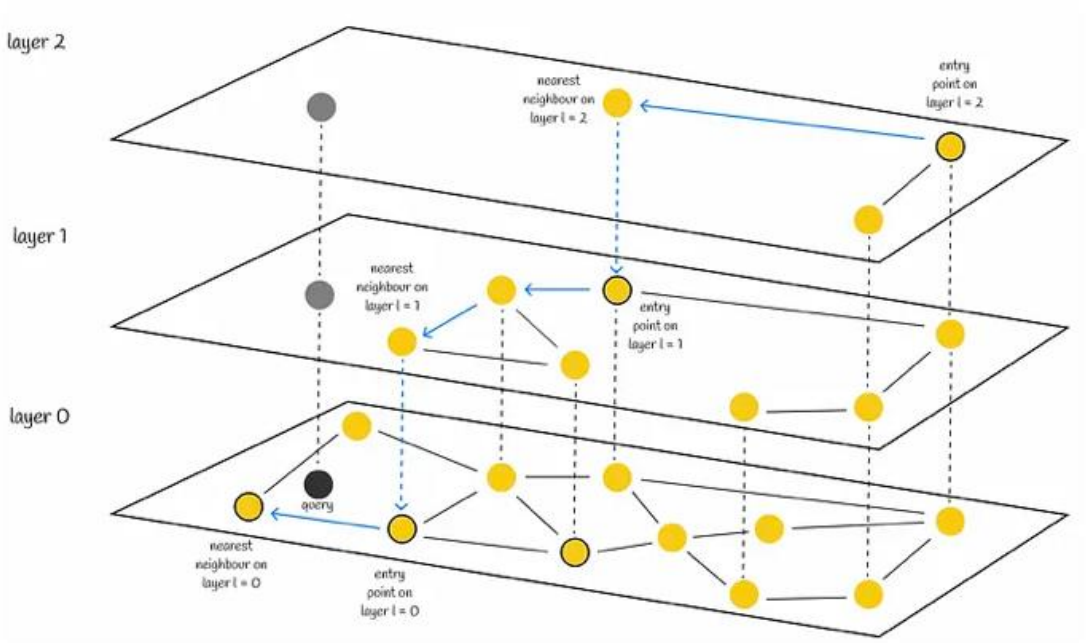


$$\vec{v}_1 \cdot \vec{v}_2 = |\vec{v}_1||\vec{v}_2| \cos\theta$$

Dense Passage Retriever (DPR)

Overview

- **Encoder**: query & passage encoder는 원칙적으로 모든 neural net으로 구현 가능
  두개의 독립적인 BERT를 사용하여 [CLS] token에서 representation을 출력으로 가져와 $d = 768$로 설정
- **Inference**: inference 시, 모든 passage에 대해 passage encoder $E_P(\cdot)$를 적용하고, FAISS를 이용하여 offline 상태로 indexing
  질문 q가 주어졌을 때, 실시간으로 질문의 embedding $v_q = E_Q(q)$를 생성하고 v_q에 가장 가까운 embedding을 가진 top-k개의 passage 검색



Search in HNSW

Dense Passage Retriever (DPR)

Training

- Inner Product가 검색을 위한 좋은 rank function이 되도록 encoder를 훈련 → Metric learning
- 목표는 질문과 문단의 관련 pair가 비관련 pair보다 더 가까운 거리를 갖도록 vector space를 만드는 것

$$D = \{(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-\}_{i=1}^m$$

- m개의 instance로 구성된 training data, 각 instance는 하나의 query q_i와 하나의 관련된 positive passage p_i^+, 관련 없는 negative passage $p_{i,j}^-$들을 포함
- positive passage의 negative log likelihood를 loss function으로 optimization

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

Dense Passage Retriever (DPR)

Training

- **Positive and Negative Passages**: Retrieve task에서 positive example은 명확하게 제공되는 경우가 많음

