Part V

Dense Retriever and **End-to-end** Training

Key questions

Q: How many of Warsaw's inhabitants spoke Polish in 1933?

Document Retriever

Article Talk Read Edit View history

Warsaw

From Wikipedia, the free encyclopedia

This article is about the Polish capital. For other uses, see Warsaw (disambiguation).
"Warszawa" redirects here. For other uses, see Warszawa (disambiguation).
"City of Warsaw" redirects here. For the Second World War fighter squadron, see No. 316 Polish Fighter Squadron. F
1934, see Adamowicz brothers.

Warsaw (Polish: *Warszawa* [varˈʂava] (listen); see also other names) is the capital and largest city of Poland. It stands on the Vistula River in east-central Poland, roughly 260 kilometres (160 mi) from the Baltic Sea and 300 kilometres (190 mi) from the Carpathian Mountains. Its population is estimated at 1.750 million residents within a greater metropolitan area of 3.105 million residents, which makes Warsaw the 9th most-populous capital city in the European Union.[2][3][4] The city limits cover 516.9 square kilometres (199.6 sq mi), while the metropolitan area covers 6,100.43 square kilometres (2,355.39 sq mi).[5]

In 2012 the Economist Intelligence Unit ranked Warsaw as the 32nd most liveable city in the world.[6] It was also ranked as one of the most liveable cities in Central Europe. Today Warsaw is considered an "Alpha–" global city, a major international tourist destination and a significant cultural, political and economic hub.[7][8][9] Warsaw's economy, by a wide variety of industries, is characterised by FMCG manufacturing, metal processing, steel and electronic manufacturing and food processing. The city is a significant centre of research and development, BPO, ITO, as well as of the Polish media industry. The Warsaw Stock Exchange is one of the largest and most important in Central and Eastern Europe.[10] Frontex, the European Union agency for external border security, has its headquarters in Warsaw. It has been said that Warsaw, together with Frankfurt, London, Paris and Barcelona is one of the cities with the highest number of skyscrapers in the European Union.[11] Warsaw has also been called "Eastern Europe's chic cultural capital with thriving art and club scenes and serious restaurants".[12]

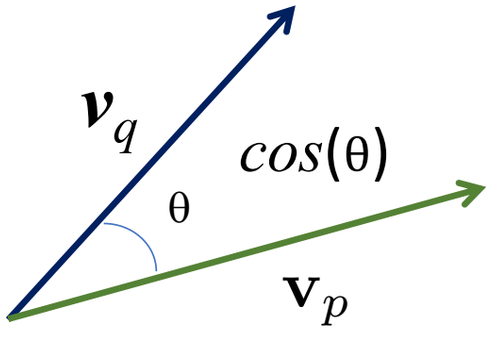
Document Reader

833,500

Can we also train the retriever component?

Can we use dense representations for retrieval?

Sparse vs dense representations for retrieval



$d_1 >> d_2$

sparse repr: [0… 1 …. 1 …0..1] $\in \mathbb{R}^{d_1}$

dense repr: [1.03, -5.72, 6.42, .., 9.91] $\in \mathbb{R}^{d_2}$

sparse

"How many provinces did the Ottoman empire contain in the 17th century?"

"What part of the atom did Chadwick discover?"

dense

"Who is the **bad guy** in lord of the rings?"

*Sala Baker is an actor and stuntman from New Zealand. He is best known for portraying the **villain** Sauron in the Lord of the Rings trilogy by Peter Jackson.*

They capture complementary information; Dense representations have never been shown to outperform sparse representations in open-domain QA before 2019!

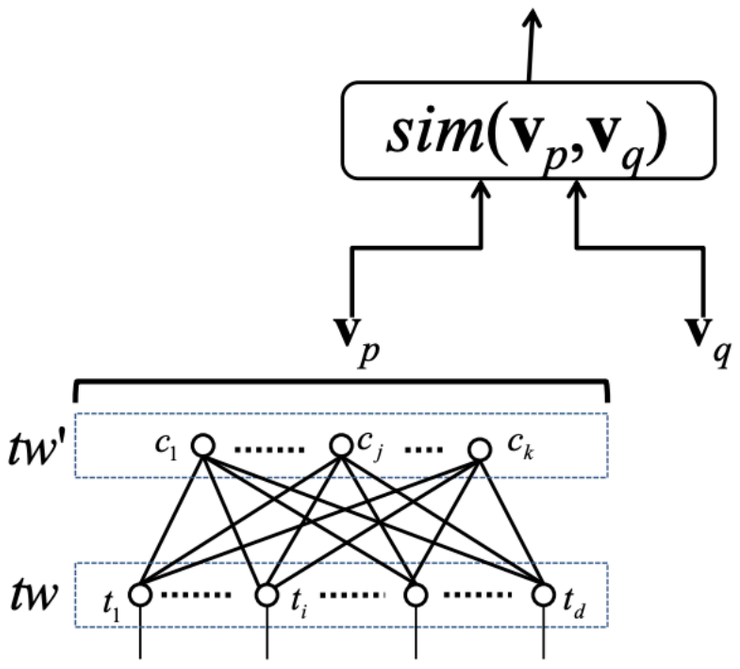
Why dense retrieval **now**?

It is a difficult problem: we need to *encode*, *index* and *search* from 5M documents or 30M paragraphs or 60B phrases.

Learning Discriminative Projections for Text Similarity Measures

Wen-tau Yih **Kristina Toutanova** **John C. Platt** **Christopher Meek**
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA

CoNLL'2011 best paper



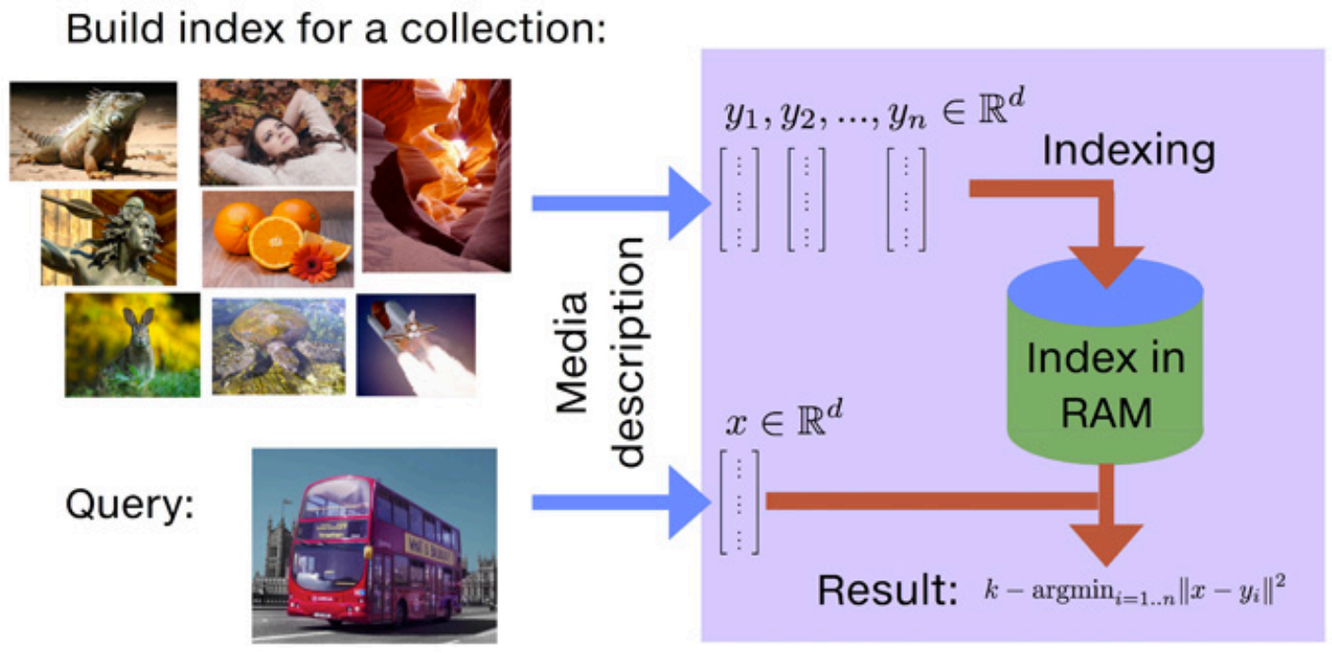
- Cross-lingual document retrieval
- Ad relevance prediction
- Web search ranking

Why dense retrieval **now**?

- It is actually not easy to make these dense models "work".
  - Needs large enough labeled data (e.g., 82M query-doc pairs from user clicks).

We have pre-trained models now!

