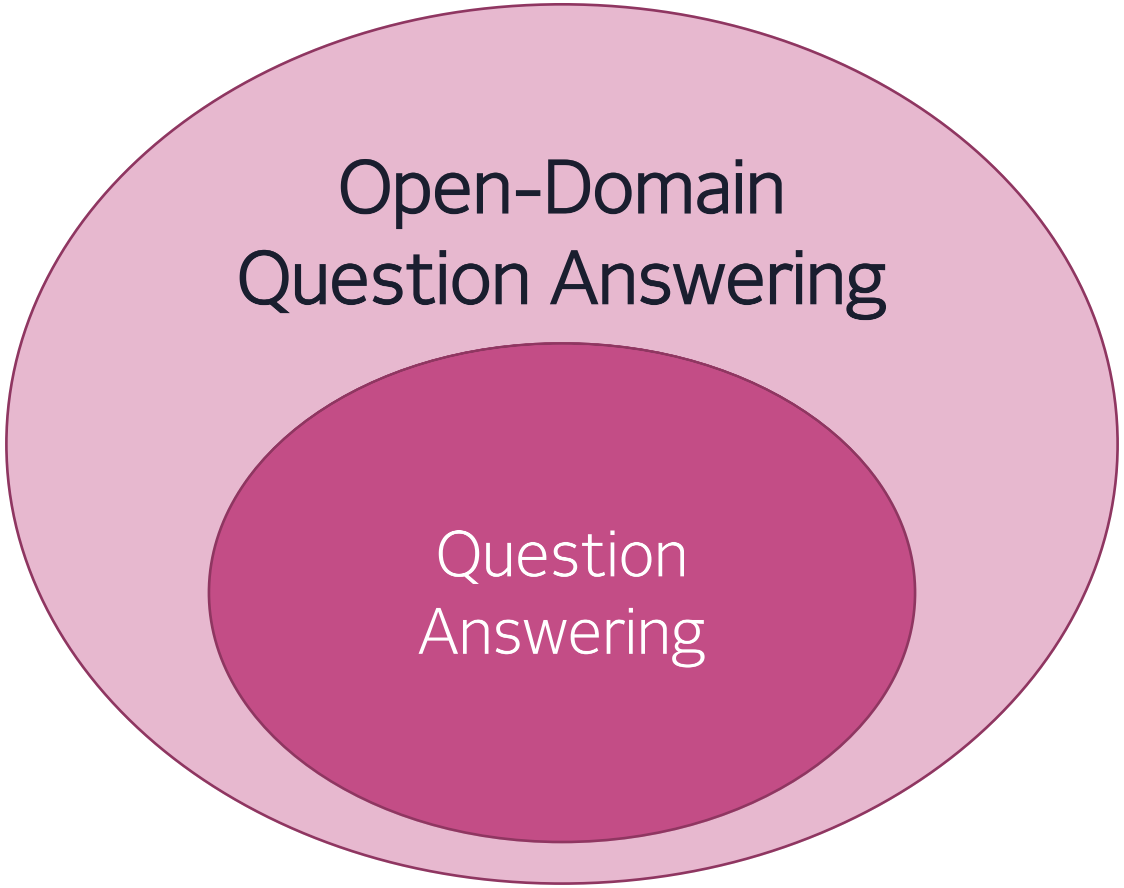
DPR : Dense Passage Retrieval for Open-Domain Question Answering