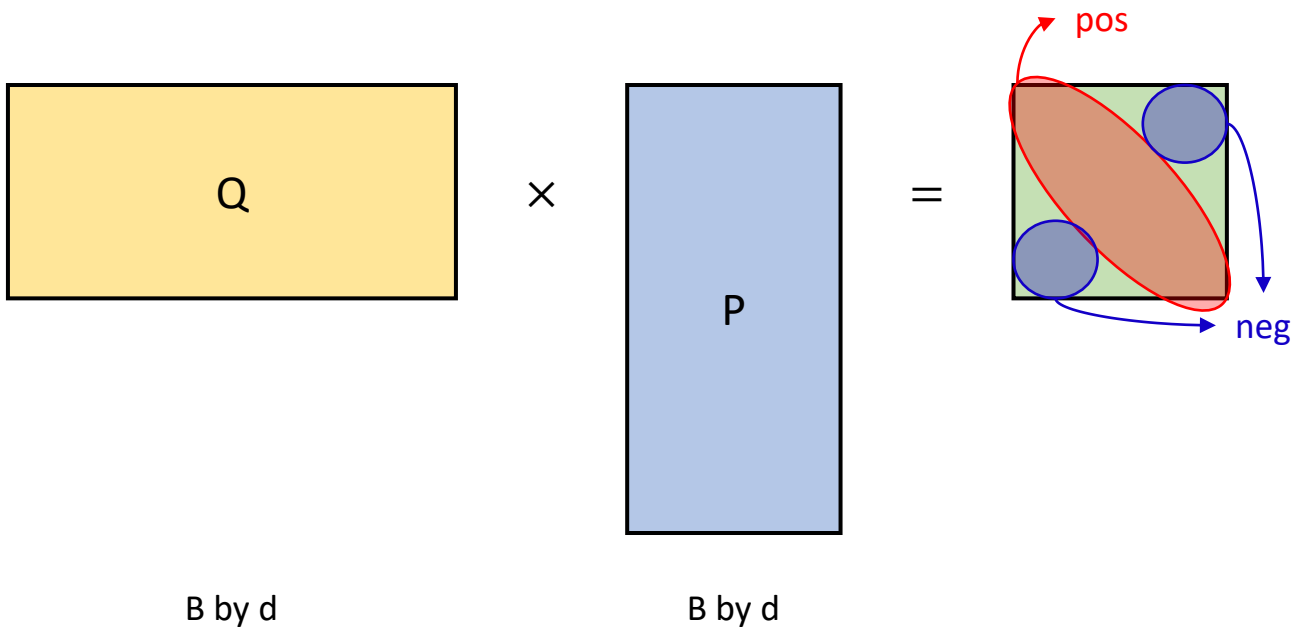
  → Negative example은 매우 큰 풀에서 선택

  - 세가지의 negative example 고려
  - Random: collection에서 임의로 선택한 문단
  - BM25: BM25가 반환한 상위 문단으로, 답변은 포함하진 않으나 대부분의 질문 토큰과 일치
  - Gold: train set에 있는 다른 질문과 짝지어진 positive 문단

Dense Passage Retriever (DPR)

Training

- **In-batch negatives**: mini-batch에 B개의 질문이 있고, 각 질문에 관련된 하나의 문단이 연결되어있다고 가정
- Q와 P는 각각 질문과 문단 embedding matrix이고, 배치 크기는 B
- $S = QP^T$는 question & passage pair를 이루는 유사성 score matrix → computation을 재사용할 수 있으며, 각 batch에서 B^2개의 (query, passage) pair에 대해 효과적으로 학습



Experiment Setup

Experiment Setup

Question Answering Datasets

- Natural Questions (NQ): 실제 Google 검색 질문과 Wikipedia 답변으로 구성된 데이터셋
- TriviaQA: 다양한 웹 출처의 답변을 포함한 질문 데이터셋
- WebQuestions(WQ): Google Suggest API 기반 질문과 Freebase 엔티티 답변 데이터셋
- CuratedTREC(TREC): TREC QA 트랙 질문을 웹 소스에서 가져와 구성한 오픈 도메인 QA 데이터셋
- SQuAD v1.1: 위키백과 문단 기반 질문-답변 데이터셋. 제공된 문단이 없는 경우가 있어 이상적이지 않음

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

Table 1: Number of questions in each QA dataset. The two columns of **Train** denote the original training examples in the dataset and the actual questions used for training DPR after filtering. See text for more details.

Experiment Setup

Question Answering Datasets

- **Selection of postive passages**
  - TREC, WebQuestions, TriviaQA에서는 BM25로 상위 100개 문단 중 답변이 있는 문단을 양성 문단으로 선택하고, 답변이 없으면 해당 질문을 제외
  - SQuAD와 Natural Questions에서는 생성한 문단 후보와 원본 문단을 매칭하여 후보에 추가하며, 매칭 실패 시 질문을 제외

Experiments: Passage Retriever

Experiments: Passage Retriever

Main Results

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

Table 2: Top-20 & Top-100 retrieval accuracy on test sets, measured as the percentage of top 20/100 retrieved passages that contain the answer. *Single* and *Multi* denote that our Dense Passage Retriever (DPR) was trained using individial or combined training datasets (all the datasets excluding SQuAD). See text for more details.

