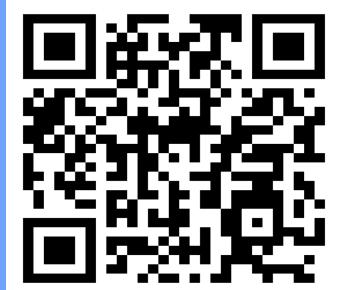


Reproducible and Efficient Multi-modal Open Retrieval Question Answering

powered by **PrimeQA**



Aug 20
IJCAI 2023

About the Presenters



Bhavani Iyer,
Research Engineer, IBM Research



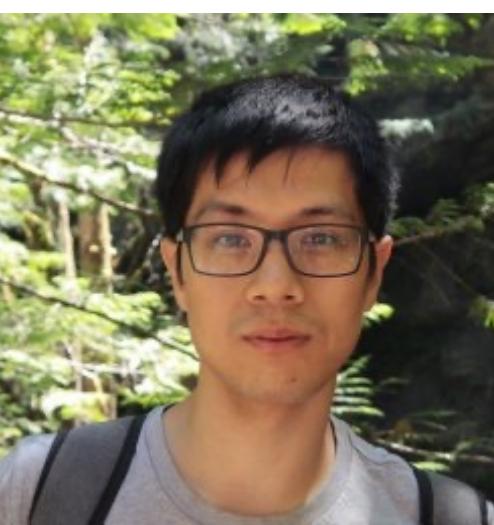
Revanth Gangi Reddy,
PhD student University of Illinois, Urbana Champaign



Jaydeep Sen
Senior Research Scientist, IBM Research



Avi Sil,
Principal Research Scientist, IBM Research



Wenhua Chen
Assistant Professor at Computer Science at
University of Waterloo, Research Scientist at
Google Deepmind



Christopher Potts,
Professor and Chair
Department of Linguistics, Stanford

Participation and Q&A

QR for live Q&A



Online Q&A



Coffee Break....

1st Half

Q&A (15 mins)

1 Hr 30 min

Interactive Q&A

2nd Half

Q&A (15 mins)

1 Hr 30 min

Question Answering

I have a question



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

Here is the document



Warsaw

From Wikipedia, the free encyclopedia

Coordinates: 52°14'N 21°1'E

"Warszawa", "Warschau", and "City of Warsaw" redirect here. For other uses, see [Warsaw \(disambiguation\)](#), [Warszawa \(disambiguation\)](#), [Warschau \(disambiguation\)](#), and [City of Warsaw \(disambiguation\)](#).

Warsaw,^[a] officially the **Capital City of Warsaw**,^{[4][b]} is the capital and largest city of Poland. The metropolis stands on the River Vistula in east-central Poland and its population is officially estimated at 1.8 million residents within a greater metropolitan area of 3.1 million residents,^[9] which makes Warsaw the 7th most populous capital city in the European Union. The city area measures 517 km² (200 sq mi) and comprises 18 boroughs, while the metropolitan area covers 6,100 km² (2,355 sq mi).^[6] Warsaw is an alpha-global city^[7] a major cultural, political and economic hub, and the country's seat of government. Its historical Old Town was designated a UNESCO World Heritage Site.

Warsaw traces its origins to a small fishing town in Masovia. The city rose to prominence in the late 16th century, when Sigismund III decided to move the Polish capital and his royal court from Kraków. Warsaw served as the de facto capital of the Polish–Lithuanian Commonwealth until 1795, and subsequently as the seat of Napoleon's Duchy of Warsaw. The 19th century and its Industrial Revolution brought a demographic boom which made it one of the largest and most densely-populated cities in Europe. Known then for its elegant architecture and boulevards, Warsaw was bombed and besieged at the start of World War II in 1939.^{[9][10]} Much of the historic city was destroyed and its diverse population decimated by the Ghetto Uprising in 1943, the general Warsaw Uprising in 1944 and systematic razing.

Warsaw is served by two international airports, the busiest being Warsaw Chopin and the smaller Warsaw Modlin intended for low-cost carriers. Major public transport services operating in the city include the Warsaw Metro, buses, urban-light railway and an extensive tram network. In 2012, the Economist Intelligence Unit ranked Warsaw as the 32nd most liveable city in the world.^[11] In 2017, the city came 4th in the "Business-friendly", 8th in "Human capital and life style" and topped the quality of life rankings in the region.^[12] The city is a significant centre of research and development, business process outsourcing, and information technology outsourcing. The Warsaw Stock Exchange is the largest and most important in Central and Eastern Europe.^{[13][14]} Frontex, the European Union agency for external border security as well as ODIHR, one of the principal institutions of the Organization for Security and Cooperation in Europe have their headquarters in Warsaw. Jointly with Frankfurt and Paris, Warsaw features one of the highest numbers of skyscrapers in the European Union.^[15]

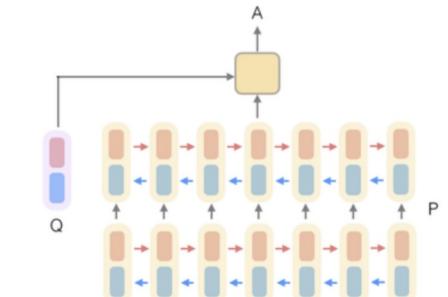
The city hosts the Polish Academy of Sciences, National Philharmonic Orchestra, University of Warsaw, the Warsaw University of Technology, the National Museum, Zachęta Art Gallery and the Warsaw Grand Theatre, the largest of its kind in the world.^[16] The reconstructed Old Town, which represents examples of nearly every European architectural style and historical period,^[17] was listed as a World Heritage Site by UNESCO in 1980. Other main architectural attractions include the Royal Castle and the iconic King Sigismund's Column, the Wilanów Palace, the Palace on the Isle, St. John's Cathedral, Main Market Square, as well as numerous churches and mansions along the Royal Route. Warsaw possesses thriving arts and club scenes, gourmet restaurants and large urban green spaces, with around a quarter of the city's area occupied by parks.^{[18][19]}



Get the answer



833,500



Question Answering → Open Domain

I have a question



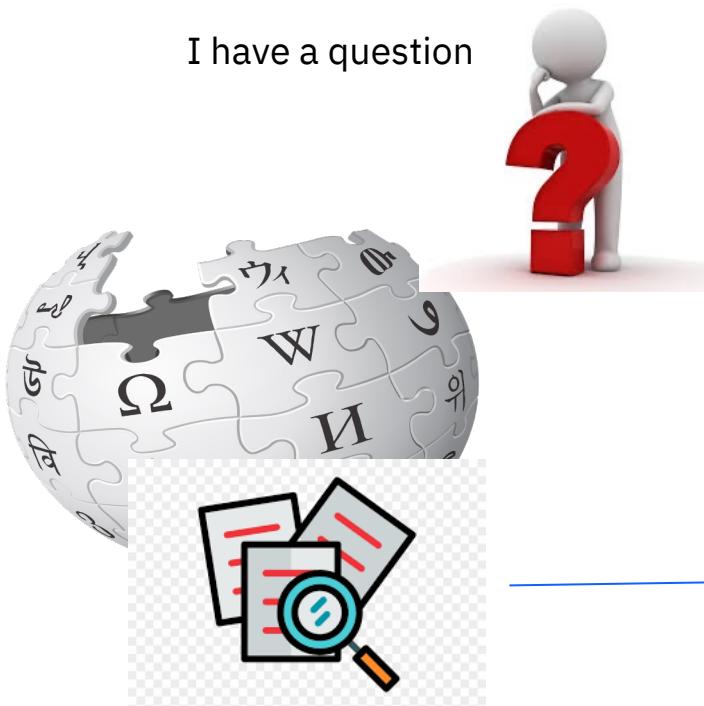
Q: How many of Warsaw's inhabitants spoke Polish in 1933?



Document is **NOT** given

Open Retrieval QA

I have a question



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

Warsaw

From Wikipedia, the free encyclopedia

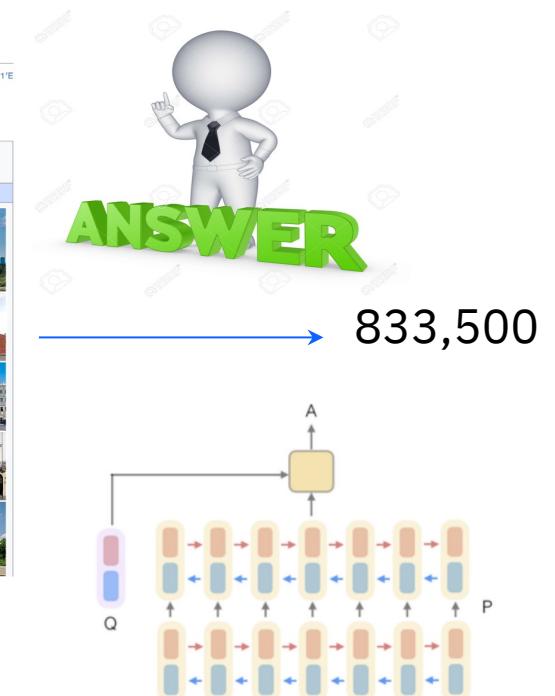
"Warszawa", "Warschau", and "City of Warsaw" redirect here. For other uses, see [Warsaw \(disambiguation\)](#), [Warszawa \(disambiguation\)](#), [Warschau \(disambiguation\)](#), and [City of Warsaw \(disambiguation\)](#).

Warsaw,^[4] officially the **Capital City of Warsaw**,^{[4][5]} is the capital and largest city of Poland. The metropolis stands on the River Vistula in east-central Poland and its population is officially estimated at 1.8 million residents within a greater metropolitan area of 3.1 million residents,^[5] which makes Warsaw the 7th most populous capital city in the European Union. The city area measures 517 km² (200 sq mi) and comprises 18 boroughs, while the metropolitan area covers 6,100 km² (2,355 sq mi).^[6] Warsaw is an alpha-global city,^[7] a major cultural, political and economic hub, and the country's seat of government. Its historical Old Town was designated a UNESCO World Heritage Site.

Warsaw traces its origins to a small fishing town in Masovia. The city rose to prominence in the late 16th century, when Sigismund III decided to move the Polish capital and his royal court from Kraków. Warsaw served as the de facto capital of the Polish–Lithuanian Commonwealth until 1795, and subsequently as the seat of Napoleon's Duchy of Warsaw. The 19th century and its Industrial Revolution brought a demographic boom which made it one of the largest and most densely populated cities in Europe. Known then for its elegant architecture and boulevards, Warsaw was bombed and besieged at the start of World War II in 1939.^{[8][9][10]} Much of the historic city was destroyed and its diverse population decimated by the Ghetto Uprising in 1943, the general Warsaw Uprising in 1944 and systematic razings.

Warsaw is served by two international airports, the busiest being Warsaw Chopin and the smaller Warsaw Modlin intended for low-cost carriers. Major public transport services operating in the city include the Warsaw Metro, buses, urban-light railway and an extensive tram network. In 2012, the Economist Intelligence Unit ranked Warsaw as the 32nd most liveable city in the world.^[11] In 2017, the city came 4th in the "Business-friendly", 8th in "Human capital and life style" and topped the quality of life rankings in the region.^[12] The city is a significant centre of research and development, business process outsourcing, and information technology outsourcing. The Warsaw Stock Exchange is the largest and most important in Central and Eastern Europe.^{[13][14]} Frontex, the European Union agency for external border security as well as ODIHR, one of the principal institutions of the Organization for Security and Cooperation in Europe have their headquarters in Warsaw. Jointly with Frankfurt and Paris, Warsaw features one of the highest number of skyscrapers in the European Union.^[15]

The city hosts the Polish Academy of Sciences, National Philharmonic Orchestra, University of Warsaw, the Warsaw University of Technology, the National Museum, Zachęta Art Gallery and the Warsaw Grand Theatre, the largest of its kind in the world.^[16] The reconstructed Old Town, which represents examples of nearly every European architectural style and historical period,^[17] was listed as a World Heritage Site by UNESCO in 1980. Other main architectural attractions include the Royal Castle and the iconic King Sigismund's Column, the Wilanów Palace, the Palace on the Isle, St. John's Cathedral, Main Market Square, as well as numerous churches and mansions along the Royal Route. Warsaw possesses thriving arts and club scenes, gourmet restaurants and large urban green spaces, with around a quarter of the city's area occupied by parks.^{[18][19]}



Open-Retrieval QA (ORQA)

Note: ORQA is aka Open Domain QA [Lee et al., 2019] and/or End-2-end QA [Reddy et al., 2021].

Open-Retrieval QA: Components

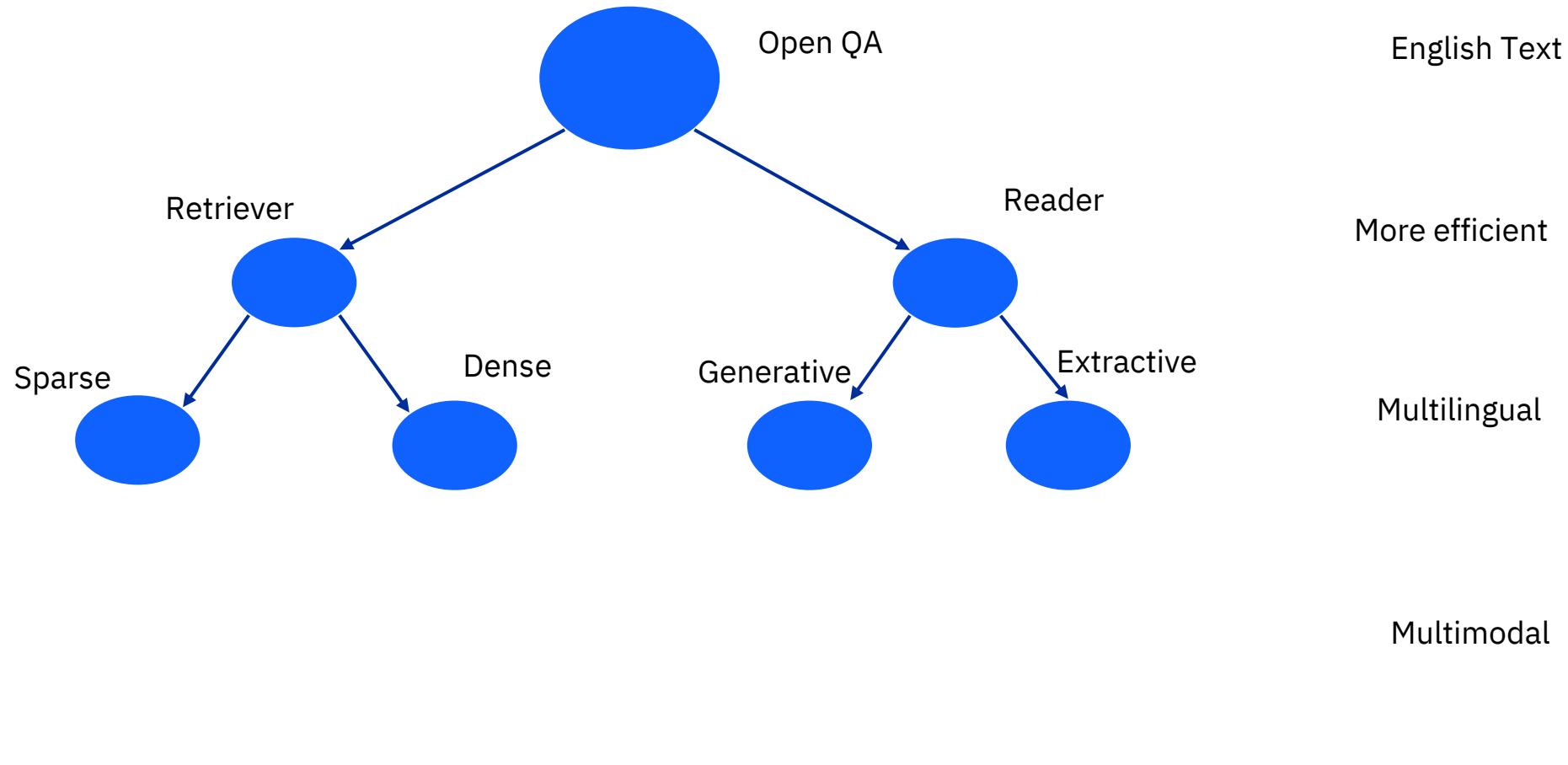


**Open-Retrieval QA
(ORQA)**

Note: ORQA is aka Open Domain QA [Lee et al., 2019] and/or End-2-end QA [Reddy et al., 2021].

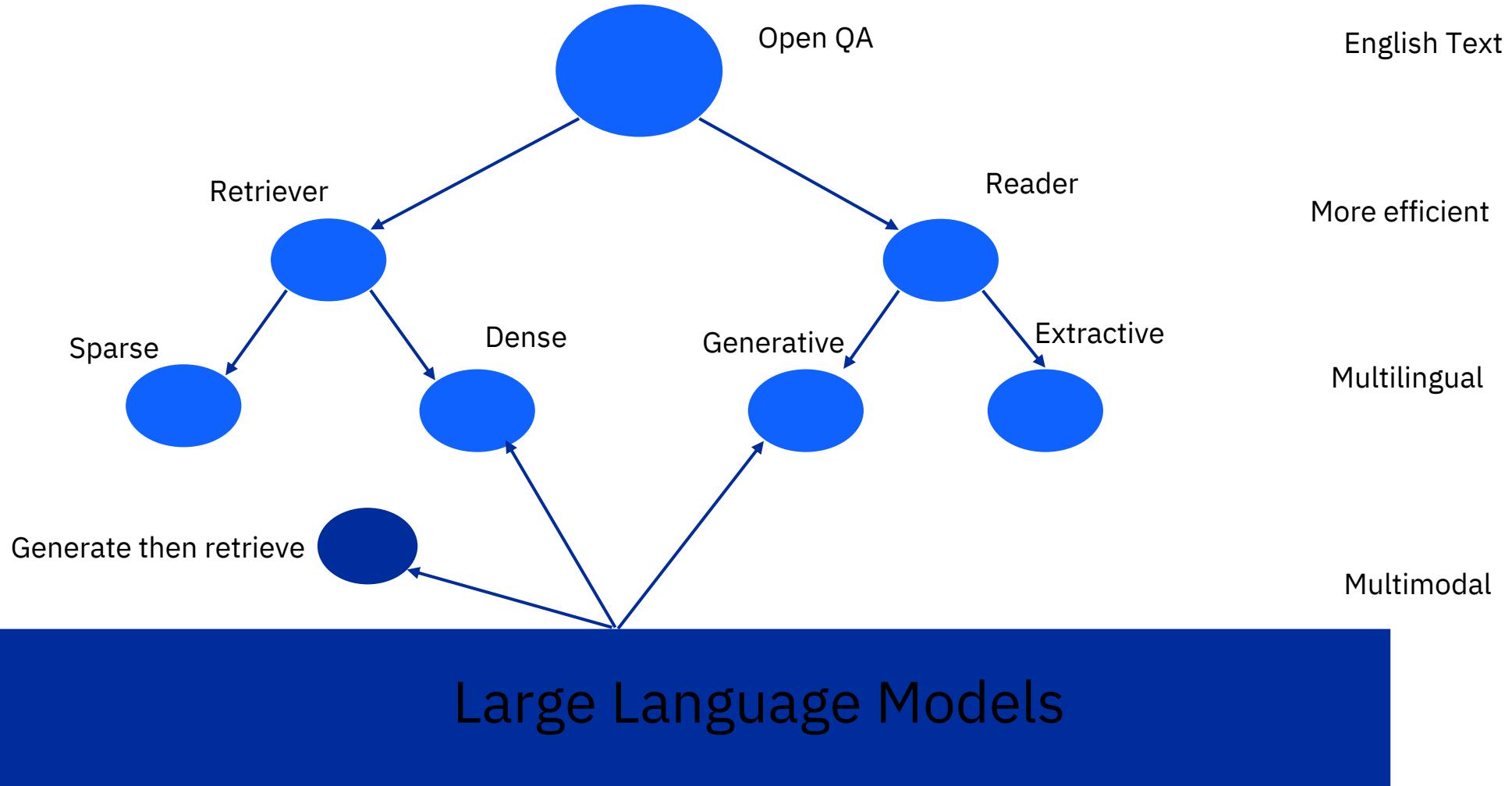
Why this Tutorial ?

- Research has progressed by leaps and bounds for both Retrievers and Readers

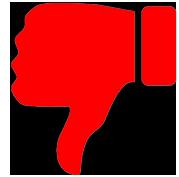


Why this Tutorial ?

Research has progressed by leaps and bounds for both Retrievers and Readers



Why this Tutorial ?



Reproducibility is still an issue



Reusability, customization : often a steep curve



This is important:
**“Towards Reproducible Machine Learning Research
in Natural Language Processing”, tutorial at ACL 22**



Outline

-  Motivation of OpenQA
-  Retrievers: Sparse and Dense
-  Recent Advances for OpenQA Efficient Neural Retrievers
-  Recent Advances in Efficient Multilingual Retrieval
-  Reproducibility in OpenQA: Hands-On Guide I
-  Q&A: [15 min]

1st Half



Coffee Break.....

-  Multilingual Readers
-  Multi-modal Readers: Text, Table, Visual QA
-  Large Language Models as Retrievers/Readers
-  Reproducibility in OpenQA: Hands-on Guide II
-  Pipelines, Service and Deployment
-  Q&A: [15 min]

2nd Half

Sparse IR

Outline

- Traditional Sparse Retrieval
 - BM25 Scoring Algorithm
 - Limitations
- Learned Sparse Representation
 - SPLADE and SPLADEV2
- BEIR Benchmark

A Traditional Retriever

A TF-IDF [Robertson 2004] weighted term vector model over unigrams/ bi-grams

BM25: Smoothed IDF weighted scoring function

IDF more importance to less frequent words in the collection

Flattens out higher frequencies
k controls the term frequency saturation

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

Penalizes long documents
b controls the document length penalty

Limitations

Sparse IR algorithms are fast and interpretable

However, they

1. Can NOT answer questions when there's little or no **lexical overlap**

“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 **LOTR** trilogy by Peter Jackson...”*

not trainable

2. Can NOT retrieve **cross-lingual** passages without translation (needs special models)

Q (in Ja): “ロン・ポールの学部時代の専攻は？”

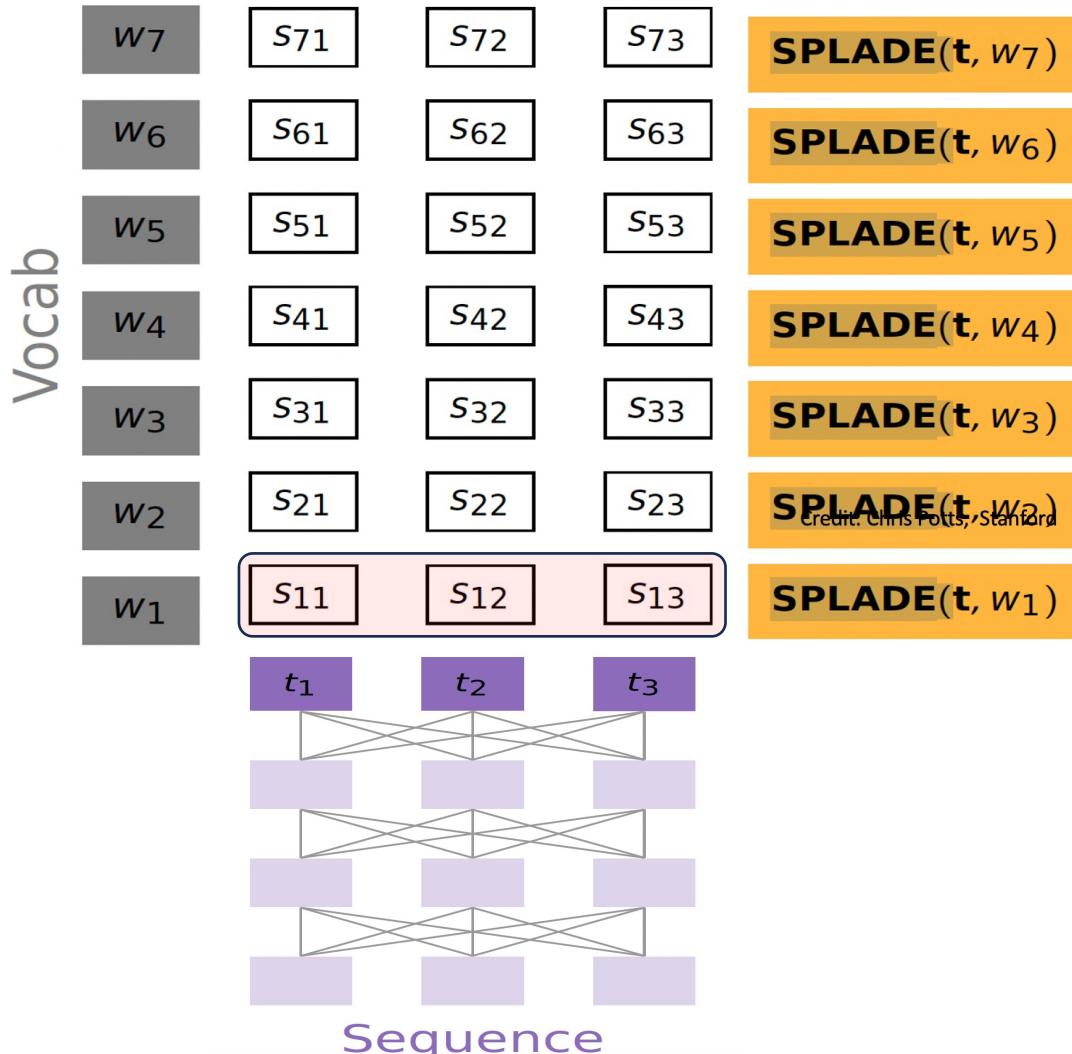
Ron Paul (en.wikipedia)

Paul went to Gettysburg College, where he was a member of the Lambda Chi Alpha fraternity. He graduated with a B.S. degree in **Biology** in 1957.

生物学 (Biology)

SPLADE

Learn (BERT based) sparse representations for first-stage retrieval



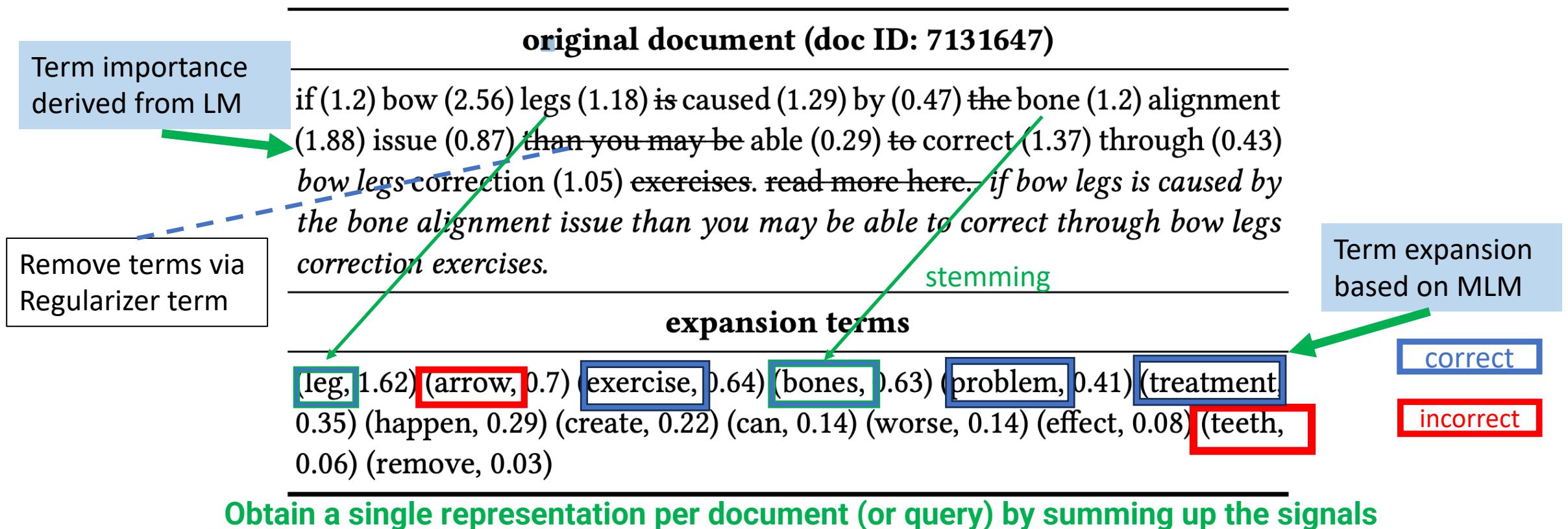
1. $s_{ij} = \text{transform}(\text{Enc}(t)_{N,i})^\top \text{Emb}(w_j) + b_j$
where
 $\text{transform}(x) = \text{LayerNorm}(\text{GeLU}(xW + b))$
and $\text{Emb}(w)$ is the embedding for w .
2. $\text{SPLADE}(t, w_j) = \sum_i^M \log(1 + \text{ReLU}(s_{ij}))$
3. $\text{Sim}_{\text{SPLADE}}(q, \text{doc}) = \text{SPLADE}(q)^\top \text{SPLADE}(\text{doc})$
4. Loss: Usual negative log-likelihood plus a regularization term that leads to sparse, balanced scores.

Slide Credit: Chris Potts, Stanford

SPLADE: An Example

Predict importance of each term in the vocabulary space

Model Exact Match

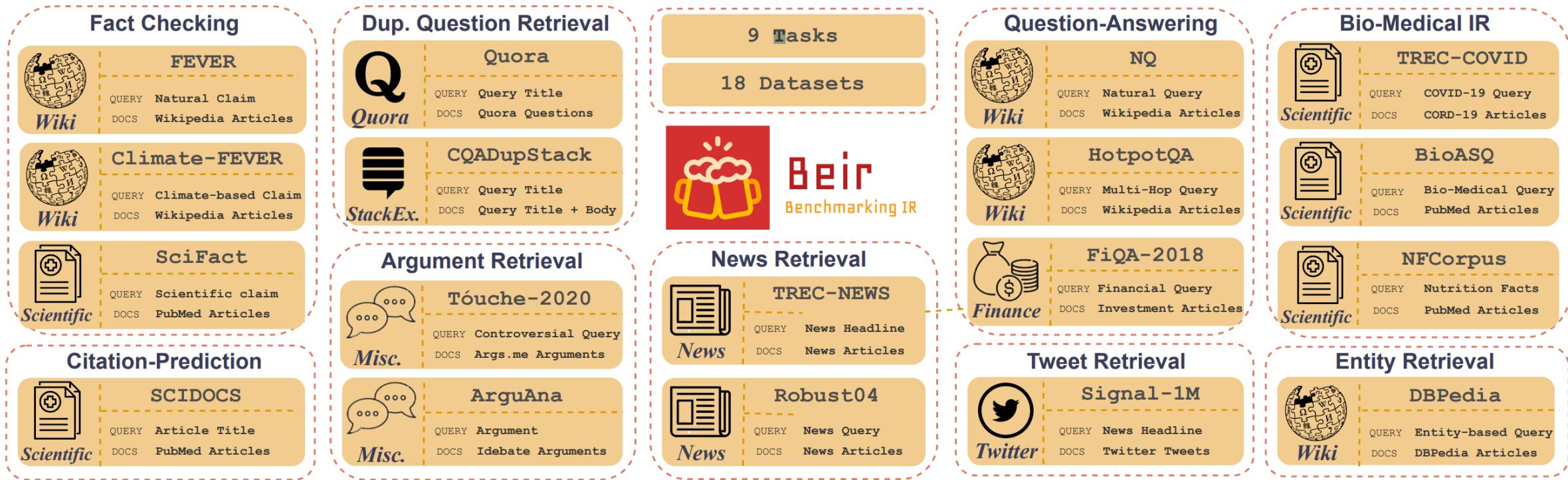


Formal, et al., "SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking" (2021), SIGIR 21

Formal et al., "SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval" (2021)

Slide inspired by <https://zzun.app/repo/naver-splade>

BEIR Benchmark



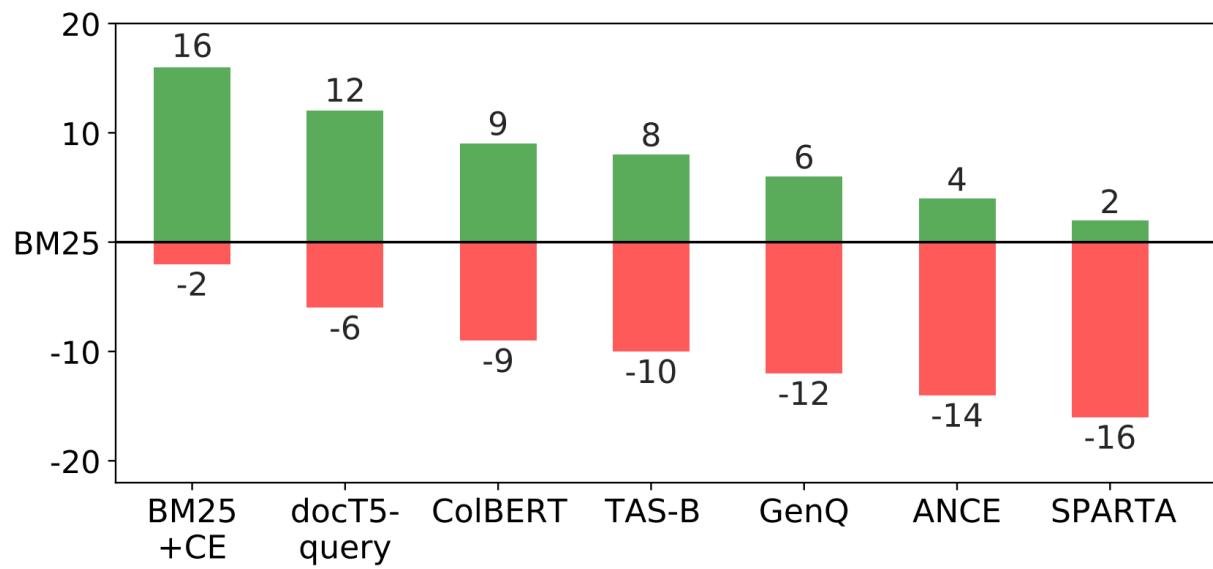
18 Diverse tasks and domains

Eval Zero-shot setting

Metrics: nDCG@10

BEIR Benchmark (Thakur, et al., 2021)

Key Takeaways



Comparison of neural retrieval model performance with BM25

- BM25 Strong Baseline
 - Sparse retrievers are the fastest
- Cross-Encoder Rerankers generalize well BUT come at the cost of high latency
- Dense single vector models optimized for specific datasets do not do well out of domain
- Late interaction (ColBERT) has better generalization

Dense IR

Why Dense Retrievers?

Need to extend beyond lexical matching

What is the body of water between England and Ireland?

Title: British Cycling

... **England** is not recognised as a region by the UCI, and there is no English cycling team outside the Commonwealth Games. For those occasions, British Cycling selects and supports the **England** team. Cycling is represented on the Isle of Man by the Isle of Man Cycling Association. Cycling in Northern **Ireland** is organised under Cycling Ulster, part of the all-Ireland governing **body** Cycling **Ireland**. Until 2006, a rival governing **body** existed, ...

Lexical Matching

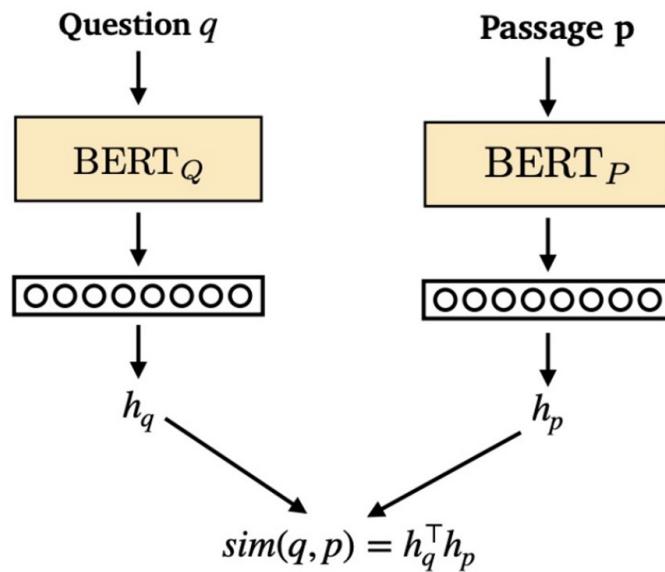


Semantic Matching

Title: Irish Sea

... Annual traffic between Great Britain and **Ireland** amounts to over 12 million passengers and of traded goods. **The Irish Sea** is connected to the North Atlantic at both its northern and southern ends. To the north, the connection is through the North Channel between Scotland and Northern **Ireland** and the Malin Sea. The southern end is linked to the Atlantic through the St George's Channel between Ireland and Pembrokeshire, and the Celtic Sea. ...

How to train Dense Retrievers?