- Now we have much better techniques and tools to support fast maximum inner product search (MIPS):
  - In-memory data structure and indexing schemes



e.g., FAISS
[Johnson et al., 2017]

ORQA: **O**pen-**R**etriever **Q**uestion **A**nswering

[Lee et al., 2019]

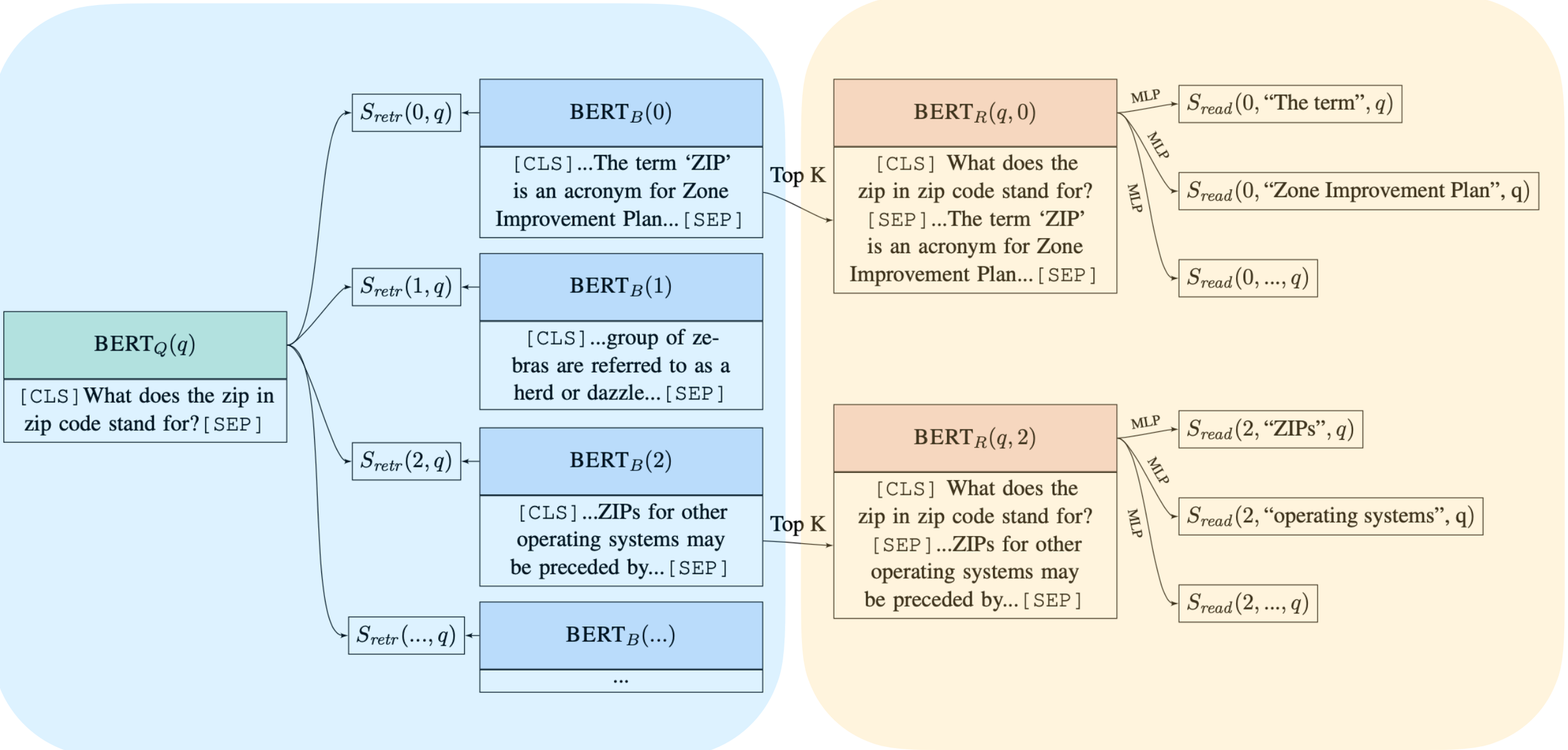
First model to learn retriever and reader jointly

Key contributions:

- Both retriever and reader are learnable with NNs (= BERT)
- Only learned from question-answer pairs; No reading comprehension datasets! $\mathcal{D}_{\text{QA}} = \{(Q_i, A_i)\}$
- A new pre-training task called Inverse Cloze Task (ICT) to address the challenging retrieval problem.

Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

ORQA: Open-Retriever Question Answering



Information Retrieval

Reading Comprehension

Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

ORQA: Overview

Notations: b - text block, s - a span of text within b, q - question

$$S(b, s, q)$$

Fixed-length blocks as "passages": 288 wordpieces $\Longrightarrow$13M in total

Each block has 2000 possible answer spans

Modeling

$$S(b, s, q) = S_{retr}(b, q) + S_{read}(b, s, q)$$

Inference

$$a^* = \text{TEXT}(\underset{b,s}{\text{argmax}}\, S(b, s, q))$$

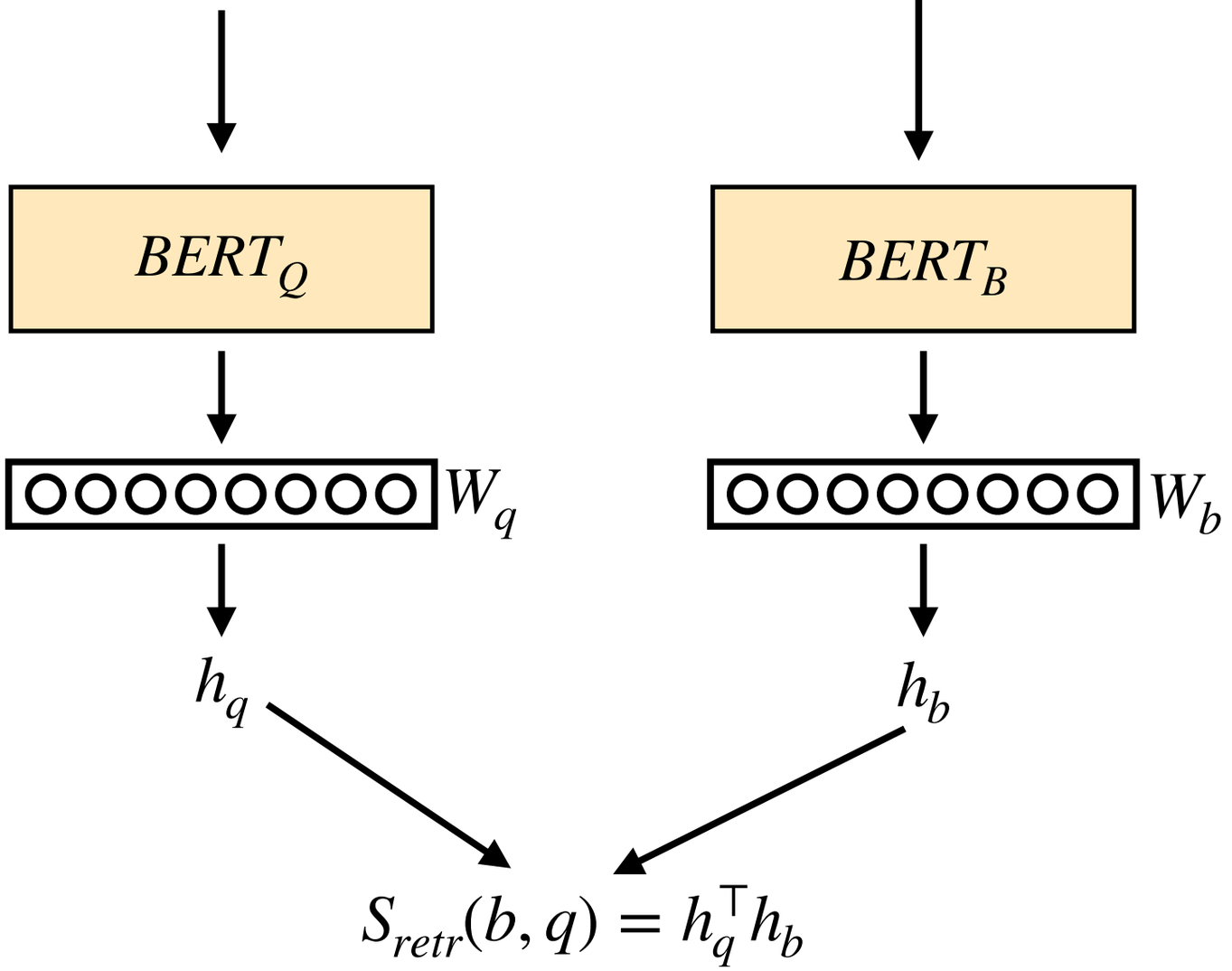
Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

Retriever score: $S_{retr}(b, q)$

$$h_q = \mathbf{W_q}\mathbf{BERT}_Q(q)[\texttt{CLS}]$$
$$h_b = \mathbf{W_b}\mathbf{BERT}_B(b)[\texttt{CLS}]$$
$$S_{retr}(b, q) = h_q^\top h_b$$