JUHYEONG LEE, KNUAIR

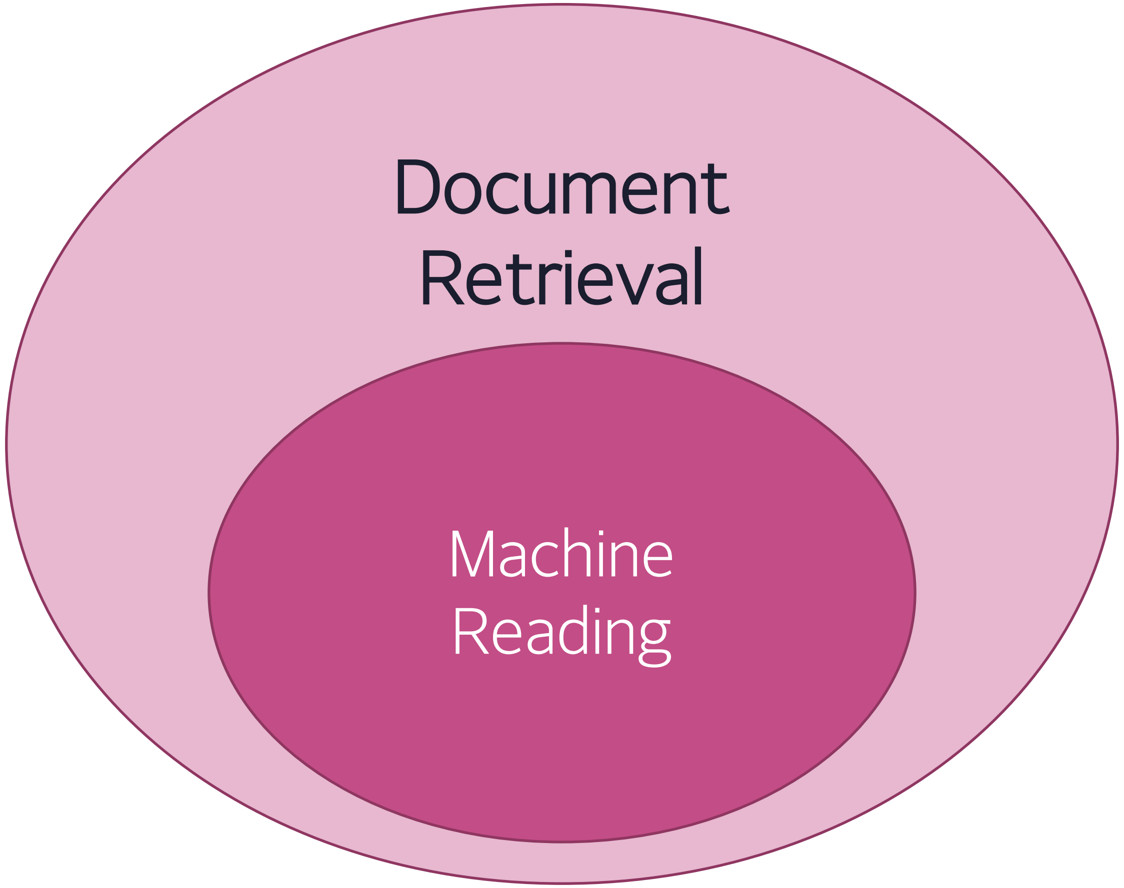
Contents

Karpukhin, Vladimir, et al. "Dense Passage Retrieval for Open-Domain Question Answering." EMNLP, 2020.

Question Answering



Question Answering



Open-domain QA

- Open-domain Question Answering
  - Retrieval + Machine Reading
  - Answers factoid questions using a large collection of docs

Open-domain QA

- Challenges in Open-domain Question Answering
    - Given a factoid question, a system is required to answer it using a large corpus of diversified topics
    - needs to include *efficient retriever component* that can select a small set of relevant texts

Brief History of Open-QA

- Early QA Systems
  - Complicated and consist of multiple components
  - Question Analysis + Document Retreival + Answer Extraction
- Modern QA Systems
  - Context Retriever + Machine Reader

- Performance of both systems *have dependency for previous components*
  → *Pipelining*

Brief History of Open-QA

- Traditional Approach
  - TF-IDF or BM25(upgraded ver. of TF-IDF)
  - *Sparse vector space models*
  - Statistical approach(count-based)

