自然語言處理與應用
Natural Language Processing and Applications

Retrieval-Augmented Generation (RAG)

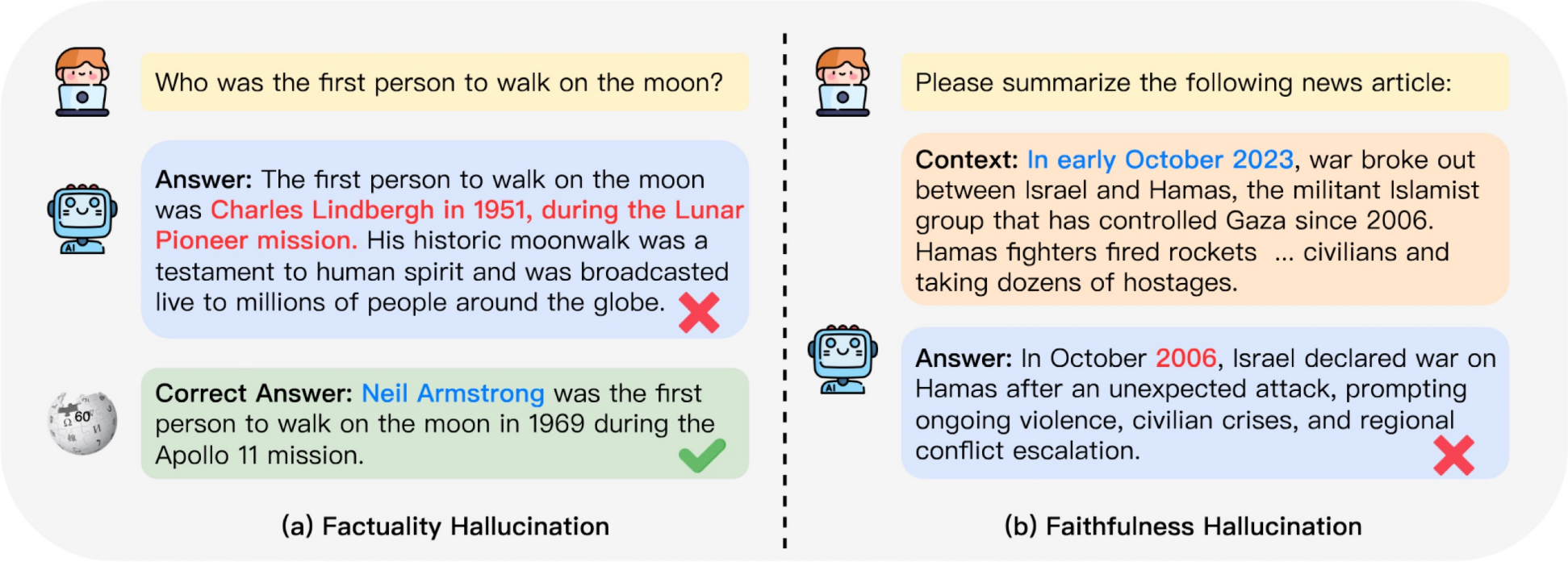
Instructor: 林英嘉 (Ying-Jia Lin)
2025/05/19

Course GitHub

Slido # NLP_0519

Hallucination of LLM

- It is discovered that NLG models often generate text that is nonsensical, or unfaithful to the provided input. Such undesirable generation is referred to Hallucination (Ji et al., 2023).