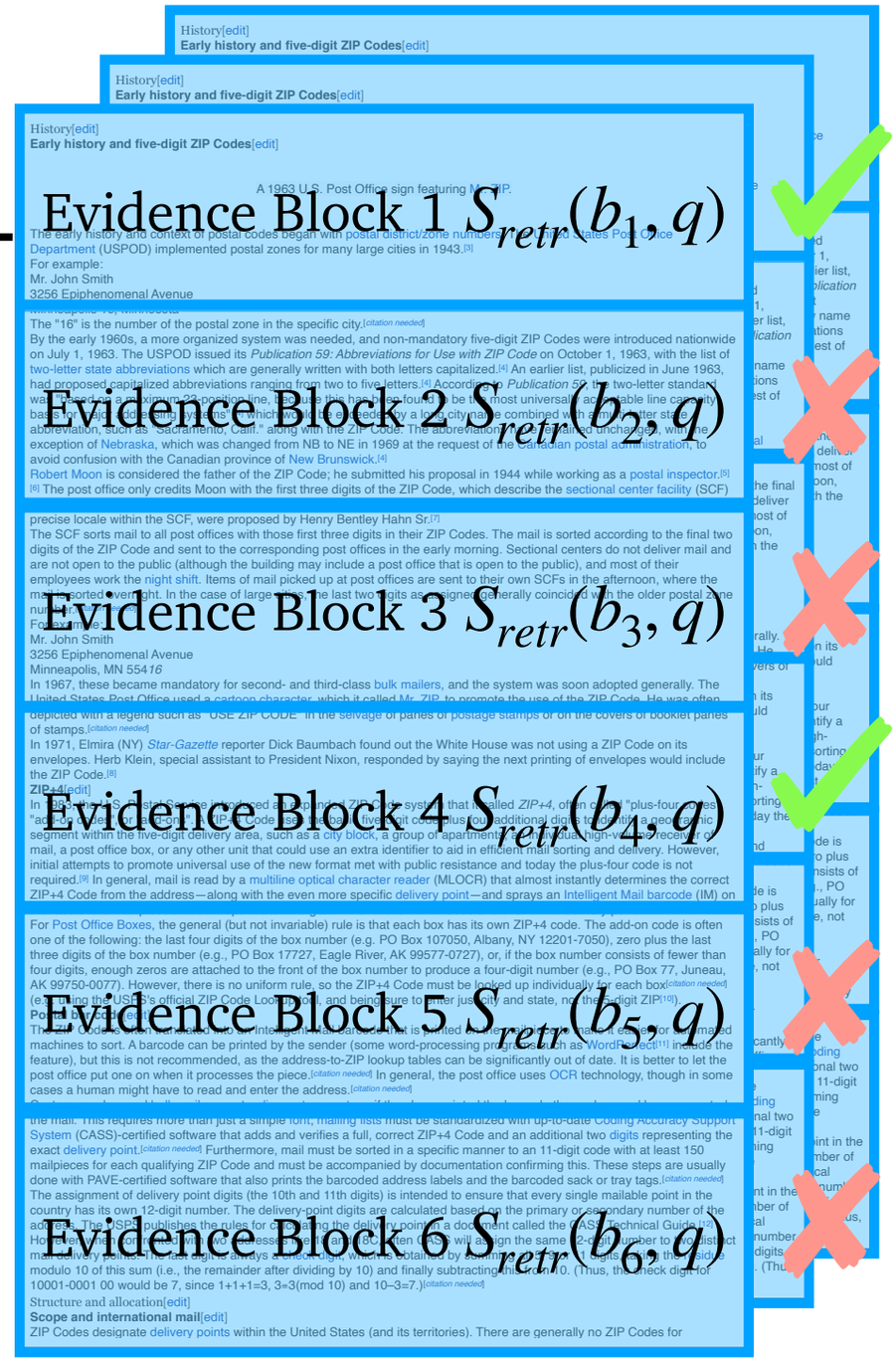
Question q

What does the zip in zip code stand for?



Each evidence block b

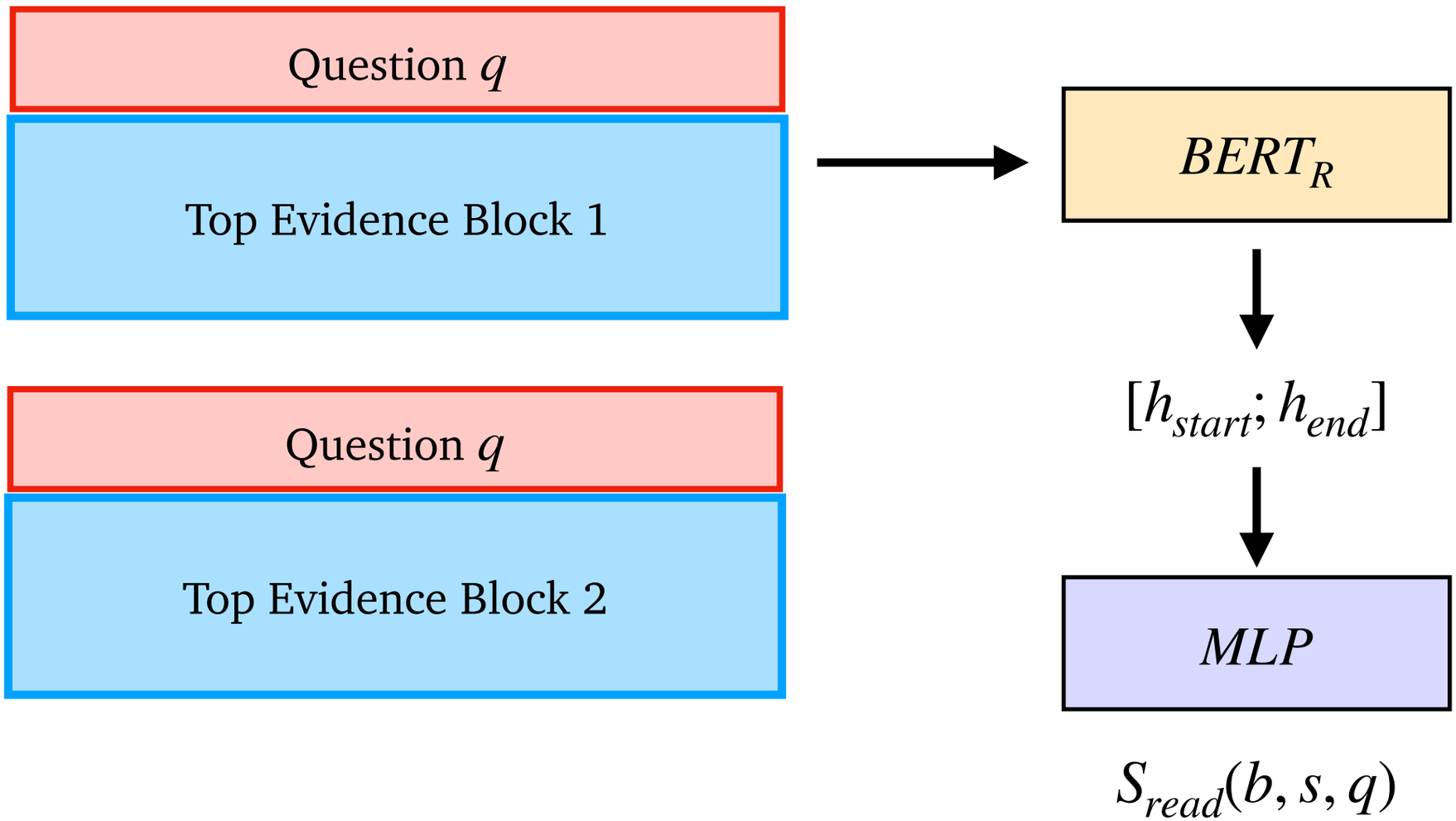
All of Wikipedia: select top K



Reader score: $S_{read}(b, s, q)$

$$h_{start} = \text{BERT}_R(q, b)[\text{START}(s)]$$
$$h_{end} = \text{BERT}_R(q, b)[\text{END}(s)]$$
$$S_{read}(b, s, q) = \text{MLP}([h_{start}; h_{end}])$$

Top-K evidence blocks concatenated with question



Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

How is this model learned?

$$P(b, s|q) = \frac{\exp(S(b, s, q))}{\sum_{b' \in \text{TOP}(k)} \sum_{s' \in b'} \exp(S(b', s', q))}$$

k=5, TOP(k) = the top k retrieved blocks according to $S_{\text{retr}}(b, q)$

How to update BERT_B for all the blocks??

How if top 5 (out of 13M) blocks don't contain the answer at all?

Loss function $L_{\text{full}}(q, a) = -\log \sum_{b \in \text{TOP(k)}} \sum_{s \in b, a = \text{TEXT}(s)} P(b, s \mid q)$

Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

How is this model learned?

Early learning: consider a larger set of evidence blocks

$c = 5000$

$$P_{\text{early}}(b|q) = \frac{\exp(S_{retr}(b,q))}{\sum_{b' \in \text{TOP}(c)} \exp(S_{retr}(b',q))}$$

$$L_{\text{early}}(q,a) = -\log \sum_{b \in \text{TOP}(c),\ a \in \text{TEXT}(b)} P_{\text{early}}(b|q)$$

Final loss

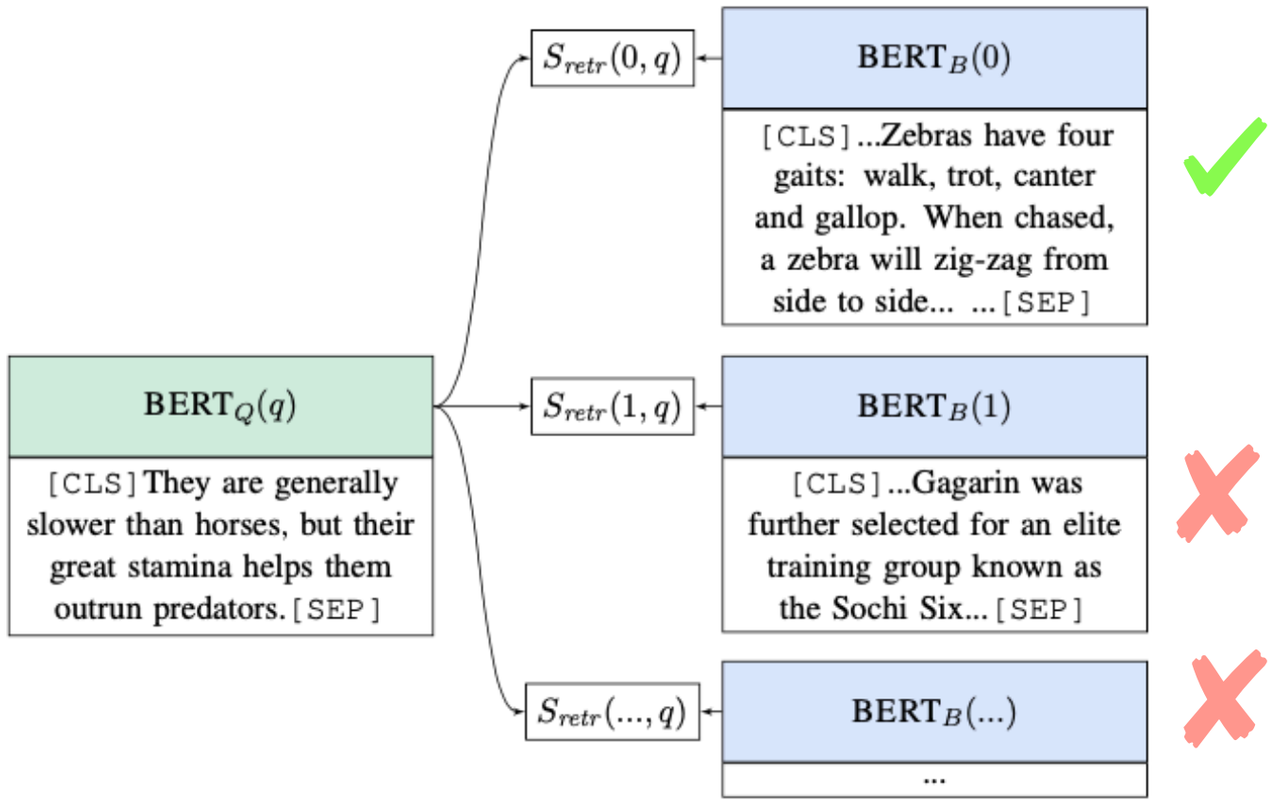
$$L(q,a) = L_{\text{early}}(q,a) + L_{\text{full}}(q,a)$$

Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

Pre-training: Inverse Cloze Task (ICT)

Key idea: a sentence is treated as a *pseudo-question*, and its context is treated as *pseudo-evidence*. The goal is to predict the correct context among a set of other random options.