Brief History of Open-QA

Sparse Representation

- Represent sequence in high dimension

Dense Representation

- Represent sequence in lower dimension

Brief History of Open-QA

Sparse Representation

- Represent sequence in high dimension
- Synonyms, paraphrases – mapped independently

Dense Representation

- Represent sequence in lower dimension
- Synonyms, paraphrases – mapped closely

KNUAIR

Brief History of Open-QA

Sparse Representation

- Represent sequence in high dimension
- Synonyms, paraphrases – mapped independently
- Always provides same representations

Dense Representation

- Represent sequence in lower dimension
- Synonyms, paraphrases – mapped closely
- Learnable by adjusting the embedding functions, and provides additional flexibility to have a task-specific representation

Brief History of Open-QA

- Then, does dense representations always outperform sparse one?

- Answer : NO (until 2019)

- Learning a good dense vector representation
  - Needs a large number of labeled examples

Brief History of Open-QA

- ORQA : Open-Retrieval QA using *dense representation*
  - Outperforms BM25 for the first time
  - Proposal : Inverse Cloze Training
    - predicting the blocks that contain the masked sentence, for additional pretraining
  - Two Weeknesses
    - ICT pretraining : consumes intensive computation
    - Context encoder is not fine-tuned

Lee, Kenton, Ming-Wei Chang, and Kristina Toutanova. "Latent Retrieval for Weakly Supervised Open Domain Question Answering." Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019.

Introduction

- Question : Can we train a **better dense embedding model**
  using ① only (a few) pairs of questions and passages,
  ② without additional pretraining?

