DL for NLP, S02

Agenda

November, December 2021

1. Open domain question answering
2. Information extraction (NER, relation extraction, entity linking)
3. Summarization
4. Syntax parsing
5. Task-oriented dialogue systems
6. Multi-lingual applications
7. Black-box interpretation

Open Domain Question Answering

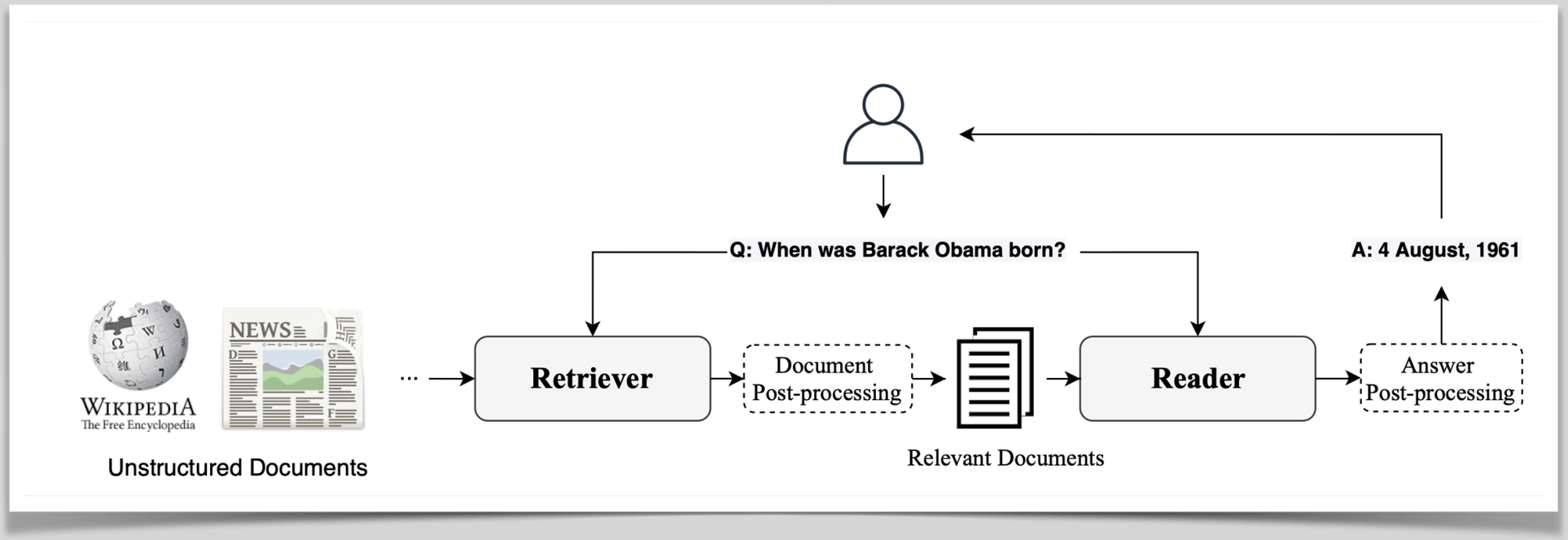
Problem definition

Open domain question answering (ODQA) is an important task in Natural Language Processing (NLP), which aims to answer a question in the form of natural language based on large-scale unstructured documents.

Zhu, F., Lei, W., Wang, C., Zheng, J., Poria, S., & Chua, T. S. (2021). Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

Retriever-Reader models

- ***Retriever*** is aimed at retrieving relevant documents w.r.t. a given question, which can be regarded as an IR system
- ***Reader*** aims at inferring the final answer from the received documents



Zhu, F., Lei, W., Wang, C., Zheng, J., Poria, S., & Chua, T. S. (2021). Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

Datasets

Stanford Question Answering Dataset (SQuAD)

Oxygen

The Stanford Question Answering Dataset

Oxygen is a chemical element with symbol O and atomic number 8. It is a member of the chalcogen group on the periodic table and is a highly reactive nonmetal and oxidizing agent that readily forms compounds (notably oxides) with most elements. By mass, oxygen is the third-most abundant element in the universe, after hydrogen and helium. At standard temperature and pressure, two atoms of the element bind to form dioxygen, a colorless and odorless diatomic gas with the formula O
2. Diatomic oxygen gas constitutes 20.8% of the Earth's atmosphere. However, monitoring of atmospheric oxygen levels show a global downward trend, because of fossil-fuel burning. Oxygen is the most abundant element by mass in the Earth's crust as part of oxide compounds such as silicon dioxide, making up almost half of the crust's mass.

What type of compounds does oxygen most commonly form?
Ground Truth Answers: oxides | oxides | oxides | oxide compounds | oxide

Compared to other elements, how abundant does oxygen rank?
Ground Truth Answers: third | third-most | third | third-most | third

Under normal conditions, what do two atoms of oxygen form?
Ground Truth Answers: dioxygen | diatomic gas | dioxygen | dioxygen | dioxygen

What element has an atomic symbol of O?
Ground Truth Answers: `<No Answer>`

What element has a symbol number of 8?
Ground Truth Answers: `<No Answer>`

What is the most abundant element in the universe followed by hydrogen and helium?
Ground Truth Answers: `<No Answer>`

https://rajpurkar.github.io/SQuAD-explorer/

Datasets

Natural Questions

Question:

how many episodes in season 2 breaking bad?