Question *Positive P* *Negative P*
 $\mathcal{D} = \{\langle \underline{q_i}, \underline{p_i^+}, \underline{p_{i,1}^-}, \dots, \underline{p_{i,n}^-} \rangle\}_{i=1}^m$

NLL of positive passage

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

Contrastive Learning

Recent Advances Overview



Data Augmentation

- Unsupervised Data *Obtaining Positives in an Unsupervised Fashion*
- Denoising Negatives *Obtaining Better Negatives for Contrastive Learning*

Training Strategies

- Distillation
- Better Encoding Strategies

Improved Generalization

- Generalization to New Domains
- Generalization to New Tasks

Data Augmentation

Unsupervised Data

Obtaining Positives in an Unsupervised Fashion

- Contriever: Unsupervised Dense Information Retrieval with Contrastive Learning (Izacard et al 2021)
- Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval (Gao et al 2022)
- SPIDER: Learning to Retrieve Passages without Supervision (Ram et al 2022)
- AugTriever: Unsupervised Dense Retrieval by Scalable Data Augmentation (Meng et al 2023)

Data Augmentation

Unsupervised Data

Use **recurring spans** across passages in a document to create pseudo examples for contrastive learning



Data Augmentation

Unsupervised Data

Unsupervised Spider model shows comparable generalization as DPR dataset finetuned models

Model	# Examples	NQ	TriviaQA	WQ	TREC	SQuAD	EntityQs
DPR-NQ	58,880	-	69.0	68.8	85.9	48.9	49.7
DPR-TriviaQA	60,413	67.5	-	71.4	87.9	55.8	62.7
DPR-WQ	2,474	59.4	66.7	-	82.0	52.3	58.3
DPR-TREC	1,125	57.9	64.0	61.7	-	49.4	46.9
DPR-SQuAD	70,096	47.0	60.0	56.0	77.2	-	30.9
DPR-Multi	122,892	-	-	-	-	52.0	56.7
BM25	None	62.9	76.4	62.4	81.1	71.2	71.4
ICT	None	50.6	57.5	43.4	-	45.1	-
Spider	None	68.3	75.8	65.9	82.6	61.0	66.3
Spider-NQ	58,880	-	77.2	74.2	89.9	57.7	61.9
Spider-TriviaQA	60,413	75.5	-	73.7	91.2	68.1	72.9

After finetuning, Spider shows better cross-dataset generalization than DPR

Data Augmentation

Unsupervised Data

Obtaining Positives in an Unsupervised Fashion

- Contriever: Unsupervised Dense Information Retrieval with Contrastive Learning (Izacard et al 2021)
- Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval (Gao et al 2022)
- SPIDER: Learning to Retrieve Passages without Supervision (Ram et al 2022)
- AugTriever: Unsupervised Dense Retrieval by Scalable Data Augmentation (Meng et al 2023)

Denoising Negatives

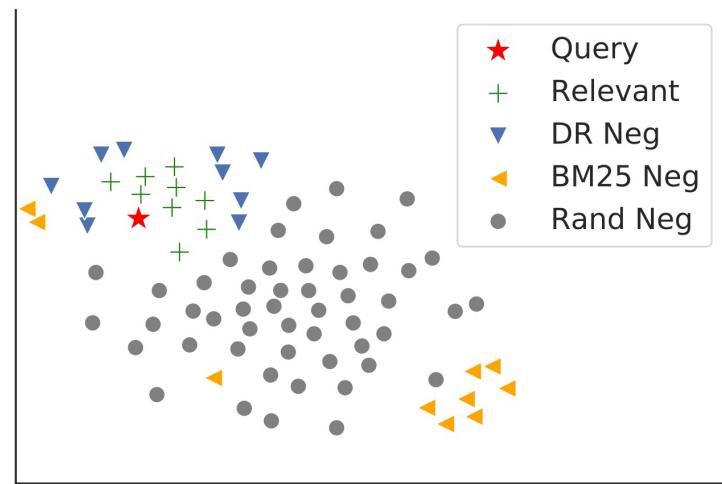
Obtaining Better Negatives for Contrastive Learning

- ANCE: Unsupervised Dense Information Retrieval with Contrastive Learning (Xiong et al 2020)
- RocketQA: An Optimized Training Approach to Dense Passage Retrieval (Qu et al 2021)
- ANCE-Tele: Reduce Catastrophic Forgetting of Dense Retrieval Training with Teleportation Negatives (Sun et al 2022)

Data Augmentation

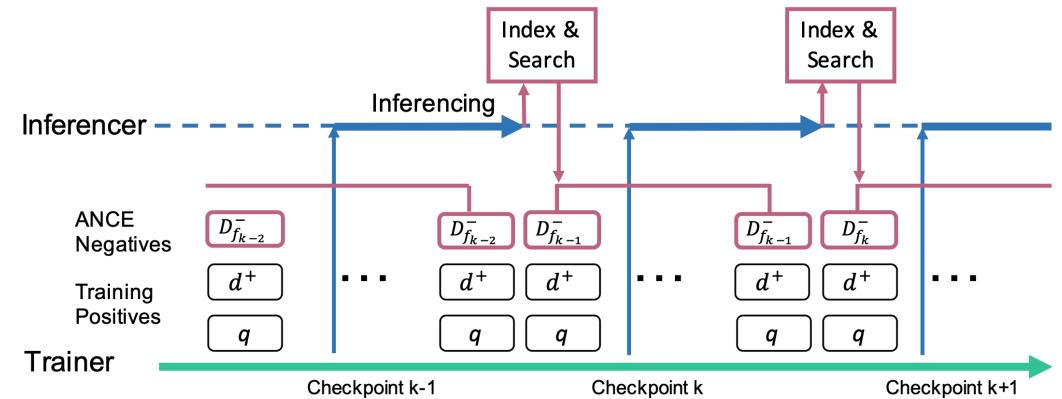
Denoising Negatives

Dense retriever's negatives are closer to query and relevant positives compared to BM25 and random negatives



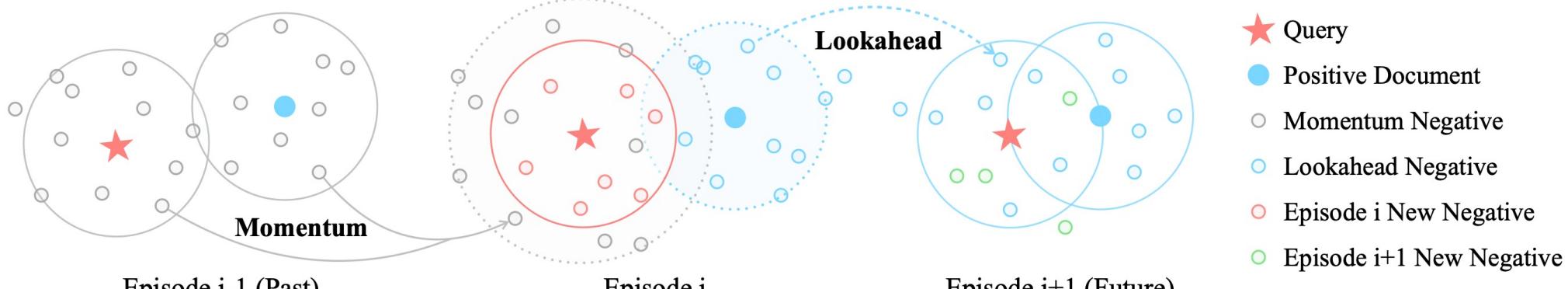
Sampling harder negatives helps the model learn to differentiate better!

Iteratively sample from model itself!



Data Augmentation

Denoising Negatives



Momentum Negatives

Training negatives from past episode.

- **Reduce Catastrophic Forgetting**
- **Improve Learning Stability**

Lookahead Negatives

Future hard negatives are neighbors of positive document.

- **Efficient future forecast**
- **Improve convergence speed**

Recent Advances Overview

```
graph LR; A[Recent Advances Overview] --> B[Data Augmentation]; A --> C[Training Strategies]; A --> D[Improved Generalization]
```

Data Augmentation

- Unsupervised Data *Obtaining Positives in an Unsupervised Fashion*
- Denoising Negatives *Obtaining Better Negatives for Contrastive Learning*

Training Strategies

- Distillation *Using Cross-Encoder as Teacher*
- Better Encoding Strategies *Architecture improvements*

Improved Generalization

- Generalization to New Domains
- Generalization to New Tasks

Training Strategies

Distillation

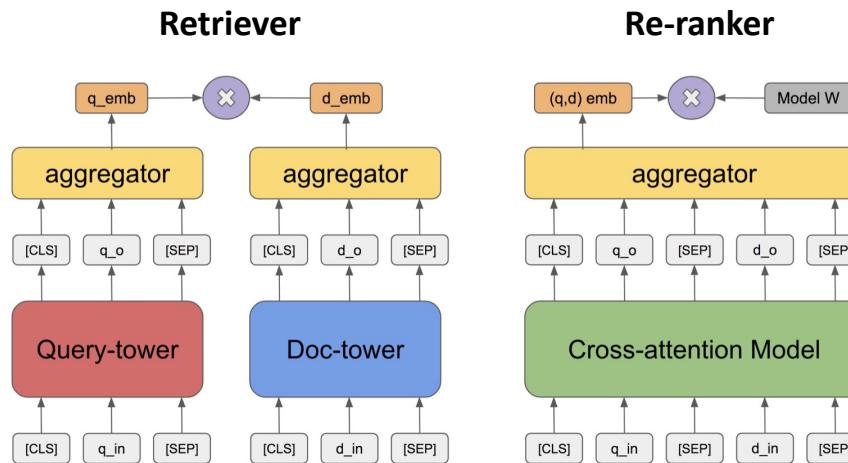
- Distilling Knowledge from Reader to Retriever for Question Answering (Izacard et al 2020)
- RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking (Ren et al 2021)
- LEAD: Liberal Feature-based Distillation for Dense Retrieval (Sun et al 2022)
- Curriculum Learning for Dense Retrieval Distillation (Zeng et al 2022)
- PROD: Progressive Distillation for Dense Retrieval (Lin et al 2023)

Training Strategies

Distillation

Cross-attention in re-ranker makes it more powerful than the retriever.

*Document representation
is obtained independent of
the query*

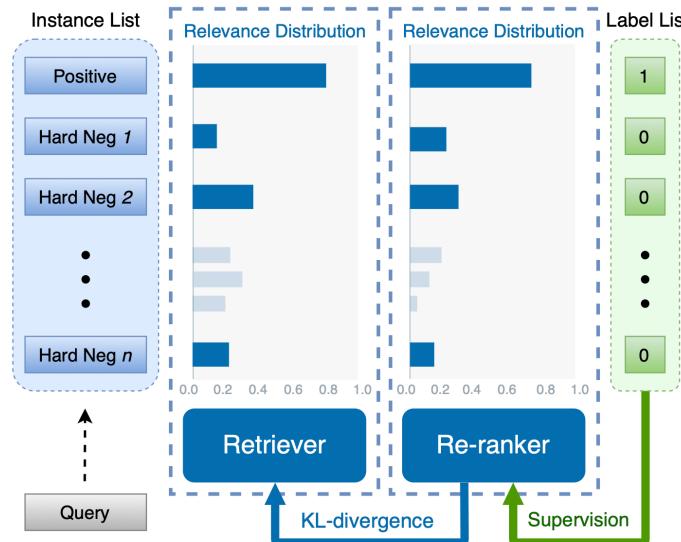


*More fine-grained attention
between query and document
in re-ranker*

Training Strategies

Distillation

Use KL-Divergence loss between re-ranker and retriever distributions to directly train the retriever.



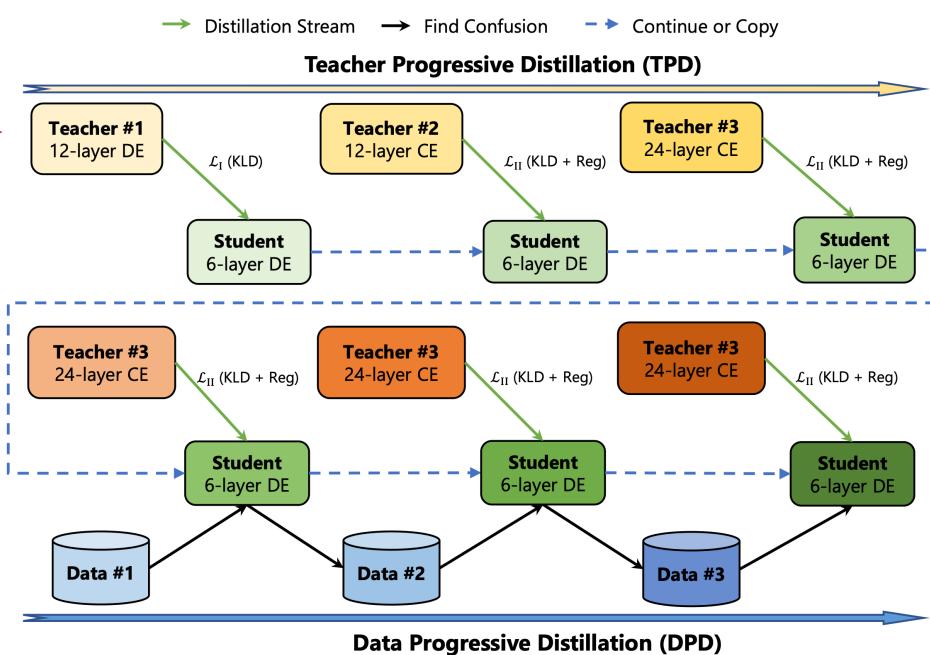
RocketQAv2 (Ren et al 2021)

However, distillation can still lead to a bad student due to non-negligible gap between capabilities of teacher and student.

Training Strategies

Distillation

Teacher progressive distillation and a data progressive distillation to gradually improve the student.



PROD: Progressive Distillation for Dense Retrieval (Lin et al 2023)

A 6-layer PROD model outperforms all 12-layer baselines!

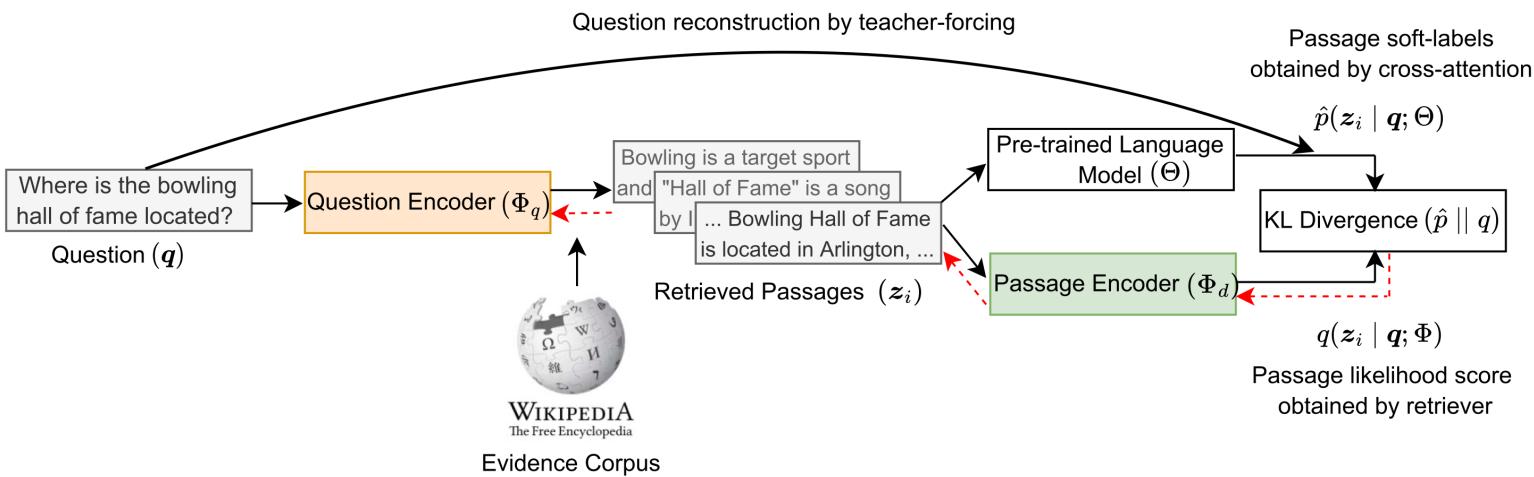
Method	Recall@5	Recall@20	Recall@100
BM25 [47]	-	59.1	73.7
GAR [27]	60.9	74.4	85.3
DPR [17]	-	78.4	85.4
ANCE [46]	-	81.9	87.5
RDR [48]	-	82.8	88.2
Joint Top- k [38]	72.1	81.8	87.8
DPR-PAQ [31]	74.5	83.7	88.6
Ind Top- k [38]	75.0	84.0	89.2
RocketQA v1 [33]	74.0	82.7	88.5
PAIR [34]	74.9	83.5	89.1
RocketQA v2 [35]	75.1	83.7	89.0
PROD	75.6*	84.7*†	89.6*†

Numbers on the Natural Questions dataset

Training Strategies

Distillation

Use a Pre-trained Language Model to obtain the (query-passage) relevance score



Use an instruction-tuned model and append a simple natural language instruction "**Please write a question based on this passage.**" to the passage text.

Training Strategies

Distillation

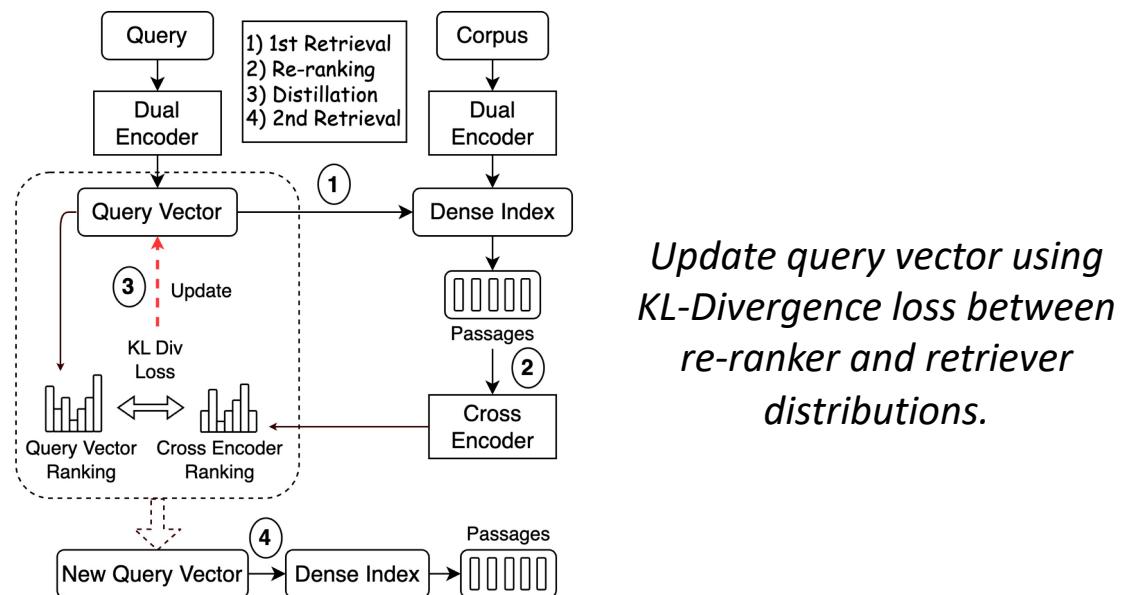
Retriever	Cross-Attention Language Model	SQuAD-Open		TriviaQA		NQ-Open		WebQ	
		Top-20	Top-100	Top-20	Top-100	Top-20	Top-100	Top-20	Top-100
<i>Unsupervised Approaches (trained using Wikipedia / Internet data)</i>									
BERT	T5* (220M)	5.2	13.5	7.2	17.8	9.4	20.3	3.7	12.8
ICT		45.1	65.2	57.5	73.6	50.6	66.8	43.4	65.7
MSS		51.3	68.4	68.2	79.4	59.8	74.9	49.2	68.4
BM25		71.1	81.8	76.4	83.2	62.9	78.3	62.4	75.5
Contriever		63.4	78.2	74.2	83.2	67.8	82.1	74.9	80.1
Spider		61.0	76.0	75.8	83.5	68.3	81.2	65.9	79.7
cpt-text S [†]		—	—	75.1	81.7	65.5	77.2	—	—
HLP		—	—	76.9	84.0	70.2	82.0	66.9	80.8
<i>Supervised Approaches (trained using question-passage aligned data)</i>									
DPR	ERNIE* (110M)	63.2	77.2	79.4	85.0	78.4	85.4	73.2	81.4
DPR-Multi [‡]		51.6	67.6	78.8	84.7	79.4	86.0	75.0	82.9
ANCE		—	—	80.3	85.3	81.9	87.5	—	—
ICT-DPR		—	—	81.7	86.3	81.8	88.0	72.5	82.3
MSS-DPR [◦]		73.1	84.5	81.8	86.6	82.1	87.8	76.9	84.6
coCondenser		—	—	83.2	87.3	84.3	89.0	—	—
RocketQAv2		—	—	—	—	83.7	89.0	—	—
EMDR ^{2◦}		—	—	83.4	87.3	85.3	89.7	79.1	85.2
AR2		—	—	84.4	87.9	86.0	90.1	—	—
<i>Our Approach (trained using questions and Wikipedia text)</i>									
ART	T5-lm-adapt (11B)	74.2	84.3	82.5	86.6	80.2	88.4	74.4	82.7
ART-Multi	T5-lm-adapt (11B)	72.8	83.2	82.2	86.6	81.5	88.5	74.8	83.7
ART	T0 (3B)	75.3	85.0	82.9	87.1	81.6	89.0	75.7	84.3
ART-Multi	T0 (3B)	74.7	84.5	82.9	87.0	82.0	88.9	76.6	85.0

Outperforms most models trained with question-passage aligned data.

Training Strategies

Distillation (but at inference!)

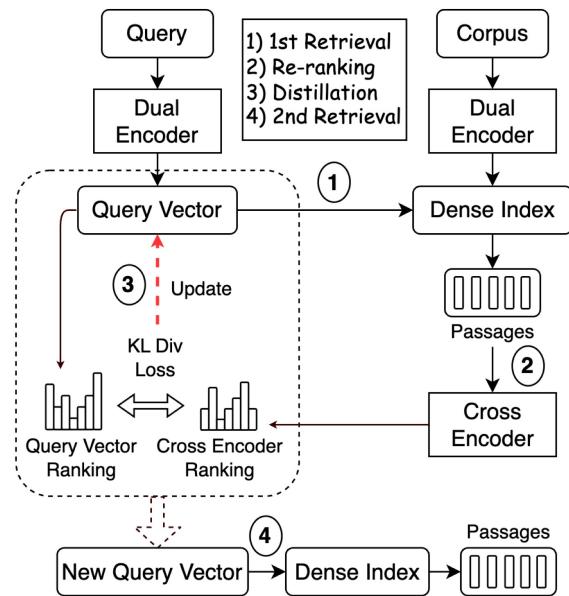
Distillation from Re-ranker can be done at *inference* too!



Training Strategies

Distillation (but at inference!)

Distillation from Re-ranker can be done at *inference* too!



- Minimal additional latency at inference.
- Performs better than reranking more passages.

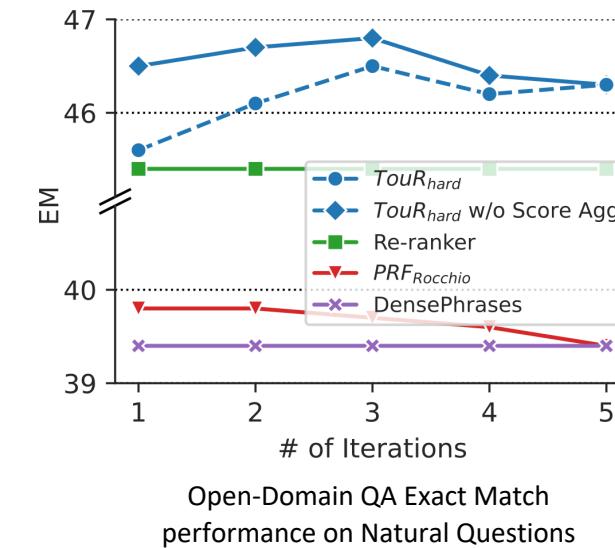
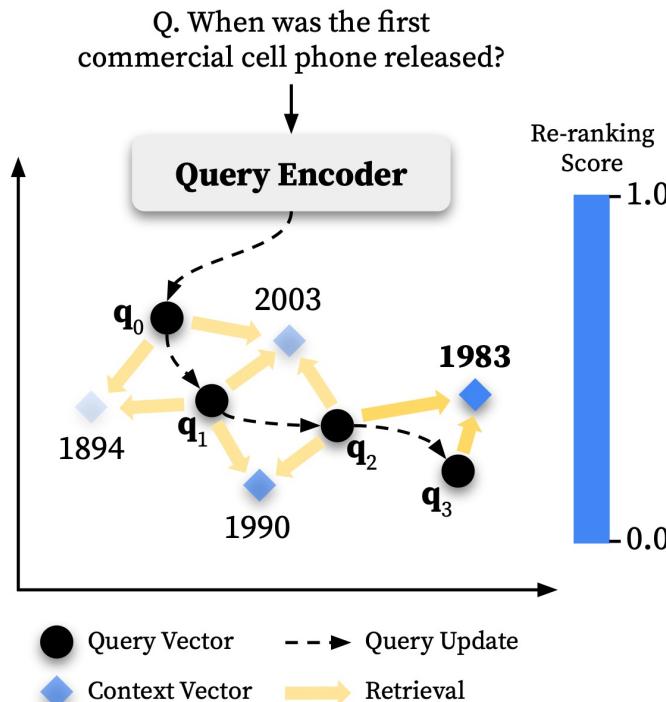
Method	R@100	Extra Latency
Retrieve and Re-rank (100 passages)	66.8	0%
Retrieve and Re-rank (125 passages)	67.6	24.3%
Inference-time distillation (25 updates)	68.3	3.0% (7.5+40)ms
Inference-time distillation (50 updates)	68.8	3.5% (15+40)ms
Inference-time distillation (100 updates)	69.2	4.4% (30+40)ms
Inference-time distillation (500 updates)	69.1	7.3% (75+40)ms

Numbers on the BEIR Benchmark

Training Strategies

Distillation (but at inference!)

Process can be repeated multiple times to iteratively update the query vector.



Training Strategies

Distillation

- Distilling Knowledge from Reader to Retriever for Question Answering (Izacard et al 2020)
- RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking (Ren et al 2021)
- LEAD: Liberal Feature-based Distillation for Dense Retrieval (Sun et al 2022)
- Curriculum Learning for Dense Retrieval Distillation (Zeng et al 2022)
- PROD: Progressive Distillation for Dense Retrieval (Lin et al 2023)

Better Encoding Strategies

- Condenser: a pre-training architecture for dense retrieval (Gao et al 2021)
- RetroMAE: Pre-Training Retrieval-oriented Language Models Via Masked Auto-Encoder (Xiao et al 2022)
- ConTextual Masked Auto-Encoder for Dense Passage Retrieval (Wu et al 2022)
- Led: Lexicon-enlightened dense retriever for large-scale retrieval (Zhang et al 2023)

Recent Advances Overview

```
graph LR; A[Recent Advances Overview] --> B[Data Augmentation]; A --> C[Training Strategies]; A --> D[Improved Generalization]
```

Data Augmentation

- Unsupervised Data *Obtaining Positives in an Unsupervised Fashion*
- Denoising Negatives *Obtaining Better Negatives for Contrastive Learning*

Training Strategies

- Distillation *Using Cross-Encoder as Teacher*
- Better Encoding Strategies *Architecture improvements*

Improved Generalization

- Generalization to New Domains *Handling Distribution Shifts*
- Generalization to New Tasks *Handling Different Search Intents*

Improved Generalization

To New Domains *Handling Distribution Shifts*

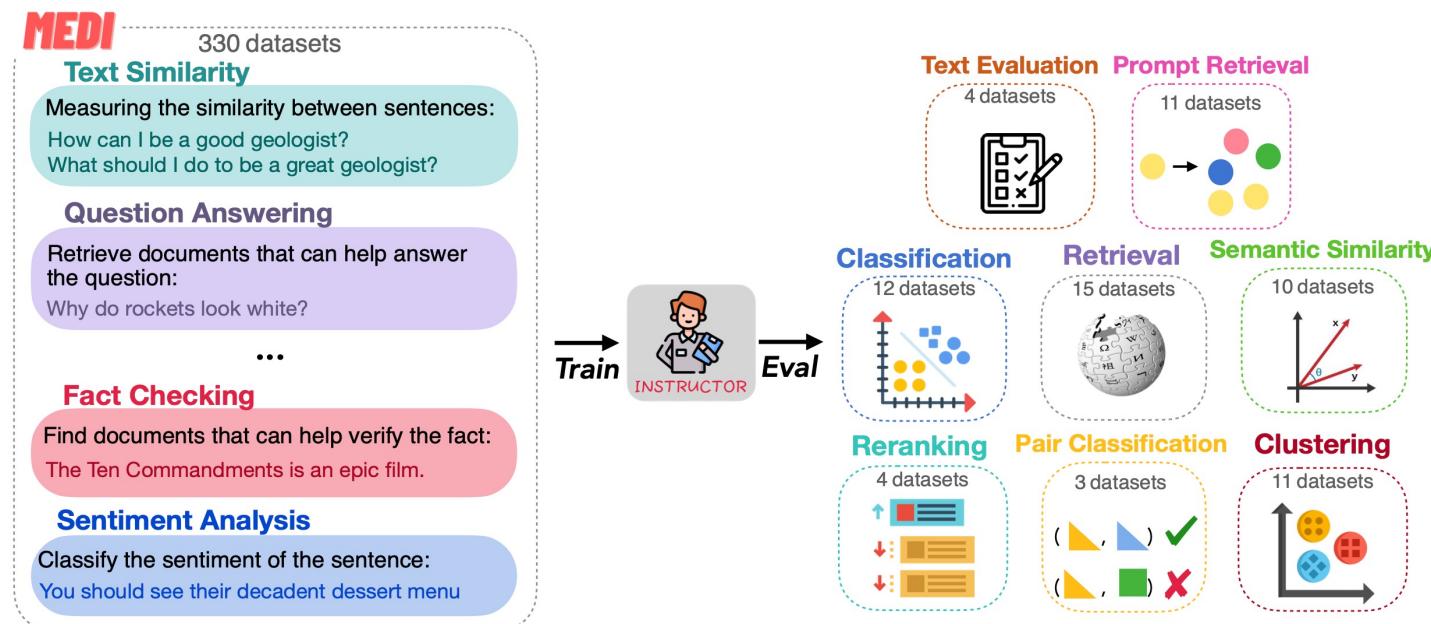
- GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval (Wang et al 2021)
- Zero-Shot Dense Retrieval with Momentum Adversarial Domain Invariant Representations (Xin et al 2021)
- COCO-DR: Combating Distribution Shifts in Zero-Shot Dense Retrieval with Contrastive and Distributionally Robust Learning (Yu et al 2022)
- Continually Adaptive Neural Retrieval Across the Legal, Patent and Health Domain (Althammer et al 2022)
- MoMA: Augmenting Zero-Shot Dense Retrievers with Plug-in Mixture-of-Memories (Suyu et al 2023)
- How to Train Your DRAGON: Diverse Augmentation Towards Generalizable Dense Retrieval (Lin et al 2023)

Improved Generalization

To New Tasks

Handling Different Search Intents

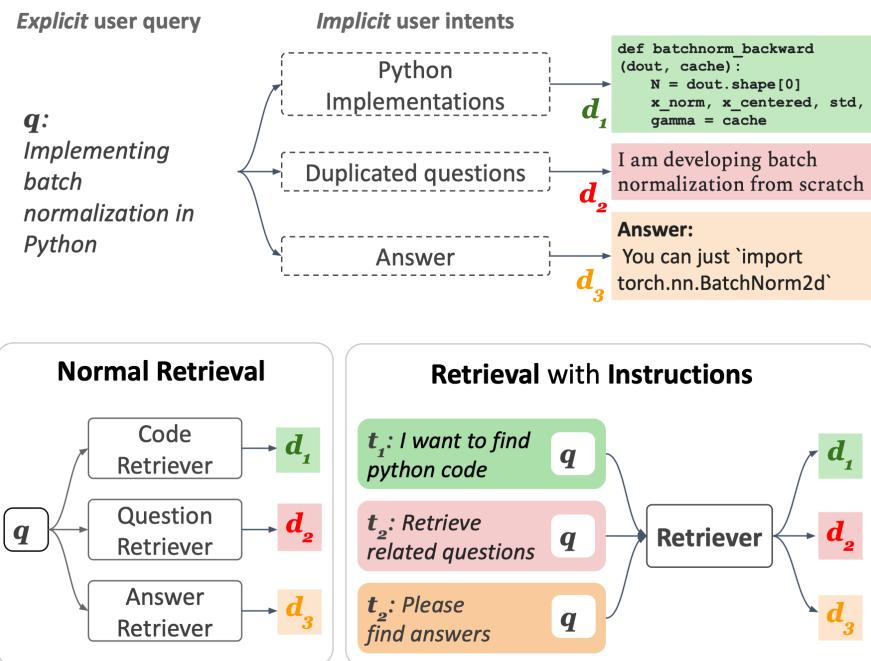
- Promptagator: Few-shot Dense Retrieval from 8 Examples (Dai et al 2022)
- TART: Task-aware Retrieval with Instructions (Asai et al 2022)
- TACO: Improving Multitask Retrieval by Promoting Task Specialization (Zhang et al 2023)



Instructor: One Embedder, Any Task: Instruction-Finetuned Text Embeddings (Su et al 2023)

Improved Generalization

To New Tasks

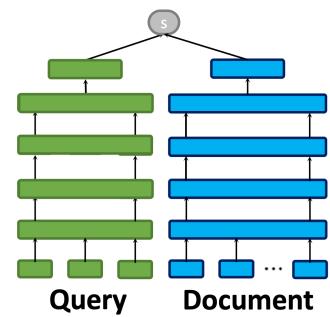


Dataset	Instruction
NQ	Retrieve a Wikipedia paragraph that answers this question.
Med Simple	Your task is to find a simplified paragraph of this paragraph from a medical paper.
QReCC	Find a dialogue response from dialogue history to answer the user's question.
Arguana	Retrieve a paragraph from an argument website that argues against the following argument.
SciFact	Find a sentence from a scientific paper to check if the statement is correct or not.
MultiLexSum	I want to find the one-sentence summary of this legal case.

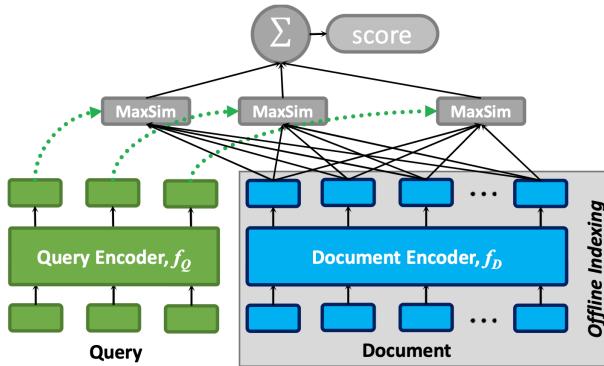
Each instruction defines *intent*, *domain* and *unit*.

Overcomes the need for a separate retriever for each task

Single-Vector vs Multi-Vector

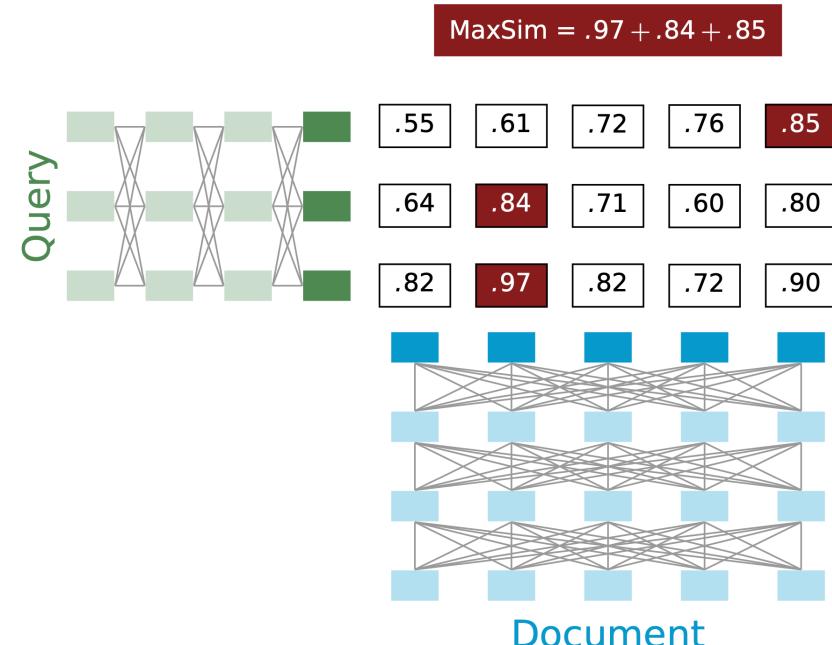


Single Representation for Query and Document



Token-level Representations for Query and Document

$$S_{q,d} := \sum_{i \in [|E_q|]} \max_{j \in [|E_d|]} E_{qi} \cdot E_{dj}^T$$



Late Contextual Interactions!

Outline

-  Motivation of OpenQA
-  Retrievers: Sparse and Dense
-  Recent Advances for OpenQA Efficient Neural Retrievers
-  Recent Advances in Efficient Multilingual Retrieval
-  Reproducibility in OpenQA: Hands-On Guide I
-  Q&A: [15 min]

1st Half



Coffee Break.....