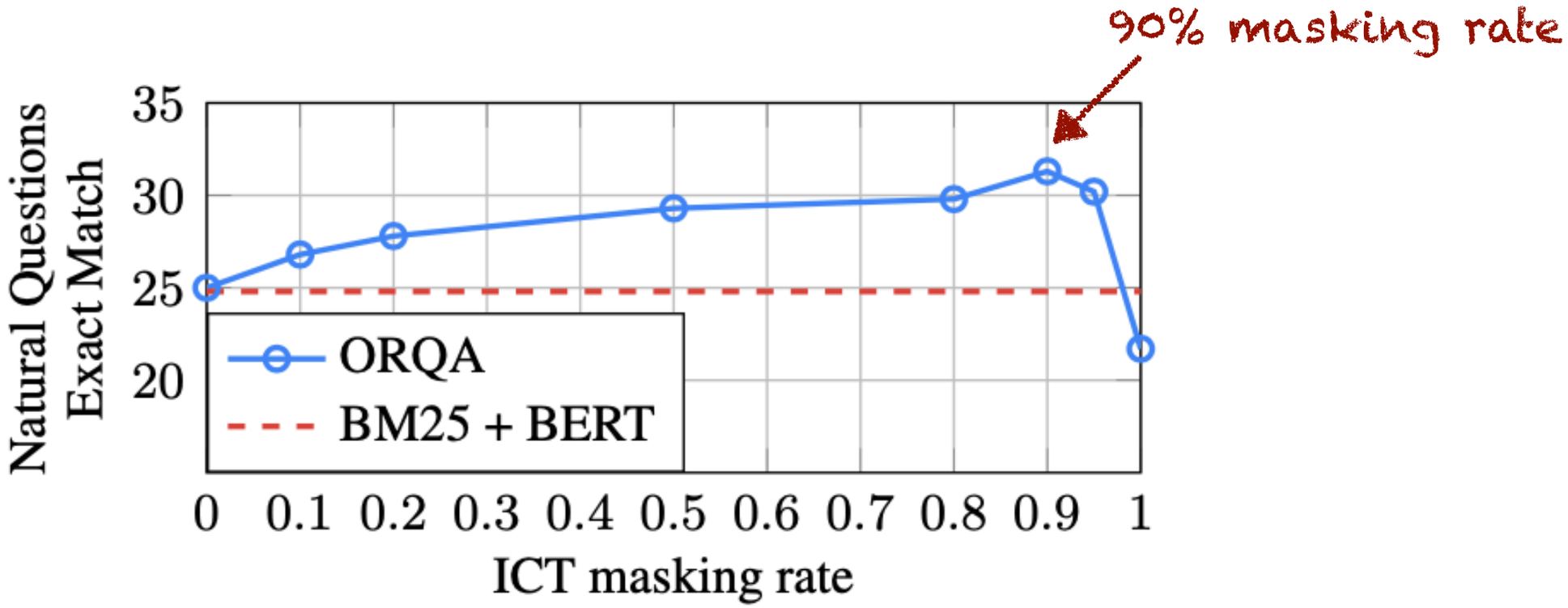
...Zebras have four gaits: walk, trot, canter and gallop. **They are generally slower than horses, but their great stamina helps them outrun predators.** When chased, a zebra will zig- zag from side to side...



Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

Pre-training: Inverse Cloze Task (ICT)

Still encourages the model to learn word matching — only remove sentences from its context in 90% of the examples!



Important - After pre-training, **BERT_B is fixed** and all the block representations can be pre-computed and search efficiently using existing tools (e.g., Locality Sensitive Hashing).

Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

ORQA: experimental results

	Model	BM25 +BERT	ORQA
Test	Natural Questions	26.5	**33.3**
	WebQuestions	17.7	**36.4**
	CuratedTrec	21.3	**30.1**
	TriviaQA	**47.1**	45.0
	SQuAD	**33.2**	20.2

ORQA wins!

BM25 + BERT wins!

TriviaQA/SQuAD — question writer knows the answer

NQ/WebQ/TREC — genuine information-seeking questions

Dataset bias? SQuAD: only 536 documents, violating IID assumption?

Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering

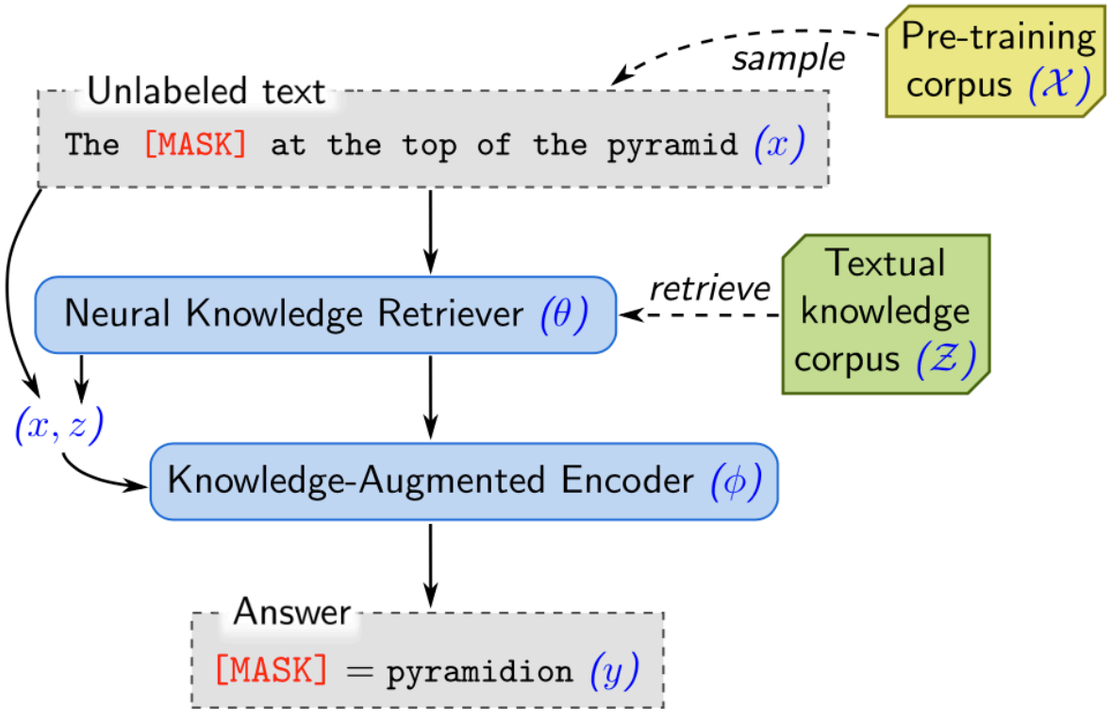
REALM: **Re**trieval-**a**ugmented **L**anguage **M**odel [Guu et al., 2020]

Pre-training on both retriever and reader!

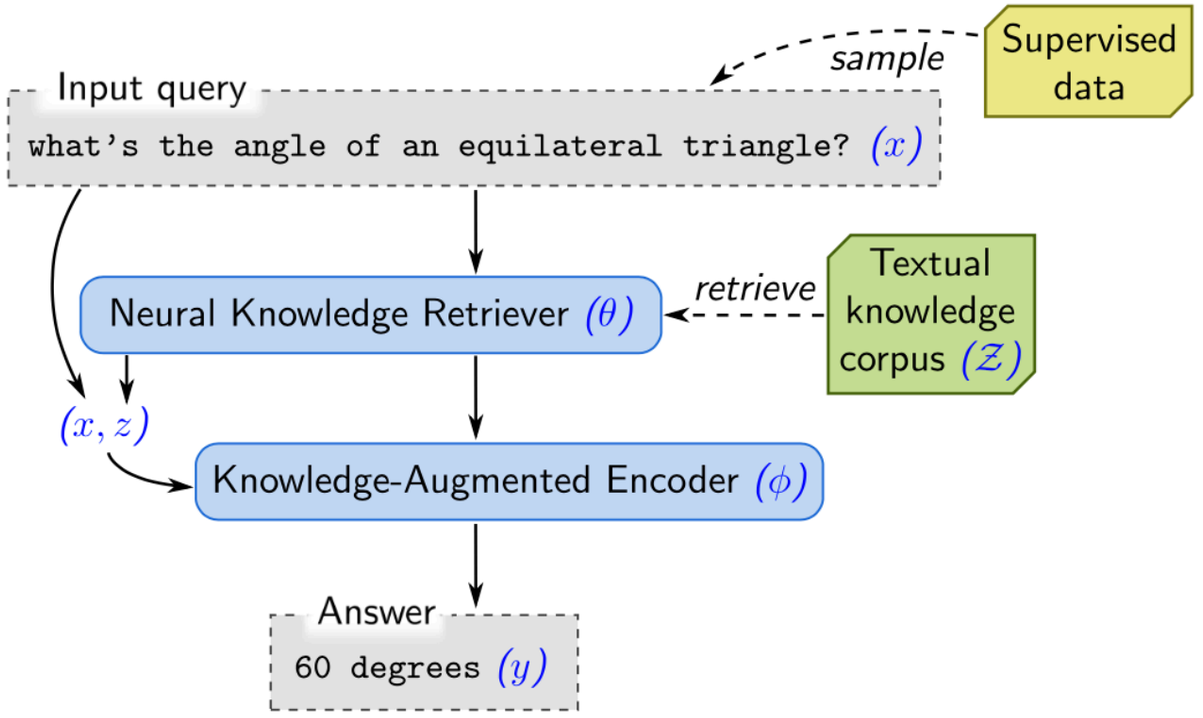
$$p(y \mid x) = \sum_{z \in \mathcal{Z}} p(y \mid z, x)\, p(z \mid x)$$