Experiments: Passage Retriever

Ablation Study on Model Training

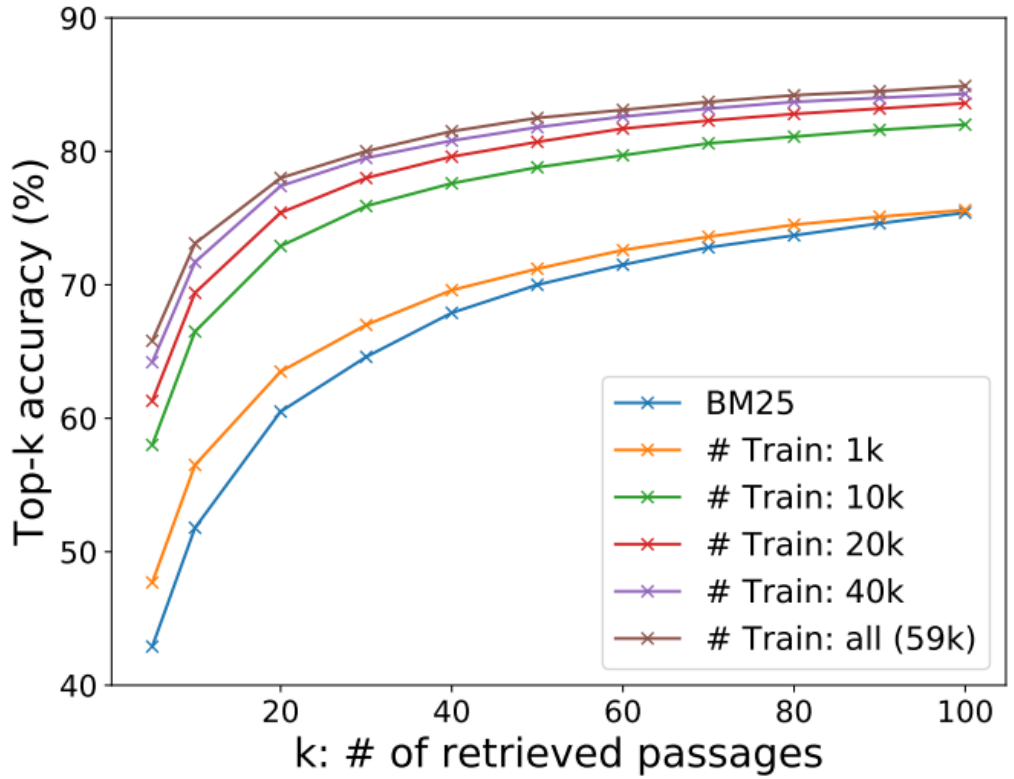


Figure 1: Retriever top-k accuracy with different numbers of training examples used in our dense passage retriever vs BM25. The results are measured on the development set of Natural Questions. Our DPR trained using 1,000 examples already outperforms BM25.

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**

Table 3: Comparison of different training schemes, measured as top-k retrieval accuracy on Natural Questions (development set). #N: number of negative examples, IB: in-batch training. G.+BM25$^{(1)}$ and G.+BM25$^{(2)}$ denote in-batch training with 1 or 2 additional BM25 negatives, which serve as negative passages for all questions in the batch.

Experiments: Question Answering

Experiments: Question Answering

Results

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

Table 4: End-to-end QA (Exact Match) Accuracy. The first block of results are copied from their cited papers. $\text{REALM}_{\text{Wiki}}$ and $\text{REALM}_{\text{News}}$ are the same model but pretrained on Wikipedia and CC-News, respectively. *Single* and *Multi* denote that our Dense Passage Retriever (DPR) is trained using individual or combined training datasets (all except SQuAD). For WQ and TREC in the *Multi* setting, we fine-tune the reader trained on NQ.

Conclusion

Conclusion

- 밀집 검색이 전통적인 희소 검색을 능가하며 대체할 수 있음을 입증했고, 복잡한 모델보다는 기본적인 듀얼 인코더 접근 방식이 효과적임을 확인

- 이를 통해 오픈 도메인 질문 응답 벤치마크에서 새로운 SOTA 달성