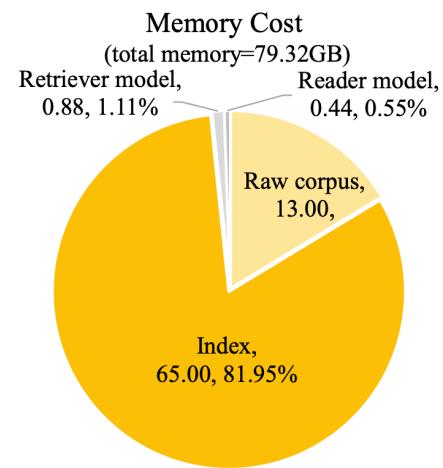
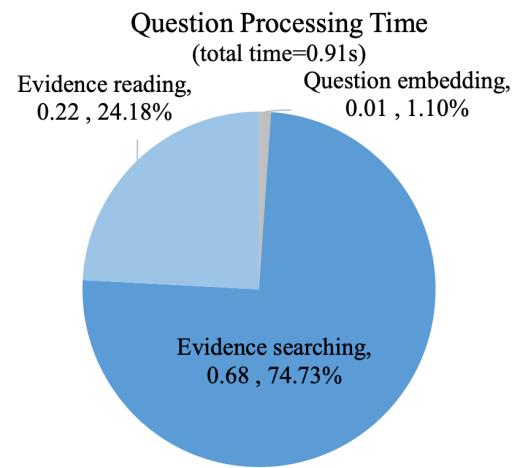
-  Multilingual Readers
-  Multi-modal Readers: Text, Table, Visual QA
-  Large Language Models as Retrievers/Readers
-  Reproducibility in OpenQA: Hands-on Guide II
-  Pipelines, Service and Deployment
-  Q&A: [15 min]

2nd Half

Recent Advances in Efficient Retrievers

Why is efficiency important?

Evidence searching does take up considerable time in the pipeline



Index size dominates the overall storage cost

Question processing time and memory cost for DPR on NQ test set.

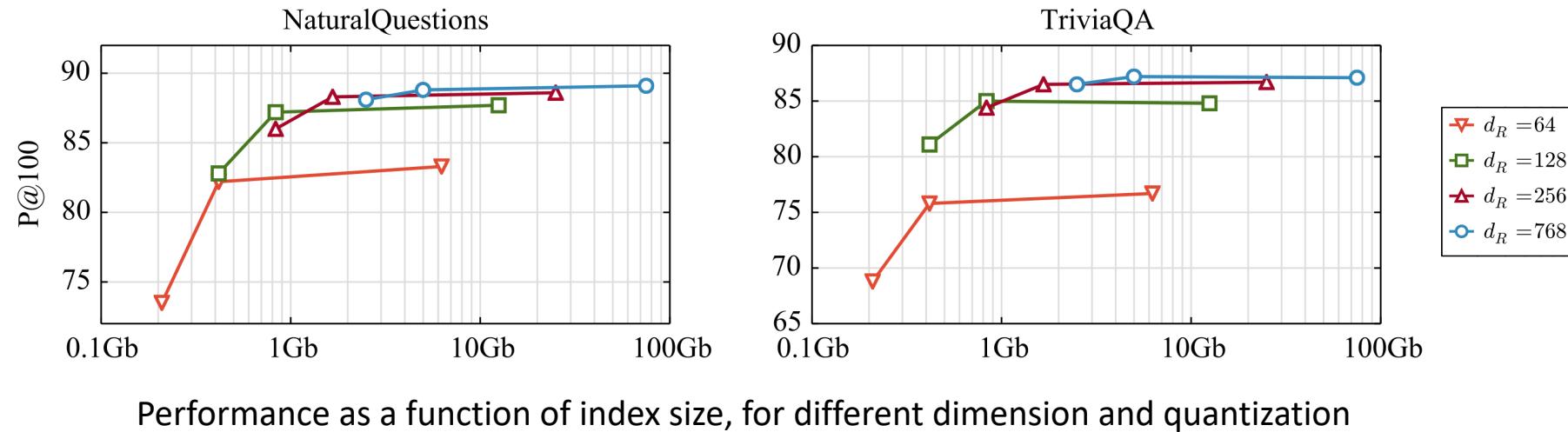
Need for Multi-dimensional Benchmarking

*Latency and hardware cost
are important considerations
for retrieval systems*

*SOTA can be different under
various constraints*

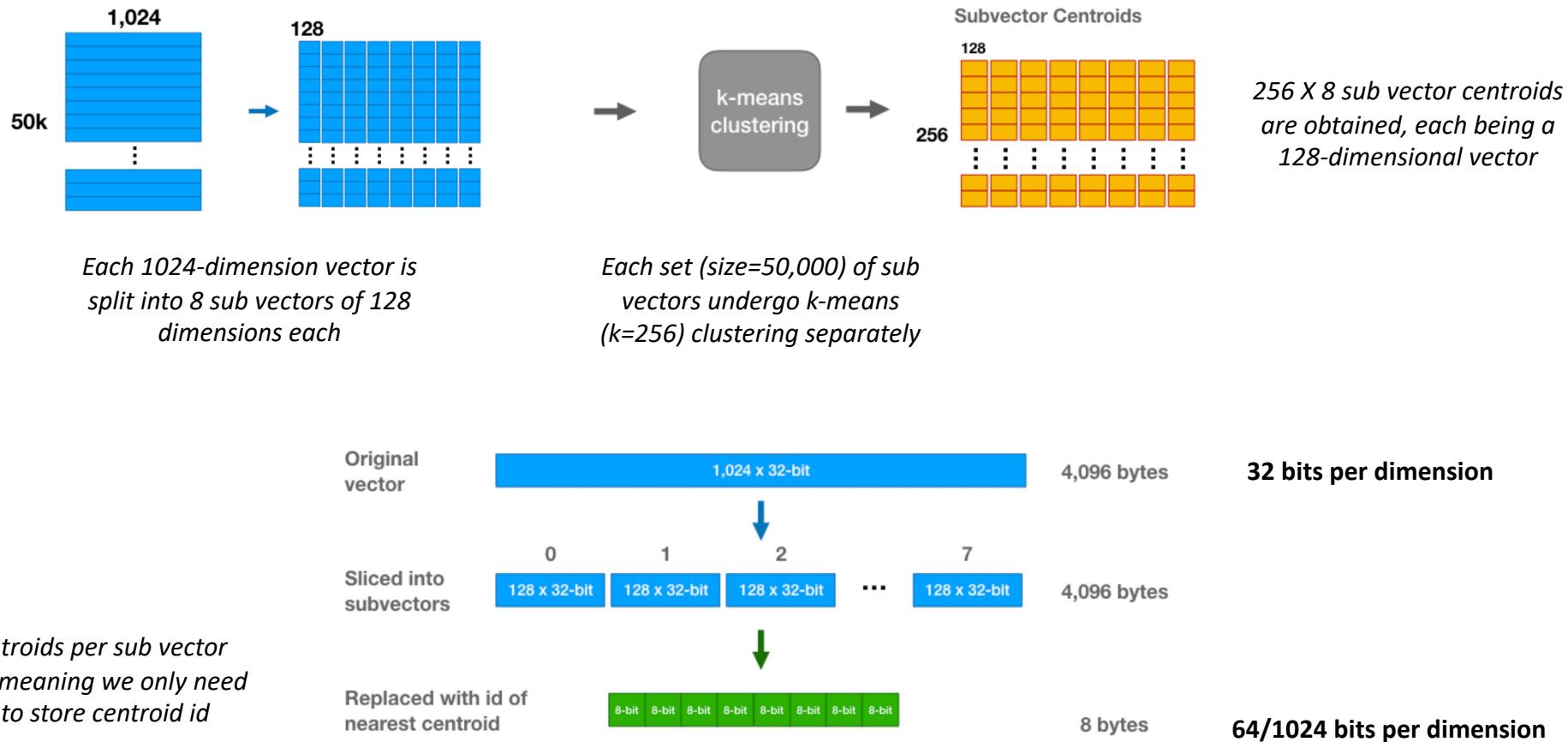
	Hardware				Performance		
	GPU	CPU	RAM	Instance	Latency	Cost	Success@10
BM25	0	1	4	m6gd.med	11	\$0.14	38.6
BM25	0	1	32	x2gd.lrg	10	\$0.48	38.6
DPR					146	\$6.78	52.1
ColBERTv2-S					206	\$9.58	68.8
ColBERTv2-M					321	\$14.90	69.6
ColBERTv2-L					459	\$21.30	69.7
BT-SPLADE-L					46	\$2.15	66.3
BM25	1	16	32	p3.8xl	9	\$29.94	38.6
DPR					18	\$61.06	52.1
ColBERTv2-S					27	\$90.41	68.8
ColBERTv2-M					36	\$123.35	69.6
ColBERTv2-L					55	\$187.24	69.7
BT-SPLADE-L					33	\$112.87	66.3

Reducing the Index Size

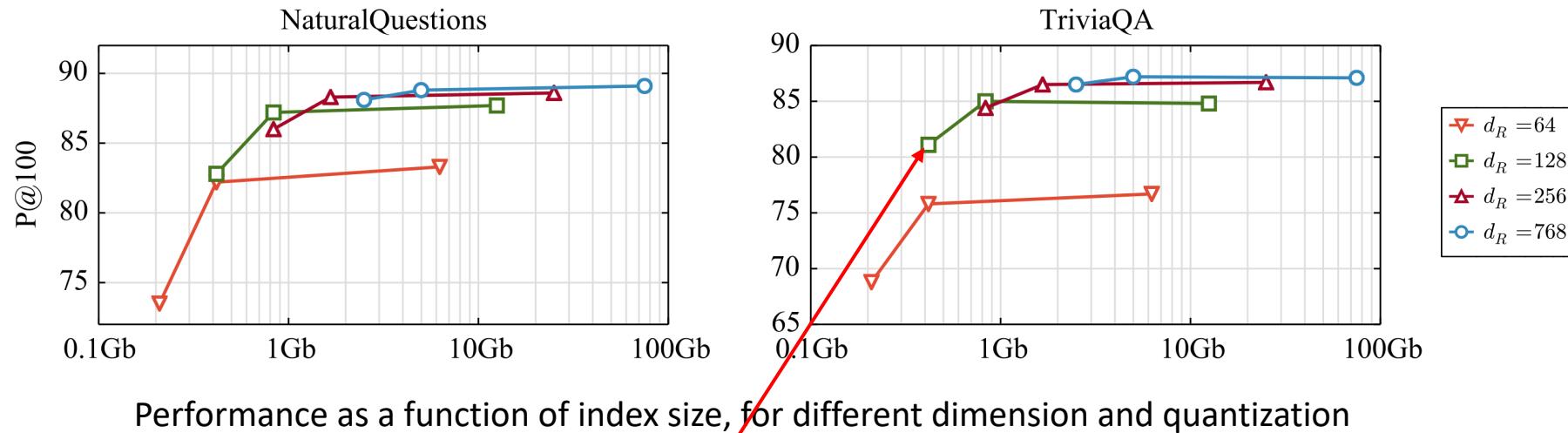


- **Dimension Reduction:**
 - Reduce dimension of dense representations
 - Achieved by adding a linear layer to output of encoder
- **Product Quantization:**
 - Quantized to 1 bit (left), 2 bit (middle) and 32 bit (right) per dimension respectively.

Product Quantization



Product Quantization



Product Quantization

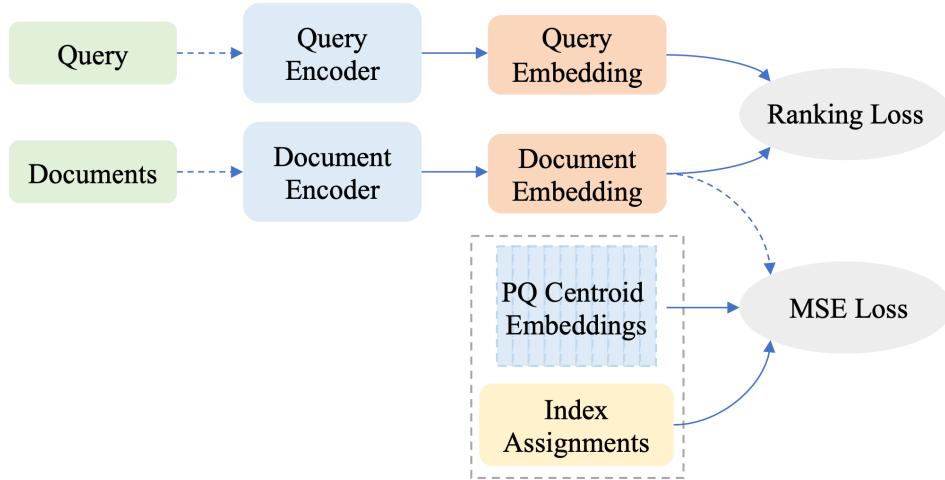
Number of dimensions = 128

Number of sub vectors = 16

Number of centroids = 256

Final Size = $(16*8)/128 = 1$ bit per dimension

Optimizing Product Quantization



$$s(q, d) = \langle \vec{q}, \vec{d} \rangle \approx s^\dagger(q, d) = \langle \vec{q}, \vec{d}^\dagger \rangle$$

Quantizing Document Embeddings

Let $\vec{c}_{i,j}$ be the j^{th} centroid embedding from the i^{th} set

$$\vec{c}_{i,j} \in R^{\frac{D}{M}}$$

$\Phi_i(d)$ is the index assignment for i^{th} subvector of d

$$\vec{d} \rightarrow \vec{d}^\dagger = \vec{c}_{1,\Phi_1(d)}, \vec{c}_{2,\Phi_2(d)}, \dots, \vec{c}_{M,\Phi_M(d)} \in R^D$$

Optimization Objective

Train $\{\vec{c}_{i,j}\}$ and ϕ to minimize the MSE loss.

$$\vec{c}_{i,j}, \phi = \operatorname{argmin} \|\vec{d} - \vec{d}^\dagger\|^2$$

Search Procedure

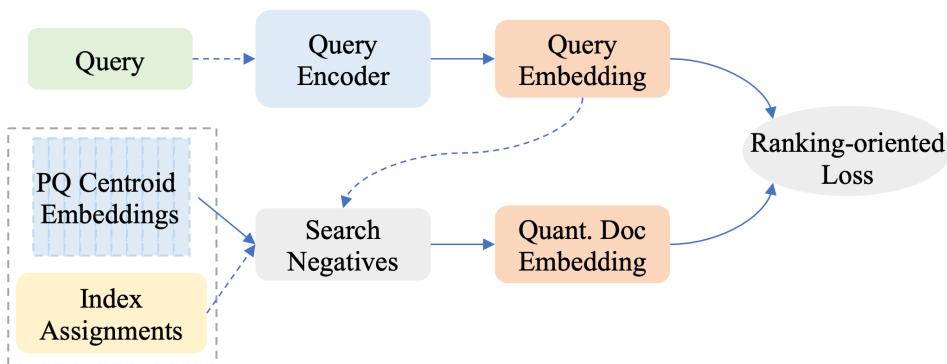
Query embedding gets split into M subvectors:

$$\vec{q} = \vec{q}_1, \vec{q}_2, \dots, \vec{q}_M$$

$$s^\dagger(q, d) = \sum_{i=1}^M \langle \vec{q}_i, \vec{c}_{i,\phi_i(d)} \rangle$$

Jointly Optimizing Product Quantization

Jointly optimize PQ with Query Encoder!



$$s(q, d) = \langle \vec{q}, \vec{d} \rangle \approx s^\dagger(q, d) = \langle \vec{q}, \vec{d}^\dagger \rangle$$

Traditional pair-wise ranking loss

$$l(s(q, d^+), s(q, d^-))$$

Use $s^\dagger(q, d)$ to compute the new ranking-oriented loss

$$l(s^\dagger(q, d^+), s^\dagger(q, d^-))$$

However, Index Assignments are not differentiable
Thus, only do Centroid Optimization

Optimization Objective

$$f^*, \{\vec{c}_{i,j}\}^* = \arg \min_{f, \{\vec{c}_{i,j}\}} \sum_q \sum_{d^+ \in \mathcal{D}_q^+} \sum_{d^- \in \mathcal{D}_q^-} \ell(s^\dagger(q, d^+), s^\dagger(q, d^-))$$

$$\frac{\partial s^\dagger(q, d)}{\partial \vec{c}_{i,j}} = \begin{cases} \vec{q}_i, & \text{if } j = \varphi_i(d). \\ 0, & \text{otherwise.} \end{cases}$$

CoBERTv2

Residual compression for Scalable Neural Information Retrieval

v is encoded as index of closest centroid C_t and a *quantized* vector \tilde{r} .

Residual $\longrightarrow \tilde{r} \approx r = v - C_t$

To encode \tilde{r} , every dimension of r is *quantized* to 1 bit

For a 128-dimensional vector

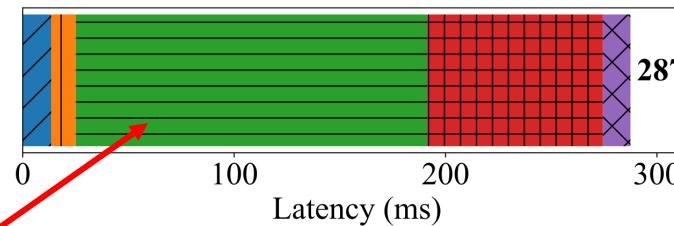
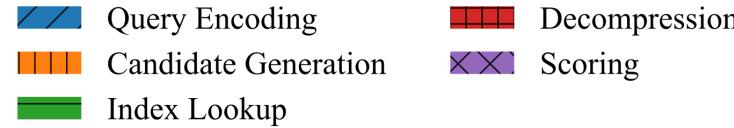
Normal Size = 256 bytes
at 16-bit precision

Total Size = 4 bytes + 16 bytes

For upto 2^{32} centroids

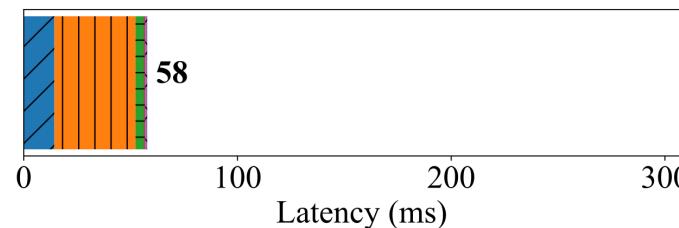
1 bit per dimension

PLAID



(a) Vanilla ColBERTv2 ($n_{probe}=4$, $n_{candidates}=2^{16}$).

Memory overhead from centroid and residual retrieval over a huge index



(b) PLAID ColBERTv2 ($k = 1000$)

Outline

-  Motivation of OpenQA
-  Retrievers: Sparse and Dense
-  Recent Advances for OpenQA Efficient Neural Retrievers
-  Recent Advances in Efficient Multilingual Retrieval
-  Reproducibility in OpenQA: Hands-On Guide I
-  Q&A: [15 min]

1st Half



Coffee Break.....

-  Multilingual Readers
-  Multi-modal Readers: Text, Table, Visual QA
-  Large Language Models as Retrievers/Readers
-  Reproducibility in OpenQA: Hands-on Guide II
-  Pipelines, Service and Deployment
-  Q&A: [15 min]

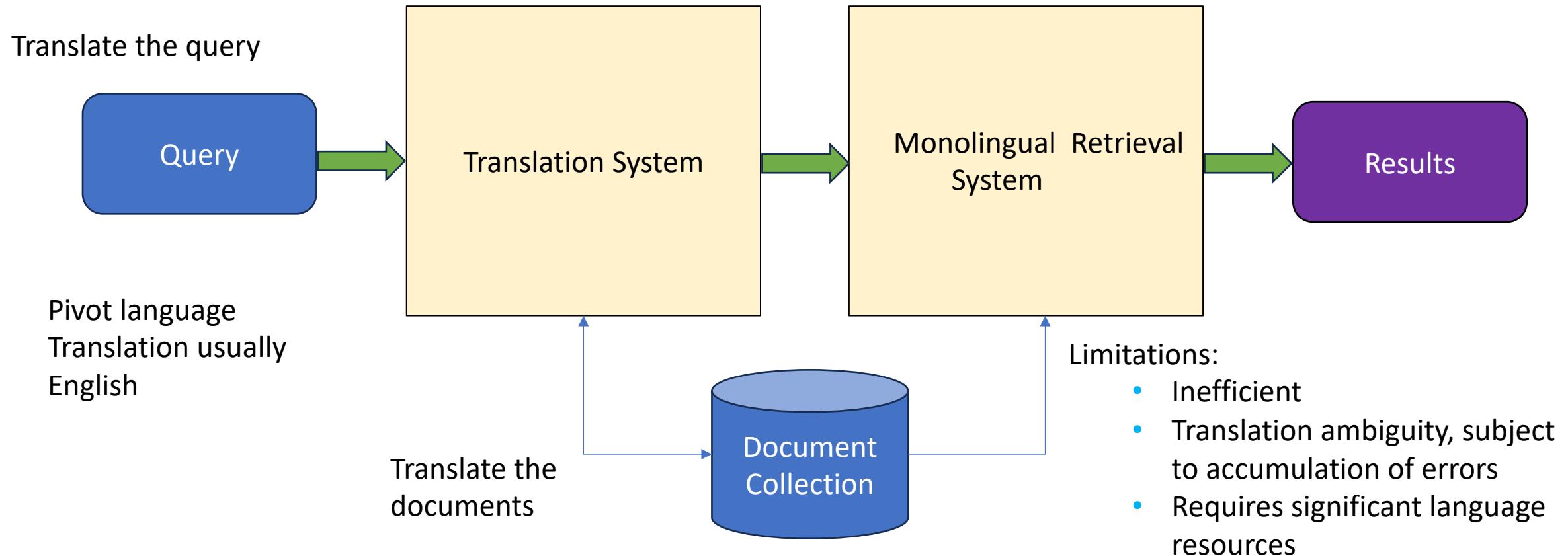
2nd Half

Multilingual IR

Outline

- Initial systems based on Machine Translation
- Datasets
 - MrTyDI
 - MKQA
 - XOR-TyDI QA
- Approaches
 - Multilingual BERT based neural ranking
 - Cross-lingual alignment using Parallel Corpora
 - Joint IR and LM Pretraining with parallel corpora
 - DrDECR SOTA model on XOR-TyDI QA Retrieve Task

Machine Translation based approach



Martin, McCarley, Roukos. "Ad hoc and multilingual information retrieval at IBM." *NIST SP* (1999)
Dwivedi and Chandra, "A Survey on Cross Language Information Retrieval" (2016)

Multilingual IR Datasets Comparison

Dataset	Source	Translated	Information Seeking	Cross-lingual	Answer Only	Parallel Questions	Languages	Total Examples
MrTyDI ¹	TyDi QA	N	Y	N	N	N	11	204K
MKQA ²	NQ	Y	Y	Y	Y	Y	26	260K
XOR-TyDI QA ³	TyDi QA	N	Y	Y	N	N	7	40K

¹Zhang et al., “Mr. TyDi: A Multi-lingual Benchmark for Dense Retrieval”, EMNLP 2021

²Longpre et al., “MKQA: A Linguistically Diverse Benchmark for Multilingual Open Domain Question Answering”, 2021

³Asai et al., “XOR QA: Cross-lingual Open-Retrieval Question Answering”, NAACL 2021

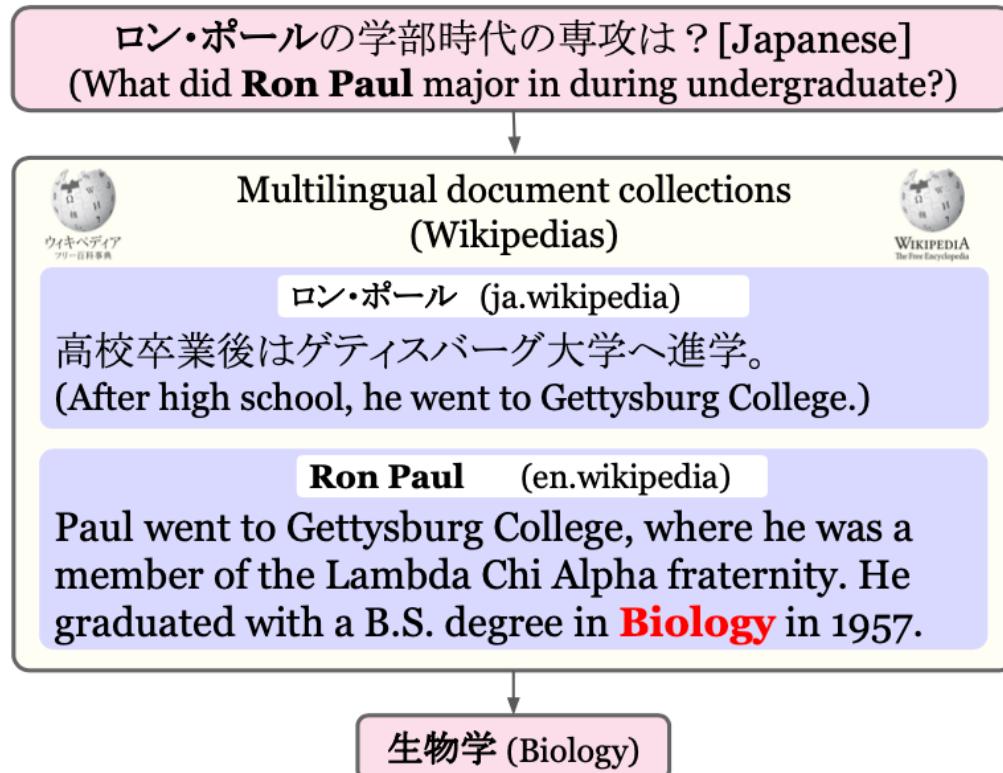
Multilingual IR Datasets from QA dataset

- MrTyDI (Zhang et al., 2021)¹
 - Consists of questions and gold passages from TyDIQA (Clark et al., 2020)
 - Best used for Monolingual retrieval in non-English languages
- MKQA (Longpre, et al., 2021)²
 - 10K examples from Natural Questions (NQ) (Kwiatkowski et al., 2019)
 - Question and Answers are human translated to 26 languages
 - Best used for zero-shot evaluation of cross-lingual retrieval

¹Zhang et al., “Mr. TyDi: A Multi-lingual Benchmark for Dense Retrieval”, EMNLP 2021

²Longpre et al., “MKQA: A Linguistically Diverse Benchmark for Multilingual Open Domain Question Answering”, 2021

XOR-TyDI QA



Cross-lingual IR

Three Subtasks:

- **XOR-Retrieve** – cross lingual passage retrieval from English Wikipedia
- **XOR-EnglishSpan** - cross lingual retrieval from English Wikipedia and answer span selection
- **XOR-Full** – cross lingual retrieval from Wikipedia in multiple languages and answer in the language of the question

Leaderboard

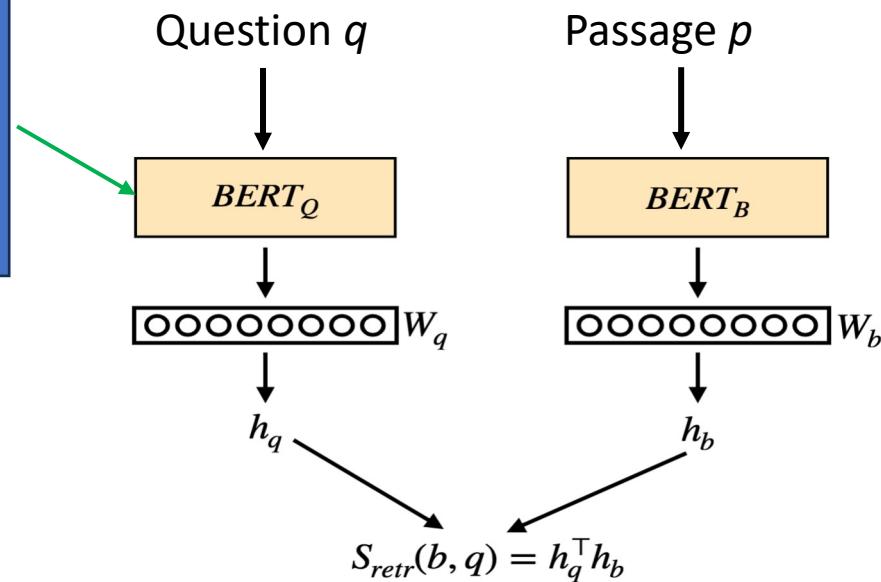
<https://nlp.cs.washington.edu/xorqa/>

Multilingual BERT based neural ranking

Multilingual Adaptation of DPR [Karpukhin et al., 2020]

- Initialize the model with multilingual BERT (mBERT)¹
- Train with English (NQ) data identical to DPR

Zero-shot baseline for MrTyDI dataset



- Cross-lingual capabilities of mBERT²
- Strong zero-shot cross-lingual transfer and ability to generalize across languages
 - Pretraining on English training data such as NQ, MSMarco helps
 - Using data from all the languages beats using data only from the target language

¹Zhang, Xinyu and Ma, Xueguang and Shi, Peng and Lin, Jimmy “Mr. {T}y{D}i: A Multi-lingual Benchmark for Dense Retrieval”, ACL 2021

²Zhang}, Xinyu and {Oguejji}, Kelechi and {Ma}, Xueguang and {Lin}, Jimmy, “Towards Best Practices for Training Multilingual Dense Retrieval Models”, 2022

Limitations: BERT-based approach

- Do not explicitly learn the alignment across different languages
- Unbalanced pre-training data in different languages results in performance gap between high and low-resource languages
- Annotated training data is costly to obtain in many languages

Leverage Parallel and Monolingual Corpora

Joint IR and LM Pretraining

- ✓ Hu, Xiayang et. al., “Language Agnostic Multilingual Information Retrieval with Contrastive Learning” (ACL 2023)

Knowledge Distillation

- ✓ Li et al., “Learning Cross-Lingual IR from an English Retriever” (NAACL,2022)

Monolingual Pretraining Task transfers to Cross-lingual pairs

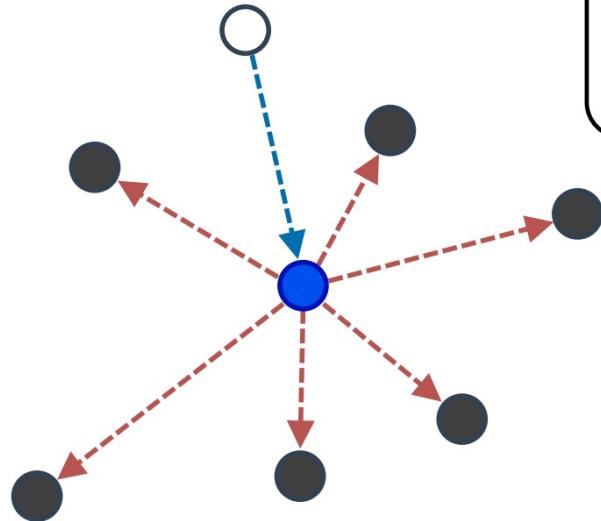
- Wu et al., "Unsupervised context aware sentence representation for multi-lingual dense retrieval" (IJCAI 2022)

Multilingual IR with Contrastive Learning

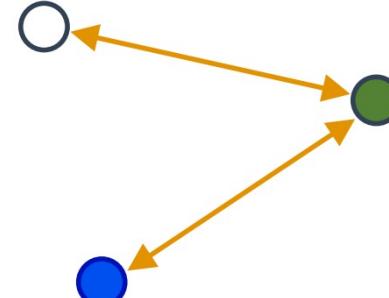
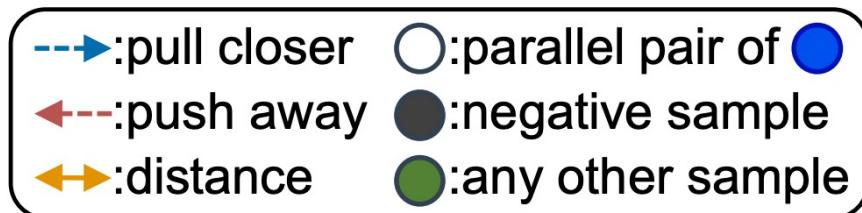
Annotated training data is costly to obtain in many languages

Use parallel corpora between English and other languages

Use only
English IR
training data



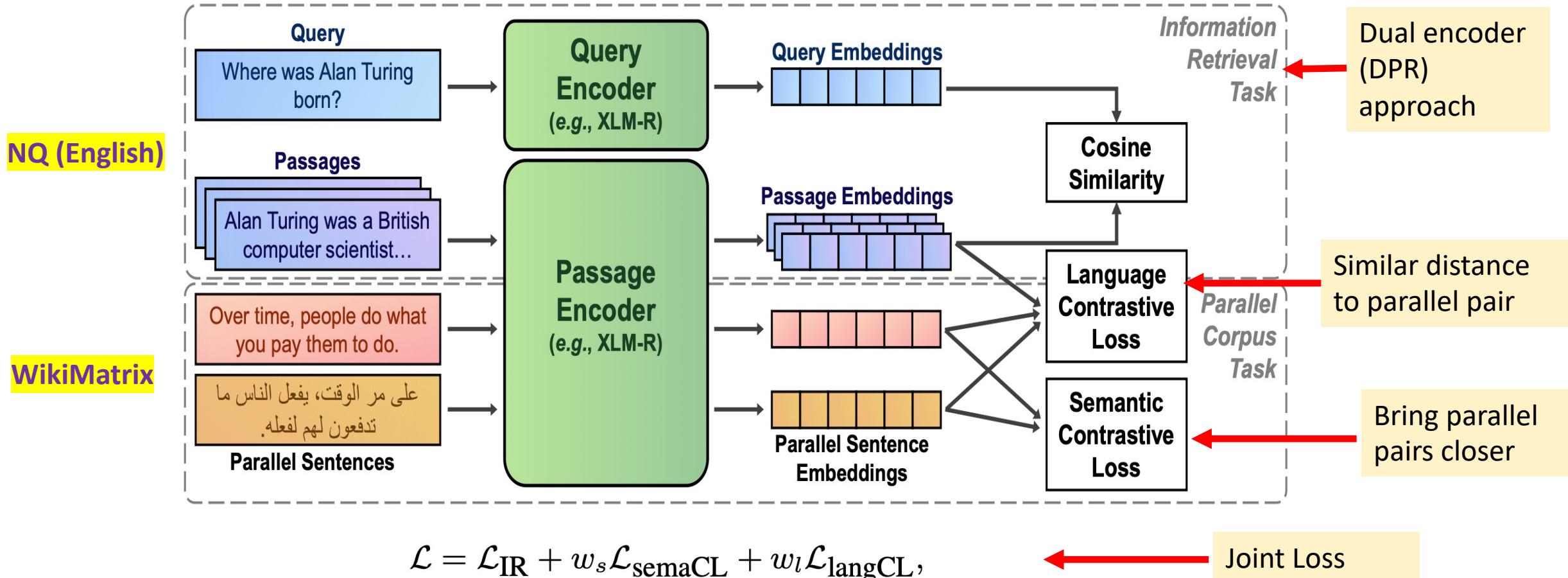
(a) Semantic Contrastive Loss



(b) Language Contrastive Loss

Zero-shot eval
on MrTyDI

Multilingual IR with Contrastive Loss Training



DrDECR (Li et al., 2021)

Dense Retrieval with Distillation Enhanced Cross-lingual Representation

XOR-TyDi v1.1 Leaderboard

Task 1: XOR-Retrieve

XOR-Retrieve is a cross-lingual retrieval task where a question is written in a target language (e.g., Japanese) and a system is required to retrieve English paragraphs that answer the question. The scores are macro-average over the 7 target languages.

Although we see the effectiveness of blackbox systems (e.g., Google Translate), we encourage the community to use white-box systems so that all experimental details can be understood. The systems using external blackbox APIs are highlighted in gray and ranked in the table of "Systems using external APIs" for reference.

Metrics: R@5kt, R@2kt (the recall by computing the fraction of the questions for which the minimal answer is contained in the top 5,000 / 2,000 tokens selected.)

Rank	Model	R@5kt	R@2kt
1 October 28, 2022	PrimeQA (DrDecr-large with PLAID + Colbert V2) <i>IBM Research AI</i>	74.7	69.2

Empirical study of the effectiveness of ColBERT on CLIR

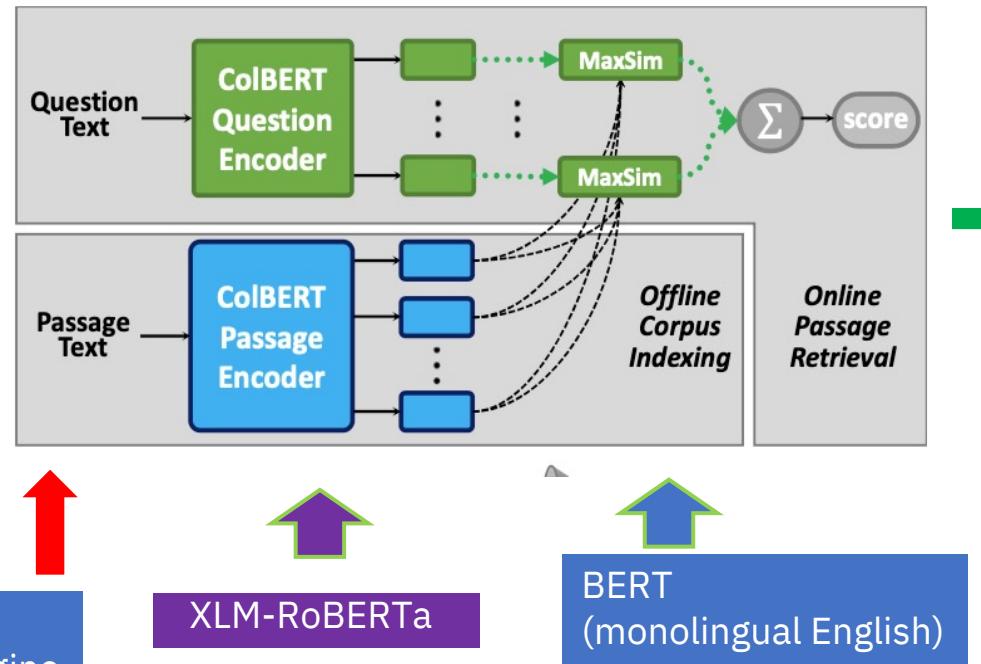
1993년 프랑스 총리는 누구 인가요? (Who was the French Prime Minister in 1993?)