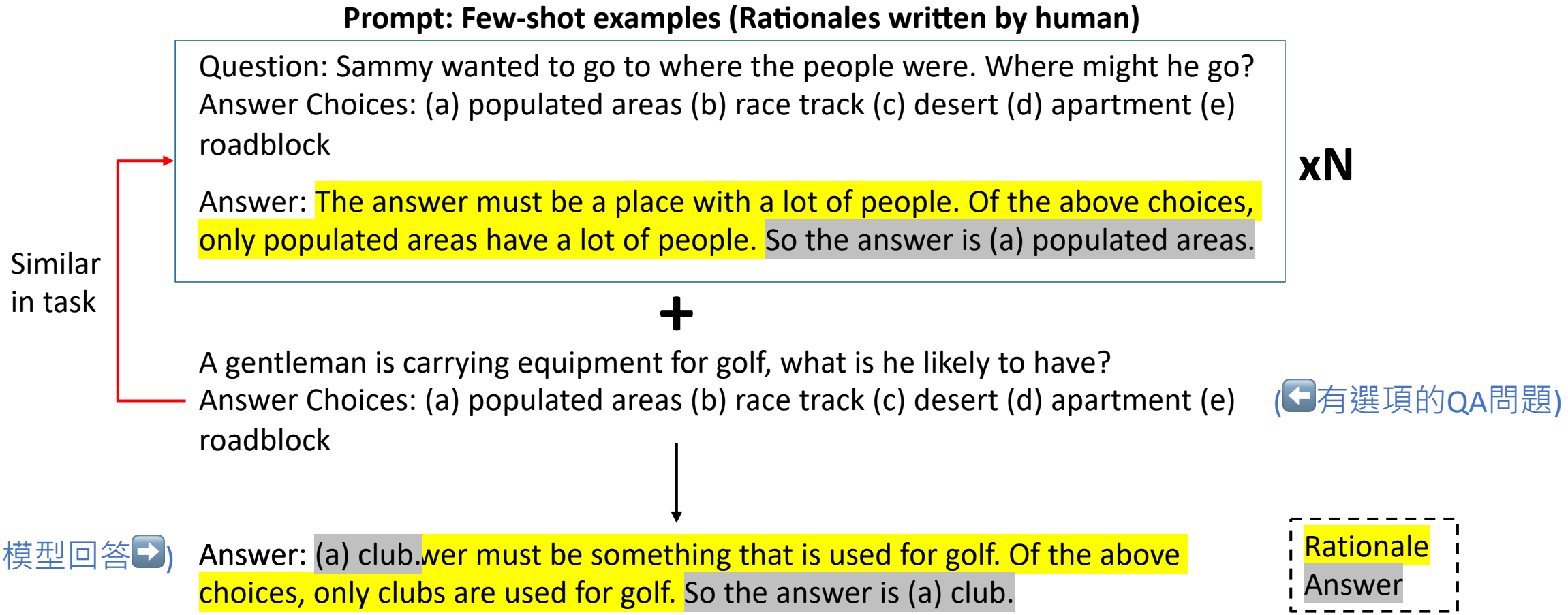
Ji et al. "Survey of hallucination in natural language generation." ACM Computing Surveys 55.12 (2023): 1-38.
Figure source: Munkhdalai, Tsendsuren, Manaal Faruqui, and Siddharth Gopal. "Leave no context behind: Efficient infinite context transformers with infini-attention." arXiv preprint arXiv:2404.07143 (2024).

Solutions to Mitigating Hallucinations

- Chain-of-Thought Prompting (CoT)
- Retrieval-Augmented Generation (RAG)
- …

Chain-of-Thought Prompting (CoT)

- Wei et al., Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NeurIPS 2022. By Google Brain.



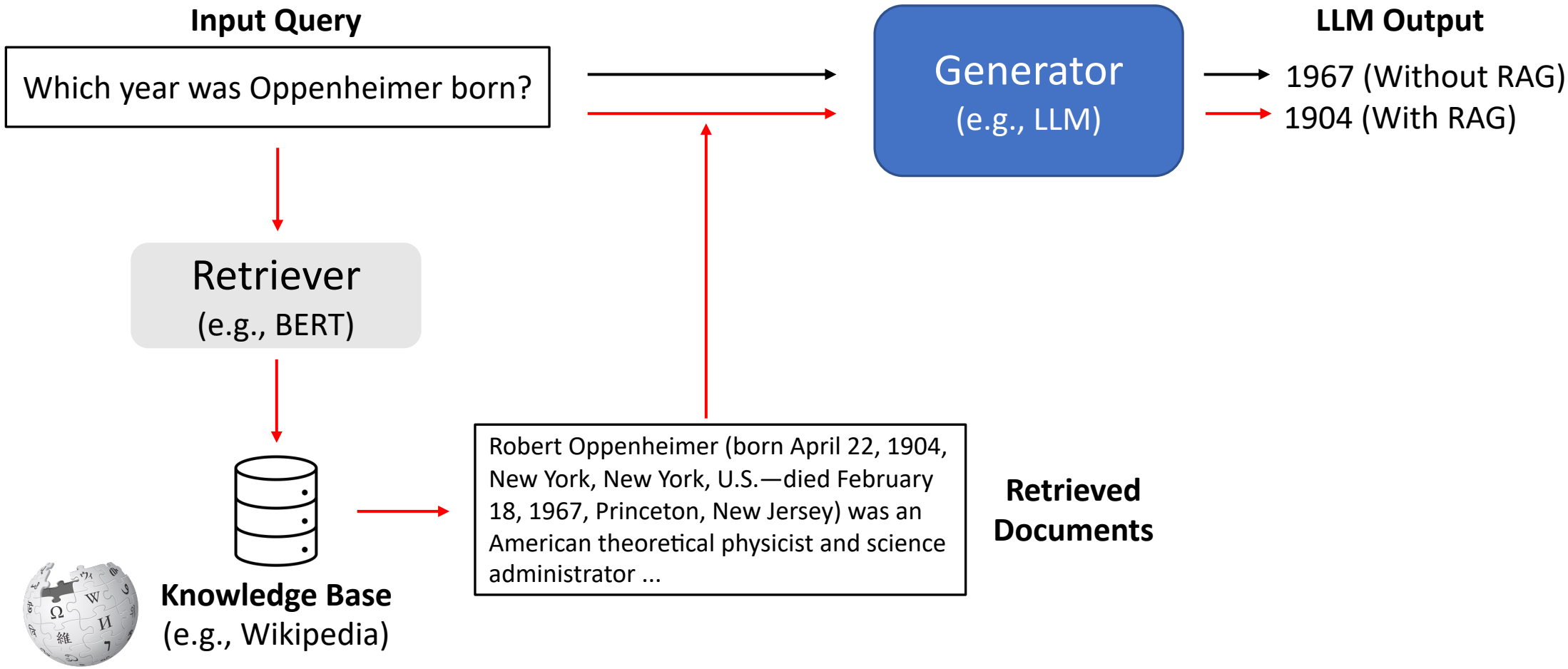
Retrieval-Augmented Generation (RAG)