Short Answer:

13

Long Answer:

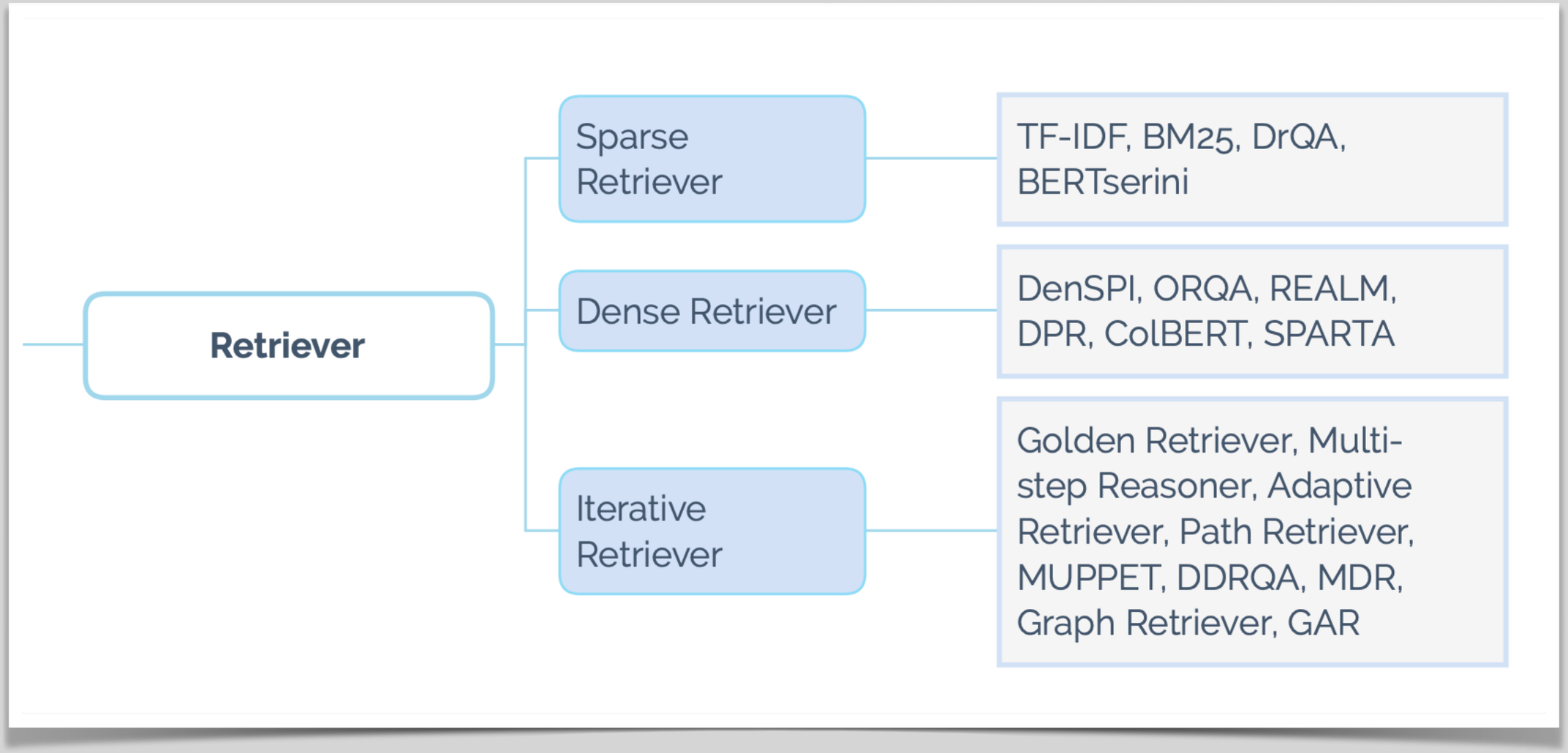
The second season of the American television drama series Breaking Bad premiered on March 8 , 2009 and concluded on May 31 , 2009 . It consisted of 13 episodes , each running approximately 47 minutes in length . AMC broadcast the second season on Sundays at 10 : 00 pm in the United States . The complete second season was released on Region 1 DVD and Region A Blu - ray on March 16 , 2010.

https://ai.google.com/research/NaturalQuestions/dataset

Retriever

Retriever

aims to get a lot of relevant documents



Sparse retriever

Recap: TF-IDF

- V – vocabulary, $d \in D$ – documents, q – question
- $\overrightarrow{v_q}$ – question vector, $\overrightarrow{v_d}$ – document vector, $dim(v) = |V|$
- Rank documents by cosine similarity and take top-N documents

$$v_d^i = \text{TF} \times \text{IDF} = \frac{\text{count}(w_i \in d_j)}{\sum_k \text{count}(w_k \in d_j)} \times \log \frac{|D|}{|d : w_i \in d|}$$