Z=Top(K) passages

Pre-training: masked language model (MLM)

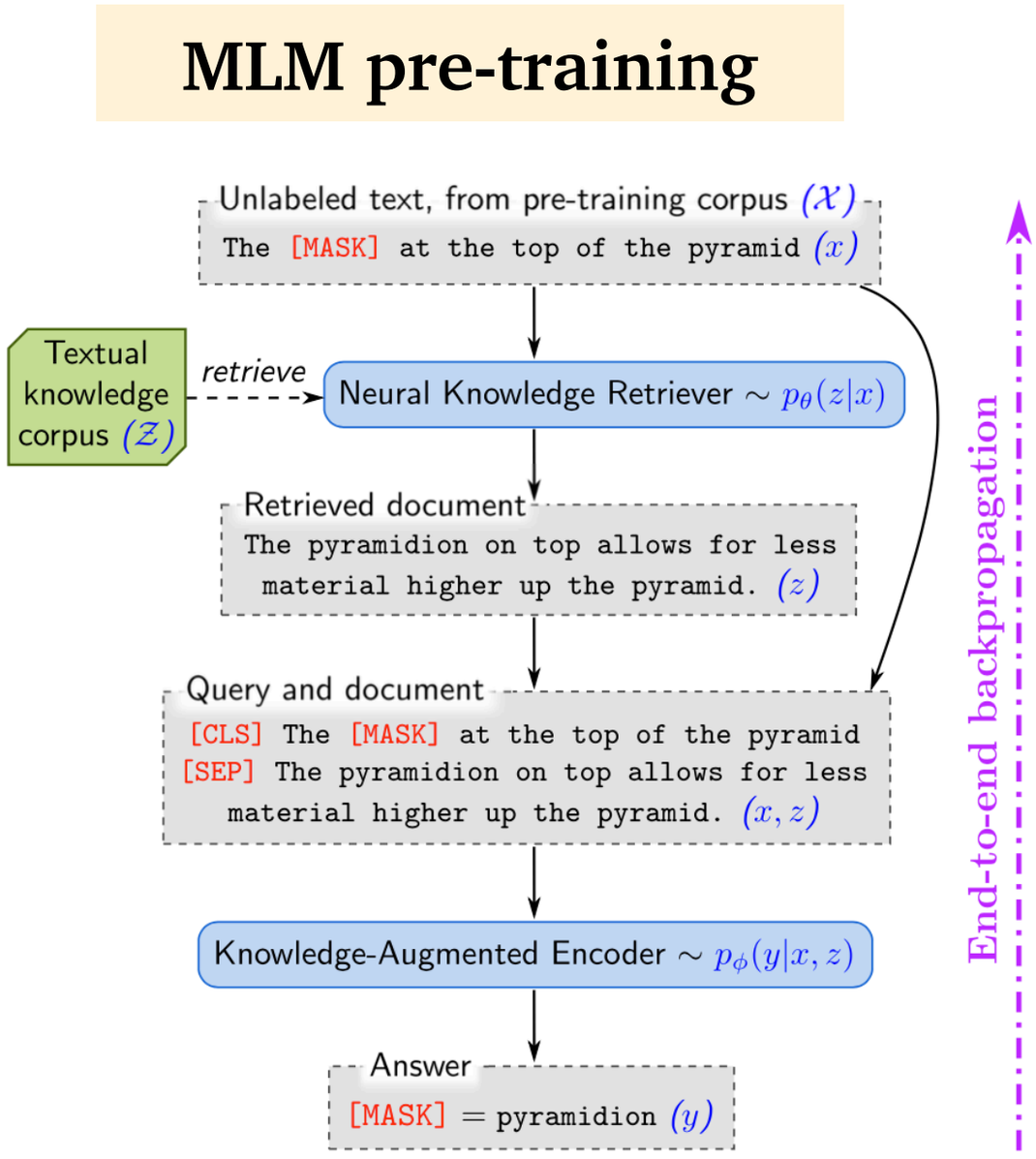


Fine-tuning: QA



Guu et al., 2020. REALM: Retrieval-Augmented Language Model Pre-Training

REALM: MLM pre-training



- Cold-start is hard => Use ICT as a first-stage pre-training

- Salient span masking: similar to span masking [Joshi et al., 2020] but only mask named entities + dates.

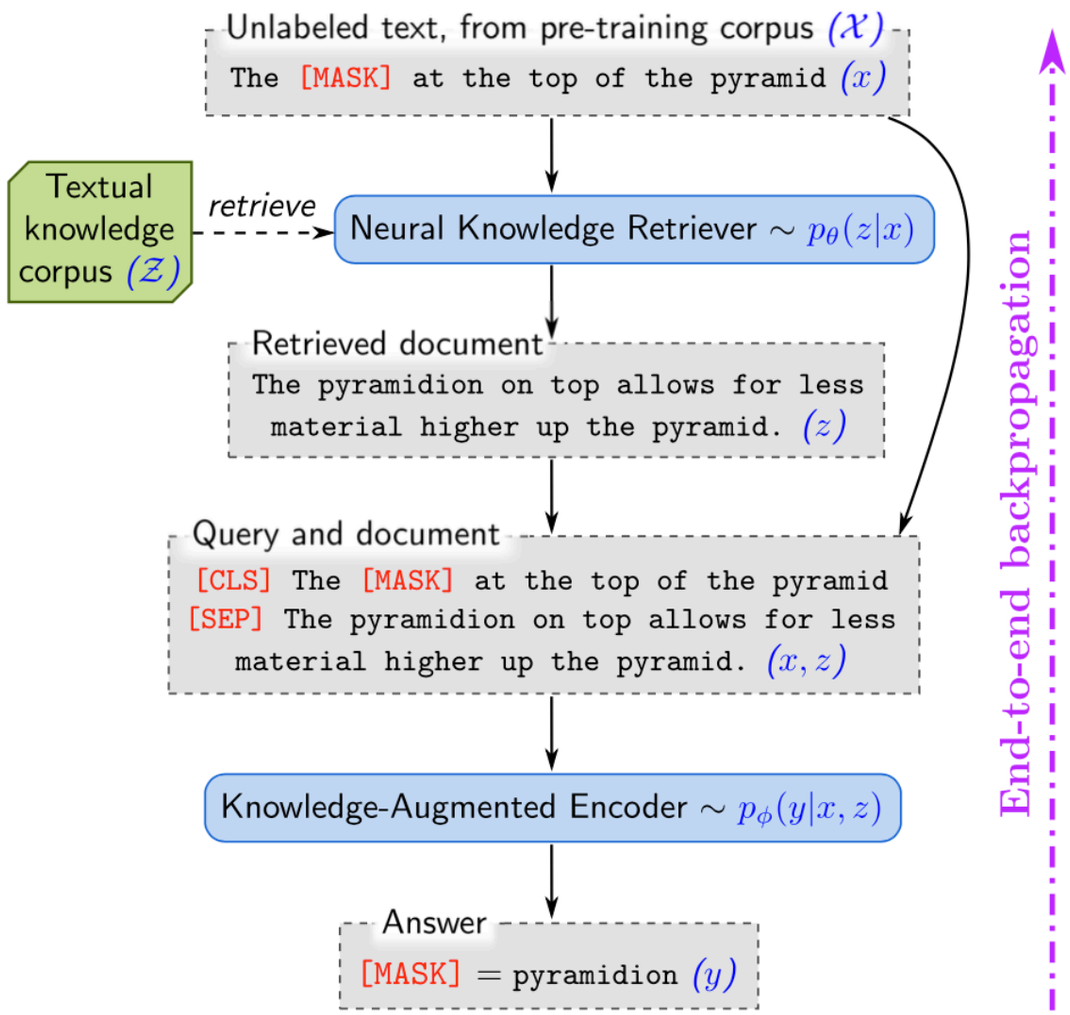
Evaluated on NQ

Random **uniform** masks: 32.3
Random **span** masks: 35.3
Salient span masks: 38.2

Guu et al., 2020. REALM: Retrieval-Augmented Language Model Pre-Training

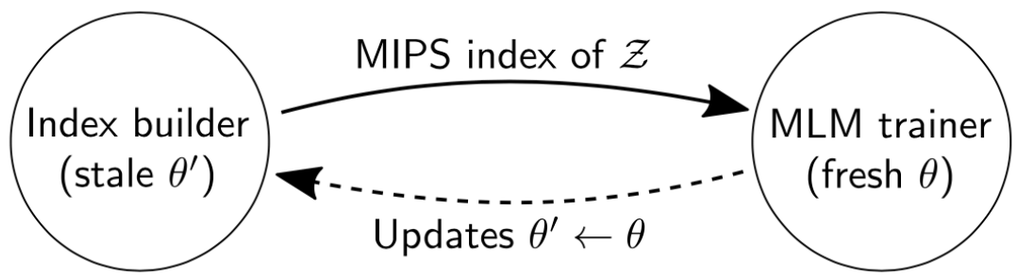
REALM: MLM pre-training

MLM pre-training



- Can use corpora even larger than Wikipedia for pre-training
  - CC-News vs Wikipedia: 40.4 vs 39.2

- Can allow updating evidence encoder (BERT_B) asynchronously



ORQA vs REALM: 33.3 vs 40.4 on NQ

Guu et al., 2020. REALM: Retrieval-Augmented Language Model Pre-Training

Take-aways from ORQA/REALM

- It is possible to jointly train the retriever and reader, without any sparse IR components.