Какая средняя зарплата в Краснодаре на сегодняшний день? (What is the average wage in Krasnodar?)

速水堅曹はどこで製糸技術を学んだ? (Where did Kenso Hayami learn the silk-reeling technique?)

IBM / Google Translation Engine

Monolingual ColBERT which needs translated data
r@5kt: 75.0



Multilingual ColBERT (does NOT need translated data)
r@5kt: 54.7

Mayor of Neuilly-sur-Seine from 1983 to 2002, he was Minister of the Budget under Prime Minister Édouard Balladur (1993–1995).

Krasnodar has the lowest unemployment rate among the cities of the Southern Federal District at 0.3% of the total working-age population. In addition, Krasnodar holds the first place in terms of highest average salary—21,742 rubles per capita.

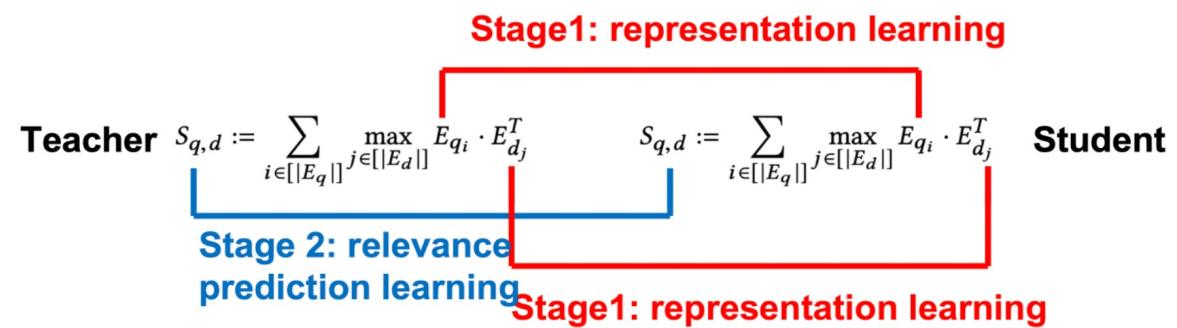
founded Hanei Maebashi Silk Mill and learned instrumental silk reeling techniques directly from Caspal Müller)

How can we improve the multilingual model?

DrDECR Approach: Knowledge Distillation

Two-stage knowledge distillation from model trained with English data (teacher) to Cross-lingual model (student)

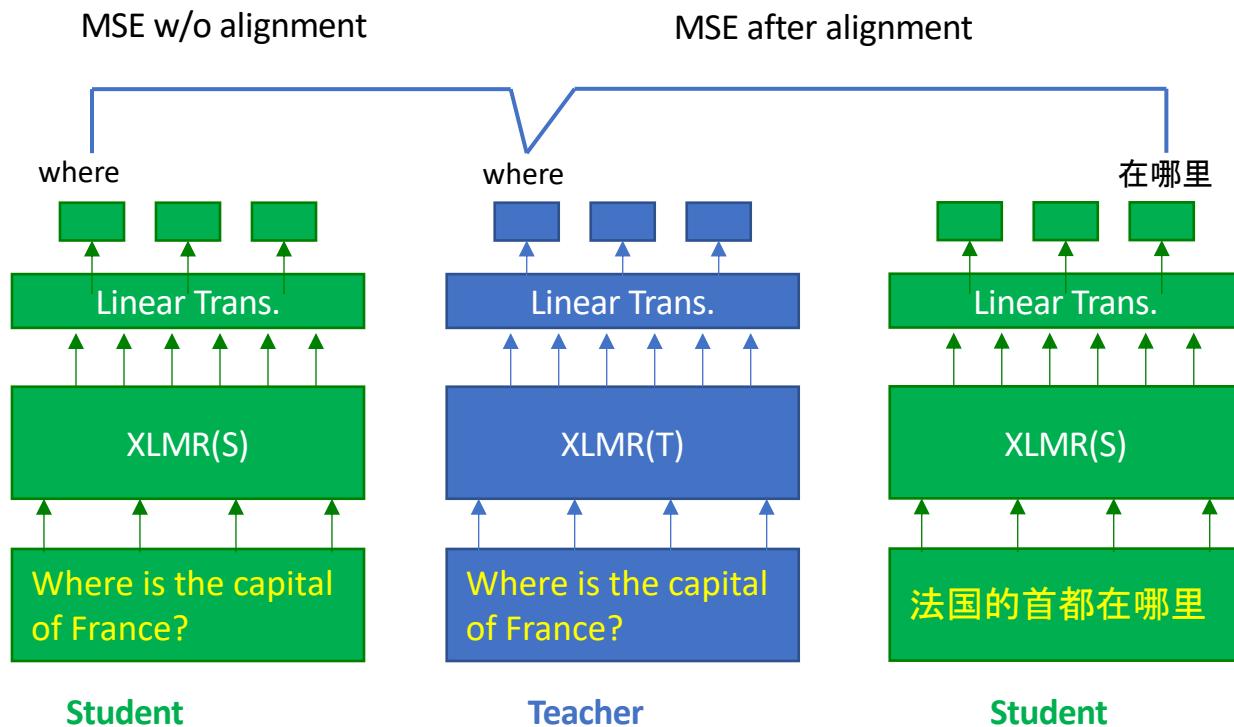
alignment between contextualized vectors



minimize student's KL divergence from teacher's softmax distribution

Representation Learning with Token Alignment

- Have the student learn from teacher's vector representation
- Utilize parallel corpus as data
- Align teacher's output token with student, based on their cosine distance



	<S>	法国	的	首都	在哪里
where	0.017	0.022	0.023	0.025	0.014
is	0.014	0.023	0.017	0.026	0.015
the	0.019	0.031	0.011	0.039	0.028
capital	0.031	0.018	0.030	0.015	0.019
of	0.025	0.034	0.014	0.043	0.032
France	0.017	0.009	0.026	0.021	0.016

During distillation, student sees both the Eng and Non-Eng version of the content

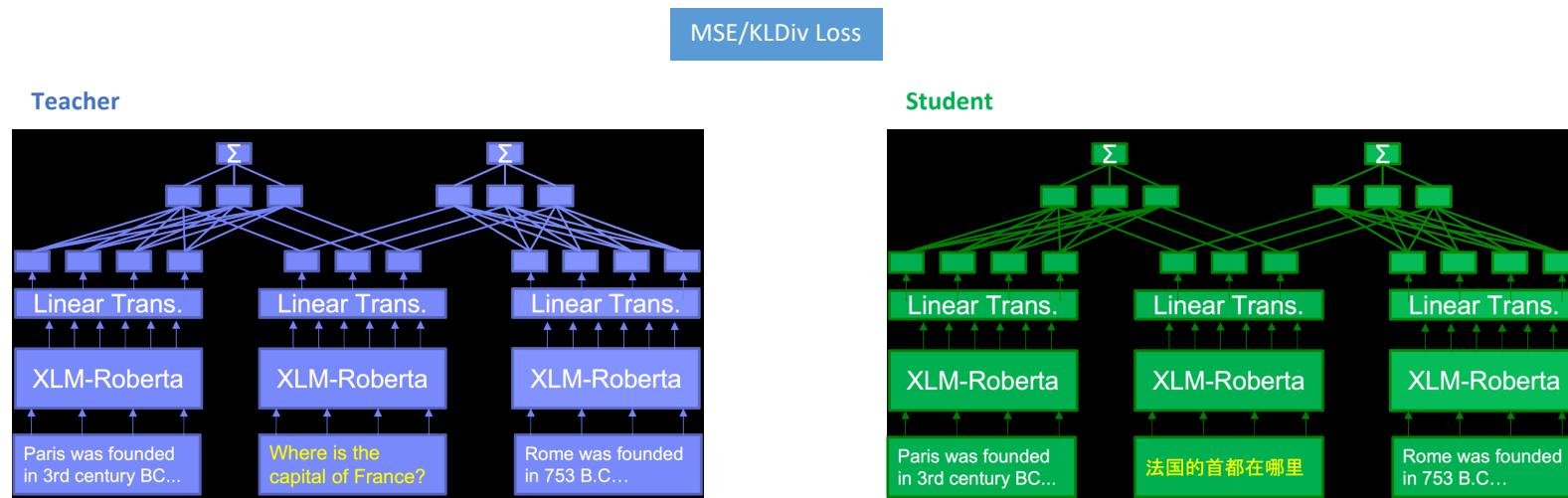
Relevance prediction learning

Teacher: (Eng q, Eng d+, Eng d-)

Student: (**Non-Eng** q, Eng d+, Eng d-)

minimize student's KL divergence from teacher's softmax distribution

- **4.4 points** improvement from parallel corpus
- **11.2 points** improvement from XOR-TyDI data



<https://github.com/primeqa/primeqa/tree/main/extensions/drdecr>

Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- Reproducibility in OpenQA: Hands-On Guide I
- Q&A: [15 min]

1st Half

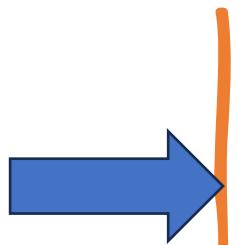


Coffee Break.....

- Multilingual Readers
- Multi-modal Readers: Text, Table, Visual QA
- Large Language Models as Retrievers/Readers
- Reproducibility in OpenQA: Hands-on Guide II
- Pipelines, Service and Deployment
- Q&A: [15 min]

2nd Half

Reproducibility in OpenQA: Hands On Guide



Why reproducibility is important

Hands On Guide with PrimeQA

- PrimeQA Repo
- PrimeQA Design

Working with Retrievers

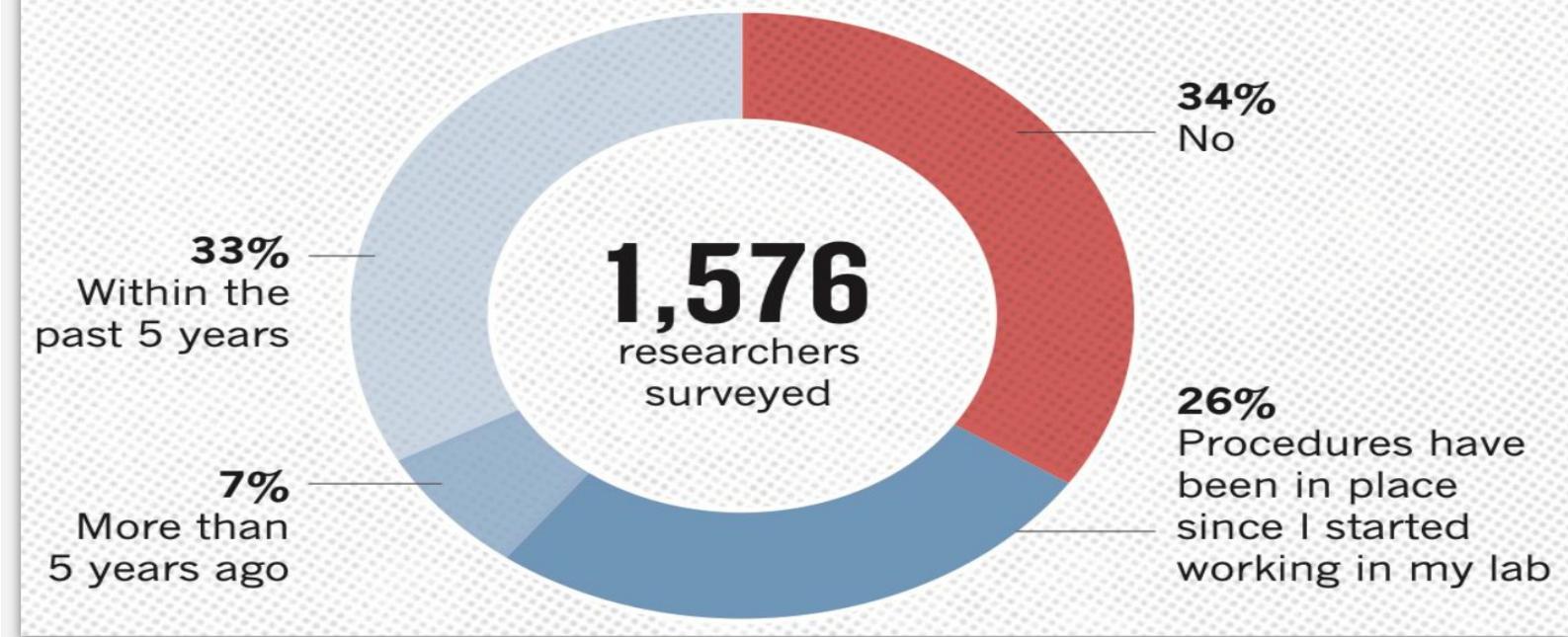
- Python scripts for reproducibility and customization
- Built-in classes in PrimeQA Retriever

Reproducibility

“

HAVE YOU ESTABLISHED PROCEDURES FOR REPRODUCIBILITY?

Among the most popular strategies was having different lab members redo experiments.

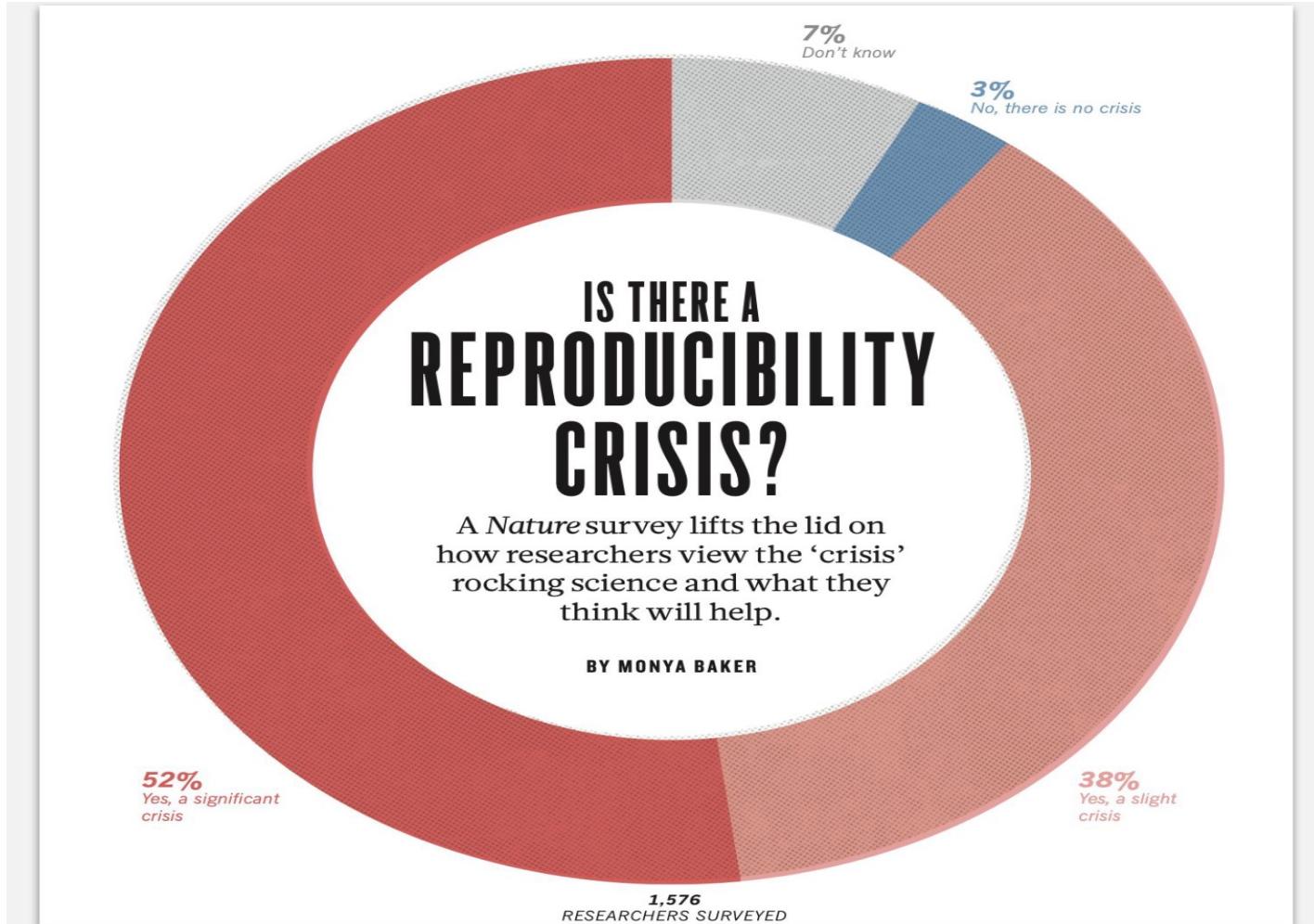


Baker 2016, Is there a reproducibility crisis, Nature

”

Reproducibility

“



Baker 2016, Is there a reproducibility crisis, Nature

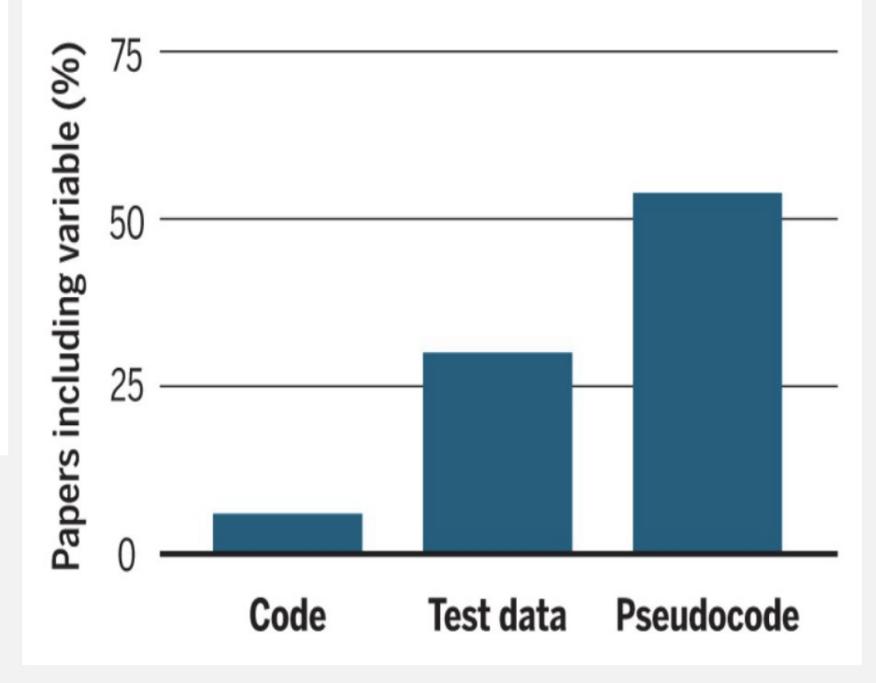
”

Reproducibility

“

Code break

In a survey of 400 artificial intelligence papers presented at major conferences, just 6% included code for the papers' algorithms. Some 30% included test data, whereas 54% included pseudocode, a limited summary of an algorithm.



”

Baker 2016, Is there a reproducibility crisis, Nature

Welcome Cecilia ! A new graduate student

- Cecilia has taken ML 101, NLP 101 and knows basic QA details.
- She has read about: the academic benchmarks for performing QA:
 - Multilingual Machine Reading Comprehension: TyDI [Clark2019]
 - Cross-lingual Open Retrieval: XOR-TyDI [Asai2020]
 - Table QA: WikiSQL [Zhong2017]



Cecilia does some literature survey!

- Cecilia wants to get the latest greatest SOTA models to start with!

She sees the following leaderboards ->

She reads the following papers: SOTA on the tasks

Tapex --[Liu2022_ICLR] – SOTA on WikiSQL

Dr. Decr -- [Li2022_NAACL, Bornea2020_AAAI]–
SOTA on XOR TyDI



TyDi QA

Rank	Model	Participant	Affiliation	Attempt Date	F1	Precision	Recall
1	GAAMA-Syn-Bool-Single-Model	GAAMA	IBM Research AI	9/7/2021	72.35	73.07	71.88
2	GAAMA-DM-Syn-ARES						
XOR-TyDi v1.1						Rank	Model
February 11, 2022						1	DrDecr
March 14, 2022						2	Senti 2.0 base
January 7, 2022						3	Huawei Noah's Ark lab
August 26, 2021						4	Contrastive Context-aware Pretraining Model (CCP)
October 7, 2021						5	Anonymous
							61.0
							60.7
Model	Dev	Test					
Wang et al. (2019)	79.4%	79.3%					
Min et al. (2019)	84.4%	83.9%					
TAPAS _{large}	88.0%	86.4%					

Cecilia looks for the **source code** to replicate these models

Papers with Code?

Question Answering Models

Natural Language Processing • 3 methods

Edit

Methods

Add a Method

Method

Year

Papers

Macaw

General-Purpose Qu

2021

2

TransferQA

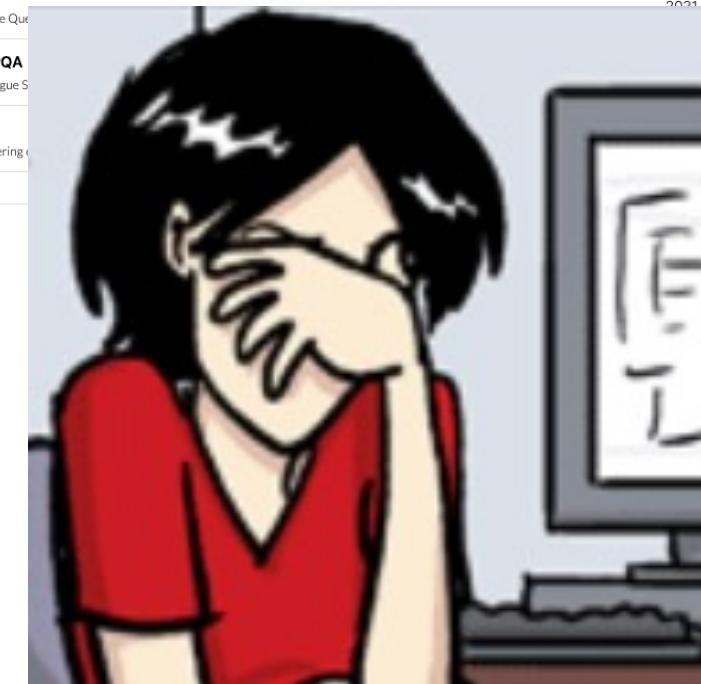
Zero-Shot Dialogue S

1

EMQAP

Question Answering

1



What about HuggingFace?



Need of the hour: Democratize & Replicate QA research

- ONE toolkit that encompasses all Open QA needs and hosts state-of-the-art models.
- Should be aimed to make both reproducibility and reusability easy.
- Can be used as a base to build more complex application.

PrimeQA: Democratize & Replicate QA research

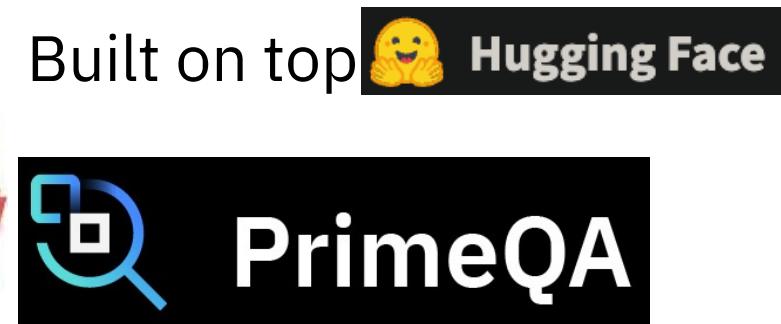
- End-user can replicate advanced research papers and leaderboard submissions quickly
- End-user can modify them as per their own needs
- End-user can use them as Lego blocks for QA problems



Lots of stand-alone Github repos



A final end2end QA solution



An alliance of QA researchers



SOTA contributions from:
**Stanford, CMU, OSU,
UIUC**.... to name a few.

Reproducibility in OpenQA: Hands On Guide



Why reproducibility is important

PrimeQA as a toolkit

- PrimeQA Repo
- PrimeQA Design

Working with Retrievers in PrimeQA

- Scripts for reproducibility and customization
- Built-in classes for PrimeQA Retrievers

PrimeQA : Toolkit for Hands On OpenQA

PrimeQA: Repository

Core Models	Extensions
Retriever	
BM25 (Robertson and Zaragoza, 2009)	Dr.DECR * (Li et al., 2022)
DPR (Karpukhin et al., 2020)	
ColBERT (Santhanam et al., 2022b)	
Reader	
General MRC* (Alberti et al., 2019b)	ReasonBERT (2021)
FiD (Izacard and Grave, 2020)	OmniTab (Jiang et al., 2022a)
Boolean* (McCarley et al., 2023)	MITQA* (Kumar et al., 2021)
Lists	
Tapas (Herzig et al., 2020a)	
Tapex (Liu et al., 2021)	
Question Generation	
Table QG (Chemmengath et al., 2021)	
Passage QG	
Table+Passage QG	

- Covers both retrievers and readers.
- Includes leaderboard winning state-of-the-art systems.
- Extensions, which are built on core components.

PrimeQA : helps everyone !!

Researcher



Q: I need to reproduce results, finetune with my own data.
A: Run our python scripts

Developer



Q: I want to use it as SDK, integrate it in my application.
A: Check the notebooks on how to use built-in classes.

Q: I like it, but how do I set it up as a service?
A: Use our services layer to create serviceable endpoints

Analysts



Q: Can I use it from Huggingface?
A: Our models are accessible via Huggingface APIs

End users



Q: I'm non-technical, can I see it working ?
A: Get a UI up using PrimeQA UI layer.

PrimeQA: Design Principles



Top level Scripts

Entry Points



Notebooks



Inference API



Services



UI

Ideal For:

Researchers

Developers

Developers

Developers

End-Users

Open Retrieval QA Pipelines

Core Components

Retriever

Sparse

Dense

Reader

Extractive

Text
Boolean
Table
List

Question Generator

Generative

Seq2Seq

AI Libraries

HuggingFace
Transformers

Pyserini

HuggingFace
Datasets

FAISS

Stanza

Spacy

Reproducibility in OpenQA: Hands On Guide



Why reproducibility is important

PrimeQA as a toolkit

- PrimeQA Repo
- PrimeQA Design

Working with Retrievers in
PrimeQA

- Scripts for reproducibility and customization
- Built-in classes for PrimeQA Retrievers

Hands On Guide with PrimeQA

Retriever

- python scripts

Retriever- Steps

Step-1 Train your retriever model with <q,p+,p-> triples

Step-2 Index a document corpus using trained model

Step-3 Search from your indexed corpus

Note:

PrimeQA supports DPR, ColBERT, BM25.

We will use DPR engine as an example retriever to show the steps.

One can find equivalent scripts for other retriever engines in :

https://github.com/primeqa/primeqa/blob/prompt_reader/primeqa/ir/README.md

Retriever- Training

Step-1 Train your retriever model with <q,p+,p-> triples

Save your training data in a .tsv

query	positive_passage	negative_passage
중국에서 가장 오랜기간 왕위를 유지한 인물은 누구인가?	"Kangxi Emperor The Kangxi Emperors reign of 61 years ...	Chiddy Bang new songs from the duo and in November 2009 debuted...
중국에서 가장 오랜기간 왕위를 유지한 인물은 누구인가?	Kangxi Emperor The Kangxi Emperors reign of 61 years ...	Emperor Zhi Yao. The Bamboo Annals says that when Emperor Zhuanxu died ...

(English translation of the original Korean query is "Who maintained the throne for the longest time in China?")

Retriever- Training

Step-1 Train your retriever model with <q,p+,p-> triples

DPR

```
python primeqa/ir/run_ir.py \
--do_train \
--engine_type DPR \
--train_dir <training_file_or_directory> \
--output_dir <output_directory> \
--epochs <number_of_training_epochs> \
--bsize <training_batch_size> \
--training_data_type text_triples
```

Other engine types are “colBERT”

Single .tsv or multiple

Output models in: <output_directory>/qry_encoder, ctx_encoder

Scripts look similar, only differ in “engine type” and hyper params each engine allows

Details at: <https://github.com/primeqa/primeqa/tree/main/primeqa/ir/ReadMe.md>

Retriever- Indexing

Create your document collection as a .tsv

id	text	title
1	"The Kangxi Emperor's reign of 61 years ...	Kangxi Emperor
2	Yao. The Bamboo Annals says that when Emperor Zuanxu died ...	Emperor Zhi

- .tsv file should contain collection records in the form of [ID, text, title] triples.
- The first line of the file contains a header record

Retriever- Indexing

Step-2 Use your retriever model to index a document collection

DPR

```
python primeqa/ir/run_ir.py \
    --engine_type DPR \ ← Other engine types are “colBERT”, “BM25”
    --do_index \
    --ctx_encoder_name_or_path <context_encoder_model> \
    --embed <part_number>of<parts_total> \
    --sharded_index \
    --collection <document_collection> \
    --output_dir <output_directory> \
    --bsize <indexing_batch_size> \
```

The index is stored in : output_directory

Scripts look similar, only differ in “engine type” and hyper params each engine allows

Retriever- Searching

Step-2 Use your retriever model to index a document collection

DPR

```
python primeqa/ir/run_ir.py \
  --do_search \
  --engine_type DPR \           ← Other engine types are “colBERT”, “BM25”
  --queries <query_file> \
  --model_name_or_path <query_encoder_model> \
  --bsize <search_batch_size> \
  --index_location <directory_containing_index_files> \   ← Output location used during indexing operation
  --top_k <number_of_items_per_query_retrieved> \          ← How many docs you want to retrieve
  --output_dir <output_directory>
```

Scripts look similar, only differ in “engine type” and hyper params each engine allows

Reproducing Dr Decr

Dr. Decr is an extension over PrimeQA core functionalities and leaderboard winner for Xor-Tydi

Step0: Data preparation

→ PrimeQA provides readymade .sh scripts.

Step 1: Fine tuning a ColBERT model for both student and teacher training

→ Supported by run_ir script with “do_train” and right parameters.

Step 2: Two-stage knowledge distillation to train the final (student) model

→ “distillation” is also supported by run_ir script with “do_train” and specific flags for KD like “teacher_triples”.

Step 3: Indexing the corpus using the trained student model

→ Supported by run_ir with “do_index”.

Step 4: Retrieving the relevant passages using the index

→ Supported by run_ir with “do_search”.

Step 5: Relevance scoring

→ PrimeQA provides readymade .sh scripts.

Reproducibility Test:



Dr Decr can be reproduced using PrimeQA core functionalities and readymade scripts.

More Details here:

<https://github.com/primeqa/primeqa/blob/main/extensions/drdecr/README.md>

Hands On Guide with PrimeQA

Retriever

built-in classes

Retriever- Steps

Step-1 Use the built-in class "SearchableCorpus"

Create with right parameters

```
from primeqa.components import SearchableCorpus
collection = SearchableCorpus(context_encoder_name_or_path="PrimeQA/XOR-TyDi_monolingual_DPR_ctx_encoder",
                               query_encoder_name_or_path="PrimeQA/XOR-TyDi_monolingual_DPR_qry_encoder",
                               batch_size=64, top_k=10)
```

← context encoder
← query encoder

Step-2 Create a .tsv file to which assumes the format as : id \t text \t title_of_document

```
collection.add_documents('input.tsv')
```

PrimeQA notebook offers simple scripts to convert existing files to this format

Step-3 You are ready to search

```
1 queries= ["When did Einstein receive his nobel prize ? "]
2 retrieved_doc_ids, passages = collection.search(queries)
```

One can include multiple queries together in a batch.

Retriever – In Action

snapshot of the input .tsv file (to index and search against)

1	title	text
2	0 "Albert Einstein"	to Einstein in 1922. Footnotes Citations Albert Einstein Albert Einstein (; ; 14 March 1879 , 18 April 1955) was a German-born theoretical physicist who developed the theory of relativity, one of the two pillars of modern physics (alongside quantum mechanics), and is known for his mass–energy equivalence formula E = mc² (which has been dubbed the world's most famous equation). He received the 1921 Nobel Prize in Physics "for his services to Theoretical Physics, and especially for his discovery of the law of the photoelectric effect."
3	1 "Albert Einstein"	Albert Einstein Albert Einstein (; ; 14 March 1879 , 18 April 1955) was a German-born theoretical physicist who developed the theory of relativity, one of the two pillars of modern physics (alongside quantum mechanics), and is known for his mass–energy equivalence formula E = mc² (which has been dubbed the world's most famous equation). He received the 1921 Nobel Prize in Physics "for his services to Theoretical Physics, and especially for his discovery of the law of the photoelectric effect."
4	2 "Albert Einstein"	observations were published in the international media, making Einstein world-famous. On 7 November 1919, the leading British newspaper "The Times" printed a banner headline that read: "Revolution in Science - Albert Einstein's Theory Proved by British Astronomers". The theory of relativity has become part of everyday language, with the word "Einstein" used as a verb meaning "to change something fundamentally".
5	3 "Albert Einstein"	model for depictions of mad scientists and absent-minded professors; his expressive face and distinctive hairstyle have been widely copied and exaggerated. "Time" magazine's Frederic Golden wrote that Einstein's image had become "an international icon".
6	4 "Alfred Nobel"	was adopted as the standard technology for mining in the "Age of Engineering" bringing Nobel a great amount of financial success, though at a significant cost to his health. An offshoot of this research resulted in the Nobel Prize in Physics.
7	5 "Akira Kurosawa"	for 2020. Patrick Frater writing for "Variety" magazine in May 2017 stated that another two unfinished films by Kurosawa were planned, with "Silvering Spear" to start filming in 2018. In September 2011, it was reported that Kurosawa's final film, "Red Cliff", had been completed.
8	6 "Apple Inc."	a near bezel-less design along with wireless charging. On September 12, 2018, Apple introduced the iPhone XS, iPhone XS Max and iPhone XR. The iPhone XS and iPhone XS Max features Super Retina displays, improved cameras, and faster processors.
9	7 "Akira Kurosawa"	through the Second World War and beyond. The narrative centers on yearly birthday celebrations with his former students, during which the protagonist declares his unwillingness to die just yet, a theme that has been repeated in many of his later films.
10	8 "Apple Inc."	2016, Apple introduced the iPhone 7 and the iPhone 7 Plus, which feature improved system and graphics performance, add water resistance, a new rear dual-camera system on the 7 Plus model, and, controversially, a faster processor, and brighter display. On September 12, 2017, Apple introduced the Apple Watch Series 3 featuring LTE cellular connectivity, giving the wearable independence from an iPhone except for the ability to charge it.
11	9 "Apple Inc."	the iPhone X, which features a larger screen, a new Face ID feature, and a new A11 Bionic chip. The iPhone X also features a new camera system with optical image stabilization and a new TrueDepth camera system for facial recognition.

Note: The document has an ID which is numerical, a title and a text.

Some retrieved documents for query “when did Einstein receive his Nobel prize?”

"observations were published in the international media,
In 1922, he was awarded the 1921 Nobel Prize in Physics \"for
his services to Theoretical Physics, and

"Albert Einstein Albert Einstein (; ; 14 March 1879 \u2013 18 April 1955) was a German-born theoretical physicist
He received the 1921 Nobel Prize in Physics \"for his services to theoretical physics, and

PrimeQA also supports Reranker

Step-1 Use the built-in class “ColBERTReranker”

Create with right parameters

```
# Import the ColBERT Reranker
from primeqa.components.reranker.colbert_reranker import ColBERTReranker

# Instantiate the ColBERTReranker
reranker = ColBERTReranker(reranker_model_path)
reranker.load()
```

Step-2 Create a .tsv file to which assumes the format as : id \t text \t title_of_document

```
from primeqa.ir.util.corpus_reader import DocumentCollection

collection = DocumentCollection(downloaded_corpus_file)
hits_to_be_reranked = collection.add_document_text_to_hit(
```

The documents to rerank. This can be results from another retriever such as BM25.

Step-3 You can now rerank !!

```
reranked_results = reranker.rerank(question, [hits_to_be_reranked], max_num_documents=3)
```

pick top 3 docs after reranking

Outline

-  Motivation of OpenQA
-  Retrievers: Sparse and Dense
-  Recent Advances for OpenQA Efficient Neural Retrievers
-  Recent Advances in Efficient Multilingual Retrieval
-  Reproducibility in OpenQA: Hands-On Guide I
-  Q&A: [15 min]

1st Half



Coffee Break.....

-  Multilingual Readers
-  Multi-modal Readers: Text, Table, Visual QA
-  Large Language Models as Retrievers/Readers
-  Reproducibility in OpenQA: Hands-on Guide II
-  Pipelines, Service and Deployment
-  Q&A: [15 min]

2nd Half

Coffee Break.....



Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- ✓ Reproducibility in OpenQA: Hands-On Guide I
- ✓ Q&A: [15 min]

1st Half



Coffee Break.....