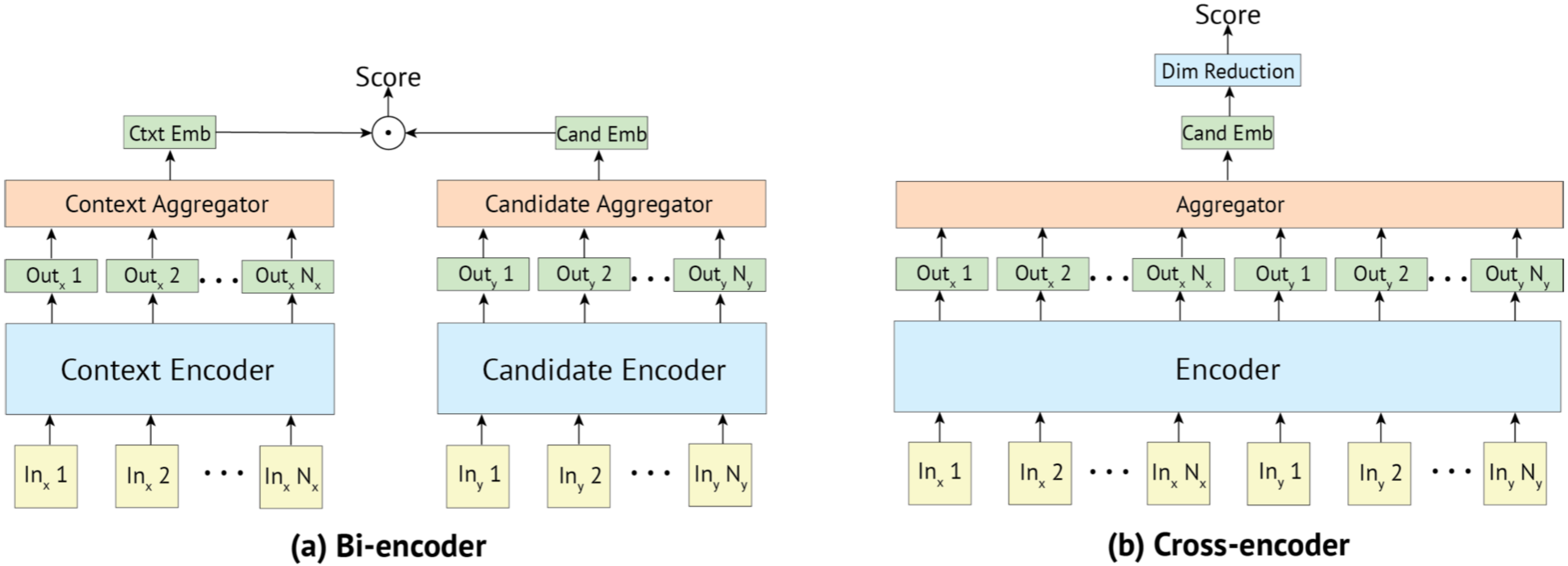
Introduction

- Question : Can we train a better dense embedding model using ① only (a few) pairs of questions and passages, ② without additional pretraining?
- Answer :
  - *Objective* - maximizing inner products of question and relevant passage vectors by comparing all pairs of questions and passages in a batch

Introduction

- Proposal : retrieval can be practically implemented using *dense* representations alone
  - Embeddings are learned from a small number of questions and passages by simple dual-encoder framework

Dual-Encoder vs Cross-Encoder



Reference : Humeau, Samuel, et al. "Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring." ICLR. 2019.

Dual-Encoder vs Cross-Encoder

- Cross-Encoder
  - Perform full (cross)self-attention over a given input and label candidate
  - *Higher accuracy, but slower than dual-encoder*

Reference : Humeau, Samuel, et al. "Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi- sentence Scoring." ICLR. 2019.

Dual-Encoder vs Cross-Encoder

- Cross-Encoder
  - Perform **full (cross)self-attention** over a given input and label candidate
  - *Higher accuracy, but slower than dual-encoder*
- (Bi)Dual-Encoder
  - Perform self-attention over the input and candidate label separately
  - Able to cache the encoded candidates,
    and reuse these representations → *fast prediction*

Reference : Humeau, Samuel, et al. "Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring." ICLR. 2019.

Contribution

- Fine-tuning the question and passage encoders on existing question-passage pairs is sufficient to outperform BM25
  - And additional pretraining may not be needed

Contribution

- **Fine-tuning the question and passage encoders** on existing question-passage pairs is sufficient to outperform BM25
  - <u>And additional pretraining may not be needed</u>

- Verify that a higher retrieval precision translates to a higher end-to-end QA accuracy

Dense Passage Retriever

- The goal of DPR :
  - Given a collection of M text passages,
    - Index all the passages in a low-dimensional and continuous space,
    - Then the top k passages can be retrieved efficiently

  ⇒ Find optimal representations of passages

- M can be very large (about 20M),
  and k is usually small (about 20 ~ 100)

DPR : Overview

- The dense encoder $E_P(\cdot)$
  - Maps any passages to a d-dimensional real-valued vectors
  - Build an index for all the M passages that are used for retrieval

DPR : Overview

- Another dense encoder $E_Q(\cdot)$
  - Maps the questions to a d-dimensional real-valued vectors
  - Retrieves k passages of which vectors are the closest to the question vector

DPR : Overview

- Similarity Measure : Dot Product
- $\mathrm{sim}(q, p) = E_Q(q)^T E_P(P)$

DPR : Overview

- Similarity Measure : Dot Product
- $\text{sim}(q, p) = E_Q(q)^T E_P(P)$
- Why Dot Product?
  - Needs to be *decomposable*
    - Another decomposable sim. func. : L2 Distance, Cosine Distance
  - So that the representations of the passages *can be pre-computed*

DPR : Overview

- Encoder architecture : BERT$_{\text{(base, uncased)}}$

- Inference time : top k passages retrieval with FAISS

- FAISS : library for similarity search and clustering of dense vectors

DPR : Training

- Goal : Create a vector space such that
  - *Relevant pairs* of questions and passages
    will have *smaller distance* than *the irrelevant ones*