- The pre-training makes this retrieval process feasible and pre-training strategies matter.

- However, pre-training is very expensive.

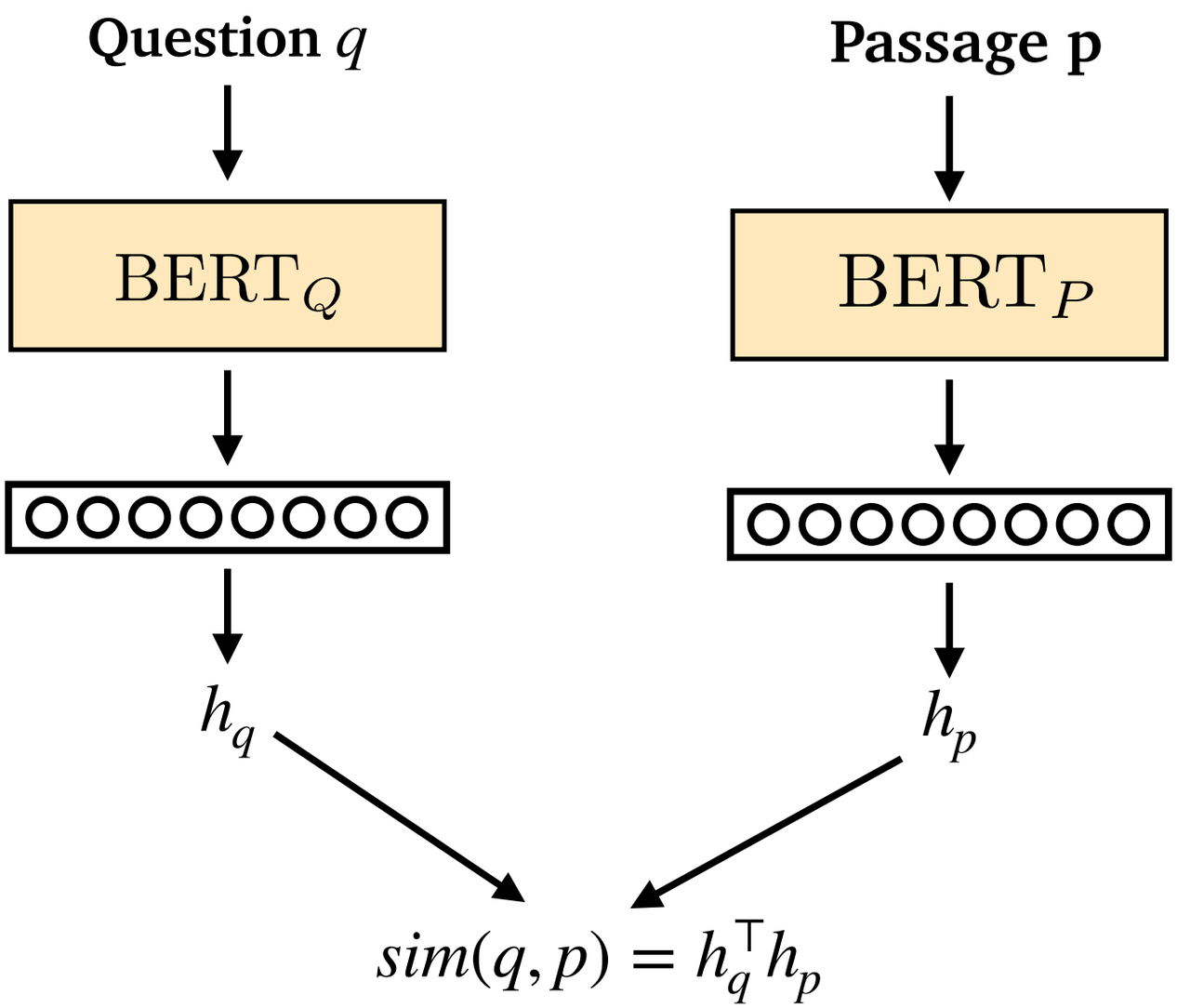
  "We pre-train for 200k steps on 64 Google Cloud TPUs, with a batch size of 512"

Next question: Is pre-training really necessary?

Dense Passage Retrieval (DPR)

[Karpukhin et al., 2020]

Key contribution: you can train a dense retrieval only from a small number of Q/A pairs, without any pre-training!



How to get positives and negatives?

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

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

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

DPR: training examples

Positives

(1) Provided in the reading comprehension datasets

(2) Passages of high BM25 scores that contain the answer string

Negatives

(1) Random passages from the corpus

(2) Passages of high BM25 scores that DO NOT contain the answer string

(3) Positive passages of OTHER questions

The best model uses (3) from the same mini-batch [in-batch negatives] and one passages from (2) [hard negatives].

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

DPR: in-batch negatives

- A small trick to effective generate more training pairs
- Suppose we have n pairs of relevant questions and passages. Let $Q_{d\times n}$ and $P_{d\times n}$ be the question and passage embeddings.
- $S = Q^\intercal P$ is a n-by-n matrix of the similarity scores. Scores of n^2 pairs of questions and passages. For each question, 1 positive passage and $n-1$ negative passages

in-batch negatives

Type	#N	IB	Top-5	Top-20
Random	7	✗	47.0	64.3
BM25	7	✗	50.0	63.3
Gold	7	✗	42.6	63.1
Gold	7	✓	51.1	69.1
Gold	31	✓	52.1	70.8
Gold	127	✓	55.8	73.0
G.+BM25$^{(1)}$	31	✓	65.0	77.3
G.+BM25$^{(2)}$	31	✓	64.5	76.4
G.+BM25$^{(1)}$	127	✓	**65.8**	**78.0**

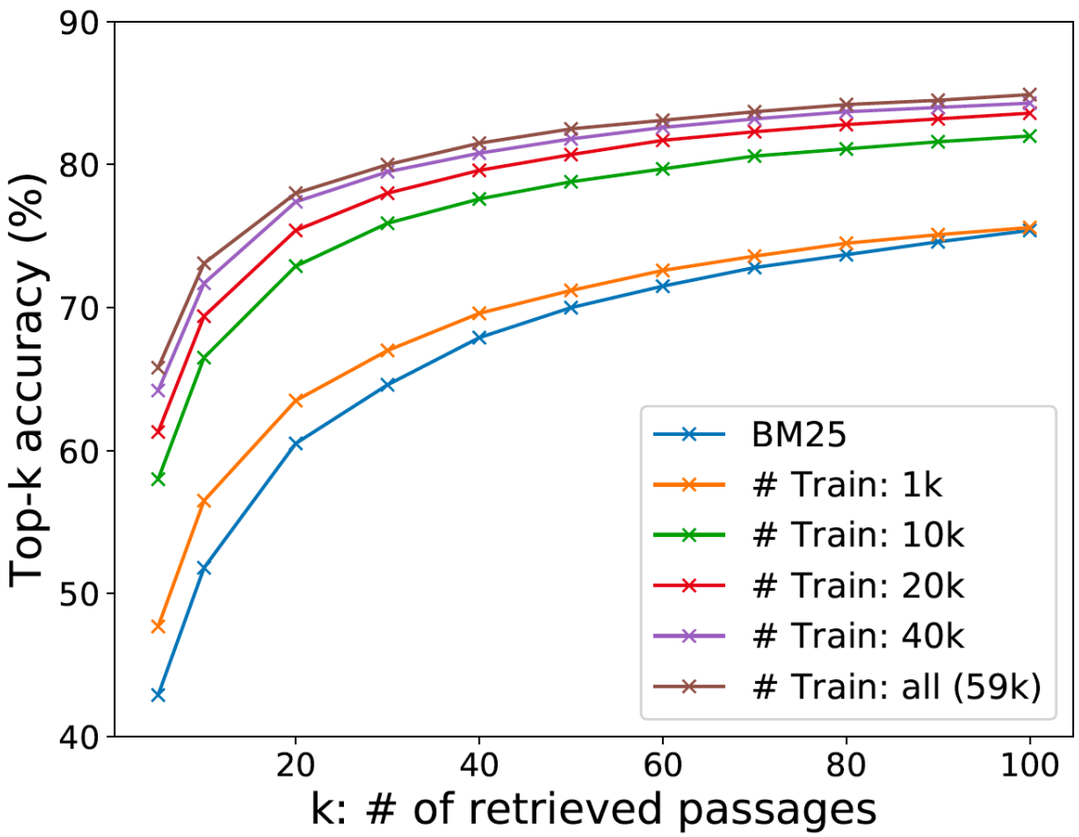
Retriever performance (Natural Questions)

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

DPR: experimental results

Retriever performance on NQ

1k Q/A pairs beat BM25!



End-to-end QA performance

With a *multi-passage BERT reader* trained on the retrieved passages from the retriever:

- DPR is better than BM25 on NaturalQuestions, WebQuestions, TREC, TriviaQA but not SQuAD.
- DPR is better than REALM on bigger datasets (41.5 vs 39.2 on NQ). For smaller datasets (WebQ, TREC), it needs a mixed training with bigger datasets to outperform REALM.