- Multilingual Readers
- Multi-modal Readers: Text, Table, Visual QA
- Large Language Models as Retrievers/Readers
- Reproducibility in OpenQA: Hands-on Guide II
- Pipelines, Service and Deployment
- Q&A: [15 min]

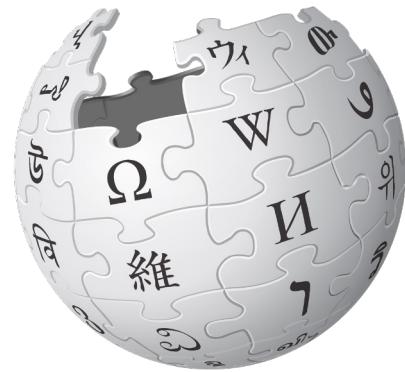
2nd Half

Multilingual Readers

Outline

- Reading Comprehension
 - Multilingual MRC
- Open-Retrieval QA
 - Fusion in Decoder
- Cross-lingual Open Retrieval QA

QA: Reading Comprehension vs Open-Retrieval QA



Document
Retriever

Open-Retrieval QA
(ORQA)

Note: ORQA is aka Open Domain QA [Lee et al., 2019] and/or End-2-end QA [Reddy et al., 2021].

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

Warsaw

From Wikipedia, the free encyclopedia

"Warszawa", "Warschau", and "City of Warsaw" redirect here. For other uses, see [Warsaw \(disambiguation\)](#), [Warszawa \(disambiguation\)](#), [Warschau \(disambiguation\)](#), and [City of Warsaw \(disambiguation\)](#).

Warsaw^[4] officially the [Capital City of Warsaw](#)^{[4][5]} is the capital and largest city of Poland. The metropolis stands on the River Vistula in east-central Poland and its population is officially estimated at 1.8 million residents within a greater metropolitan area of 3.1 million residents^[5] which makes Warsaw the 7th most populous capital city in the European Union. The city area measures 517 km² (200 sq mi) and comprises 18 boroughs, while the metropolitan area covers 6,100 km² (2,355 sq mi).^[6] Warsaw is an alpha-global city^[7] a major cultural, political and economic hub, and the country's seat of government. Its historical Old Town was designated a UNESCO World Heritage Site.

Warsaw traces its origins to a small fishing town in Masovia. The city rose to prominence in the late 16th century, when Sigismund III decided to move the Polish capital and his royal court from Kraków. Warsaw served as the de facto capital of the Polish–Lithuanian Commonwealth until 1795, and subsequently as the seat of Napoleon's Duchy of Warsaw. The 19th century and its Industrial Revolution brought a demographic boom which made it one of the largest and most densely populated cities in Europe. Known then for its elegant architecture and boulevards, Warsaw was bombed and besieged at the start of World War II in 1939.^{[8][9][10]} Much of the historic city was destroyed and its diverse population decimated by the Ghetto Uprising in 1943, the general Warsaw Uprising in 1944 and systematic razings.

Warsaw is served by two international airports, the busiest being Warsaw Chopin and the smaller Warsaw Modlin intended for low-cost carriers. Major public transport services operating in the city include the Warsaw Metro, buses, urban-light railway and an extensive tram network. In 2012, the Economist Intelligence Unit ranked Warsaw as the 32nd most liveable city in the world.^[11] In 2017, the city came 4th in the "Business-friendly", 8th in "Human capital and life style" and topped the quality of life rankings in the region.^[12] The city is a significant centre of research and development, business process outsourcing, and information technology outsourcing. The Warsaw Stock Exchange is the largest and most important in Central and Eastern Europe.^{[13][14]} Frontex, the European Union agency for external border security as well as ODIHR, one of the principal institutions of the Organization for Security and Cooperation in Europe have their headquarters in Warsaw. Jointly with Frankfurt and Paris, Warsaw features one of the highest number of skyscrapers in the European Union.^[15]

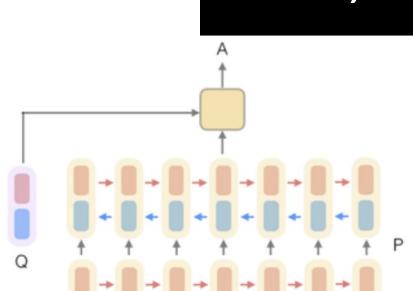
The city hosts the Polish Academy of Sciences, National Philharmonic Orchestra, University of Warsaw, the Warsaw University of Technology, the National Museum, Zachęta Art Gallery and the Warsaw Grand Theatre, the largest of its kind in the world.^[16] The reconstructed Old Town, which represents examples of nearly every European architectural style and historical period,^[17] was listed as a World Heritage Site by UNESCO in 1980. Other main architectural attractions include the Royal Castle and the iconic King Sigismund's Column, the Wilanów Palace, the Palace on the Isle, St. John's Cathedral, Main Market Square, as well as numerous churches and mansions along the Royal Route. Warsaw possesses thriving arts and club scenes, gourmet restaurants and large urban green spaces, with around a quarter of the city's area occupied by parks.^{[18][19]}

Reading comprehension



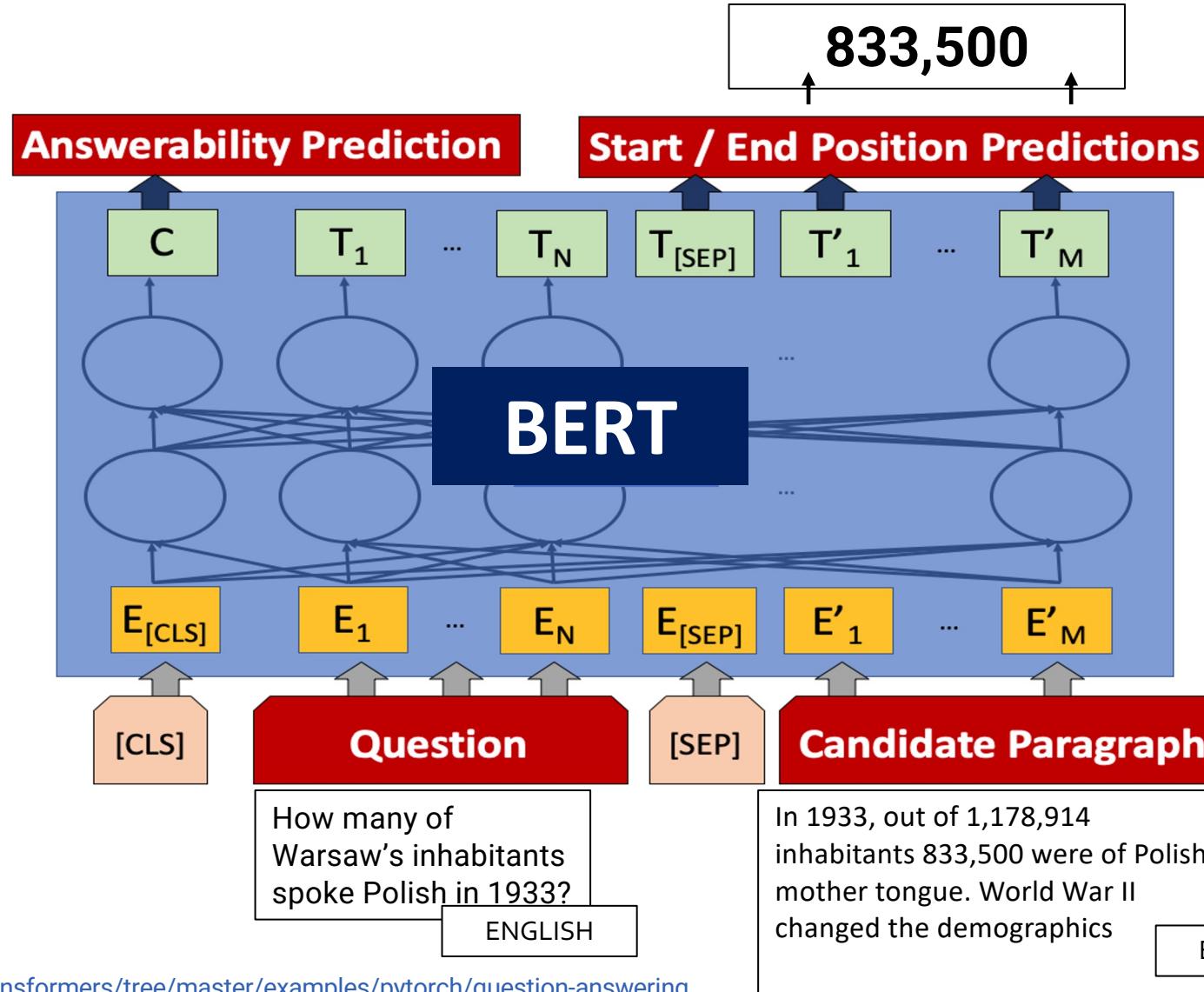
Document
Reader

833,500



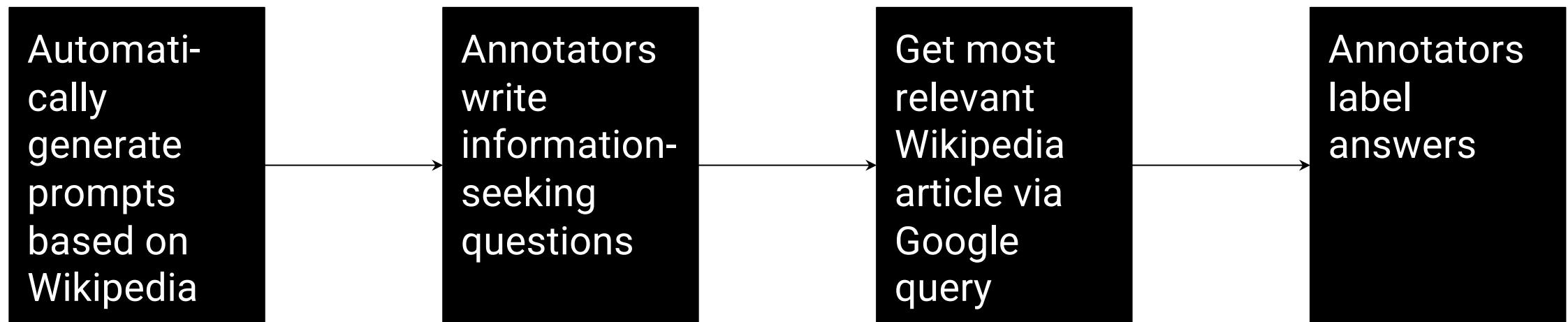
Machine Reading Comprehension (MRC)

- Popular choice: Add a fine-tuning layer on top of BERT [Devlin et al., 2019]



TyDi QA : Multilingual MRC

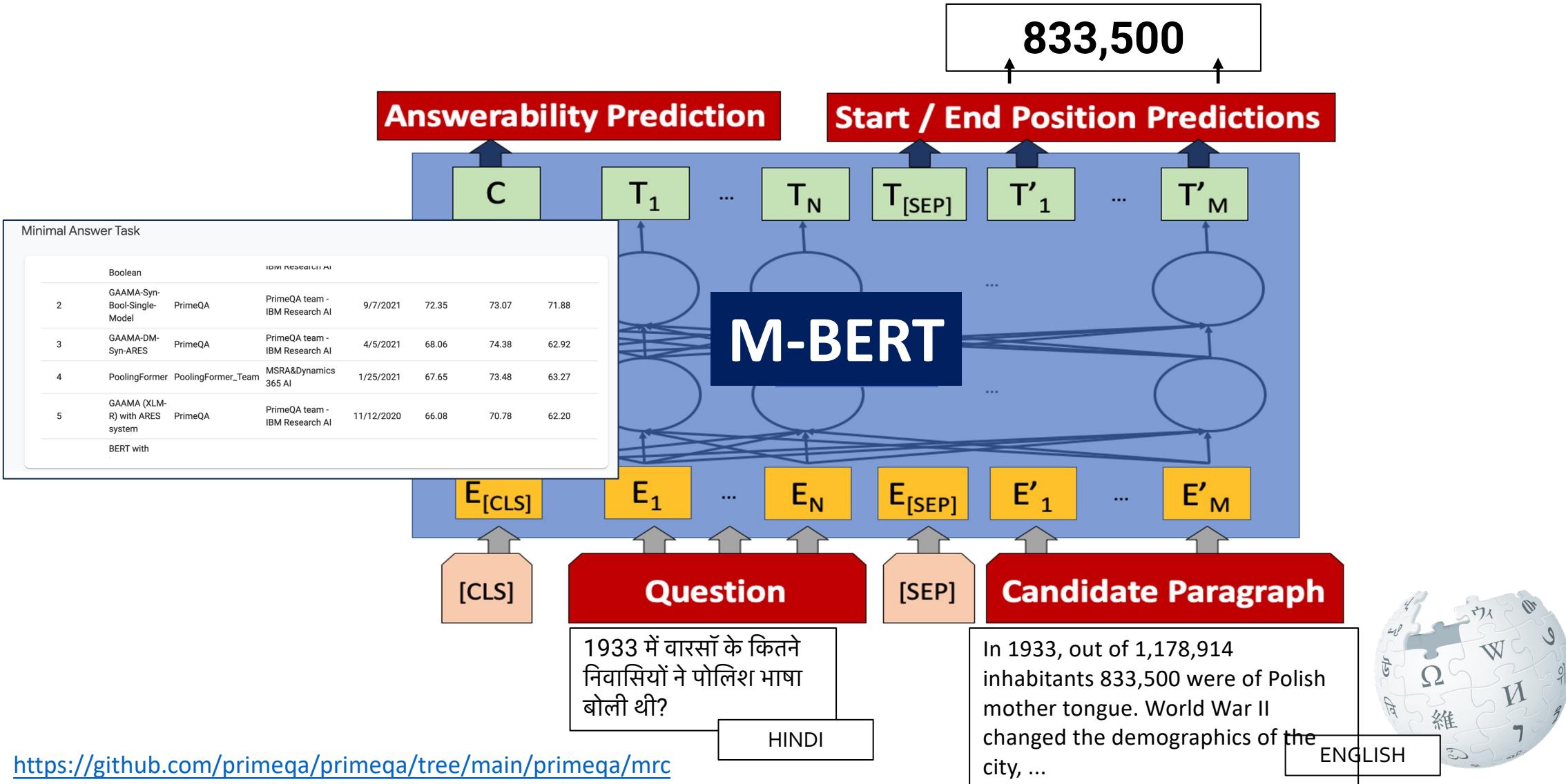
“information-seeking” questions based on short Wikipedia prompts in typologically diverse (“TyDi”) languages



- **Passage selection:** Given the question and Wikipedia page, predict the passage that contains the answer, or NULL
- **Minimal span (full version):** Given the question and Wikipedia page, identify the correct span, or NULL, YES, or NO

Multilingual Machine Reading Comprehension

Popular choice: Add a fine-tuning layer on top of M-BERT [Bornea et al., 2021]



Various Setups of ORQA

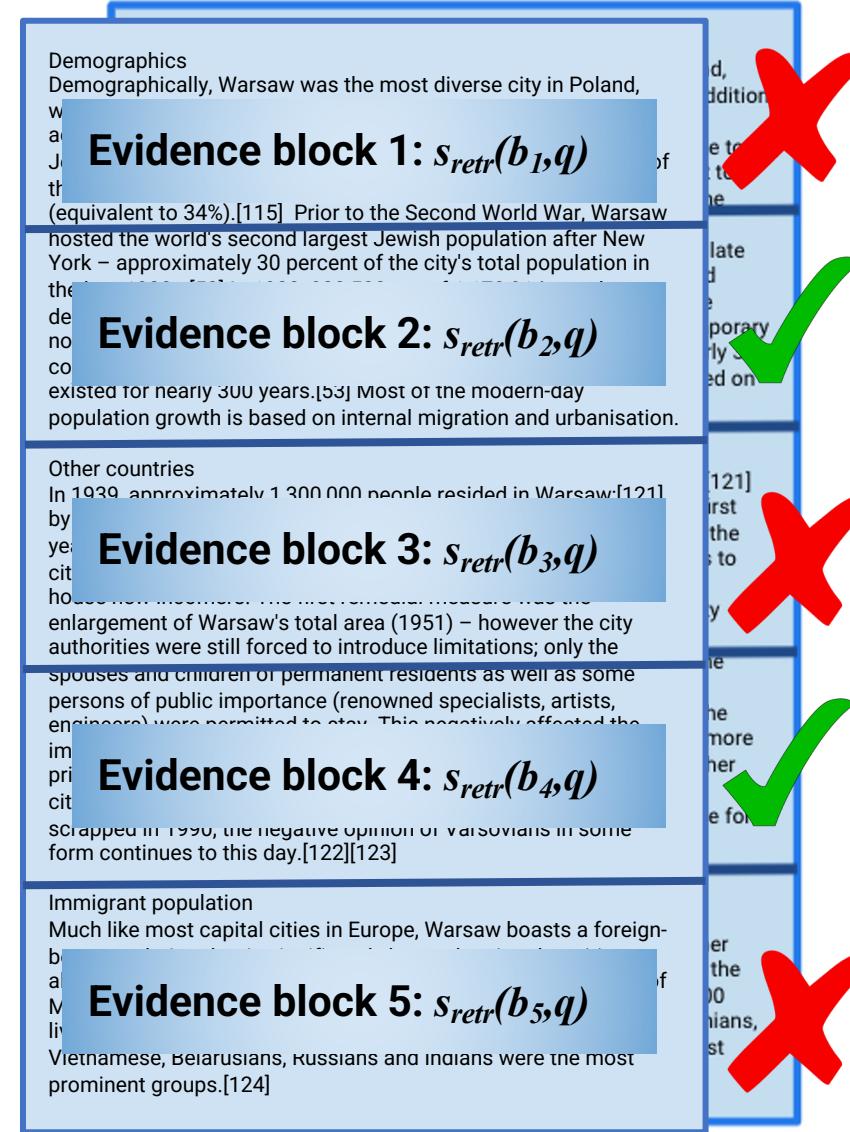
Select top-k blocks from collection (e.g. Wikipedia)

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

Retriever



- BM25
- DPR
- ColBERT



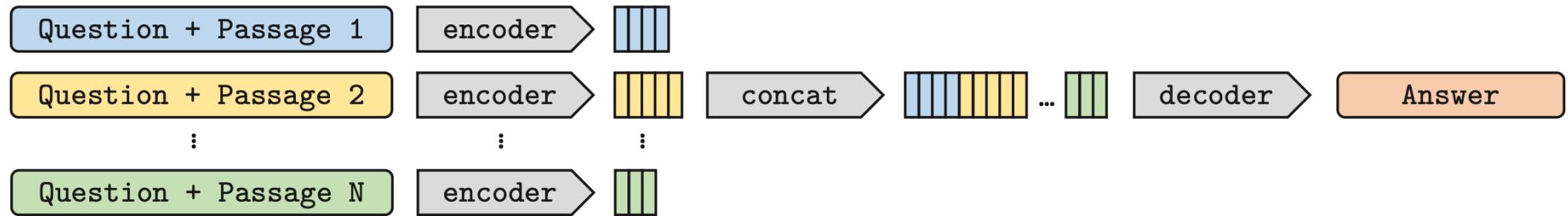
Reader

833,500

Reader

The number of inhabitants that spoke Polish in 1933 Warsaw is 833,500.

Fusion-In-Decoder



- Use a **Generative encoder-decoder** model (T5 or BART)
- Encoder processes each **Q and P pair independently** **Allows scaling to a large number of passages**
- Decoder produces the answer, **conditioned on the concatenation of the representation of all question and evidence passages**

Eval on Open NQ and TriviaQA shows generative models are **effective at aggregating evidence from multiple passages.**

Cross-lingual QA

Synthetic Question-Answer Generation

- ✓ Shakeri et al., "Towards zero-shot multilingual synthetic question and answer generation for cross-lingual reading comprehension" INLG 2021

Data Expansion and Iterative Training

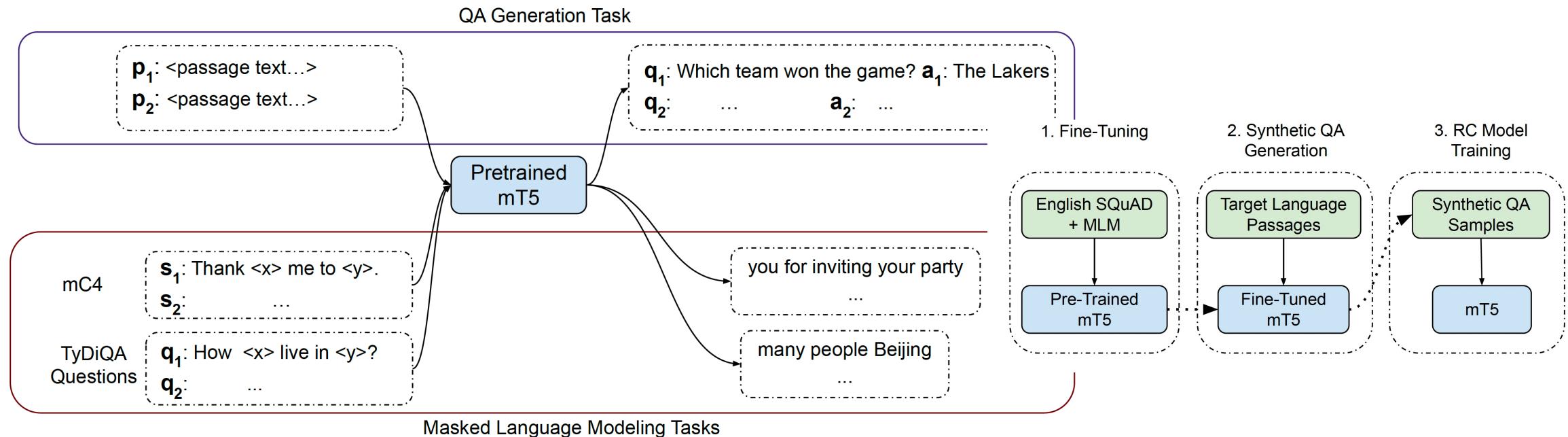
- ✓ Asai et al., "One Question Answering Model for Many Languages with Cross-lingual Dense Passage Retrieval", NeurIPS 2021
- ✓ Sorokin et al., "Ask Me Anything in Your Native Language", NAACL 2022

Synthetic QA Generation for Cross-lingual MRC

Joint training on the question-answer generation task, and the multilingual (MLM) task

MLM Task is crucial to enable zero-shot capabilities AND avoid catastrophic forgetting of multilingual generation capability

Produce non-English QA samples on non-English input when only trained on English samples



XOR-TyDI QA

Task 3: XOR-Full

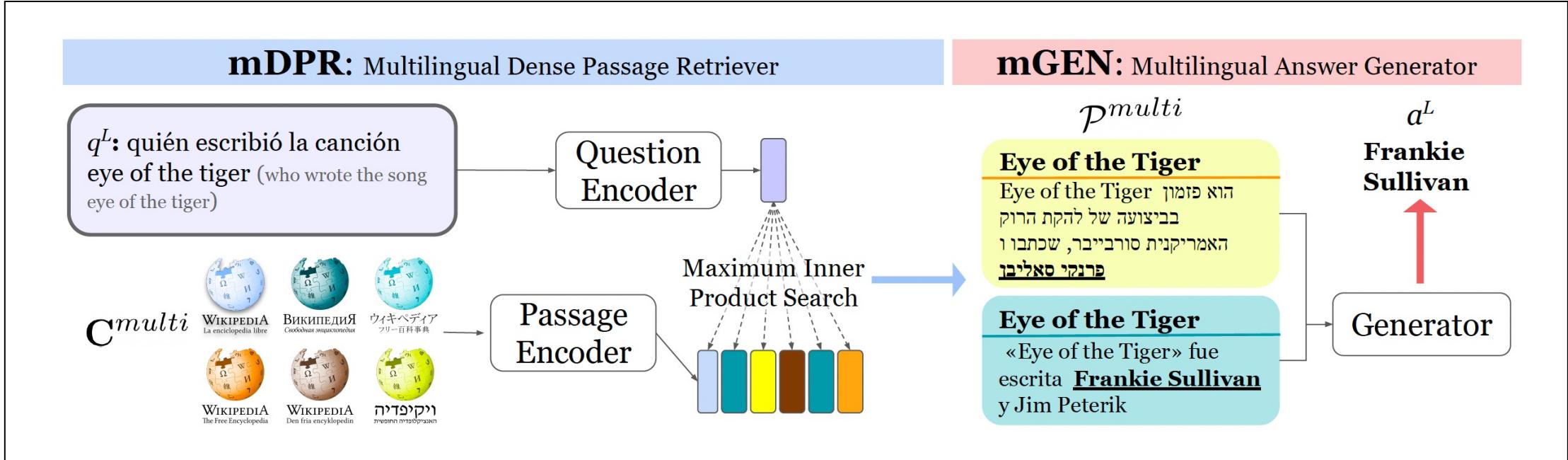
XOR-Full is a cross-lingual retrieval task where a question is written in the target language (e.g., Japanese) and a system is required to output a short answer in a target language. The scores are macro-average over the 7 target languages.

highlighted in gray and ranked in the table of "**Systems using external APIs**" for reference.

Metrics: F1, EM, BLEU over the annotated answer's token set.

Rank	Model	F1	EM	BLEU
1	Sentri + MFID base <i>Huawei Noah's Ark lab</i>	46.2	39.0	33.7
2	CORA <i>University of Washington, AI2</i>	43.5	33.5	31.1

CORA: Cross-lingual Open-Retrieval Answer Generation



Retrieve

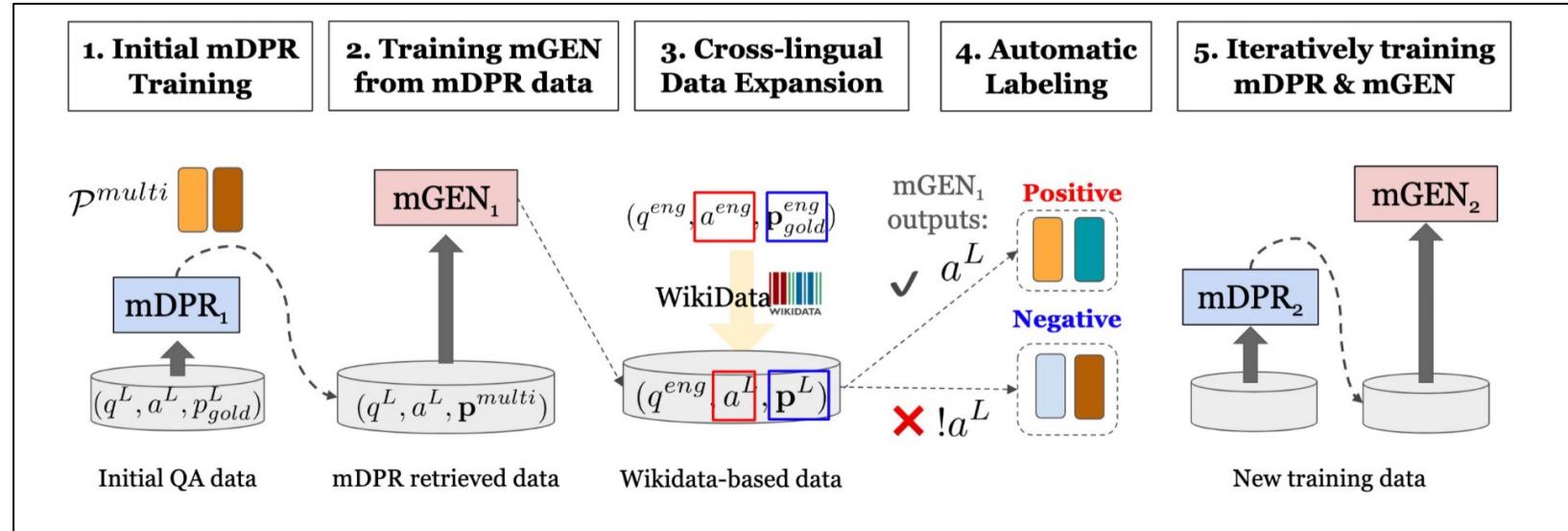
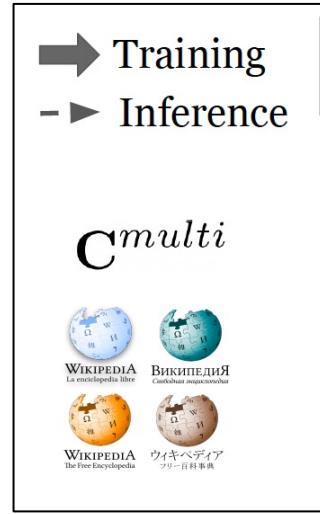
Fine-tune a multilingual LM (mBERT)

Generate

Fine-tune multilingual seq2seq model mT5

Do not rely on language-specific retrievers or machine translation modules.

CORA: Iterative training



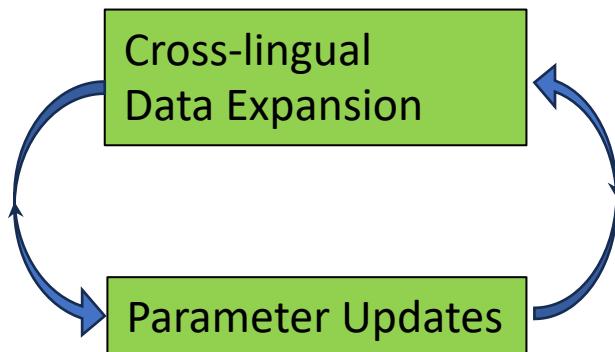
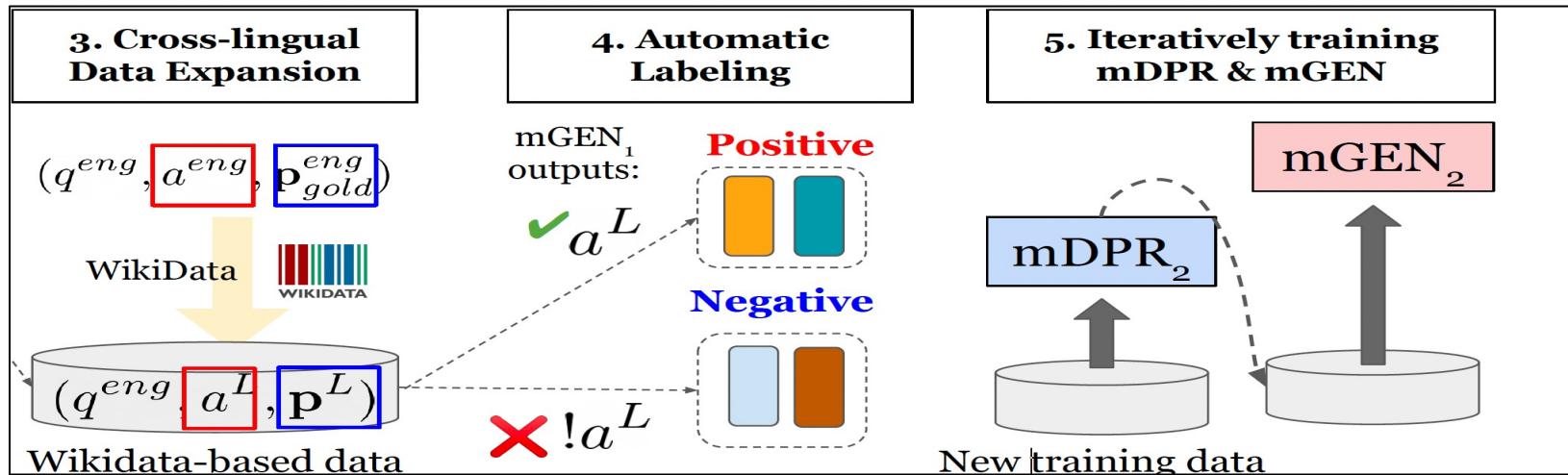
Initial Training

Train mDPR
NQ, XORQA
 $q^L; a^L; p^{gold}$

Use mDPR to
Retrieve topK
docs

Train mGEN to generate answer a^L
conditioned on topK docs
concatenation of q^L and P^{multi} .

CORA: Training data expansion



translate a^L to other languages using Wikipedia language links and find corresponding non-English Wikipedia articles

Train mDPR with expanded data

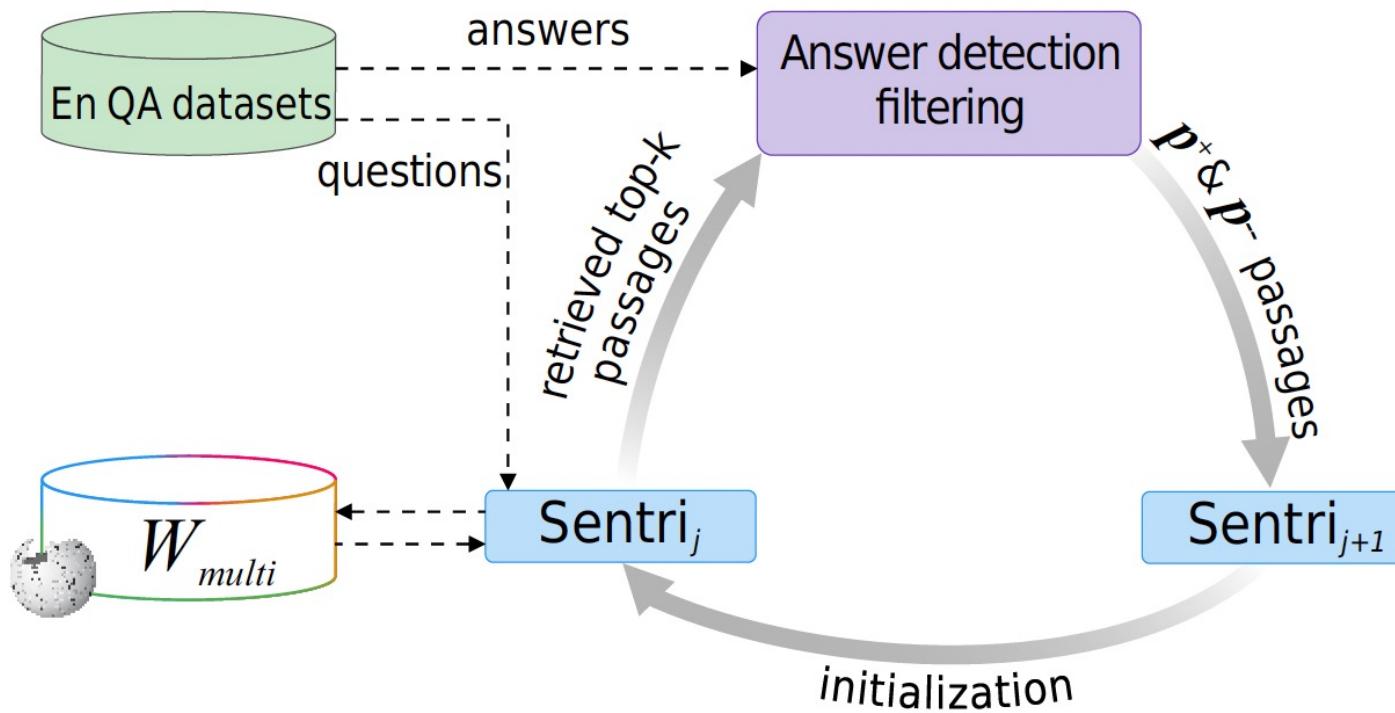
Use mDPR to retrieve topK docs

Use mGEN to label *pos* and *neg* passages using a^L

Train mGEN with expanded data

Ask Me Anything in Your Native Language

Iterative Training Framework



Reader is a FiD model

Iteratively train only the retriever

Initial retriever is BM25

Single Encoder Retriever for both Q and P_s

Ask Me Anything in Your Native Language

Differentiating features in Data Expansion

1. Usage of machine translation for data expansion
 - Use NQ and TriviaQA and translate to XOR-TyDI languages
2. Use the Retriever to mine additional positive and hard negatives
 - a sample is positive if it is ranked high **and** includes the answer
 - Morphology-aware answer detection technique
3. In-batch Negatives and False Negative Filtering
 - Multiple questions in a batch could share the same positive passage

Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- ✓ Reproducibility in OpenQA: Hands-On Guide I
- ✓ Q&A: [15 min]

1st Half



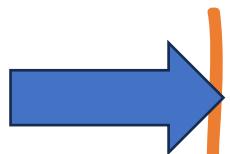
Coffee Break.....