- Training data

$$\mathcal{D} = \{\langle \underbrace{q_i}_{\text{question}}, \underbrace{p_i^+}_{\text{one relevant passage}}, \underbrace{p_{i,1}^-, \cdots, p_{i,n}^-}_{n \text{ irrelevant passages}} \rangle\}_{i=1}^{m}$$

DPR : Training

- Loss function : NLL Loss

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

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

DPR : Training

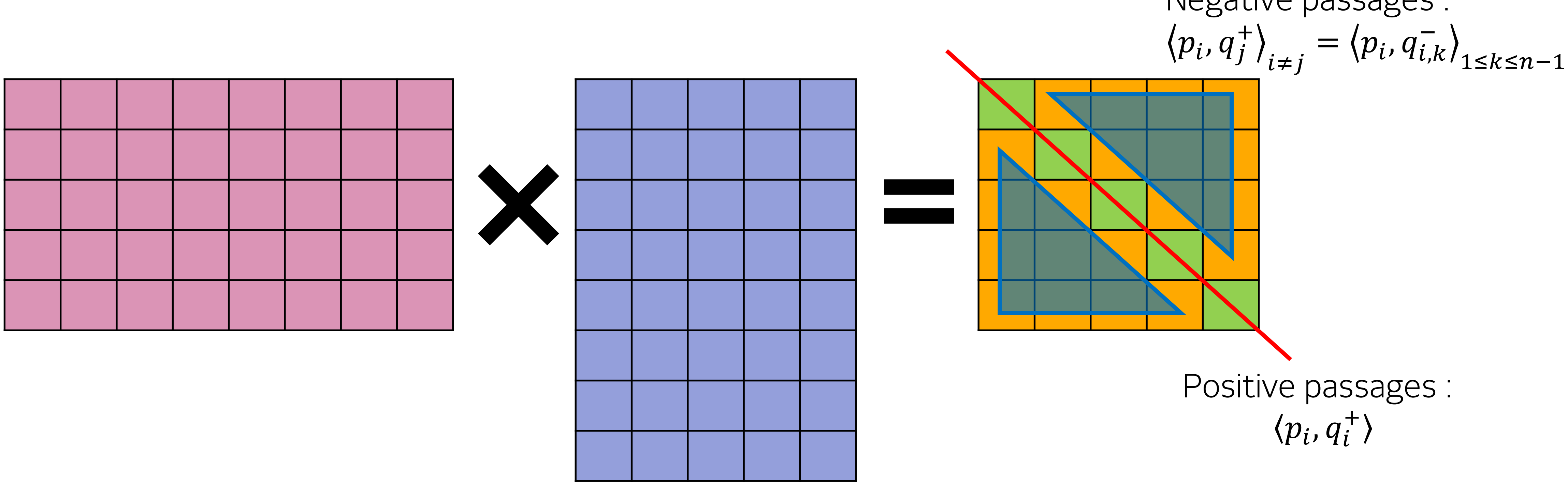
- Selecting negative examples
  - Assume that there exists *100,000 q-p pairs.*
  - For each questions, there are one positive passage
  - Rest of 99,999 passages are not positive passage
  - ∴ There exists *99,999 negative passages* per question.
    - *Need to be selected from an extremely large pool*

DPR : Training

- Selecting negative examples
  - Could be decisive for learning a high-quality encoder
  - Method 1 : Random
  - Method 2 : BM25
    - top passages from BM25 which do not contain the answer but match most question tokens
  - Method 3 : Gold
    - positive passages paired with other questions which appear in the training set