Information Retrieval

- Retrieval: get relevant information from a pool (like a search engine)

- Retrieval-Augmented Generation (RAG): Perform generation with additional retrieved information

Retrieval-Augmented Generation (RAG) 框架



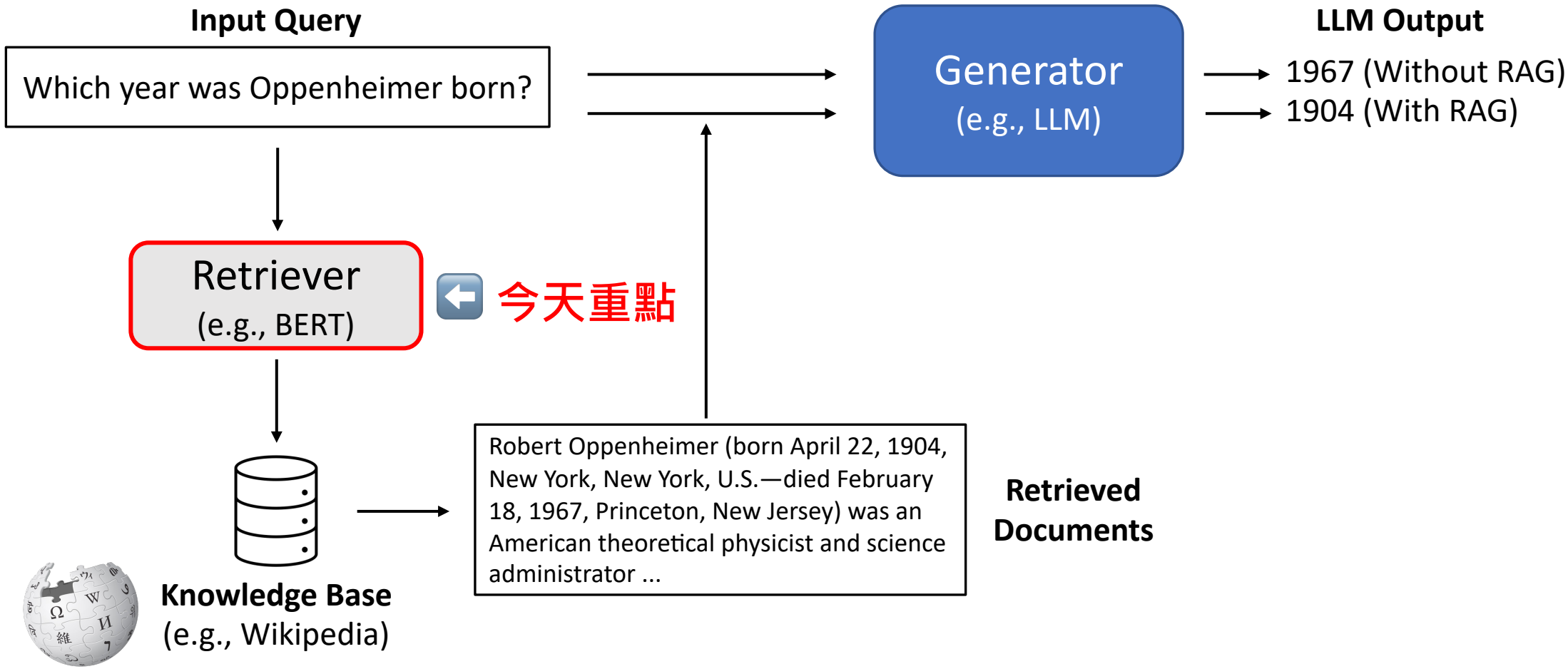
Gao, Yunfan, et al. "Retrieval-augmented generation for large language models: A survey." *arXiv preprint arXiv:2312.10997* (2023).