- ✓ Multilingual Readers
- Multi-modal Readers: Text, Table, Visual QA
- Large Language Models as Retrievers/Readers
- Reproducibility in OpenQA: Hands-on Guide II
- Pipelines, Service and Deployment
- Q&A: [15 min]

2nd Half

Multimodal readers

Multimodal Readers



Benchmarks and systems in Table+Text

- MIT-QA
- CARP

Numerical Reasoning

- Benchmarks
- TagOP
- FinMATH
- MT2Net

Extending to images

- MuMuQA

Multi-modal Question Answering: Benchmarks

Table + Text

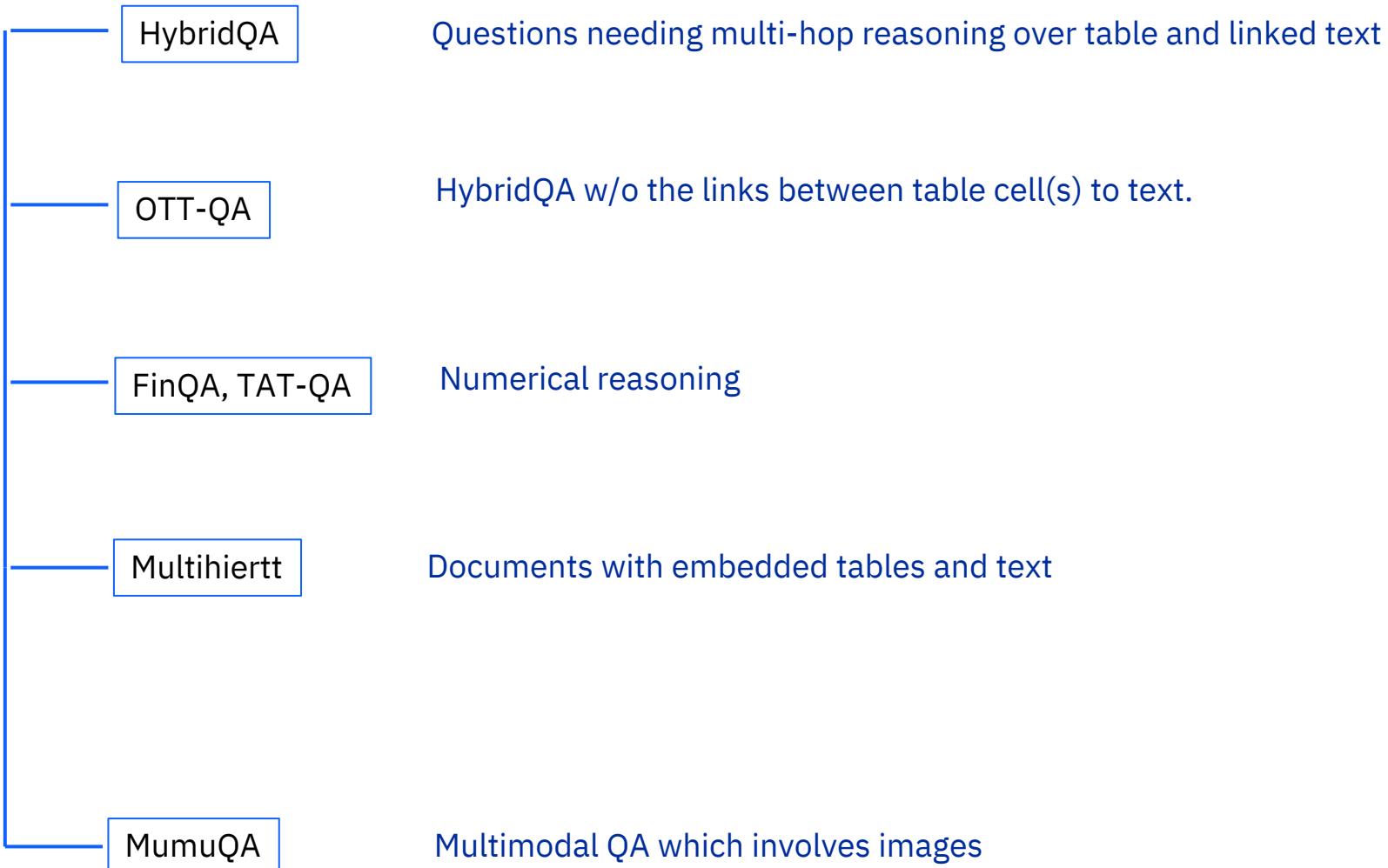


Table + Text QA: Premise

The 2016 Summer Olympics officially known as the Games of the XXXI Olympiad (Portuguese : Jogos da XXXI Olimpíada) and commonly known as **Rio 2016** , was an international multi-sport event

Name	Year	Season	Flag bearer
XXXI	2016	Summer	Yan Naing Soe
XXX	2012	Summer	Zaw Win Thet
XXIX	2008	Summer	Phone Myint Tayzar
XXVIII	2004	Summer	Hla Win U
XXVII	2000	Summer	Maung Maung Nge
XX	1972	Summer	Win Maung

Yan Naing Soe (born **31 January 1979**) is a Burmese judoka . He competed at the 2016 Summer Olympics in the **men 's 100 kg event** , He was the flag bearer for Myanmar at the **Parade of Nations** .

Zaw Win Thet (born **1 March 1991** in Kyonpyaw , Pathein District , Ayeyarwady Division , Myanmar) is a Burmese runner who

Myint Tayzar Phone (Burmese : မြင်တော်ဖွဲ့) born **July 2 , 1978**) is a sprint canoer from Myanmar who competed in the late 2000s .

.....

Win Maung (born **12 May 1949**) is a Burmese footballer . He competed in the men 's tournament at the 1972 Summer Olympics ...

Q: In which year did the judoka bearer participate in the Olympic opening ceremony? A: 2016

Q: Which event does the does the XXXI Olympic flag bearer participate in? A: men's 100 kg event

Q: Where does the Burmese jodoka participate in the Olympic opening ceremony as a flag bearer? A: Rio

Q: For the Olympic event happening after 2014, what session does the Flag bearer participate? A: Parade of Nations

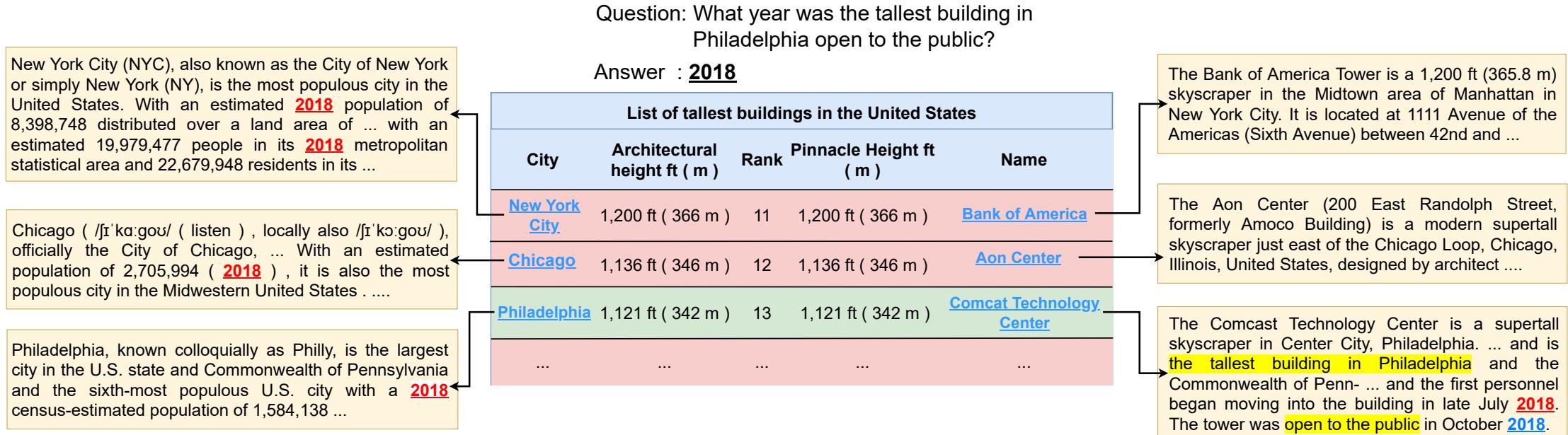
Q: For the XXXI and XXX Olympic event, which has an older flag bearer? A: XXXI

Q: When does the oldest flag Burmese bearer participate in the Olympic ceremony? A: 1972

Hardness

Table + text QA: Challenges

- Multi hop reasoning across table and text



- Only Weak supervision is available.

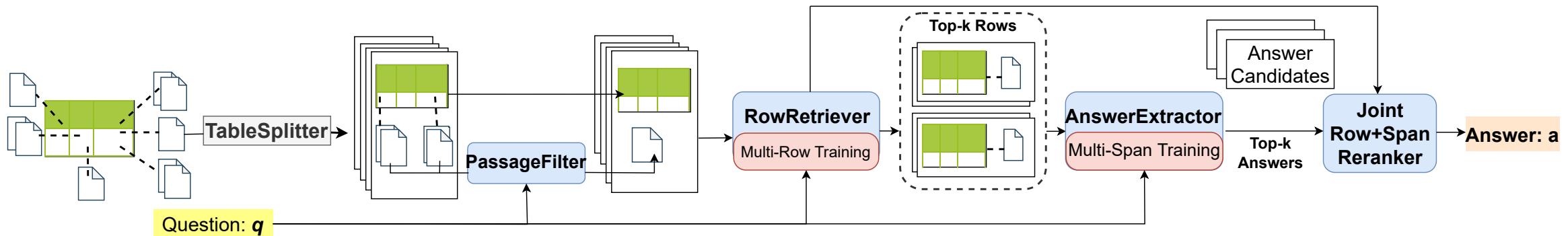
Answer can appear in multiple passages.

Answer can appear in multiple spans in same passage.

MITQA

Row is the quanta of operation.

① Each row of table may have multiple linked passages.	② TableSplitter splits each table into several records, each containing a row, table header and metadata .	③ PassageFilter prunes linked passages until they fit the input limit.	④ RowRetriever retrieves top-k rows with cells or spans containing the right answer.	⑤ AnswerExtractor performs RC over of the the k row cells and passage text to answer the query.	⑥ Using the confidence scores from RowRetriever and AnswerExtractor. Joint Row+Span Reranker selects the best possible among the k answers.
--	--	--	--	---	---



- **TableSplitter**, segments table into *retrieval units*.
- **PassageFilter**, selects a subset of passages to retain with each row.
- **RowRetriever**, Identifies the correct row from which answer can be obtained either as a table cell or as a span from a passage.
- **AnswerExtractor**, trained using multi-span training paradigm and it extracts a table cell/a span from a passage as answer.
- **Joint Row+Span Reranker**, combines the confidence scores of RowRetriever and AnswerExtractor, ranks all the answers and returns the top ranked answer as final answer.

MITQA: Key Novelties

Multi Instance Training

Row Retriever



Where B is the set of relevant rows having the gold ans.

$$\min_{r \in B} \ell(1, f(\mathbf{x}_r)) + \sum_{r' \notin B} \ell(0, f(\mathbf{x}_{r'})).$$

- Incentivizes to assign a large score to any one of the rows in B, but small score to all rows not in B

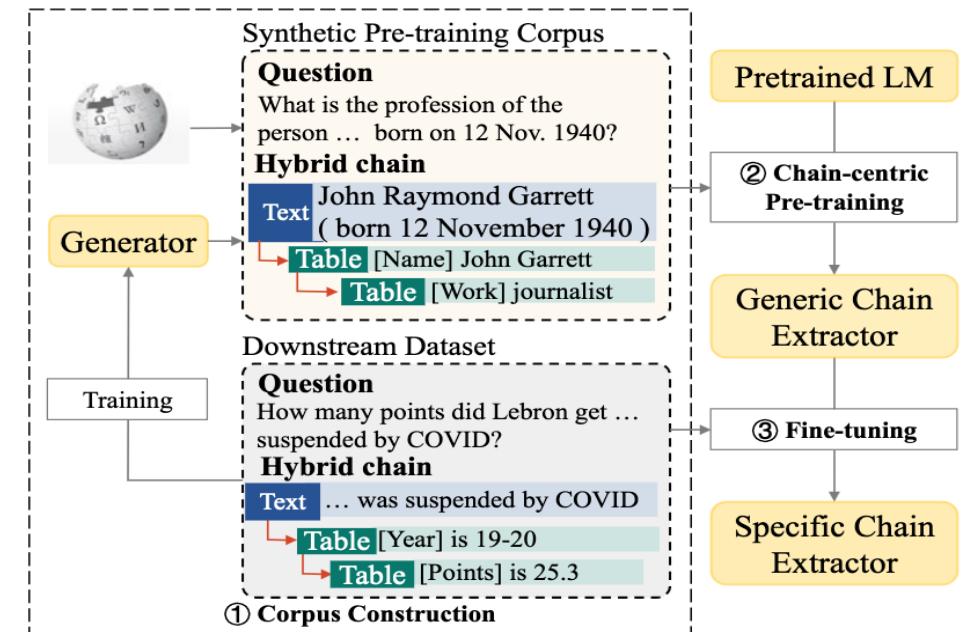
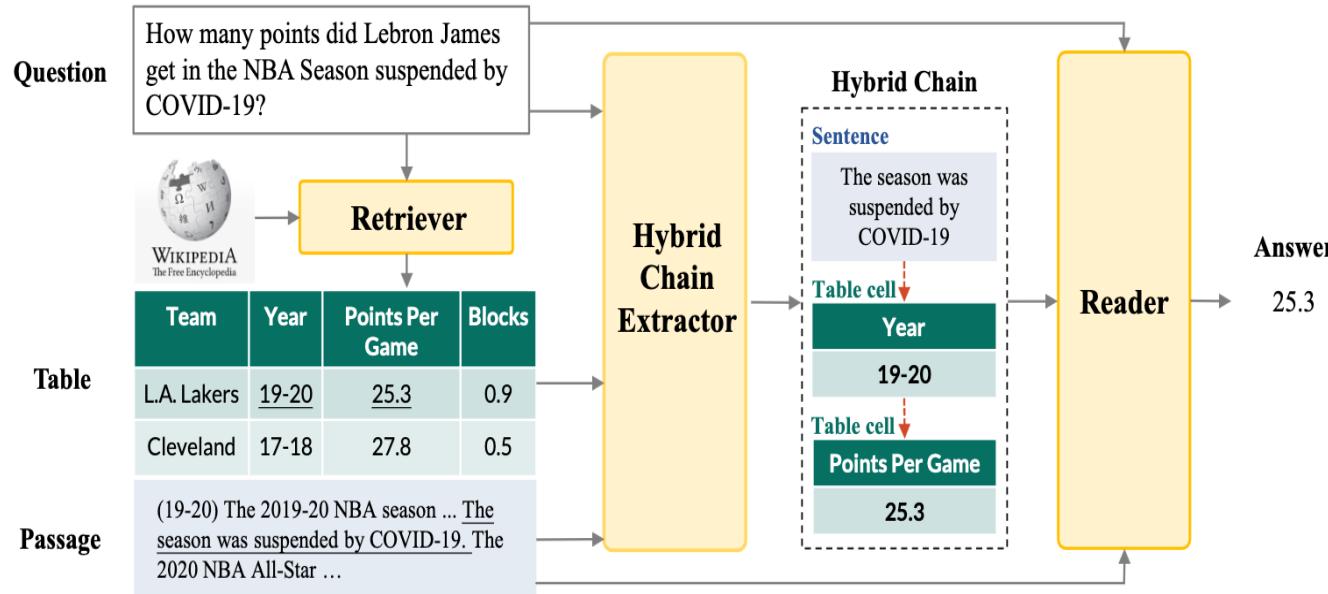
- Ablation showed ~10% (62.1->71.9) improvement in F1 with multi-instance training.

Answer Extractor (AE)



- Train an initial model AE1 on single span instances.
- Then use this initial model AE1 to score matching spans from the noisy instances.
- Treat the top ranked span as the correct one.
- Iterative training data generation makes the model stable.

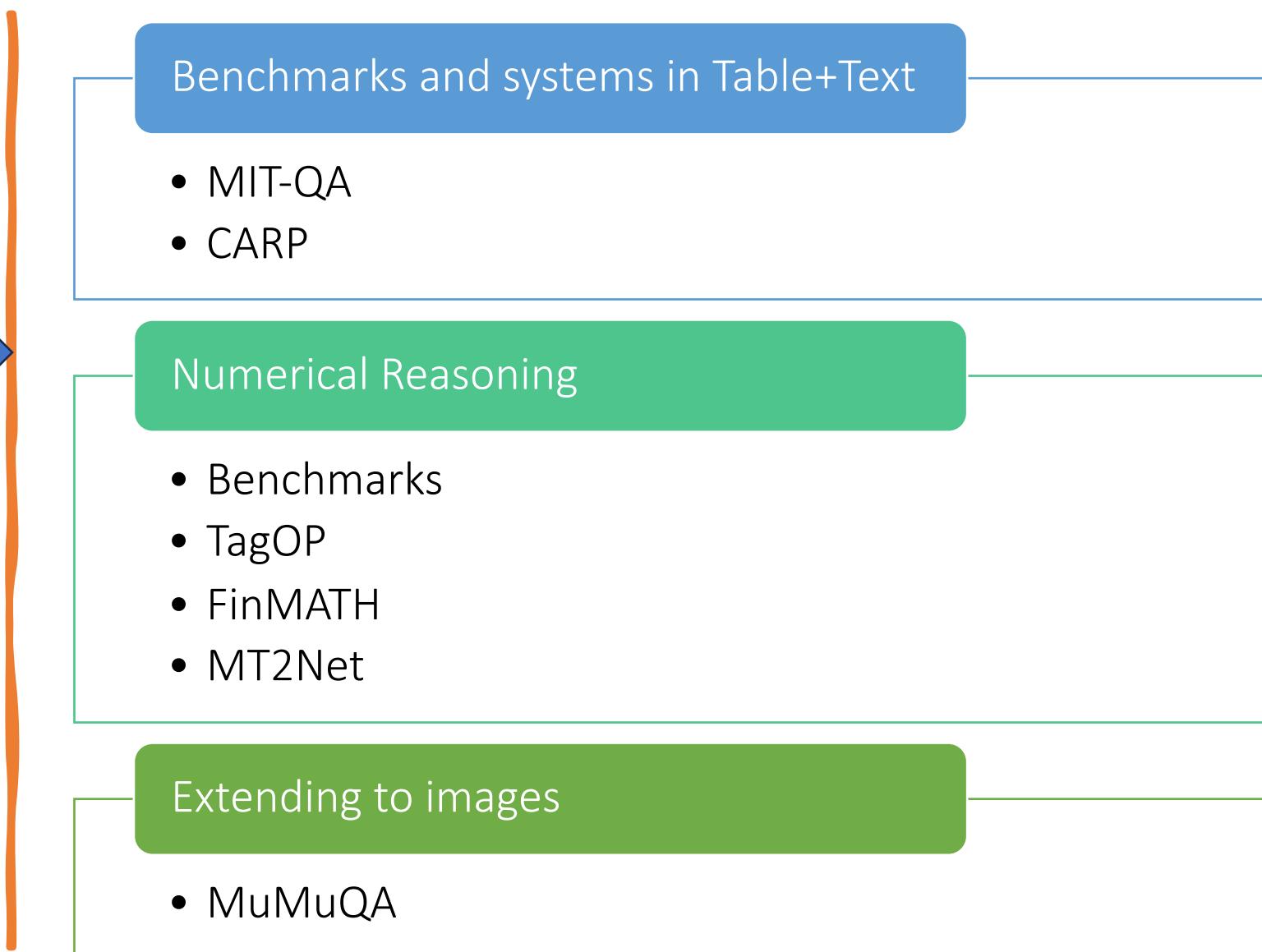
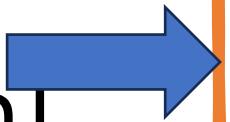
Focus on (learning) and extracting hybrid chains



- Retriever first retrieves knowledge from the corpus for the question.
- Hybrid chain extractor extracts hybrid chains from the knowledge.
- Reader answers the questions with retrieved evidence and extracted hybrid chain

- A generic chain extractor is first learned by pre-training on the synthetic corpus.
- fine-tune the extractor on the downstream dataset.

Multimodal Readers



Numerical Reasoning

TAT-QA

Revenue from external customers, classified by significant product and service offerings, was as follows:

(in millions)

Year Ended June 30,

2019

2018

2017

Server products and cloud services

32,622

26,129

21,649

Office products and cloud services

31,769

28,316

25,573

Windows

20,395

19,518

18,593

Gaming

11,386

10,353

9,051

Search advertising

7,628

7,012

6,219

LinkedIn

6,754

5,259

2,271

Enterprise Services

6,124

5,846

5,542

Devices

6,095

5,134

5,062

Other

3,070

2,793

2,611

Total

\$125,843

\$110,360

\$96,571

Our commercial cloud revenue, which includes Office 365 Commercial, Azure, the commercial portion of LinkedIn, Dynamics 365, and other commercial cloud properties, was **\$38.1 billion**, **\$26.6 billion** and **\$16.2 billion** in fiscal years 2019, 2018, and 2017, respectively. These amounts are primarily included in Office products and cloud services, Server products and cloud services, and LinkedIn in the table above.

#	Reasoning	Question	Answer	Scale	Derivation
1	Word Matching (38.06%)	How much revenue came from LinkedIn in 2018?	5,259	million	-
2	Set of spans (11.94%)	Which were the bottom 2 revenue items for 2017?	LinkedIn, Other	-	-
3	Comparison (5.65%)	Which year has the lowest revenue?	2017	-	-
4	Counting (2.28%)	How many revenue items are between 6,000 million and 6,500 million in 2019?	2	-	Devices ## Enterprise Services
5	Addition (2.37%)	What is the total revenue of commercial cloud from 2017 to 2018?	42.8	billion	26.6 + 16.2
6	Subtraction (16.17%)	How much of the total revenue in 2018 did not come from devices?	105,226	million	110,360 - 5,134
7	Division (3.84%)	How much does the commercial cloud revenue account for the total revenue in 2019?	30.28	%	38.1 billion / 125,843 million
8	Composition (19.69%)	What was the percentage change in gaming between 2018 and 2019?	9.98	%	(11,386 - 10,353) / 10,353

- Evidence that spans across tables and text, embedded in a document.
- Needs arithmetic operation.
- Annotation provided for evidence and operation.

Numerical Reasoning

Fin-QA

Page 91 from the annual reports of GRMN (Garmin Ltd.)

The fair value for these options was estimated at the date of grant using a Black-Scholes option pricing model with the following weighted-average assumptions for 2006, 2005 and 2004:

	2006	2005	2004
Weighted average fair value of options granted	\$20.01	\$9.48	\$7.28
Expected volatility	0.3534	0.3224	0.3577
Distribution yield	1.00%	0.98%	1.30%
Expected life of options in years	6.3	6.3	6.3
Risk-free interest rate	5%	4%	4%

... The total fair value of shares vested during 2006, 2005, and 2004 was \$9,413, \$8,249, and \$6,418 respectively. The aggregate intrinsic values of options outstanding and exercisable at December 30, 2006 were \$204.1 million and \$100.2 million, respectively. (... abbreviate 10 sentences ...)

Question: Considering the weighted average fair value of options , what was the change of shares vested from 2005 to 2006?

Answer: - 400

Calculations:

$$\left(\frac{9413}{20.01} \right) - \left(\frac{8249}{9.48} \right) = -400$$

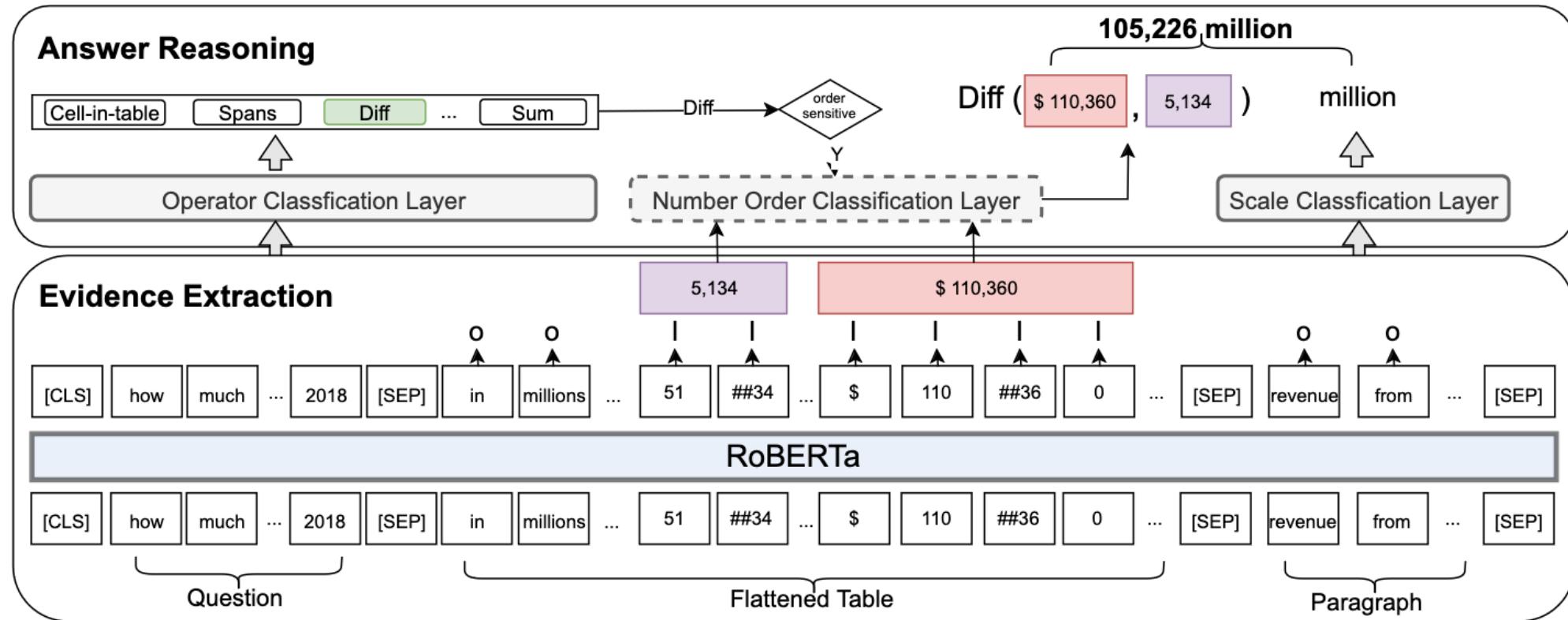
Program:

$$\begin{array}{ccc} \text{divide} (9413, 20.01) & & \text{divide} (8249, 9.48) \\ \hline & & \\ & \searrow & \end{array}$$

subtract (#0, #1)

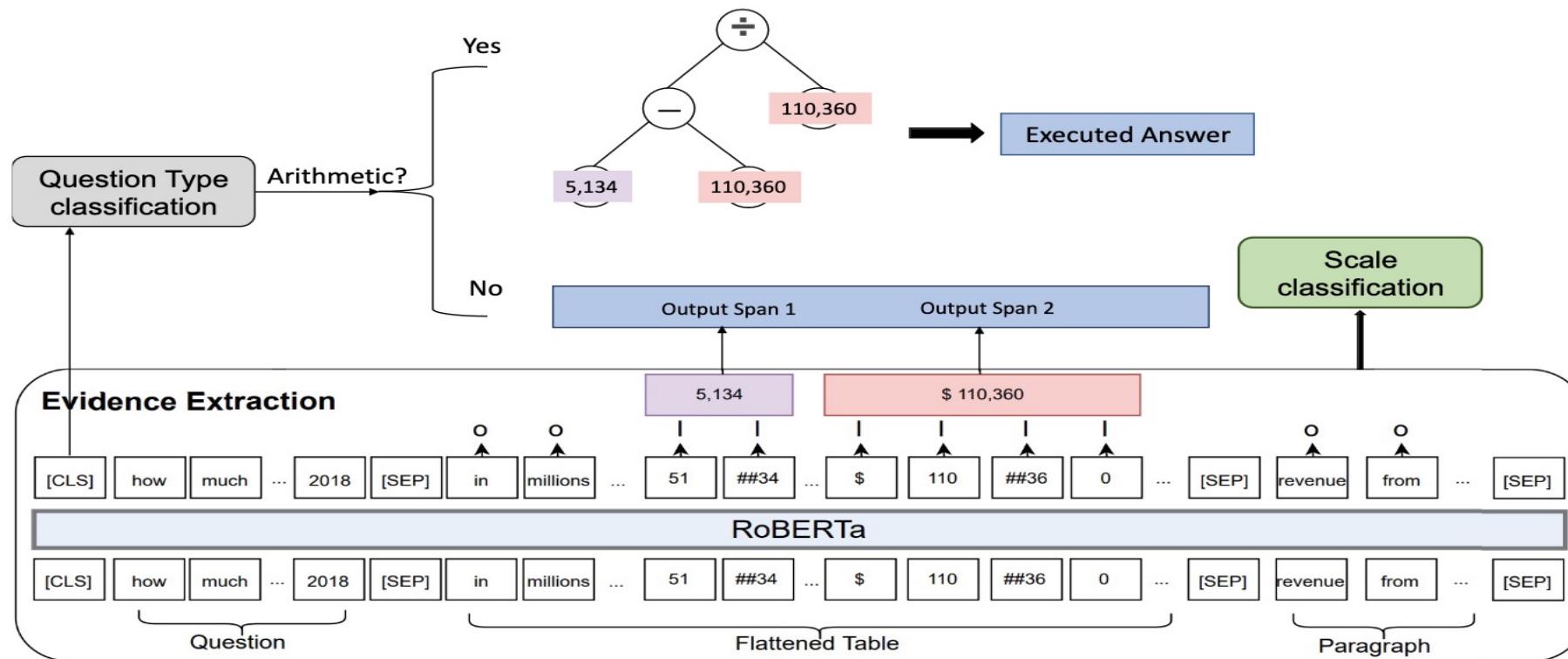
- Very similar to TAT-QA, except it is specific to Financial Documents.
- Evidence that spans across tables and text embedded in a document
- Needs arithmetic operation.
- Annotation provided for evidence and operation.

TagOP : A baseline architecture



- Sequence tagging as Inside/Outside tag → RoBERTa representation followed by a FF network to softmax.
- [CLS] is used to predict an operation among candidates: Span/cell selection and numeric operations.
- Operation order is learnt by a classifier over ‘I’ tagged evidence representations.
- Scale prediction classifier is learnt over concatenated [CLS] and [tab], [para] representations .
- Trained for minimizing negative log likelihood sum of 4 classifier losses.

FinMATH



- Follows a similar sequence tagging over RoBERTa representation.
- Uses a classifier to learn question type and scale.
- uses Seq2Tree as the decoder to generate an executable expression.
- Decoding is designed to generate operands for every predicted operation.

Multihiertt- Complex Tables

Document:

(... abbreviate...)

The following table presents product and service sales and operating expenses by segment (dollar in millions):

Segment	hierarchical column headers			
	Year Ended December 31		2018	2017
	Sales	Expenses	Sales	Expenses
Innovation Systems				
Product	2,894	2,582	—	—
Service	382	351	—	—
Aerospace Systems				
Product	11,087	9,889	10,064	8,988
Service	2,009	1,796	2,067	1,854
Mission Systems				
Product	7,329	6,335	7,012	6,088
Service	4,380	3,854	4,458	3,940
Technology Service				
Product	485	450	391	360
Service	3,812	3,404	4,296	3,878

Product sales for 2018 increased \$4.3 billion, or 25 percent, as compared with 2017. The increase was primarily due to the addition of \$2.9 billion of product sales from Innovation Systems and higher restricted and F-35 volume at Aerospace Systems.

(... abbreviate...)

The table below reconciles funds provided to each segment (dollar in millions):

Segment	2018		2017	
	Funded	Funded	% Change	
Innovation Systems	5,928	—	—	—
Aerospace Systems	11,448	9,560	19.7 %	
Mission Systems	9,676	9,277	4.3 %	
Technology Services	2,883	2,792	3.3 %	

Approximately \$26.6 billion of the \$53.5 billion total at December 31, 2018 is expected to be converted into sales in 2019. (... abbreviate...)

Question: In 2018, what was the total sales increase in the segment with most funds in 2017?

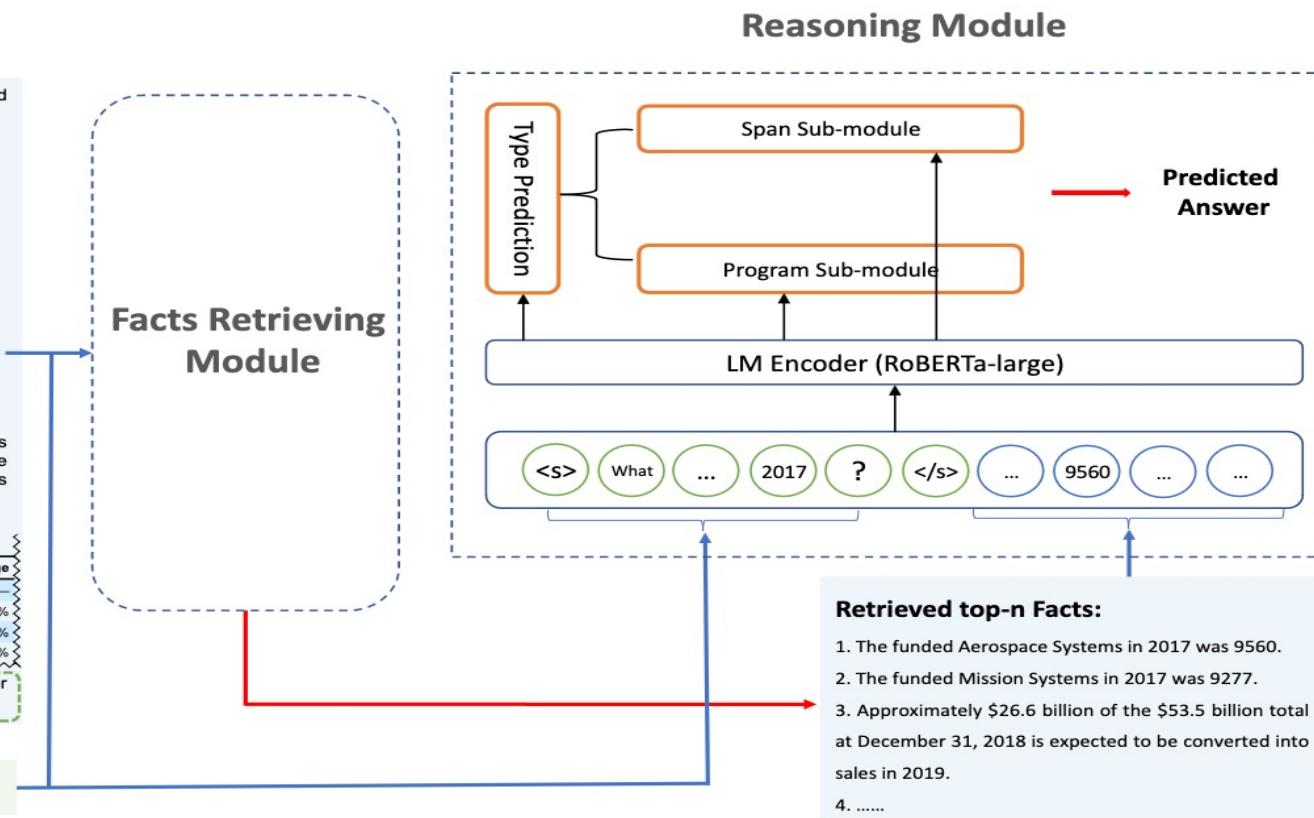
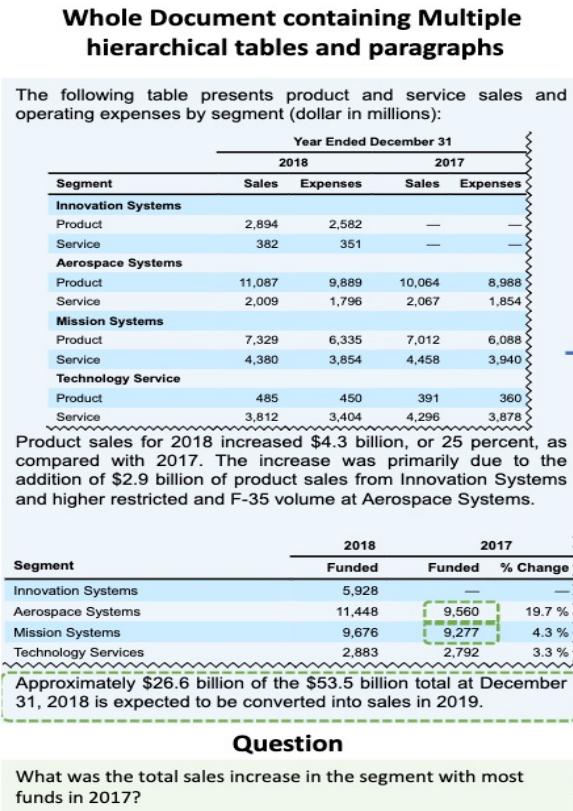
Answer: 965

Numerical expression: $(11087 - 10064) + (2009 - 2067)$

What is unique ?

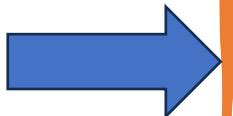
- Tables are more complex:
 - it has hierarchies, row/col headers.
- Questions need more complex reasoning chains.
 - often more than 3 hops.

MT2Net



- Relevant facts are retrieved by BERT based bi-classifier over sentences transformed from table cells ad text.
- RoBERTa based encoding layer used for (1) span extraction (2) question type classification and (3) input to the decoder for program generation.
- LSTM based decoder to generate an executable program – which can refer to tokens from the evidence, operators , previous results in memory.
- Each of the modules are trained using the annotated training data.