DPR : Training

- In-batch Negative



Experiments : Passage Retrieval

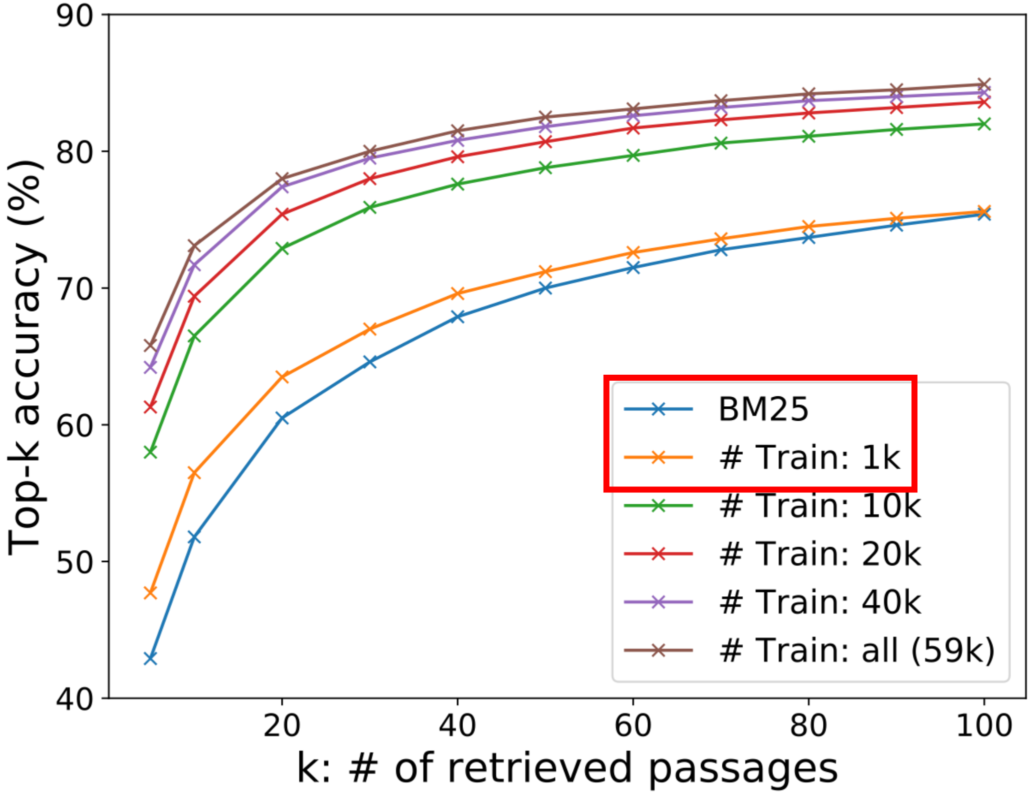
- DPR vs BM25

Training	Retriever	Top-20 NQ	Top-20 TriviaQA	Top-20 WQ	Top-20 TREC	Top-20 SQuAD	Top-100 NQ	Top-100 TriviaQA	Top-100 WQ	Top-100 TREC	Top-100 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 Retrieval

- DPR vs BM25



Experiments : Passage Retrieval

- Sample efficiency
  - DPR using only 1K examples already outperforms BM25
  - With a general pretrained LM(like BERT), it is possible to train a **high-quality dense retriever with a small number of examples**

Experiments : Passage Retrieval

- In-batch negative training
  - Adding a single BM25 negative passage improves the result substantially

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 : Passage Retrieval

- Similarity and Loss
  - Similarity
    - DP ≈ L2 > Cosine
  - Loss
    - NLL ≈ Triplet

- c.f.) Triplet Loss
  - anchor-positive-negative

Sim	Loss	Retrieval Accuracy Top-1	Top-5	Top-20	Top-100
DP	NLL	**44.9**	**66.8**	**78.1**	**85.0**
	Triplet	41.6	65.0	77.2	84.5
L2	NLL	43.5	64.7	76.1	83.1
	Triplet	42.2	66.0	**78.1**	84.9

Table 6: Retrieval Top-k accuracy on the development set of Natural Questions using different similarity and loss functions.

Experiments : End-to-End QA

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

- Dense retrieval can outperform and potentially replace the traditional sparse retrieval component in open-domain QA
- Dual-encoder
- In-batch negative approach
- Indicate that more complex model frameworks or sim. func do not necessarily provide additional values

Q&A