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

Take-aways from DPR

- No need of expensive pre-training?
  - At least for modest-sized QA datasets

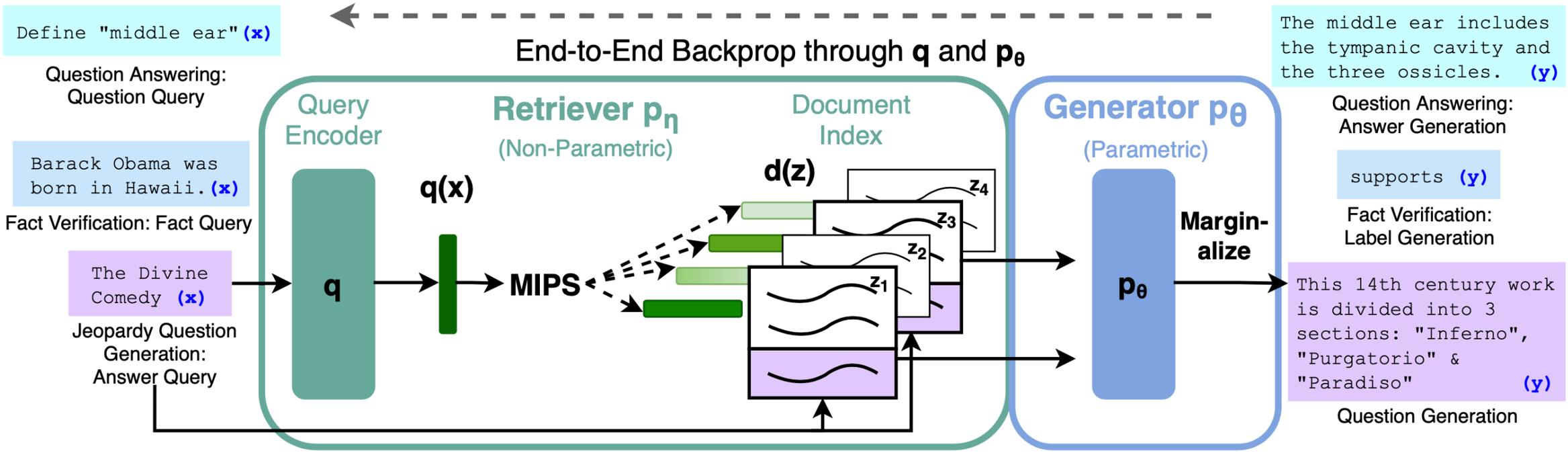
- **Joint** training vs **pipeline** training of retriever and reader?
  - Joint training didn't help DPR (41.5 → 39.8 on NQ)
  - Pipeline training is more efficient. You only build the index once!

- **Reading comprehension** vs **QA** datasets
  - Doesn't make a much difference (41.5 vs 41.0 on NQ).
  - You can train the system using only Q/A pairs!

- BM25 + DPR is slightly better than DPR but the difference is small.

Retrieval-Augmented Generation (RAG)

[Lewis et al., 2020]



$$p_{\text{RAG-Sequence}}(y|x) = \sum_{z \in \text{top-}k(p(\cdot|x))} p_\eta(z|x) \prod_i^N p_\theta(y_i|x, z, y_{1:i-1})$$

retrieval model

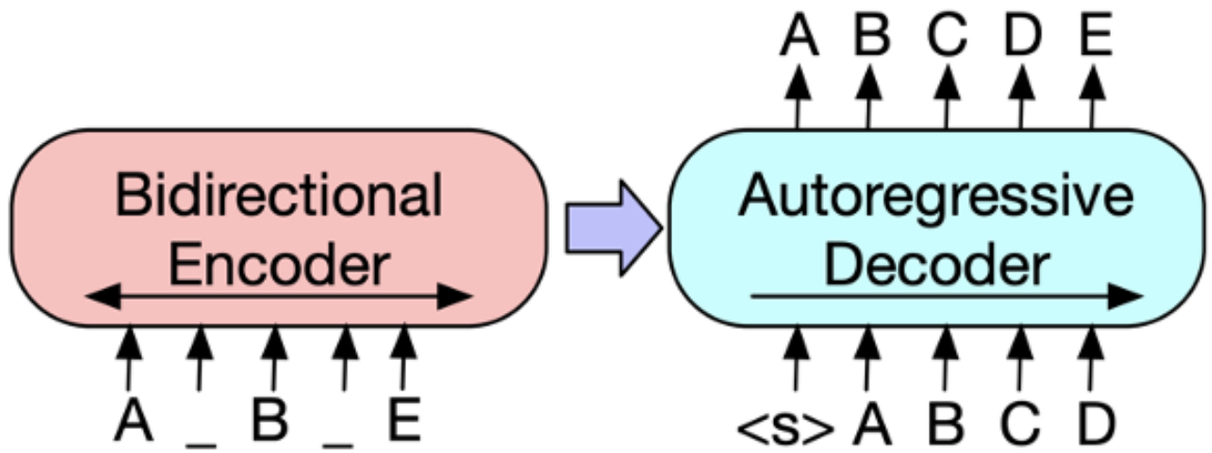
seq2seq model

Lewis et al., 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

Retrieval-Augmented Generation (RAG)

- Retrieval model: $p(z \mid x)$ = Dense passage retrieval (DPR)
- seq2seq model: $p(y \mid x, z)$ = BART [Lewis et al., 2020]

  Pre-trained on large text corpora, simply concatenates z to question x



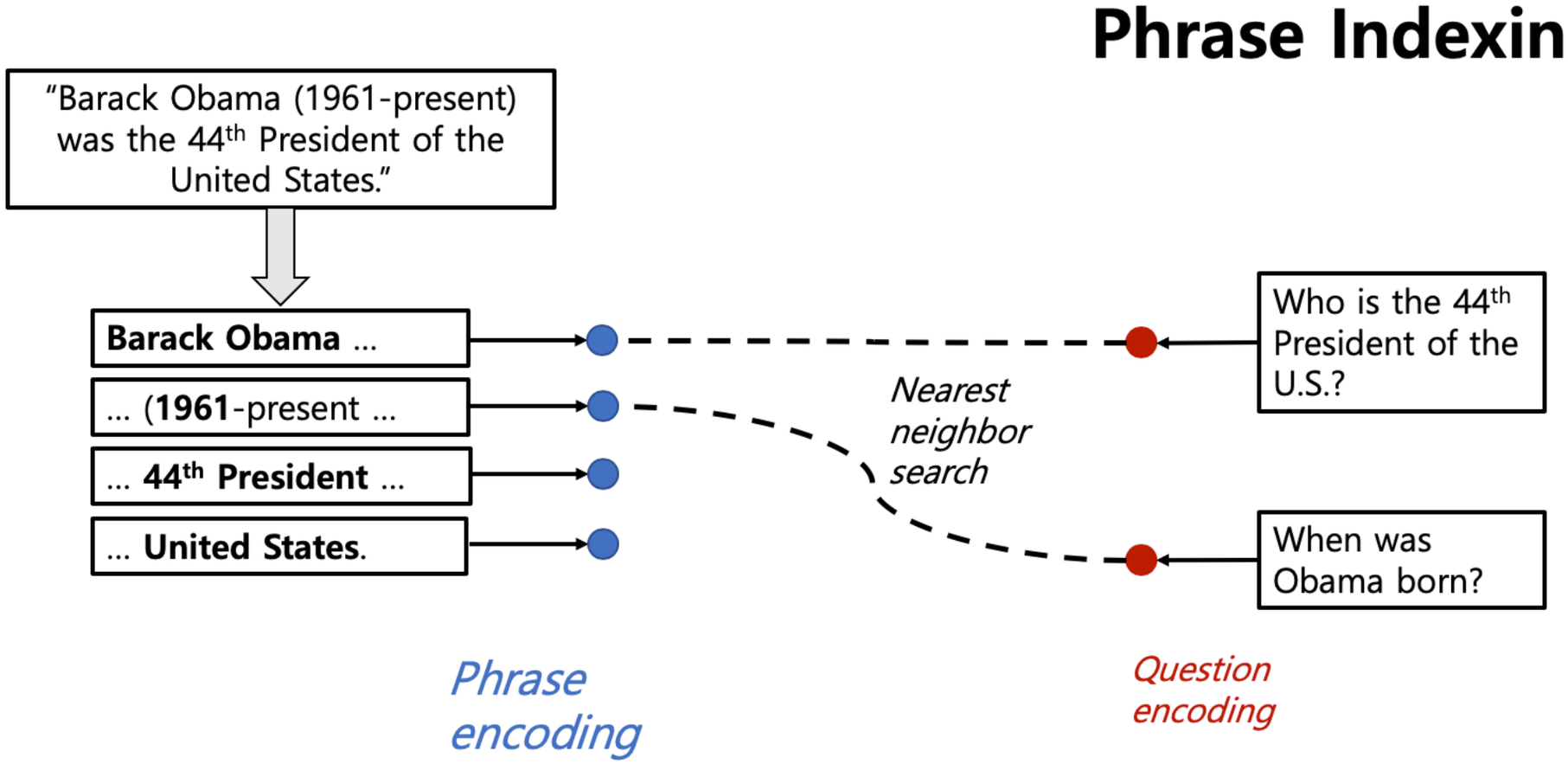
- Joint training of the two components
- Improved performance on open-domain QA evaluation and easy to extend to other generation tasks

Lewis et al., 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

DenSPI: Dense-Sparse Phrase Index

[Seo et al., 2019]

Key question: Is it possible to encode/index **at phrase level** instead of paragraphs or documents so retriever + reader will be reduced to a harder "retriever" problem?



Seo et al., 2019. Real-Time Open-Domain Question Answering with Dense-Sparse Phrase Index

DenSPI: Challenges

Decomposability gap: question and passage/answer have to be encoded independently

Model phrase question document

$$\hat{a} = \text{argmax}_a F_\theta(a, q, d)$$

We know modeling "attention" between question and passage is important..