Multimodal Readers



Benchmarks and systems in Table+Text

- MIT-QA
- CARP

Numerical Reasoning

- Benchmarks
- TagOP
- FinMATH
- MT2Net

Extending to images

- MuMuQA

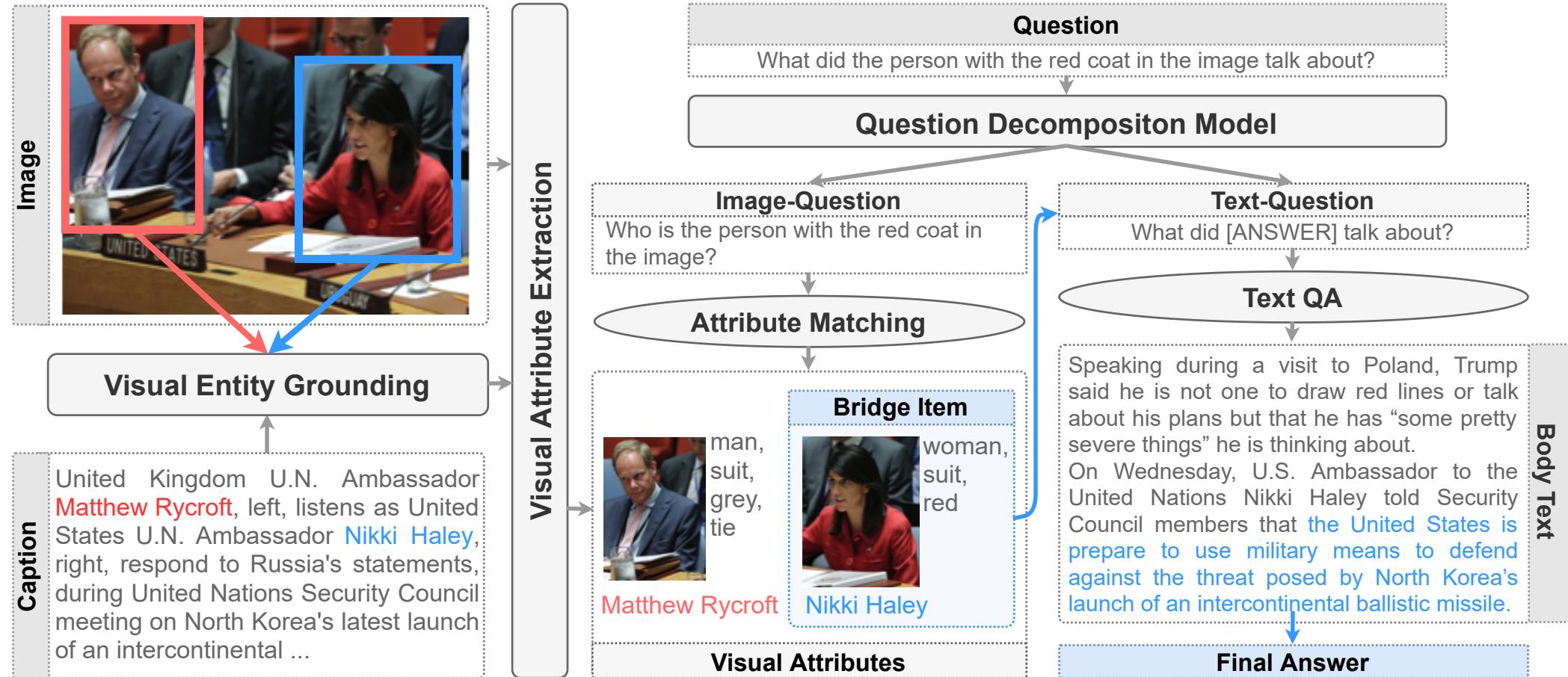
Multimodal QA

- Images can have complementary information useful to answer questions.
- For instance, images in news have objects that are coreferential to the news body text.
- It becomes crucial to identify the objects grounded across modalities to gather the necessary information.

Image - Caption	Body Text
 <p data-bbox="1326 826 1761 1005">Israeli Prime Minister Benjamin Netanyahu (R) speaks with Finance Minister Moshe Kahlon during the weekly cabinet meeting in Jerusalem</p>	<p data-bbox="1776 538 2365 740">A dispute between Israeli Prime Minister Benjamin Netanyahu and his finance minister over broadcast regulation sparked speculation on Sunday that Netanyahu could seek an election two years ahead of schedule.</p> <p data-bbox="1776 754 2365 956">... The Israeli media quoted Netanyahu as telling ministers from his Likud party that he would dissolve the government if Kahlon didn't fall into line. Kahlon heads the Kulanu party, a center-right partner in Netanyahu's</p> <p data-bbox="1776 970 2365 1005">...</p>

Question: What party does the person with the blue tie in the image belong to?
Answer: Likud

Multimodal QA



Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- ✓ Reproducibility in OpenQA: Hands-On Guide I
- ✓ Q&A: [15 min]

1st Half



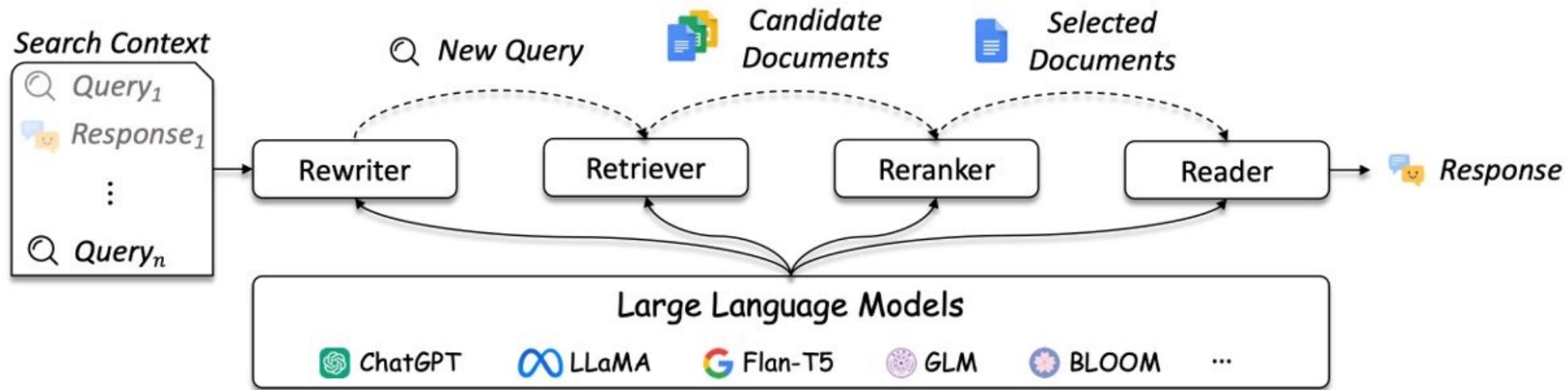
Coffee Break.....

- ✓ Multilingual Readers
- ✓ Multi-modal Readers: Text, Table, Visual QA
- Large Language Models as Retrievers/Readers
- Reproducibility in OpenQA: Hands-on Guide II
- Pipelines, Service and Deployment
- Q&A: [15 min]

2nd Half

Large Language Models as
Readers/Retrievers

Question Answering in the Era of LLMs



LLMs can be used at different steps of the QA pipeline

Recent Advances Overview



LLMs replace Retrievers

- Directly Generate Documents
- Directly Identify Documents

LLMs used with Retrievers

- Guide Retrieval with Query Rewriting
- Re-rank Retrieved Documents

LLMs as Readers

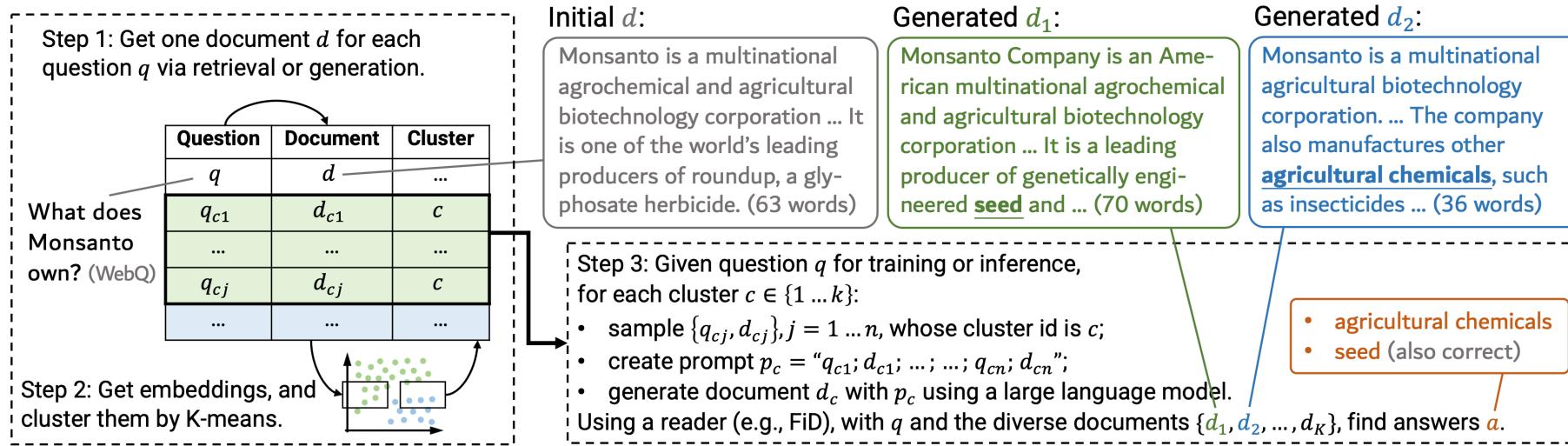
- Without Retrieval Corpus
- With Retrieval Corpus

LLMs Replace Retrievers

LLMs Directly Generate Relevant Documents

- Recite: Recitation-Augmented Language Models (Sun et al 2022)
- Generate Rather than Retrieve: Large Language Models are Strong Context Generators (Yu et al 2023)
- Self-Prompting Large Language Models for Zero-Shot Open-Domain QA (Li et al 2023)

LLMs Directly Generate Relevant Documents



LLMs Directly Generate Relevant Documents

Models	Open-domain QA			Fact Checking		Dialogue System	
	NQ	TriviaQA	WebQ	FEVER	FM2	WoW (F1 / R-L)	
<i>*with retriever, AND directly trained on these datasets</i>							
DPR + InstructGPT*	29.1	53.8	20.2	79.8	65.9	15.4	13.7
<i>*with retriever, BUT NOT trained on these datasets</i>							
BM25 + InstructGPT	19.7	52.2	15.8	78.7	65.2	<u>15.7</u>	13.7
Contriever + InstructGPT	18.0	51.3	16.6	80.4	66.6	<u>15.5</u>	<u>14.0</u>
Google + InstructGPT	28.8	<u>58.8</u>	<u>20.4</u>	82.9	<u>66.0</u>	14.8	13.2
<i>*without retriever, and not using external documents</i>							
Previous SoTA methods	24.7 ¹	56.7 ²	19.0 ¹	-	-	-	-
InstructGPT (no docs.)	20.9	57.5	18.6	77.6	59.4	15.4	13.8
GENREAD (InstructGPT)	28.0	59.0	24.6	<u>80.4</u>	65.5	15.8	14.2

Outperforms models
using a retriever

Zero-shot Open Domain QA performance

LLMs Replace Retrievers

LLMs Directly Generate Relevant Documents

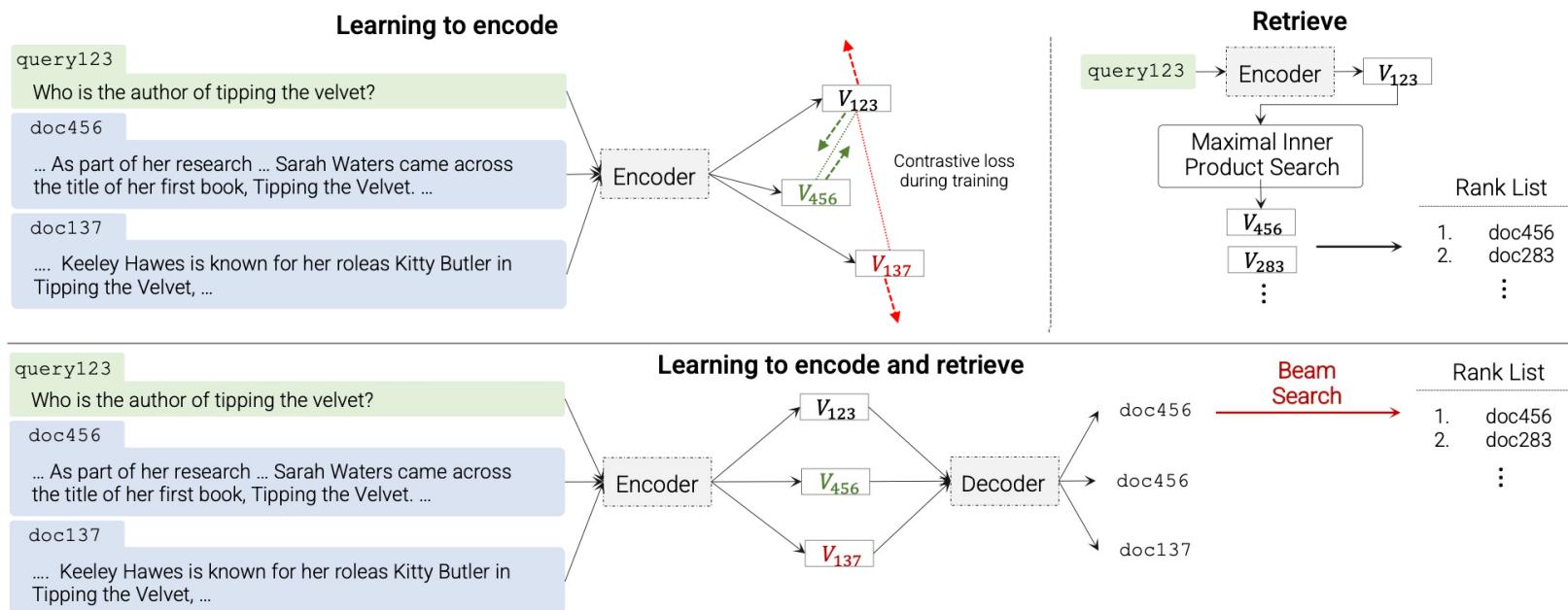
- Recite: Recitation-Augmented Language Models (Sun et al 2022)
- Generate Rather than Retrieve: Large Language Models are Strong Context Generators (Yu et al 2023)
- Self-Prompting Large Language Models for Zero-Shot Open-Domain QA (Li et al 2023)

LLMs Directly Identify Relevant Documents

- Autoregressive Search Engines: Generating Substrings as Document Identifiers (Bevilacqua et al 2022)
- Transformer Memory as a Differentiable Search Index (Tay et al 2022)
- Large Language Models are Built-in Autoregressive Search Engines (Ziems et al 2023)

LLMs Directly Identify Relevant Documents

Differentiable Search Index (DSI): A new paradigm that learns a text-to-text model that maps string queries directly to relevant document IDs



LLMs Directly Identify Relevant Documents

How to represent Document IDs?

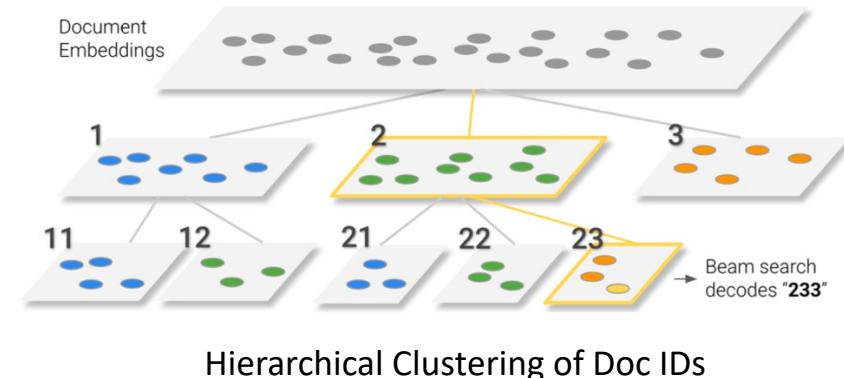
- **Atomic Identifiers:**
 - Each document is assigned a random integer ID
 - Extend the output vocabulary of language model to include Doc IDs
 - Take a softmax over the Doc IDs at inference
- **Naïve String Identifiers:**
 - Represent as tokenizable strings and decode sequentially one token at a time
 - No need to learn embeddings for each Doc ID
 - Can't obtain a Top-K ranking

LLMs Directly Identify Relevant Documents

How to represent Document IDs?

Semantically Structured Identifiers:

- Should capture some information about the semantics of its associated document
- Should be structured in a way that the search space is reduced after each decoding step
- Hierarchical clustering process over document embeddings to induce a decimal tree.
- All documents in a level are clustered into 10 clusters.
- Each document is assigned an identifier with the number of their cluster from 0-9.



LLMs Directly Identify Relevant Documents

Differentiable Search Index

Model	Size	Params	Method	NQ10K		NQ100K		NQ320K	
				Hits@1	Hits@10	Hits@1	Hits@10	Hits@1	Hits@10
BM25	-	-	-	12.4	33.5	20.9	46.4	11.6	34.4
T5	Base	220M	Dual Encoder	16.2	48.6	18.7	55.2	20.5	58.3
T5	Large	800M	Dual Encoder	18.8	55.7	22.3	60.5	22.4	63.3
T5	XL	3B	Dual Encoder	20.8	59.6	23.3	63.2	23.9	65.8
T5	XXL	11B	Dual Encoder	22.1	61.6	24.1	64.5	24.3	67.3
DSI	Base	250M	Atomic Docid	13.0	38.4	23.8	58.6	20.7	40.9
DSI	Large	800M	Atomic Docid	31.3	59.4	17.1	52.3	11.6	37.6
DSI	XL	3B	Atomic Docid	40.1	76.9	19.0	55.3	28.1	61.9
DSI	XXL	11B	Atomic Docid	39.4	77.0	25.3	67.9	24.0	55.1
DSI	Base	250M	Naive String Docid	28.1	48.0	18.7	44.6	6.7	21.0
DSI	Large	800M	Naive String Docid	34.7	60.5	21.2	50.7	13.3	33.6
DSI	XL	3B	Naive String Docid	44.7	66.4	24.0	55.1	16.7	58.1
DSI	XXL	11B	Naive String Docid	46.7	77.9	27.5	62.4	23.8	55.9
DSI	Base	250M	Semantic String Docid	33.9	57.3	19.0	44.9	27.4	56.6
DSI	Large	800M	Semantic String Docid	37.5	65.1	20.4	50.2	35.6	62.6
DSI	XL	3B	Semantic String Docid	41.9	67.1	22.4	52.2	39.1	66.8
DSI	XXL	11B	Semantic String Docid	48.5	72.1	26.9	59.5	40.4	70.3

Outperforms Standard
Dual Encoder Based
Approach

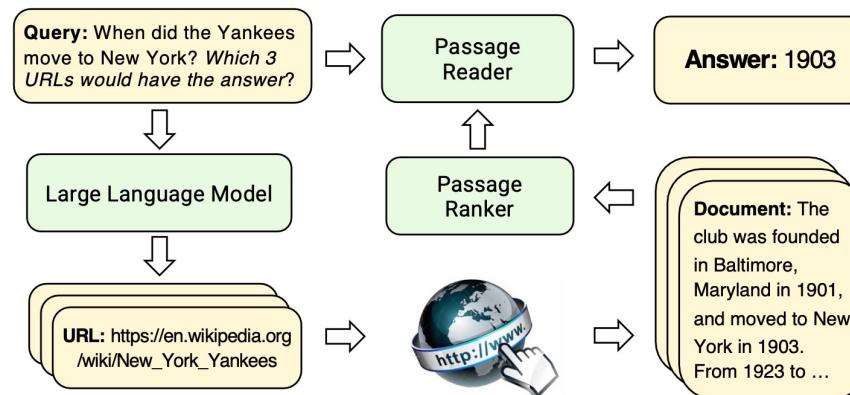
Semantic Structuring of Doc IDs
works the best

Works for a variety of corpus sizes

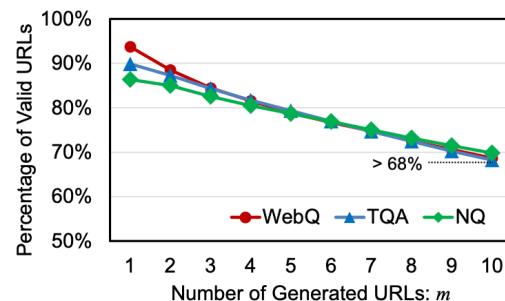
Larger models considerably
improve performance

LLMs Directly Identify Relevant Documents

Use LLM to directly generate URLs for document retrieval



Use a passage ranker to filter only the most relevant ones



As m increases, invalid URLs become more frequent

Method	Document Recall@1			Document Recall@10		
	WebQ	NQ	TriviaQA	WebQ	NQ	TriviaQA
Contriever (Izacard et al., 2021)	63.8	53.2	60.6	63.8	80.8	82.5
BM25 (Robertson and Zaragoza, 2009)	49.5	47.2	63.0	81.5	76.8	82.3
Google API	61.1	55.5	51.4	-	-	-
LLM-URL (Zero-Shot)	76.8	61.7	71.3	87.7	83.2	85.5
LLM-URL (Few-Shot)	79.7	62.6	73.5	89.9	83.9	86.8

Document retrieval performance

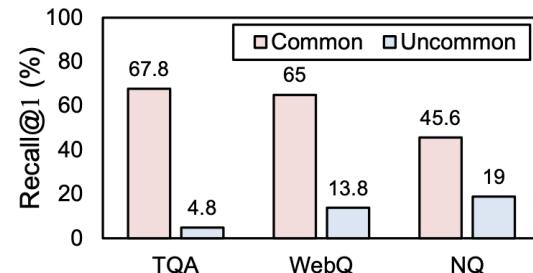
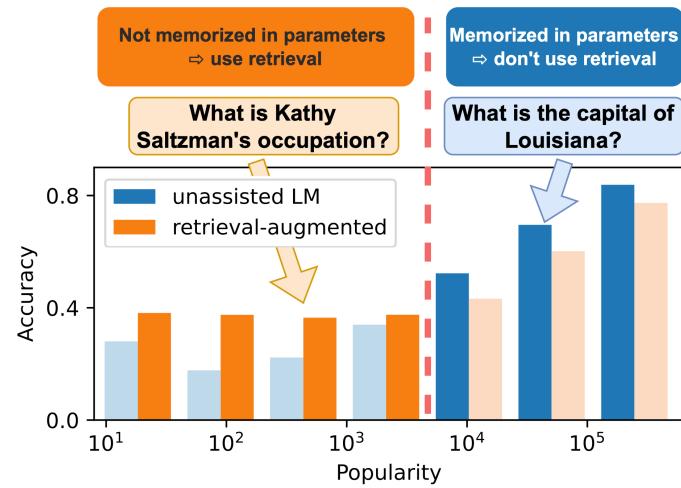
Method	Passage Recall@1			Passage Recall@10			Passage Recall@100		
	WebQ	NQ	TriviaQA	WebQ	NQ	TriviaQA	WebQ	NQ	TriviaQA
Contriever	18.2	18.8	34.0	55.7	54.8	67.9	79.8	79.6	83.3
BM25	19.1	22.8	46.2	51.8	55.6	71.7	76.6	79.6	83.9
LLM-URL (Zero-Shot)	22.2	24.0	46.7	63.1	60.6	76.6	83.8	78.3	83.6
LLM-URL (Few-Shot)	22.3	25.5	49.1	64.8	60.8	77.8	85.9	79.0	84.8

Passage retrieval performance

Considers only Wikipedia URLs (<https://en.wikipedia.org/wiki/>)

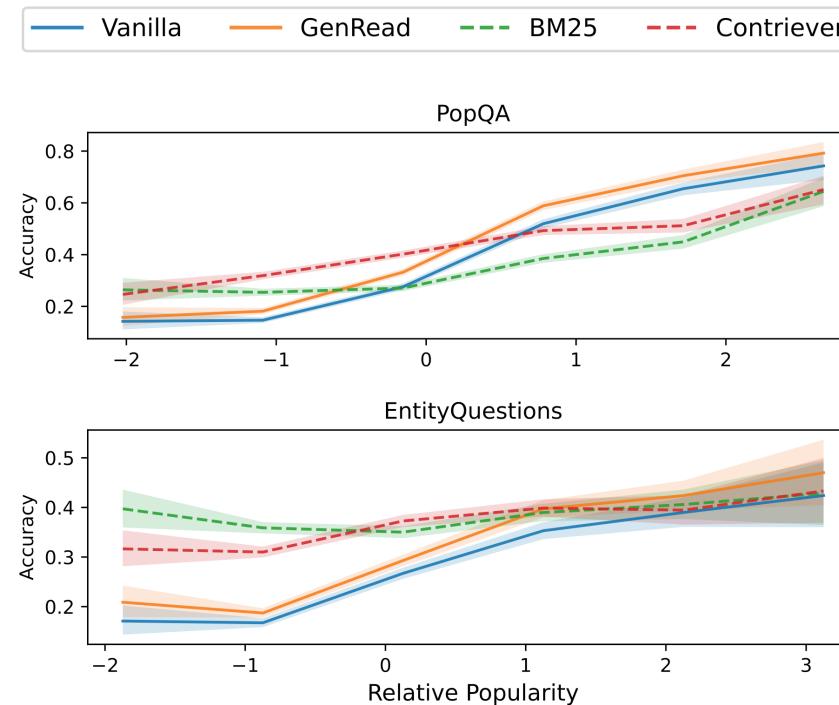
Do we still need external knowledge? Yes!

LLMs are better at directly answering (without any retrieval) questions about popular entities



LLM-URL common vs uncommon entity recall

Non-Parametric memories are useful for less popular entities



Recent Advances Overview

```
graph LR; A[Recent Advances Overview] --> B[LLMs replace Retrievers]; A --> C[LLMs used with Retrievers]; A --> D[LLMs as Readers]
```

LLMs replace Retrievers

- Directly Generate Documents
- Directly Identify Documents

LLMs used with Retrievers

- Guide Retrieval with Query Rewriting
- Re-rank Retrieved Documents

LLMs as Readers

- Without Retrieval Corpus
- With Retrieval Corpus

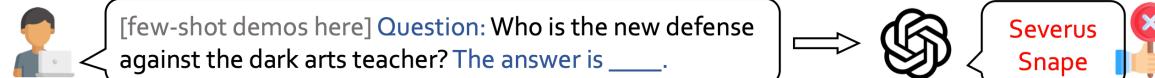
LLMs used with Retrievers

LLMs Guide the Retrieval of Relevant Documents

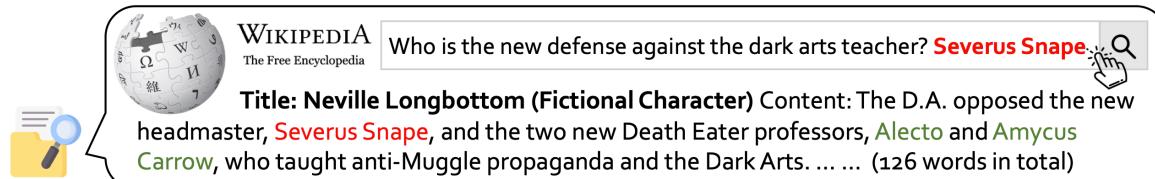
- Generation-Augmented Retrieval for Open-Domain Question Answering (Mao et al 2021)
- ReFeed: Improving Language Models via Plug-and-Play Retrieval Feedback (Yu et al 2023)
- HyDE: Precise Zero-Shot Dense Retrieval without Relevance Labels (Gao et al 2023)
- Led: Lexicon-enlightened dense retriever for large-scale retrieval (Zhang et al 2023)
- Query2doc: Query expansion with large language models (Wang et al 2023)
- Query rewriting for retrieval-augmented large language models (Ma et al 2023)

LLMs Guide Retrieval of Relevant Documents

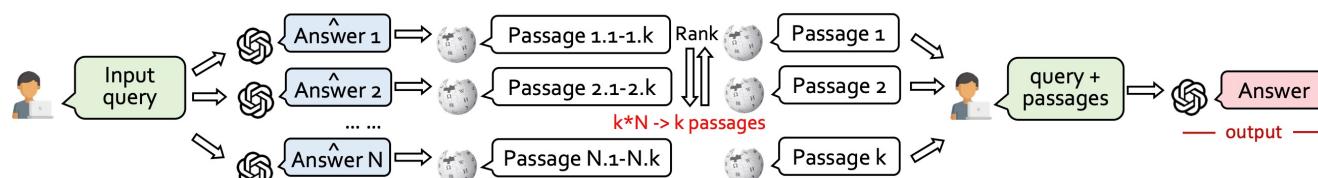
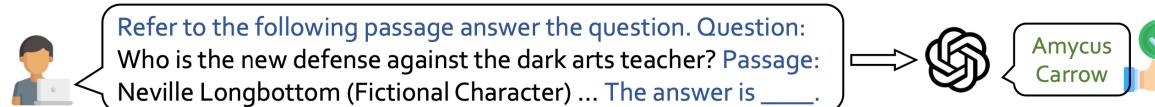
Step-1 (Generate an initial answer): $Q \xrightarrow{\text{GPT 3.5}} \hat{A}$



Step-2 (Retrieve documents): $Q, \hat{A} \xrightarrow{\text{BM25}} D_1, \dots D_n$



Step-3 (Refine the previous answer): $Q, \hat{A}, D_1, \dots D_n \xrightarrow{\text{GPT 3.5}} A$



Prompt the LLM to sample multiple answers, allowing for more comprehensive retrieval feedback

LLMs Guide Retrieval of Relevant Documents

Step-1 (Generate an initial answer): $Q \xrightarrow{\text{GPT 3.5}} \hat{A}$



[few-shot demos here] Question: Who is the new defense against the dark arts teacher? The answer is _____.



Severus Snape

Step-2 (Retrieve documents): $Q, \hat{A} \xrightarrow{\text{BM25}} D_1, \dots, D_n$



WIKIPEDIA
The Free Encyclopedia

Who is the new defense against the dark arts teacher? **Severus Snape**

Title: Neville Longbottom (Fictional Character) Content: The D.A. opposed the new headmaster, **Severus Snape**, and the two new Death Eater professors, **Alecto** and **Amicus Carrow**, who taught anti-Muggle propaganda and the Dark Arts. (126 words in total)

Step-3 (Refine the previous answer): $Q, \hat{A}, D_1, \dots, D_n \xrightarrow{\text{GPT 3.5}} A$



Refer to the following passage answer the question. Question:
Who is the new defense against the dark arts teacher? Passage:
Neville Longbottom (Fictional Character) ... The answer is _____.



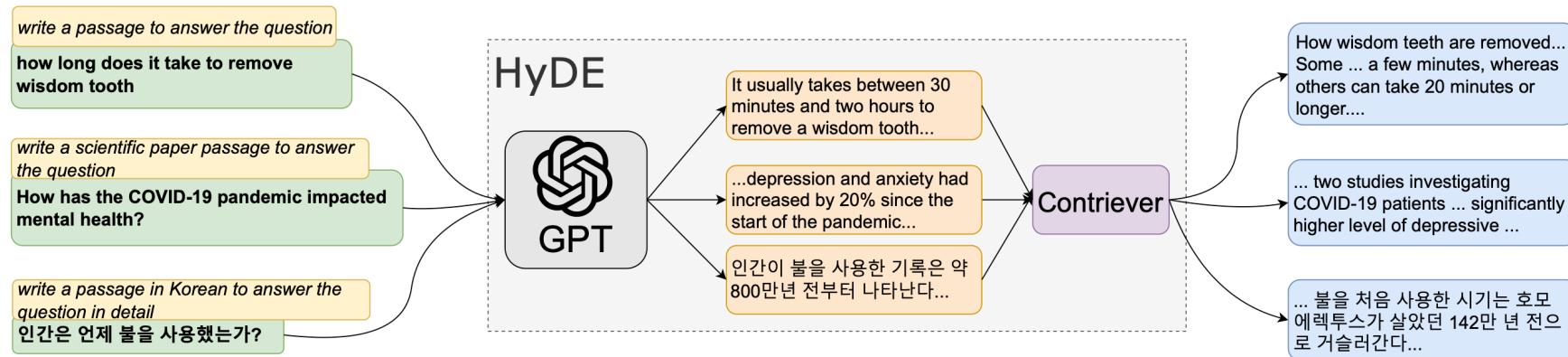
Amicus Carrow

Models	NQ		TriviaQA		HotpotQA		WoW	
	EM	F1	EM	F1	EM	F1	F1	R-L
<i>*close book methods without using retriever</i>								
TD-003 (Ouyang et al., 2022)	29.9	35.4	65.8	73.2	26.0	28.2	14.2	13.3
GenRead (Yu et al., 2023)	32.5	42.0	66.2	73.9	36.4	39.9	14.7	13.5
<i>*open book methods with using retriever</i>								
Retrieve-then-Read	31.7	41.2	61.4	67.4	35.2	38.0	14.6	13.4
ReFeed (Ours)	39.6	48.0	68.9	75.2	41.5	45.1	15.1	14.0

Performance on zero-shot knowledge intensive tasks

LLMs Guide Retrieval of Relevant Documents

*LLM generates a **hypothetical** document whose embedding is used to retrieve **relevant real** documents.*



	Scifact	Arguana	Trec-Covid	FiQA	DBPedia	TREC-NEWS
<i>nDCG@10</i>						
w/o relevance judgement						
BM25	67.9	39.7	59.5	23.6	31.8	39.5
Contriever	64.9	37.9	27.3	24.5	29.2	34.8
HyDE	69.1	46.6	59.3	27.3	36.8	44.0
w/ relevance judgement						
DPR	31.8	17.5	33.2	29.5	26.3	16.1
ANCE	50.7	41.5	65.4	30.0	28.1	38.2
Contriever ^{FT}	67.7	44.6	59.6	32.9	41.3	42.8

Even outperforms models
finetuned with domain
relevance labels

Low resource tasks from BEIR

LLMs used with Retrievers

LLMs Re-rank Relevant Documents

1. Pointwise Methods

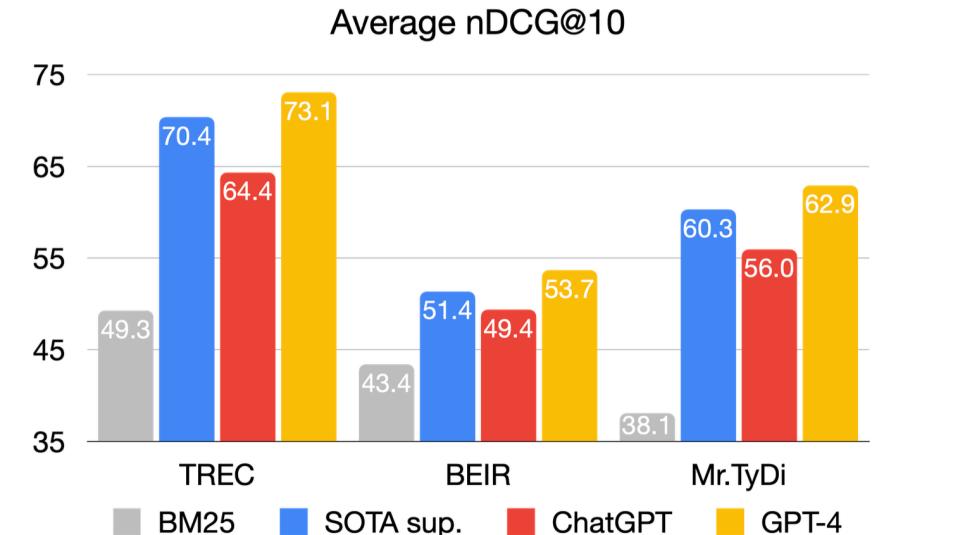
- Improving passage retrieval with zero-shot question generation (Sachan et al 2022)
- Discrete prompt optimization via constrained generation for zero-shot re-ranker (Cho et al 2023)

2. Listwise Methods

- Is chatgpt good at search? investigating large language models as re-ranking agent (Sun et al 2023)
- Zero-shot listwise document reranking with a large language model (Ma et al 2023)

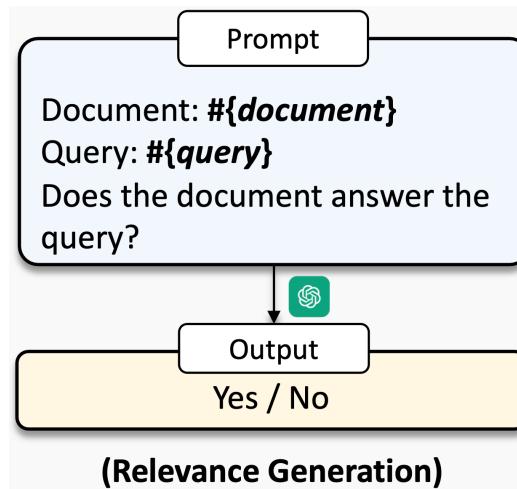
3. Pairwise Methods

- Large language models are effective text rankers with pairwise ranking prompting (Qin et al 2023)

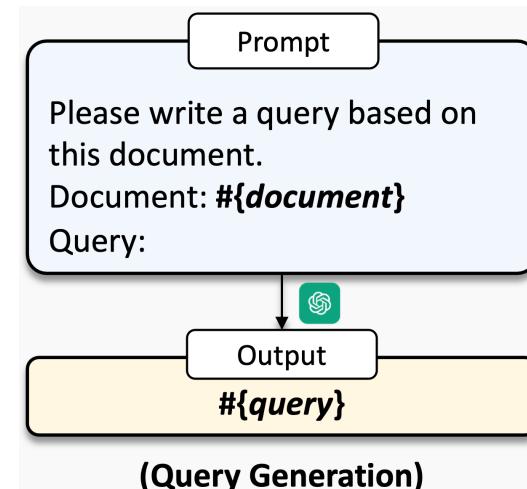


LLMs Re-rank Relevant Documents

Pointwise Methods



$$\text{score} = \begin{cases} 1 + p(\text{Yes}), & \text{if LLMs output Yes} \\ 1 - p(\text{No}), & \text{if LLMs output No} \end{cases}$$

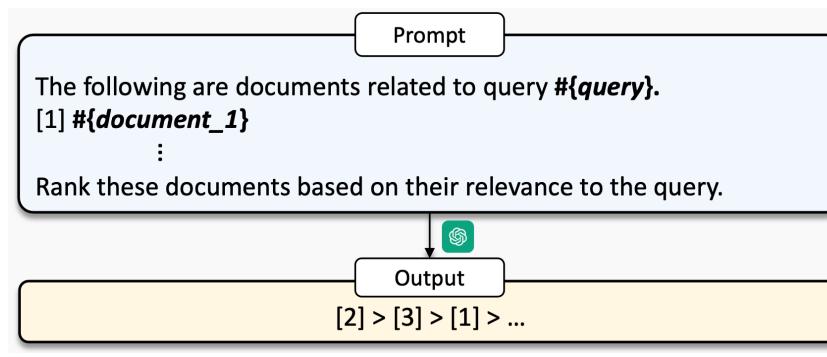


$$\text{score} = \frac{1}{|q|} \sum_i \log p(q_i | q_{<i}, d, \mathcal{P}),$$

However, the log probability of output tokens might be unavailable for LLM APIs (e.g. ChatGPT, GPT-4)

LLMs Re-rank Relevant Documents

Listwise Methods



Step 1 p1 p2 p3 p4 p5 p6 p7 p8

Step 2 p1 p2 p3 p4 p8 p5 p6 p7

Step 3 p1 p2 p8 p3 p4 p5 p6 p7

Ranking results p2 p8 p1 p3 p4 p5 p6 p7

Use sliding window approach since number of input tokens is limited

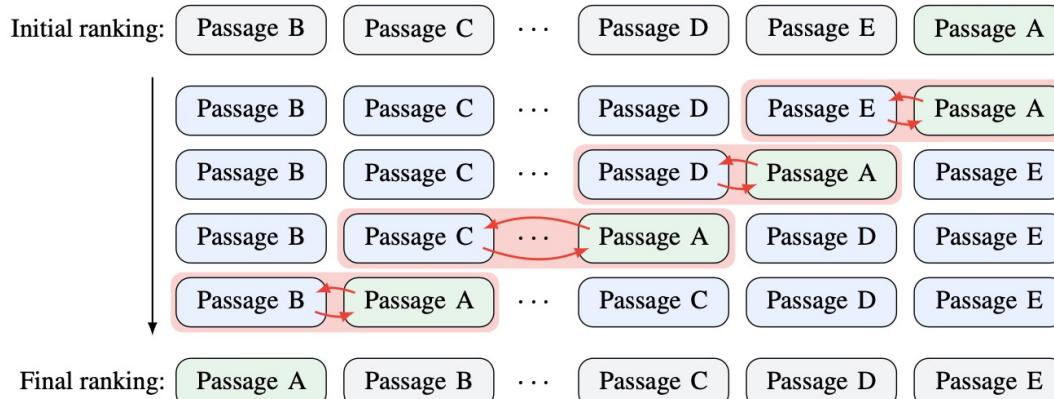
Method	DL19			DL20		
	nDCG@1	nDCG@5	nDCG@10	nDCG@1	nDCG@5	nDCG@10
BM25	54.26	52.78	50.58	57.72	50.67	47.96
Supervised						
monoBERT (340M)	79.07	73.25	70.50	78.70	70.74	67.28
monoT5 (220M)	79.84	73.77	71.48	77.47	69.40	66.99
monoT5 (3B)	79.07	73.74	71.83	80.25	72.32	68.89
Unsupervised						
UPR (FLAN-T5-XL)	51.55	53.71	53.85	63.27	59.41	56.02
InPars (monoT5-3B) [†]	-	-	-	-	-	66.12
LLM API	Instruction methods					
text-curie-001	Relevance generation (4-shot)	39.53	40.02	41.53	41.98	34.80
text-curie-001	Query generation	50.78	50.77	49.76	50.00	48.36
text-davinci-003	Query generation	37.60	44.73	45.37	51.25	47.46
text-davinci-003	Permutation generation	69.77	64.73	61.50	69.75	58.76
gpt-3.5-turbo	Permutation generation	82.17	71.15	65.80	79.32	66.76
gpt-4	Permutation generation	82.56	79.16	75.59	78.40	74.11
						70.56

- Listwise scoring outperforms pointwise scoring
- GPT-4 outperforms even supervised ranking baselines

LLMs Re-rank Relevant Documents

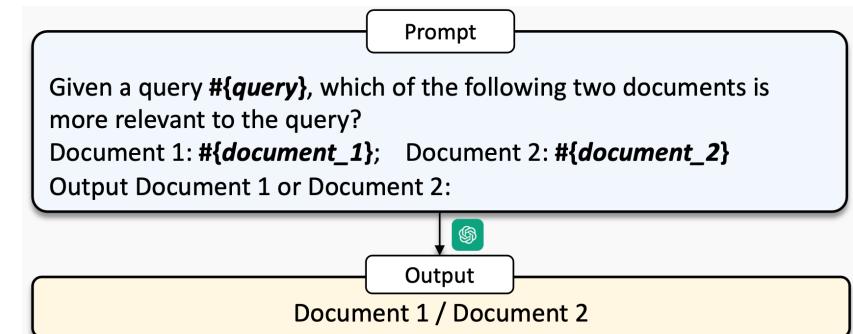
Pairwise Methods

- Listwise methods are highly sensitive to document ordering in prompt.
- Listwise methods don't work well with smaller parameterized models.
- Listwise methods can sometimes miss or repeat document IDs.