$$\text{sim(q,d)} = \cos(\overrightarrow{v_q}, \overrightarrow{v_d})$$

Sparse retriever

BM25

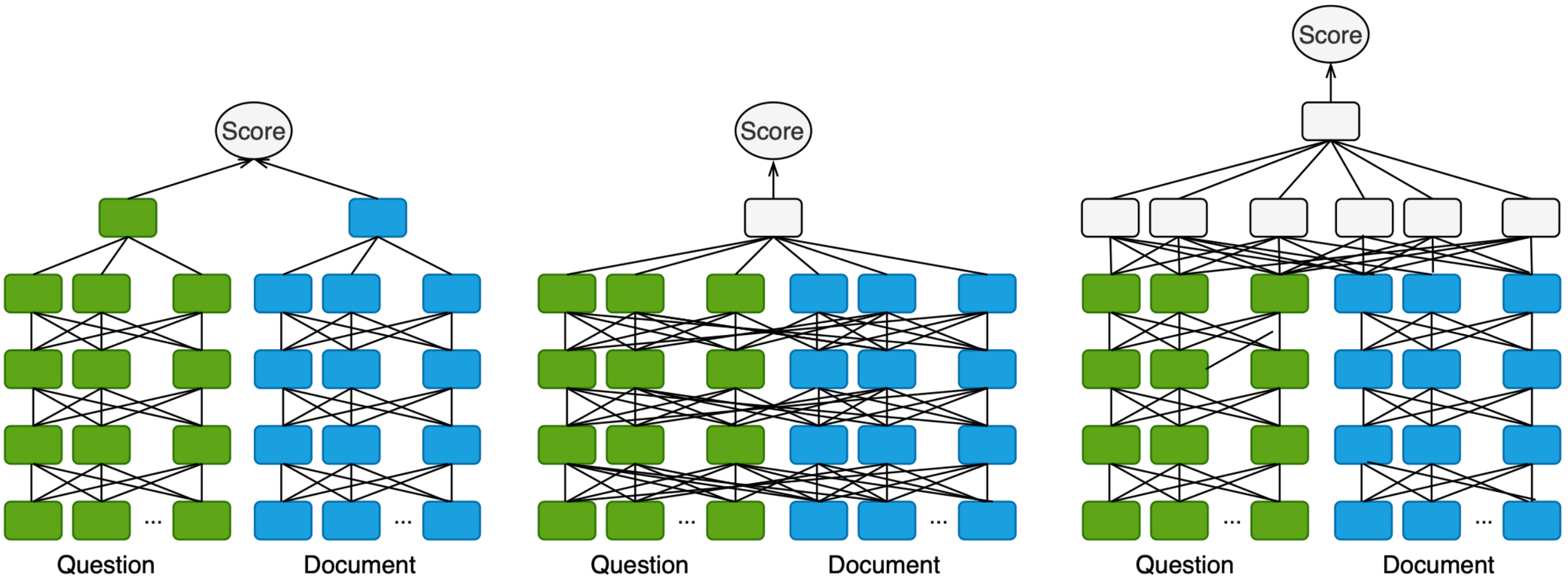
- q – question, consists of words $q_1, \ldots, q_n$
- Rank documents by score function and take top-N

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

$k_1 \in [1.2,2.0], b = 0.75$,avgdl is the average document length

$$\text{IDF}(q_i) = \ln\left(\frac{|D| - n(q_i) + 0.5}{n(q_i) + 0.5} + 1\right), n(q_i) = |d : w_i \in d|$$

Dense retriever

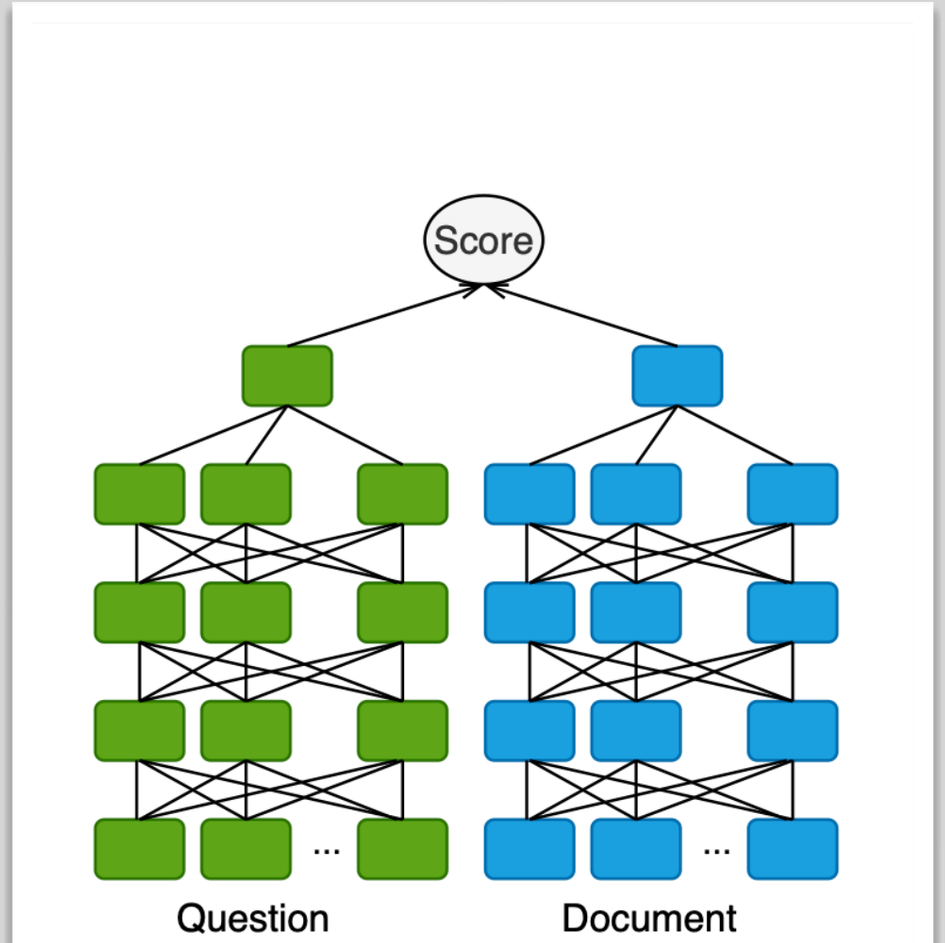


Zhu, F., Lei, W., Wang, C., Zheng, J., Poria, S., & Chua, T. S. (2021). Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

Dense retriever

Representation-based retriever (dual encoder)