Query-Agnostic Decomposition

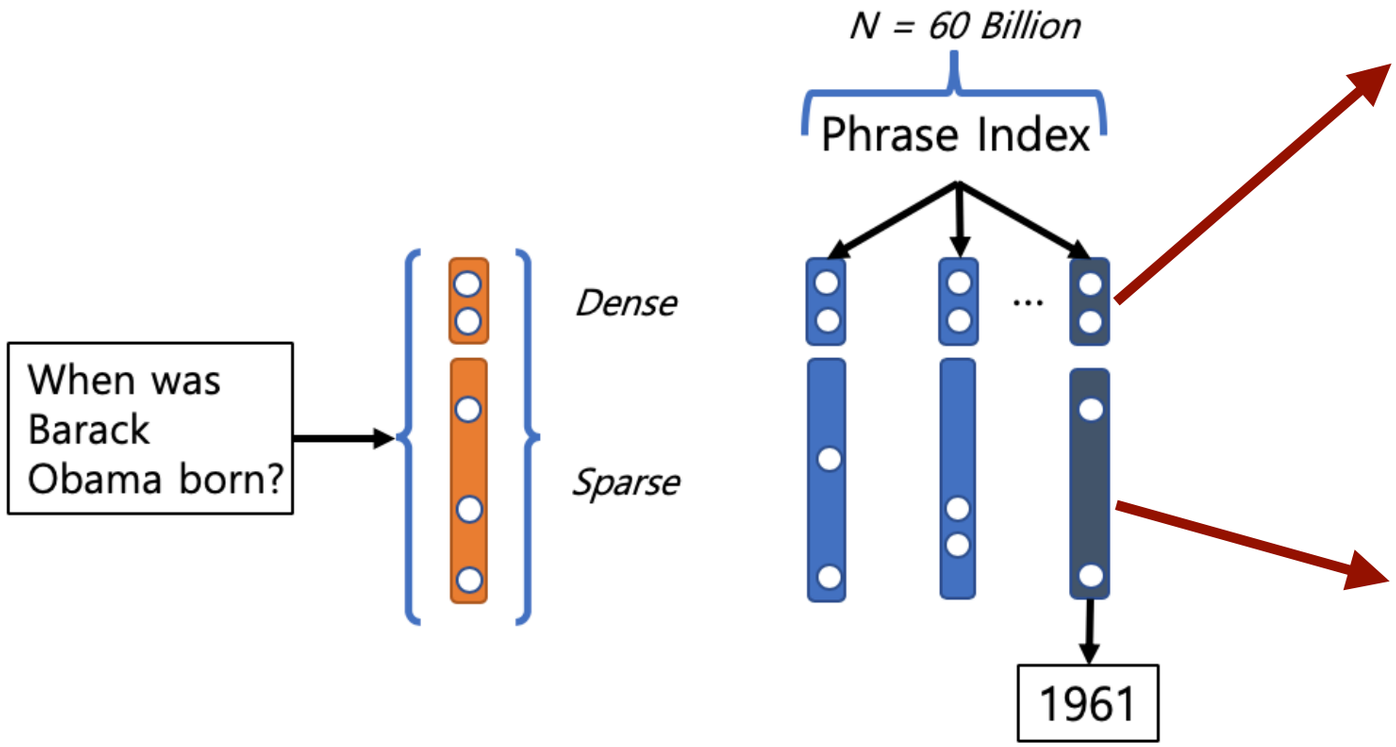
$$\hat{a} = \text{argmax}_a G_\theta(q) \cdot H_\theta(a, d)$$

Question encoder Phrase encoder

Scalability: *60 billion phrases* in Wikipedia! Storage, indexing and search all remain challenging.

Seo et al., 2019. Real-Time Open-Domain Question Answering with Dense-Sparse Phrase Index

DenSPI: Dense & sparse representations



Dense representation

Start and end vectors from BERT + coherency scalar

Sparse representation

TF-IDF vector for the corresponding document & paragraph vector

Seo et al., 2019. Real-Time Open-Domain Question Answering with Dense-Sparse Phrase Index

DenSPI: Results

Storage: pointer, filter, scalar quantization: 240T → 1.5T

Search: Dense-first search (DFS), sparse-first search (SFS), Hybrid

Performance on SQuAD

	F1	EM	s/Q
DrQA	-	29.8	35
R^3	37.5	-	-
Paragraph ranker	-	30.2	-
Multi-step reasoner	39.2	31.9	-
MINIMAL	42.5	34.7	-
BERTserini	46.1	38.6	115
Weaver	-	42.3	-
DENSPI-SFS	42.5	33.3	0.60
DENSPI-DFS	35.9	28.5	0.51
–sparse scale=0	16.3	11.2	0.40
DENSPI-Hybrid	44.4	36.2	0.81

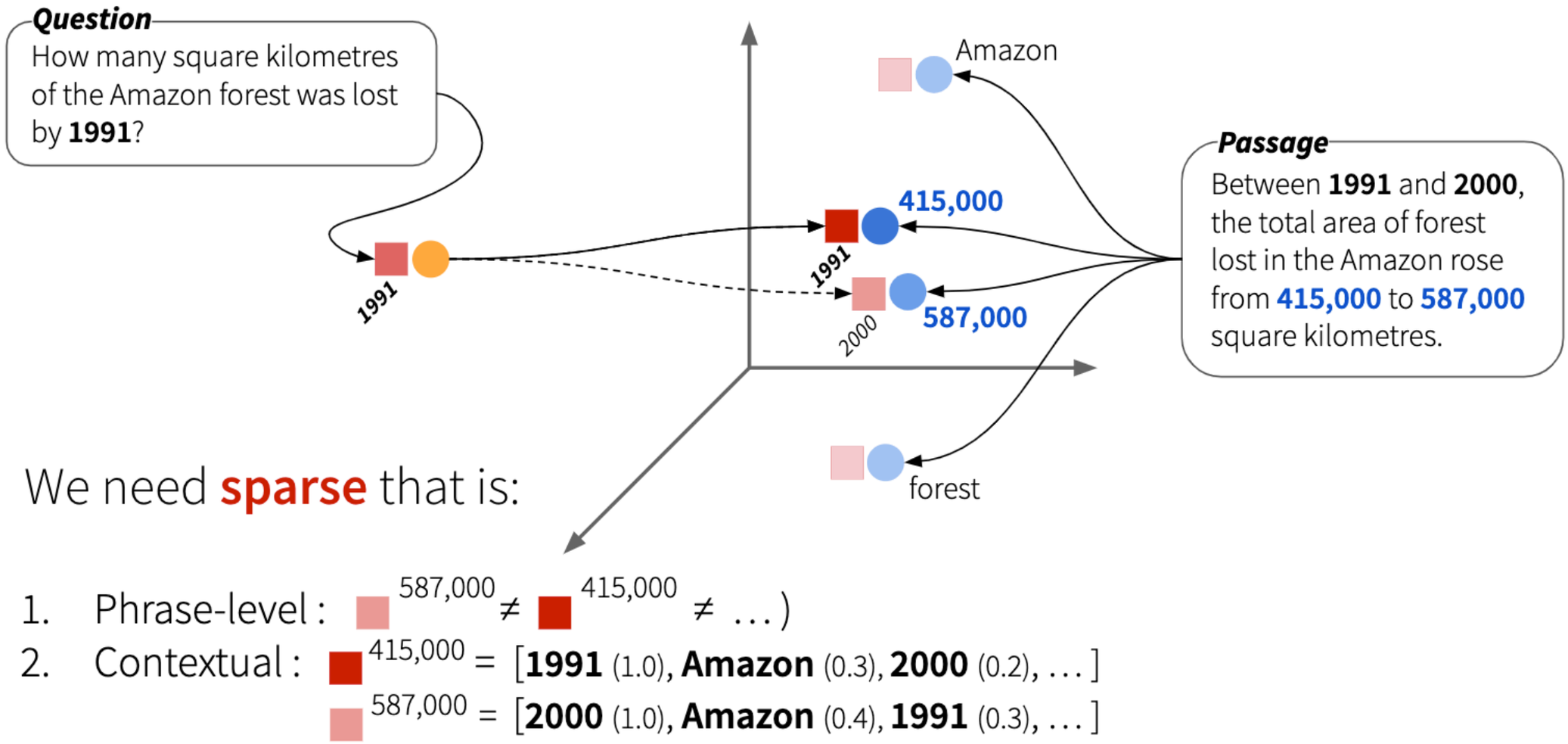
Removing sparse vector performs much worse

Very fast inference time

Seo et al., 2019. Real-Time Open-Domain Question Answering with Dense-Sparse Phrase Index

DenSPI: Contextualized sparse representations

[Lee et al., 2020]



- **Sparc:** use different weights for different sparse terms
- Improved DenSPI by +4.1% on TREC and +4.5% on SQuAD

Lee et al., 2020. Contextualized Sparse Representations for Real-Time Open-Domain Question Answering

Comparison of all models

		SQuAD	TriviaQA	NaturalQ	TREC	WebQ
2017/04	DrQA	28.4	-	-	25.7	19.5
2017/08	R^3	29.1	-	-	28.4	17.1
2018/04	DrQA*	40.4	-	-	28.8	24.3
2019/02	BERTserini	38.6	-	-	-	-
2019/08	Multi-passage BERT@	**53.0**	-	-	-	-
2019/09	Hard-EM	-	50.9	28.1	-	-
2019/06	ORQA	20.2	45.0	33.3	30.1	36.4
2020/02	REALM (Wiki)	-	-	39.2	46.8	40.2
2020/02	REALM (CC-News)	-	-	40.4	42.9	40.7
2020/04	DPR	29.8	56.8	41.5	49.4*	42.4*
2020/05	RAG@	-	56.1	**44.5**	**52.2***	**45.2***
2019/06	DenSPI	36.2	-	-	31.6#	-
2019/11	DenSPI + Sparc	40.7	-	-	35.7#	-
2020/04	BM25 + DPR	36.7	**57.0**	39.0	50.6*	41.1*

#: no supervision using target training data / *: jointly trained with bigger datasets (e.g., NQ) / @: BERT-large models

Summary

- ORQA/REALM: First demonstrate that it is possible jointly train the retriever and reader without any sparse IR components; requires novel ways of pre-training

- DPR/RAG: It is possible to learn the dense retrieval only from Q/A pairs without pre-training!

- DenSPI /Sparc: It is possible to index and retrieve at phrase level without requiring an explicit “reader”. Very fast inference time but slightly worse performance. Sparse features still matter.

Questions?