An illustration of one pass of our sliding window approach.

K such passes will ensure a high-performing top-K ranking.



Comparison of pointwise, listwise, and pairwise approaches.
N is the number of documents to be ranked for each query

Method	# of LLM API Calls	Generation API	Scoring API	Require Calibration
Pointwise	$O(N)$	No	Yes	Yes
Listwise	$O(N)$	Yes	No	No
Pairwise	$O(N^2), O(N \log N), O(N)$	Yes	Yes	No

Greedy Heapsort One Pass

LLMs Re-rank Relevant Documents

Pairwise Methods

Method	LLM	Size	TREC-DL2019			TREC-DL2020		
			NDCG@1	NDCG@5	NDCG@10	NDCG@1	NDCG@5	NDCG@10
BM25	NA	NA	54.26	52.78	50.58	57.72	50.67	47.96
Supervised Methods								
monoBERT	BERT	340M	79.07	73.25	70.50	78.70	70.74	67.28
monot5	T5	220M	79.84	73.77	71.48	77.47	69.40	66.99
monot5	T5	3B	79.07	73.74	71.83	80.25	72.32	68.89
RankT5	T5	3B	77.38	73.94	71.22	80.86	72.99	69.49
Zero-Shot LLM Methods								
LRL	text-davinci-003	175B	-	-	65.80	-	-	62.24
RankGPT	gpt-3	175B	50.78	50.77	49.76	50.00	48.36	48.73
RankGPT	text-davinci-003	175B	69.77	64.73	61.50	69.75	58.76	57.05
RankGPT	gpt-3.5-turbo	154B*	82.17	71.15	65.80	79.32	66.76	62.91
RankGPT	gpt-4	1T*	82.56	79.16	75.59	78.40	74.11	70.56
PRP-Sliding-10 FLAN-T5-XL	3B		75.58	71.23	68.66	75.62	69.00	66.59
PRP-Sliding-10 FLAN-UL2	20B		78.29	<u>75.49</u>	<u>72.65</u>	85.80	75.35	70.46

Comparable performance to listwise methods with much smaller models

Method	LLM	Init Order	NDCG@1
RankGPT	gpt-3.5-turbo	BM25	82.17
RankGPT	gpt-3.5-turbo	Inverse BM25	36.43
PRP-Sliding-10	FLAN-UL2-20B	BM25	78.29
PRP-Sliding-10	FLAN-UL2-20B	Inverse BM25	71.32

Pairwise is considerably more robust to input ordering

Recent Advances Overview



LLMs replace Retrievers

- Directly Generate Documents
- Directly Identify Documents

LLMs used with Retrievers

- Guide Retrieval with Query Rewriting
- Re-rank Retrieved Documents

LLMs as Readers

- Without Retrieval Corpus *Closed-Book QA*
- With Retrieval Corpus *Retrieval-Augmented LLMs*

Retrieval-Augmented LLMs

Methods	Backbone models	Where to incorporate retrieval	When to retrieve	How to use LLMs
REALM [137]	BERT	Input layer	In the beginning	Fine-tuning
RAG [138]	BART	Input layer	In the beginning	Fine-tuning
REPLUG [139]	GPT	Input layer	In the beginning	Fine-tuning
Atlas [140]	T5	Input layer	In the beginning	Fine-tuning
Lazaridou et al. [141]	Gopher	Input layer	In the beginning	Prompting
He et al. [142]	GPT	Input layer	In the beginning	Prompting
RETA-LLM [143]	LLaMA & GLM & GPT	Input layer	In the beginning	Prompting
RALM [144]	LLaMA & OPT & GPT	Input layer	During generation (every n tokens)	Prompting
RETRO [23]	Transformer	Attention layer	During generation (every n tokens)	Training from scratch
IRCoT [145]	Flan-T5 & GPT	Input layer	During generation (every sentence)	Prompting
FLARE [146]	GPT	Input layer	During generation (aperiodic)	Prompting

Comparison of different approaches incorporating retrieval

Periodic Retrieval

Please refer to the ACL 2023 tutorial:

Retrieval-based Language Models and Applications (Asai et al 2023)

for an exhaustive coverage of Retrieval-Augmented LLMs

Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- ✓ Reproducibility in OpenQA: Hands-On Guide I
- ✓ Q&A: [15 min]

1st Half



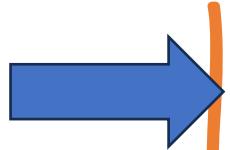
Coffee Break.....

- ✓ Multilingual Readers
- ✓ Multi-modal Readers: Text, Table, Visual QA
- ✓ Large Language Models as Retrievers/Readers
- Reproducibility in OpenQA: Hands-on Guide II
- Pipelines, Service and Deployment
- Q&A: [15 min]

2nd Half

Hands on Guide II with PrimeQA

Hands On Guide II: Readers



Readers in PrimeQA

- Python scripts for reproducibility
- Built-in classes for PrimeQA readers classes
 - Extractive
 - ListQA, BoolQA
 - TableQA
 - Generative
 - Prompt based

Retriever-Reader Pipelines

- Using "QAPipeline" from PrimeQA
- Examples

Reader – Python scripts

Script to reproduce TyDI result:

```
python primeqa/mrc/run_mrc.py --model_name_or_path xlm-roberta-large \  
    --output_dir ${OUTPUT_DIR} --fp16 --learning_rate 4e-5 \  
    --do_train --do_eval --per_device_train_batch_size 16 \  
    --per_device_eval_batch_size 128 --gradient_accumulation_steps 4 \  
    --warmup_ratio 0.1 --weight_decay 0.1 --save_steps 50000 \  
    --overwrite_output_dir --num_train_epochs 1 \  
    --evaluation_strategy no --overwrite_cache
```



```
***** eval metrics *****  
epoch = 1.0  
eval_avg_minimal_f1 = 0.6745  
eval_avg_minimal_precision = 0.7331  
eval_avg_minimal_recall = 0.628  
eval_avg_passage_f1 = 0.7215  
eval_avg_passage_precision = 0.7403  
eval_avg_passage_recall = 0.7061  
eval_samples = 18670
```

You can choose to do only eval by dropping “do_train” flag.

Reproducible scripts supported for: **SQuAD**, **XSQuAD**, **NQ**, **MLQA** with an additional argument: “**--eval_metrics <dataset>**”

Reader – Python scripts

More specific Reader Functionalities

```
python primeqa/mrc/run_mrc.py  
--model_name_or_path PrimeQA/tydi-reader_bpes-xlmr_large-20221117  
--output_dir ${OUTPUT_DIR} --fp16 --overwrite_cache  
--per_device_eval_batch_size 128 --overwrite_output_dir  
--do_boolean  
--boolean_config extensions/boolqa/tydi_boolqa_config.json
```

flag to introduce “Boolean QA”



Results for Bool QA in TyDI

```
***** eval metrics *****  
epoch = 1.0  
eval_avg_minimal_f1 = 0.7151  
eval_avg_minimal_precision = 0.7229  
eval_avg_minimal_recall = 0.7097  
eval_avg_passage_f1 = 0.7447  
eval_avg_passage_precision = 0.7496  
eval_avg_passage_recall = 0.7433  
eval_samples = 18670
```

“run_mrc” script also supports ListQA, Confidence Calibration, TableQA through specific flags.
More info at: <https://github.com/primeqa/primeqa/blob/main/primeqa/mrc/README.md>

Hands On Reader:: *any Reader

- Step-1** Import and use the built-in class
 - available for extractive qa, generative qa, table qa

- Step-2** Load the relevant model from Huggingface
 - Look at: <https://huggingface.co/PrimeQA>

- Step-3** You are ready to ask questions
 - use the .predict() function

Hands On Reader:: Using ExtractiveReader

Step-1 Import and use the built-in class “ExtractiveReader”

```
import json
from primeqa.components.reader.extractive import ExtractiveReader
```

Step-2 Load the relevant model from Huggingface

```
# load the extractive TyDi QA model that has been initialized with XLM-Roberta.
reader = ExtractiveReader(model="PrimeQA/nq_tydi_sq1-reader-xlmr_large-20221110")
reader.load()
```

Step-3 You are ready to ask questions

```
question = ["Which country is Canberra located in?"]
context = ["""Canberra is the capital city of Australia.
Founded following the federation of the colonies of Australia
as the seat of government for the new nation, it is Australia's
largest inland city"""]
answers = reader.predict(question, context)
```

Hands On Reader:: results with ExtractiveReader

Step-1 Example query and context

```
question = ["Which country is Canberra located in?"]  
context = ["""Canberra is the capital city of Australia.  
Founded following the federation of the colonies of Australia  
as the seat of government for the new nation, it is Australia's  
largest inland city"""]  
answers = reader.predict(question,context)
```

Extracted Spans are:

Candidate 1

“Australia. \nFounded following the federation of the colonies of Australia”

Candidate 2

Australia. \nFounded following the federation of the colonies of Australia \nas
the seat of government for the new nation, it is Australia

Hands On Reader:: Using ExtractiveReader for ListQA

Step-1 Import and use the built-in class “ExtractiveReader”

```
import json
from primeqa.components.reader.extractive import ExtractiveReader
```

Step-2 Load the relevant model from Huggingface

```
# load the fine-tuned Natural Questions List QA model that has been initialized with the TyDi model
list_reader = ExtractiveReader(model="PrimeQA/tydiqa-ft-listqa_nq-task-xlm-roberta-large")
list_reader.load()
```

Step-3 You are ready to ask questions

```
question = ["seven union territories of india and their capital"]
context = [{"Category : indian Union Territory capitals - wikipedia Help Category : indian Union Territory capitals"}]
answers = list_reader.predict(question, context)
```

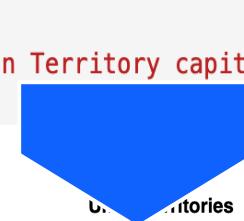
Hands On Reader:: Results for list QA with ExtractiveReader

Step-1 Example query and context

```
question = ["seven union territories of india and their capital"]
context = [{"Category : indian Union Territory capitals - wikipedia Help Category : indian Union Territory capita
answers = list_reader.predict(question,context)
..."]
```

produces Results as:

* Andaman and Nicobar Islands -- Port Blair *
Chandigarh -- Chandigarh * Dadra and Nagar Haveli -
- Silvassa * Daman and Diu -- Daman *
Lakshwadweep -- Kavaratti * National Capital
Territory -- New Delhi * Puducherry -- Pondicherry



Union Territory	Administrative/ Executive capital	Legislative capital	Judicial capital	Year of establishment
Andaman and Nicobar Islands	Port Blair	-	Kolkata	1956
Chandigarh	Chandigarh	-	Chandigarh	1966
Dadra and Nagar Haveli and Daman and Diu	Daman	-	Mumbai	2020
Jammu and Kashmir	Srinagar (summer) Jammu (winter)	Srinagar (summer) Jammu (winter)	Srinagar (summer) Jammu (winter)	2019
Ladakh	Leh (summer) Kargil (winter)	-	Srinagar (summer) Jammu (winter)	2019
Lakshadweep	Kavaratti	-	Ernakulam	1956
Delhi	New Delhi	New Delhi	New Delhi	1956
Puducherry	Pondicherry	Pondicherry	Chennai	1951

Hands On Reader:: Using GenerativeReader

Step-1 Import and use the built-in class “GenerativeReader”

```
import json
from primeqa.components.reader.generative import GenerativeFiDReader
```

Step-2 Load the relevant model from Huggingface

```
fid_reader = GenerativeFiDReader(model='PrimeQA/eli5-fid-bart-large-with-colbert-passages', num_contexts=1)
fid_reader.load()
```

Step-3 You are ready to ask questions

```
question = ["What causes the trail behind jets at high altitude?"]
context = [ """Contrails are a manmade type of cirrus cloud formed when water vapor from
the exhaust of a jet engine condenses on particles, which come from either the
surrounding air or the exhaust itself, and freezes, leaving behind a visible trail.
The exhaust can also trigger the formation of cirrus by providing ice nuclei
when there is an insufficient naturally-occurring supply in the atmosphere.
One of the environmental impacts of aviation is that persistent contrails can
form into large mats of cirrus, and increased air traffic has been implicated
as one possible cause of the increasing frequency and amount of cirrus""",
"""Associated with jet streams is a phenomenon known as clear-air turbulence
(CAT), caused by vertical and horizontal wind shear caused by jet streams.
The CAT is strongest on the cold air side of the jet, next to and just under
the axis of the jet. Clear-air turbulence can cause aircraft to plunge and so
present a passenger safety hazard that has caused fatal accidents, such as the
death of one passenger on United Airlines Flight 826.
Section: Uses.:Possible future power generation.""" ,
```

Hands On Reader:: Results with GenerativeReader

Step-1 Example query and context

```
question = ["What causes the trail behind jets at high altitude?"]  
context = ["""Contrails are a manmade type of cirrus cloud formed when water vapor from  
the exhaust of a jet engine condenses on particles, which come from either the  
surrounding air or the exhaust itself, and freezes, leaving behind a visible trail.  
The exhaust can also trigger the formation of cirrus by providing ice nuclei  
when there is an insufficient naturally-occurring supply in the atmosphere.  
One of the environmental impacts of aviation is that persistent contrails can  
form into large mats of cirrus, and increased air traffic has been implicated  
as one possible cause of the increasing frequency and amount of cirrus""",  
"""Associated with jet streams is a phenomenon known as clear-air turbulence  
(CAT), caused by vertical and horizontal wind shear caused by jet streams.  
The CAT is strongest on the cold air side of the jet, next to and just under  
the axis of the jet. Clear-air turbulence can cause aircraft to plunge and so  
present a passenger safety hazard that has caused fatal accidents, such as the  
death of one passenger on United Airlines Flight 826.  
Section: Uses.:Possible future power generation."""],
```

Generates the following response:

The water vapor in the exhaust from the engine is mixed with the cold air,
and condenses into ice crystals. \n\nThe water is then carried away by the
wind, and the air is heated up. The water vapor condenses and forms clouds

Hands On Reader:: Using TableQA

Step-1 Import and use the built-in class “TableQAModel”

```
from primeqa.tableqa.models.tableqa_model import TableQAModel
```

Step-2 Load the relevant model from Huggingface

```
#load the model from HuggingFace  
model = TableQAModel("PrimeQA/tapas-based-tableqa-wikisql-lookup")
```

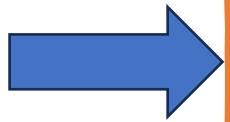
You can also use Tapex based model: "PrimeQA/tableqa_tapex_wtq"

Step-3 You are ready to ask questions

```
# Load the Table  
data = {"Actors": ["Brad Pitt", "Leonardo Di Caprio",  
                  "George Clooney"], "Number of movies": ["87", "53", "69"]}  
queries = ["What is the name of the first actor", "how many movies George Clooney acted in"]  
print(model.predict(data,queries))
```

Result ==> {'What is the name of the first actor': 'Brad Pitt', 'how many movies George Clooney acted in': '69'}

Hands On Guide2: Readers



Readers in PrimeQA

- Extractive
- ListQA, BoolQA
- TableQA
- Generative
- Prompt based

Retriever-Reader Pipelines

- Using "QAPipeline" from PrimeQA
- Examples

Hands On Retriever-Reader Pipeline

Step-1 Import the retriever and reader classes

```
from primeqa.components.retriever.dense import ColBERTRetriever
from primeqa.components.reader.extractive import ExtractiveReader
from primeqa.pipelines.qa_pipeline import QAPipeline
import json
```

Step-2 Instantiate the retriever

```
colbert_retriever = ColBERTRetriever(index_root = index_root,
                                      index_name = index_name,
                                      collection = collection,
                                      max_num_documents = 3)

colbert_retriever.load()
```

Step-3: Instantiate the reader

```
reader = ExtractiveReader()
reader.load()
```

Step-4: Setup the QA pipeline

```
# setup the pipeline
pipeline = QAPipeline(retriever, reader)
```



More examples in:
<https://github.com/primeqa/primeqa/tree/main/notebooks/retriever-reader-pipelines>

Hands On Retriever-Reader Pipeline

Step-5: You're ready to ask question..

```
questions = ["number of participating countries in tour de france 2018 ?"]
answers = pipeline.run(questions, use_retriever=True)
print(json.dumps(answers, indent=4))
```

Retrieved Passages:

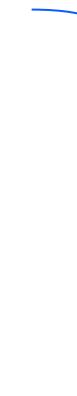
"\"2018 Tour de France\"\\n \"reduced the number of riders per team for Grand Tours from 9 to 8, resulting in a start list total of 176, instead of the usual 198. Of these, 35 competed in their first Tour de France. The total number of riders that finished the race was 145. The riders came from 30 countries. Seven countries had more than 10 riders in the race: France (35), Belgium (19), the Netherlands (13), Italy (13), Australia (11), Germany (11) and Spain (11). The average age of riders in the race was 29.37 years, ranging from the 21-year-old Egan Bernal () to the\"",

Answer

30

30 countries

The riders came from 30



These are spans in the retrieved passage

Hands On Retriever-Reader Pipeline

Step-1 Import the retriever and reader classes

```
from primeqa.components.retriever.dense import ColBERTRetriever
from primeqa.components.reader.prompt import PromptGPTReader
from primeqa.pipelines.qa_pipeline import QAPipeline
import json
```

Step-2 Instantiate the retriever

```
colbert_retriever = ColBERTRetriever(index_root = index_root,
                                      index_name = index_name,
                                      collection = collection,
                                      max_num_documents = 3)

colbert_retriever.load()
```

Step-3: Instantiate the reader

```
# setup a Prompt GPT Reader: We support gpt-3.5-turbo and text-davinci-003
reader = PromptGPTReader(model_name='gpt-3.5-turbo', api_key='API KEY HERE')
reader.load()
```

Step-4: Setup the QA pipeline

```
# setup the pipeline
pipeline = QAPipeline(retriever, reader)
```

Hands On Retriever-Reader Pipeline

Step-5: You're ready to ask question..

```
# start asking questions
questions = ["number of participating countries in tour de france 2017 ?"]
prompt_prefix = "Answer the following question after looking at the text."
answers = pipeline.run(questions, prefix=prompt_prefix)
print(json.dumps(answers, indent=4))
```

Retrieved Passages:

Produces results

"\"2017 Tour de France\"\\n \"de France. The total number of riders that finished the race was 167. *The riders came from 32 countries.* Six countries had more than 10 riders in the race: France (39), Italy (18), Belgium (16), Germany (16), the Netherlands (15), and Spain (13). The average age of riders in the race was 29.4 years, ranging from the 22-year-old \u00c9lie Gesbert () to the 40-year-old Haimar Zubeldia (). had the youngest average age while had the oldest. The teams entering the race were: In the lead up to the 2017 Tour de France, Chris Froome () was seen by many pundits\""

Answer

There were 32 participating countries in the Tour de France 2017

← A new sentence is generated

Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- ✓ Reproducibility in OpenQA: Hands-On Guide I
- ✓ Q&A: [15 min]

1st Half



Coffee Break.....

- ✓ Multilingual Readers
- ✓ Multi-modal Readers: Text, Table, Visual QA
- ✓ Large Language Models as Retrievers/Readers
- ✓ Reproducibility in OpenQA: Hands-on Guide II
- Pipelines, Service and Deployment
- Q&A: [15 min]

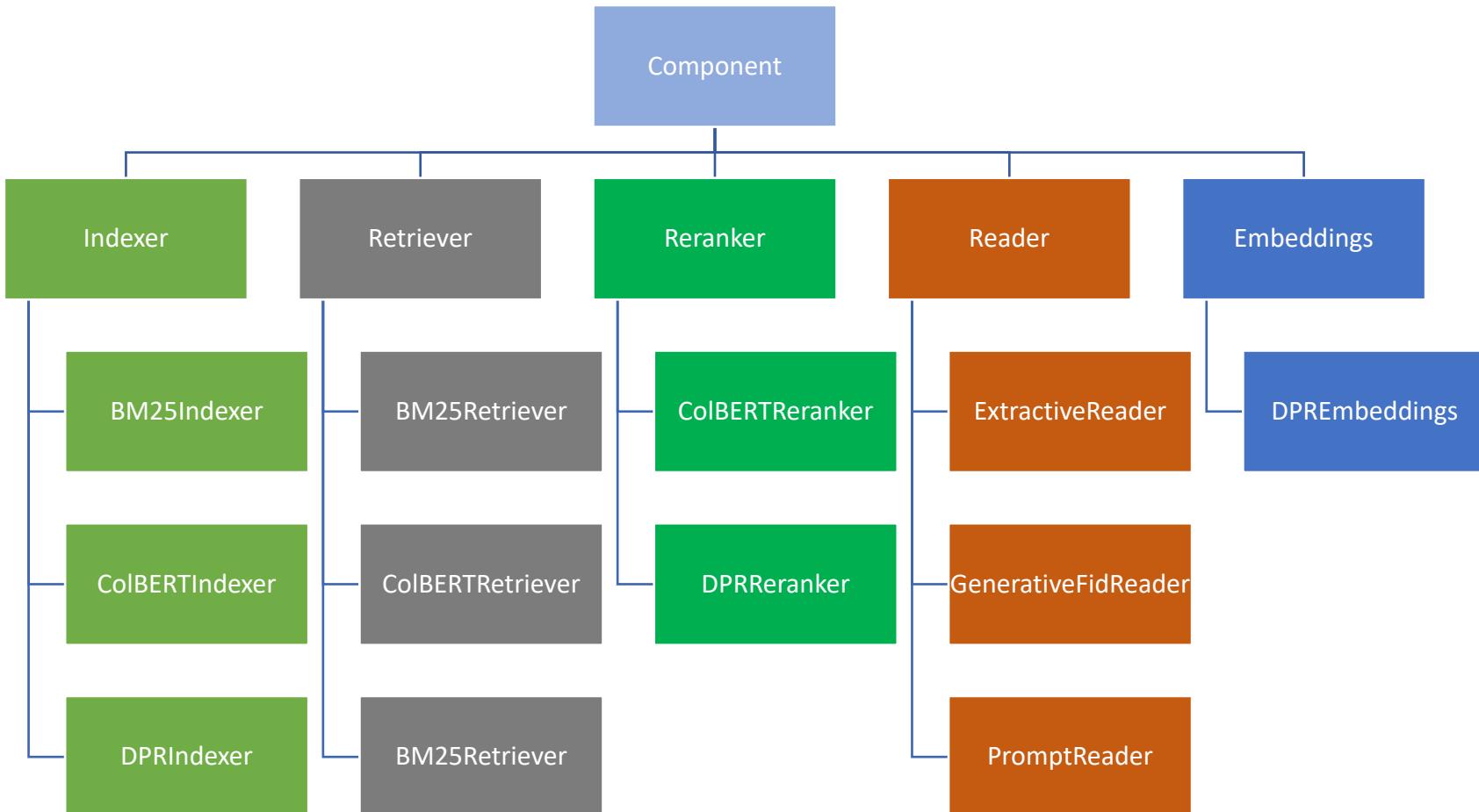
2nd Half

PrimeQA: Services and Deployment

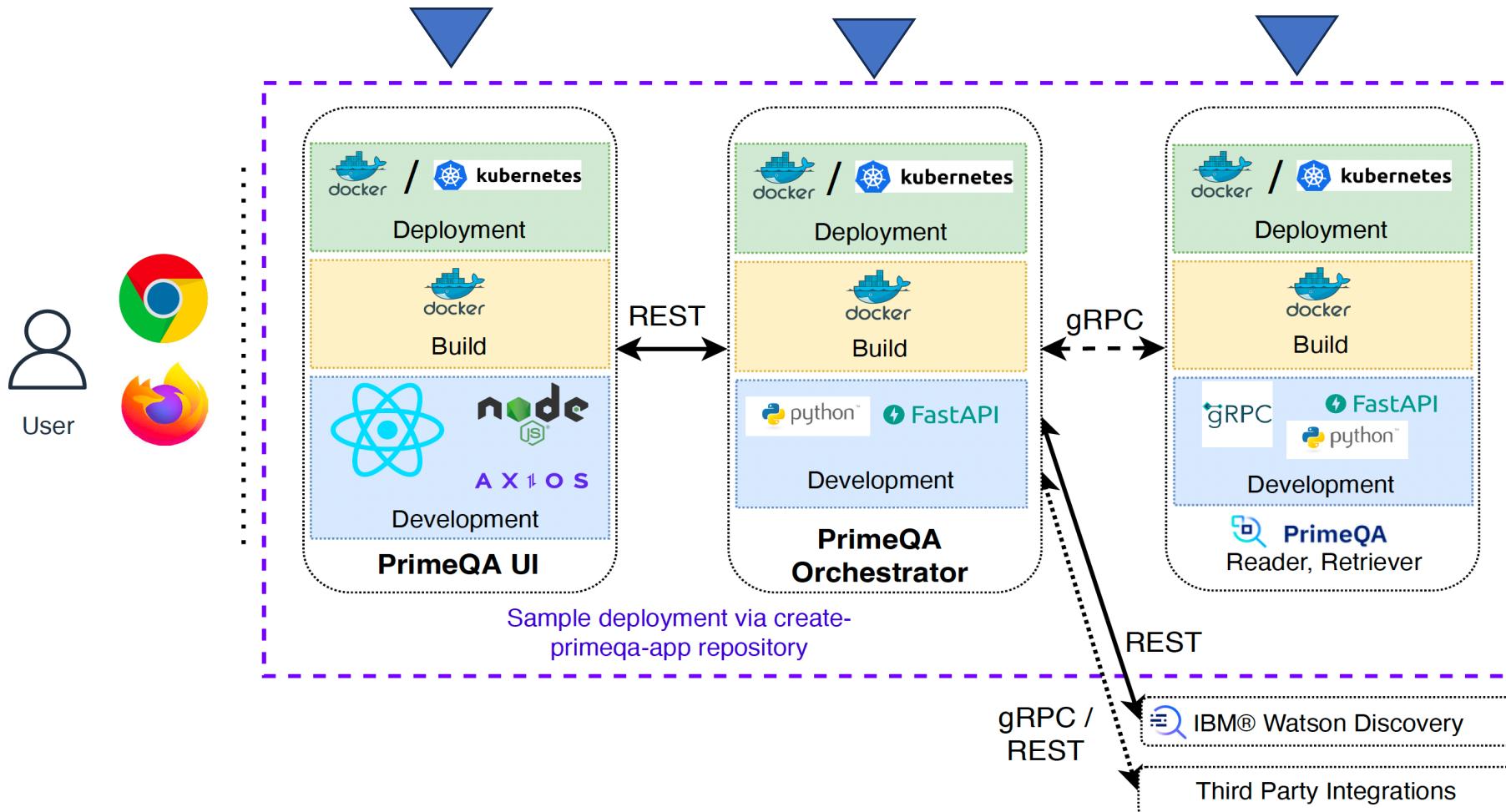
Outline

- Inference Components and Pipelines
- Service Layer
- PrimeQA Application

PrimeQA Components for Inference



PrimeQA Microservices and Application



Create a PrimeQA App

PrimeQA applications are run via  docker

Clone the create-primeqa-app repository which contains the launch scripts

```
git clone git@github.com:primeqa/create-primeqa-app.git
```

```
cd create-primeqa-app
```

Set the environment variable PUBLIC_IP to the ip address of the localhost

```
export PUBLIC_IP=<hostname>
```

Launch the container using bash in cpu (default) or gpu mode

```
./launch.sh
```

```
./launch.sh -m gpu
```

Configure PrimeQA App

Settings are defined in the file `orchestrator-store/primeqa.json`

```
{  
  "retrievers": {  
    "PrimeQA": {  
      "service_endpoint": "primeqa:50051"  
    }  
  },  
  "readers": {  
    "PrimeQA": {  
      "service_endpoint": "primeqa:50051",  
      "beta": 0.7  
    }  
  }  
}
```

```
"retrievers": {  
  "Watson Discovery": {  
    "service_endpoint": "<IBM® Watson Discovery Cloud/CP4D Instance Endpoint>",  
    "service_api_key": "<API key (ONLY If using IBM® Watson Discovery Cloud> ",  
    "service_project_id": "<IBM® Watson Discovery Project ID>"  
  }  
},
```

Drop in your own model or select a PrimeQA model from HuggingFace Model Hub

Drop in your own document collection and index

Demos built with PrimeQA



[Retrieval](#)

Search a document collection using dense and sparse information retrieval techniques

[Reading](#)

Find answer to questions based on a given context

[Question Answering](#)

Find answers to question from retrieved evidence blocks

OpenQA demo application built with PrimeQA

 **PrimeQA**

Question Answering
Find answers to question from retrieved evidence blocks

What would you like to know?

X

Ask

Found 15 answers matching your question.

Answer
Nagini

[^ Evidence](#)

Title: Book7 Paragraph 1300
Text: "Come," said Voldemort, and Harry heard him move ahead, and Hagrid was forced to follow. Now Harry opened his eyes a fraction, and saw Voldemort striding in front of them, wearing the great snake **Nagini** around his shoulders, now free of her enchanted cage. But Harry had no possibility of extracting the wand concealed under his robes without being noticed by the Death Eaters, who marched on either side of them through the slowly lightening darkness. ... "Harry," sobbed Hagrid. "Oh, Harry ... Harry ..." Harry shut his eyes tight again. He knew that they were approaching the castle and strained his ears to distinguish, above the gleeful voices of the Death Eaters and their tramping footsteps, signs of life from those within. "Stop."

Score (out of 100): 87
Was this answer useful?  

Ask a Question

Configure

Retriever
ColBERTRetriever
Select a retriever

Retriever Settings
Maximum number of retrieved documents
1 100 5

Corpus
Harry Potter
Select a corpus

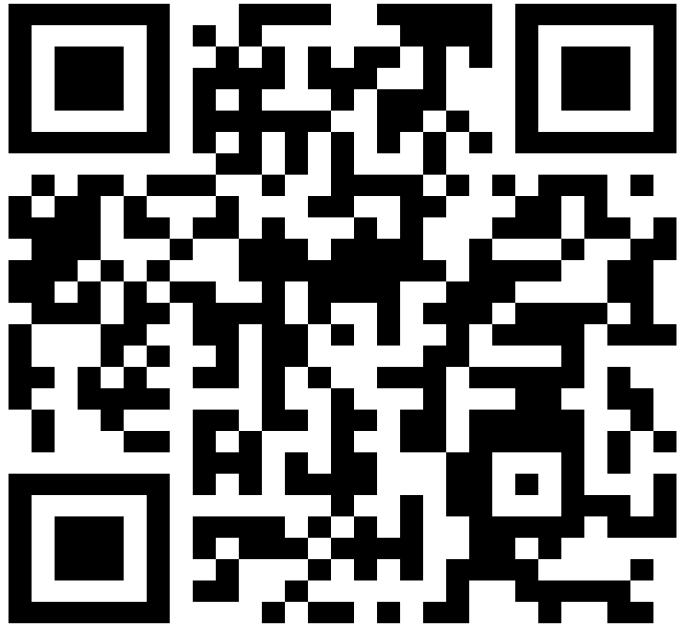
Reader
ExtractiveReader
Select a reader

Reader Settings
Model
PrimeQA/nq_tydi_sq1-reader-xlmr_large-2022

Maximum number of answers
1 5 3

Maximum answer length
2 2000 1

PrimeQA ... try it !!



<https://github.com/primeqa/primeqa>

In collaboration with:

 Stanford NLP	 University of Illinois
 University of Stuttgart	 University of Notre Dame
 Ohio State University	 Carnegie Mellon University
 University of Massachusetts	

Outline

- ✓ Motivation of OpenQA
- ✓ Retrievers: Sparse and Dense
- ✓ Recent Advances for OpenQA Efficient Neural Retrievers
- ✓ Recent Advances in Efficient Multilingual Retrieval
- ✓ Reproducibility in OpenQA: Hands-On Guide I
- ✓ Q&A: [15 min]

1st Half



Coffee Break.....

- ✓ Multilingual Readers
- ✓ Multi-modal Readers: Text, Table, Visual QA
- ✓ Large Language Models as Retrievers/Readers
- ✓ Reproducibility in OpenQA: Hands-on Guide II
- ✓ Pipelines, Service and Deployment
- ✓ Q&A: [15 min]

2nd Half

Thanks !



PrimeQA