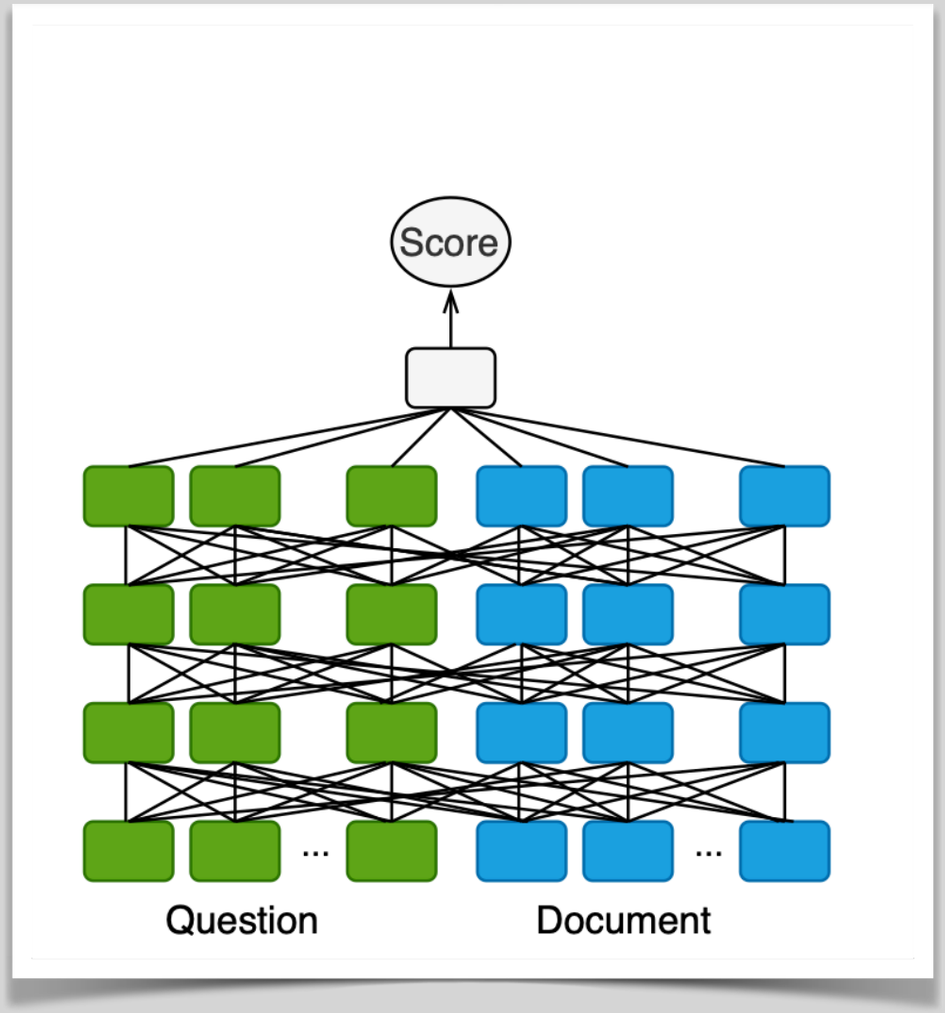
- Encode a question and a document **separately**
- Rank by cosine similarity between embeddings and take top-N documents
- Fast: there are tools for efficient computations (ElasticSearch)
- Sentence embedding models:
  - Universal sentence encoder (USE)
  - Language-Agnostic BERT Sentence Embedding (LaBSE)
  - SentenceBERT



Dense retriever

Interaction-based retriever (cross encoder)

- Encode a question and a document **together**:
  - `[CLS]` question `[SEP]` document
- Binary classification problem: is the document relevant to the question?
- Attach a classification head on top of `[CLS]` embedding
- Heavy computations, good performance



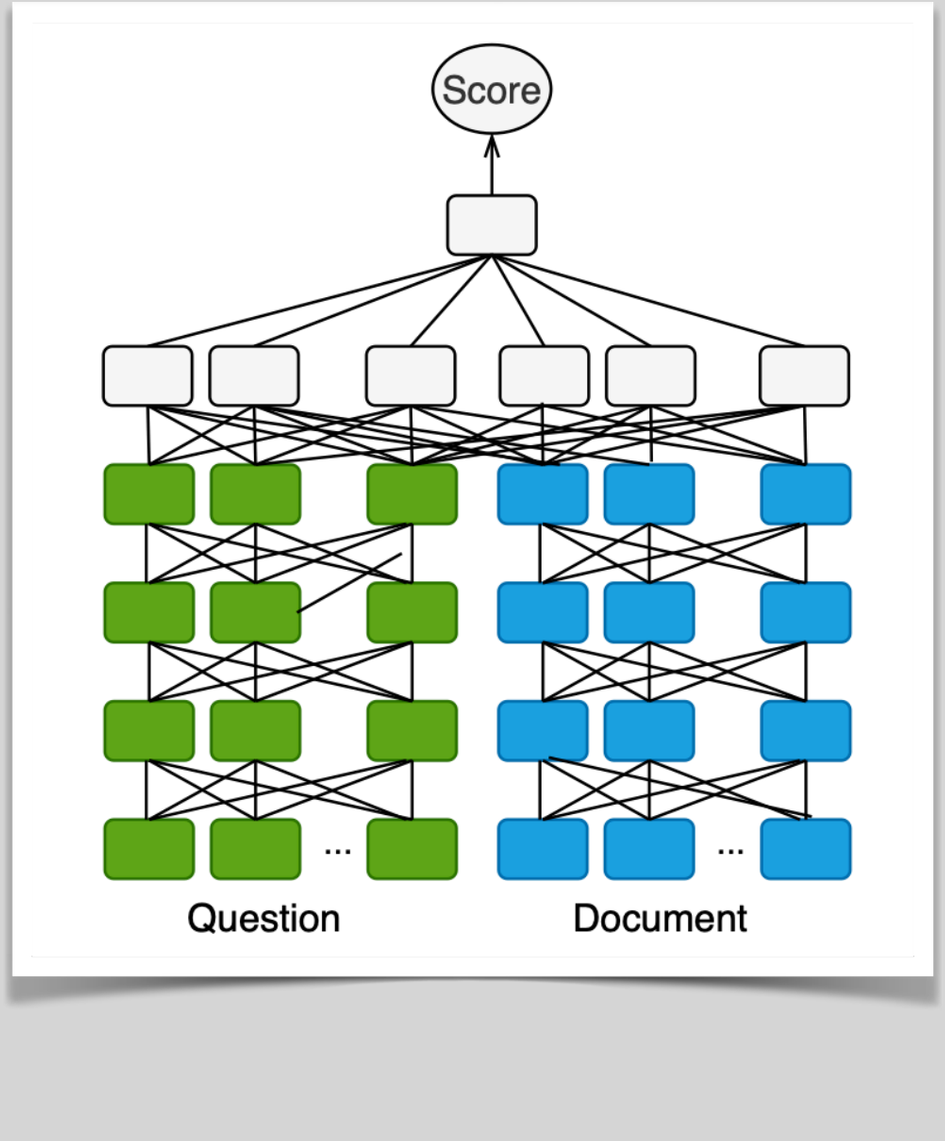
Dense retriever

Representation-interaction retriever

- Encode a question and a document separately with two BERT-like encoders
- ColBERT QA scoring accounts for soft matches between the question and the document:

$$\text{score}(q, d) = \sum_{i=1}^{N} \max_{j=1}^{m} v(q_i) \cdot v(d_j)^T,$$

v–token embeddings, $|q| = N, |d| = m$



Dense retriever

ColBERT training

- The model is trained to give higher score to positive documents (d^+), than to negative (d^-)
- Input: $< q, d^+, d^- >$
- Score each d^+ and d^-: score$(q, d^+) = 1$, score$(q, d^-) = 0$
- Binary classification task, train with cross-entropy loss

Retriever

Performance metrics

- MRR@k (mean reciprocal rank at k)

$$\text{MRR} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{\text{rank}_i}$$

Example [edit]

For example, suppose we have the following three sample queries for a system that tries to translate English words to their plurals. In each case, the system makes three guesses, with the first one being the one it thinks is most likely correct:

Query	Proposed Results	Correct response	Rank	Reciprocal rank
cat	catten, cati, **cats**	cats	3	1/3
torus	torii, **tori**, toruses	tori	2	1/2
virus	**viruses**, virii, viri	viruses	1	1

Given those three samples, we could calculate the mean reciprocal rank as (1/3 + 1/2 + 1)/3 = 11/18 or about 0.61.

https://en.wikipedia.org/wiki/Mean_reciprocal_rank

Retriever

Performance metrics

- MAP@k (mean average precision at k)

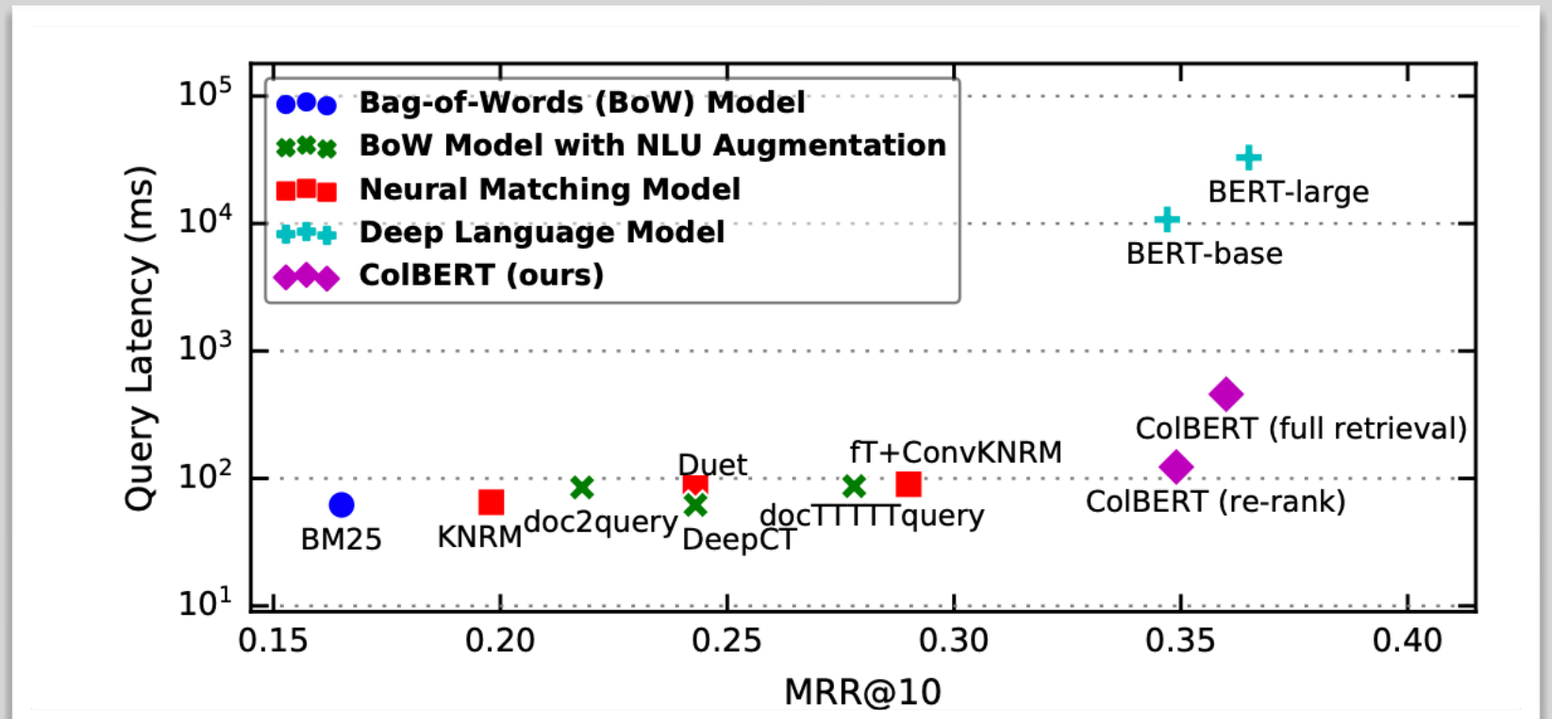
$$\text{MAP} = \frac{\sum_{q=1}^{Q} \text{AveP(q)}}{Q}, \text{AveP} = \frac{\sum_{j=1}^{k} P(j) \times \text{rel}(j)}{\text{number of relevant documents}}$$

- Other metrics, adopted from IR

https://en.wikipedia.org/wiki/Evaluation_measures_(information_retrieval)#Mean_average_precision

Retriever

Effectiveness (MRR@10) versus Mean Query Latency (log-scale) for a number of representative retrievers

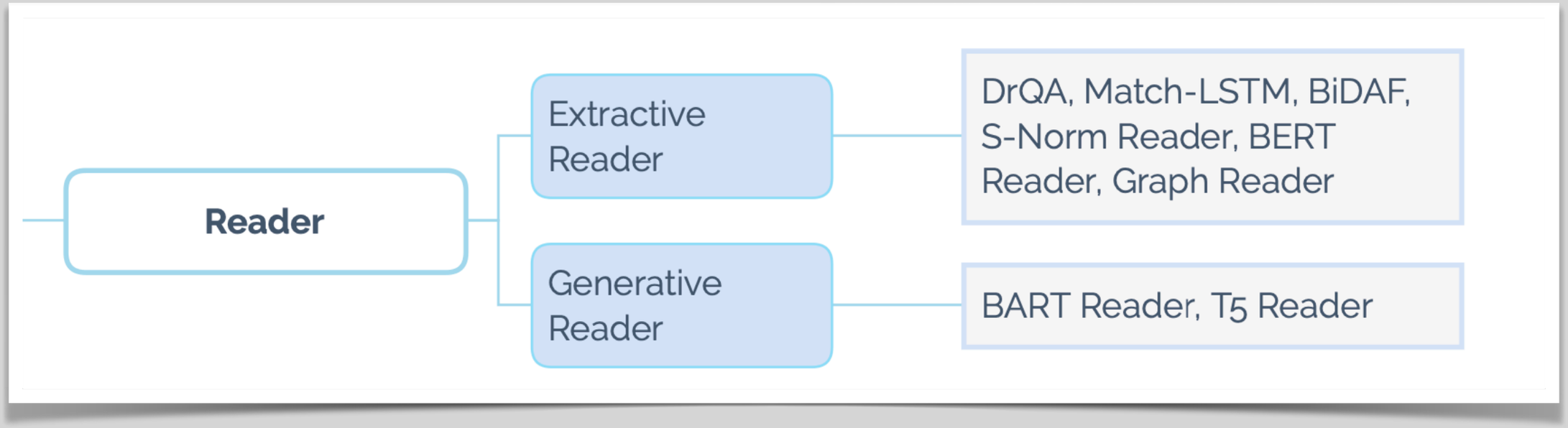


Khattab, Omar, and Matei Zaharia. "Colbert: Efficient and effective passage search via contextualized late interaction over bert." In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39-48. 2020.

Reader

Reader

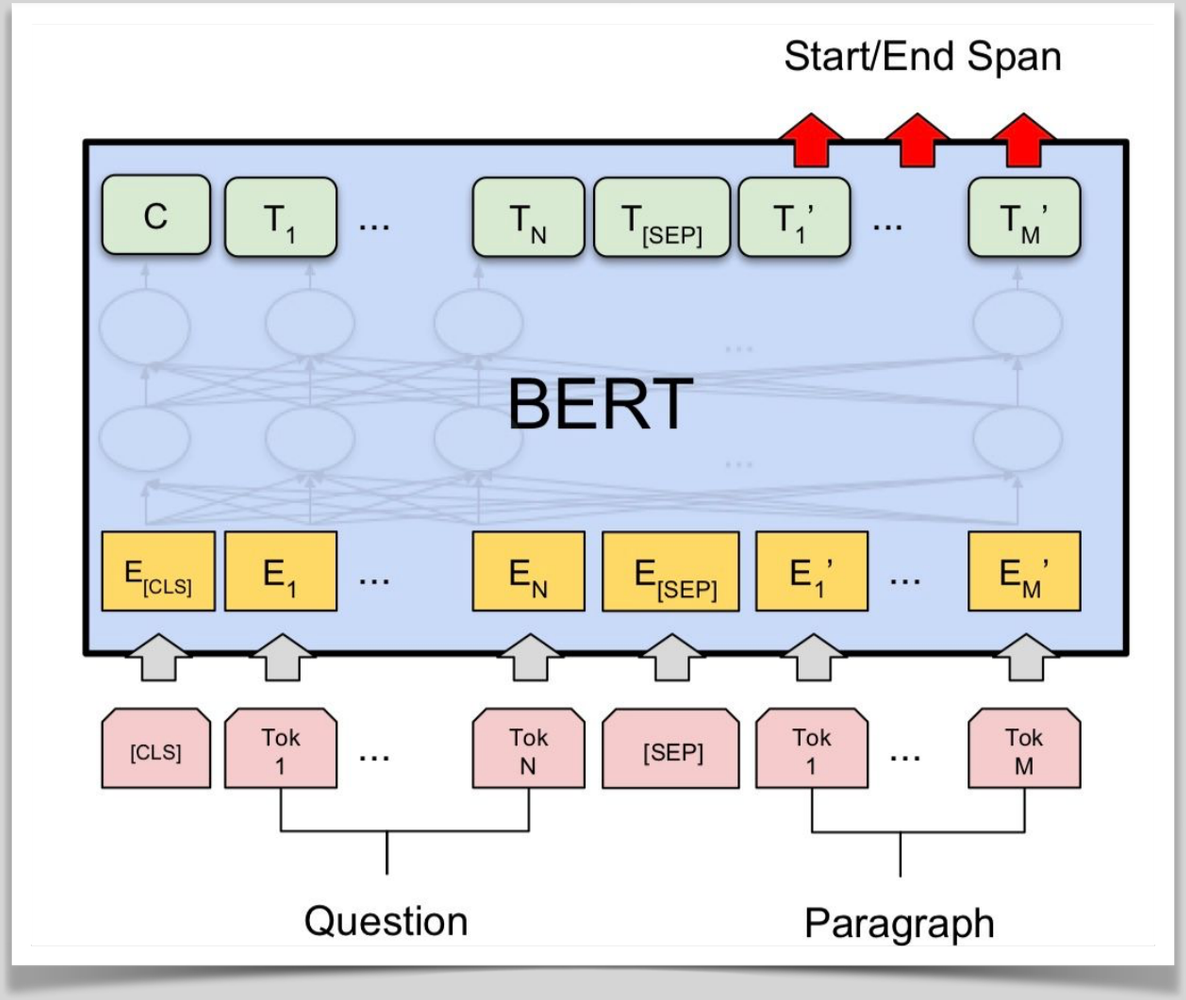
Outputs an answer span from the retrieved documents



Extractive reader

BERT for span extraction (aka machine reading comprehension)

- Introduce a start vector $S \in \mathbb{R}^H$ and an end vector $E \in \mathbb{R}^H$
- $P(w_i, \text{start}) = \dfrac{e^{S \cdot T_i}}{\sum_j e^{S \cdot T_j}}$
- $P(w_i, \text{end}) = \dfrac{e^{E \cdot T_i}}{\sum_j e^{E \cdot T_j}}$
- The score of a candidate span: $S \cdot T_i + E \cdot T_j, i \leq j$
- The training objective: the sum of the log-likelihoods of the correct start and end positions.



Generative reader

GPT-2 for answer generation

- Training:
  - Input: <|startoftext|>\n`[CONTEXT]`: … \n`[QUESTION]`: … \n`[ANSWER]`: …\n<|endoftext|>\n
- Inference:
  - Input: <|startoftext|>\n`[CONTEXT]`: … \n`[QUESTION]`: … \n`[ANSWER]`:
- Fine-tune with cross-entropy loss
- The current generation results often suffer from incoherenc, syntax or common sense errors

Reader

Performance metrics

- **Exact match (EM)**: the percentage of predictions that match any one of the ground truth answers exactly

- **(Macro-averaged) F1 score**: the average overlap between the prediction and ground truth answer. We treat the prediction and ground truth as bags of tokens, and compute their F1. We take the maximum F1 over all of the ground truth answers for a given question, and then average over all of the questions.

Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev, and Percy Liang. "SQuAD: 100,000+ Questions for Machine Comprehension of Text." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383-2392. 2016.

Efficient QA

https://efficientqa.github.io

QA Dataset Explosion

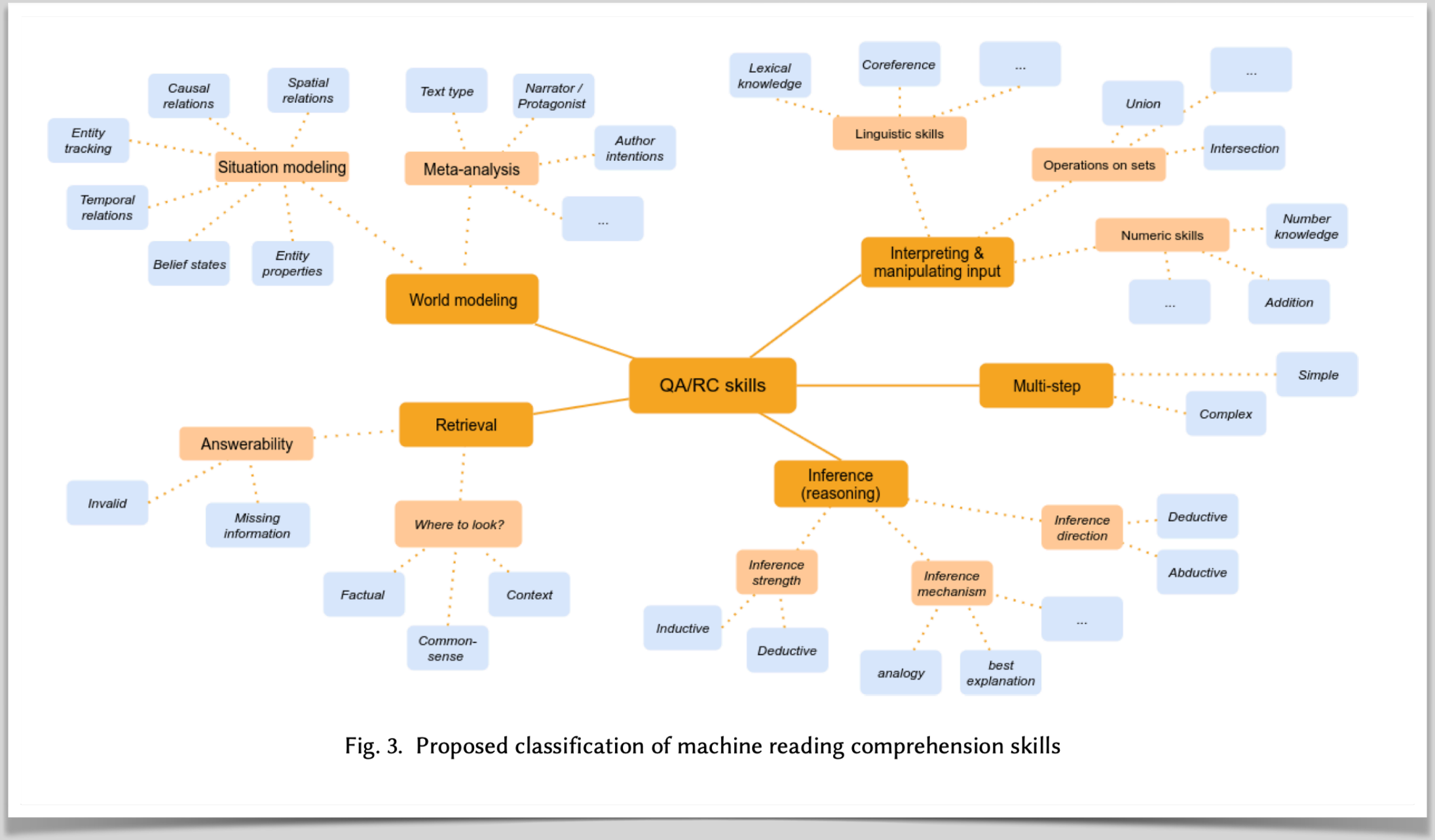


Fig. 3. Proposed classification of machine reading comprehension skills

Rogers, Anna, Matt Gardner, and Isabelle Augenstein. "QA dataset explosion: A taxonomy of NLP resources for question answering and reading comprehension." *arXiv preprint arXiv:2107.12708* (2021).

QA dataset taxonomy

I. Information-seeking VS probing questions

- Information seeking: natural questions, collected from social media, search engines, etc
- Probing questions: questions, crafted by experts or trained crowd-workers

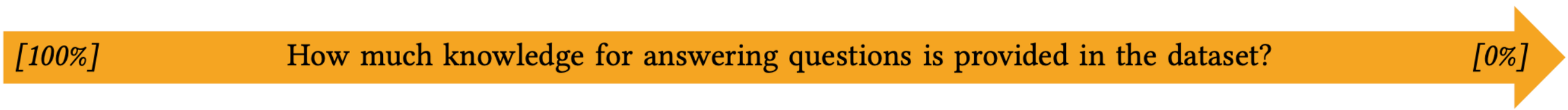
QA datasets taxonomy

IIa. Format

Fig. 1. When is question answering a task, and when is it a format?

QA datasets taxonomy

IIb. Evidence format



Single source
one document needs to be considered for answering the question

Multiple sources
evidence is provided, but it has to be ranked and found

Partial source
some evidence is provided, but it has to be combined with missing knowledge

No sources
the model has to retrieve evidence or have it memorized

Fig. 2. Sources of knowledge for answering the questions.

QA datasets taxonomy

IIc. Question and answer format

Evidence	Format	Question	Answer(s)	Example datasets
Einstein was born in 1879.	Extractive	When was Einstein born?	1879 (token 5)	SQuAD [210], NewsQA [256]
	Multi-choice	When was Einstein born?	(a) 1879, (b) 1880	RACE [138]
	Categorical	Was Einstein born in 1880?	No	BoolQ [53]
	Freeform	When was Einstein born?	1879 (generated)	MS MARCO [16], CoQA [213]

QA datasets taxonomy

III. Modality

- Unstructured text
- Semi-structured sources (tabels)
- Structured knowledge
- Images
- Audio
- Video

IV. Languages

- Monolingual VS multilingual sources

QA datasets taxonomy

V. Domains

- Encyclopedia
- Fiction
- Academic tests
- News
- E-commerce
- Expert materials
- Social media

PAQ: 65 Million Probably-Asked Questions and What You Can Do With Them

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, Sebastian Riedel; PAQ: 65 Million Probably-Asked Questions and What You Can Do With Them. *Transactions of the Association for Computational Linguistics* 2021; 9 1098–1115

PAQ: 65 Million Probably-Asked Questions and What You Can Do With Them

Generating Question-Answer Pairs

1. A *passage selection* model $p(c)$, to identify passages which humans are likely to ask questions about

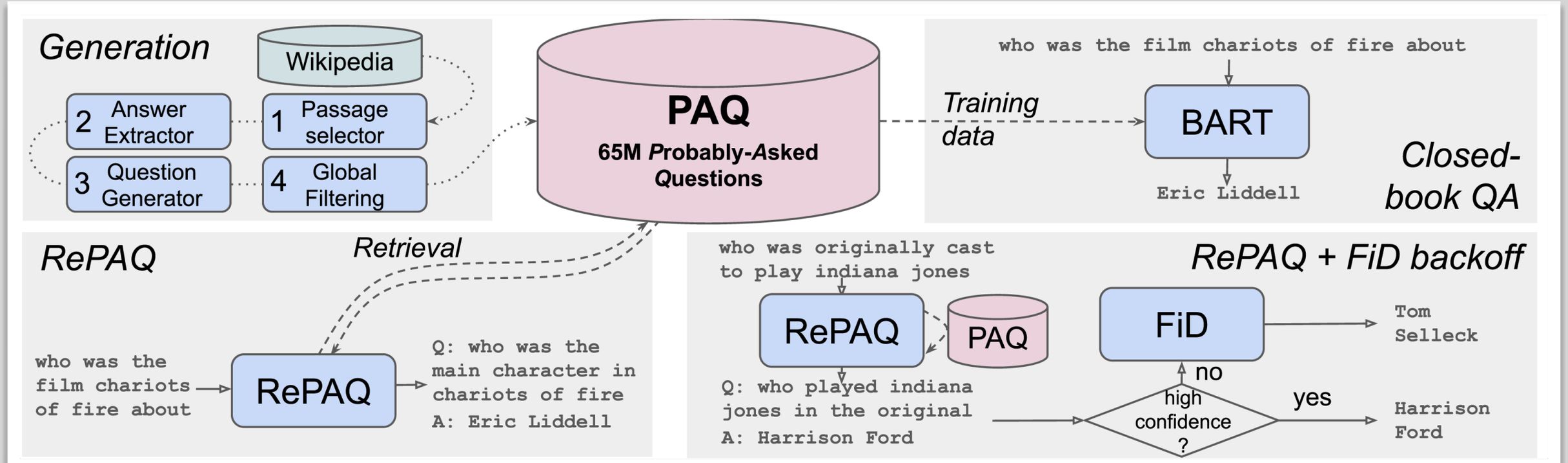
2. An *answer extraction* model $p(a \mid c)$, for identifying spans in a passage that are more likely to be answers to a question.

3. A *question generator* $p_q(q \mid a, c)$ that, given a passage and an answer, generates a question.

4. A *filtering* QA model $p_f(a \mid q, C)$that generates an answer for a given question. If an answer generated by p_f does not match the answer a question was generated from, the question is discarded. This ensures generated questions are *consistent*

PAQ: 65 Million Probably-Asked Questions and What You Can Do With Them

Generating Question-Answer Pairs



Knowledge base QA