Why do we need RAG?

- LLMs have profound parameterized knowledge that makes them useful in responding to general prompts.
- However, LLMs are error-prone due to lack of domain knowledge or outdated information (RAG 可協助更新知識).
- Standalone LLMs do not serve users who want a deeper dive into a current or more specific topic (RAG 可協助提供特定領域知識).

https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/

Retrieval-Augmented Generation (RAG)



Gao, Yunfan, et al. "Retrieval-augmented generation for large language models: A survey." *arXiv preprint arXiv:2312.10997* (2023).

Task: Open-domain question answering (ODQA)

- Given a question x such as "What is the currency of the UK?", a model must output the correct answer string y, "pound".
- "Open" refers to the fact that the model does not receive a pre-identified document that contains the answer (ODQA 任務沒有 context).
- ODQA is like Reading comprehension (RC) tasks, such as SQuAD, but no relevant articles provided.

(Example of SQuAD, 有 context)

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

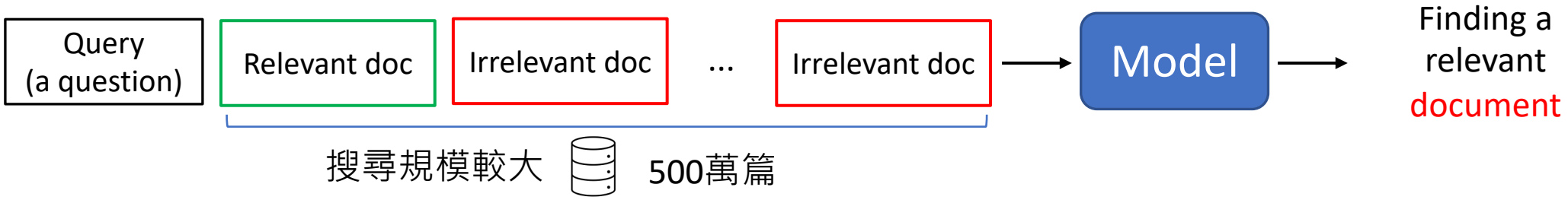
What causes precipitation to fall?
gravity

Guu, Kelvin, et al. "Retrieval augmented language model pre-training." ICML 2020.
Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. EMNLP 2016.

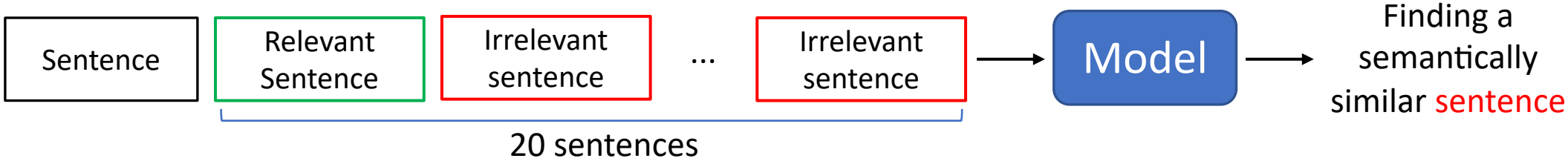
Retrievers for RAG

- Task-oriented training:
  - Retrieval for open-domain question answering (ODQA)
  - Sentence embeddings for semantic similarity tasks

Retrieval for open-domain question answering



Sentence embeddings for semantic similarity tasks



- Both approaches are suitable for retrieval (depends on your task for an LLM).

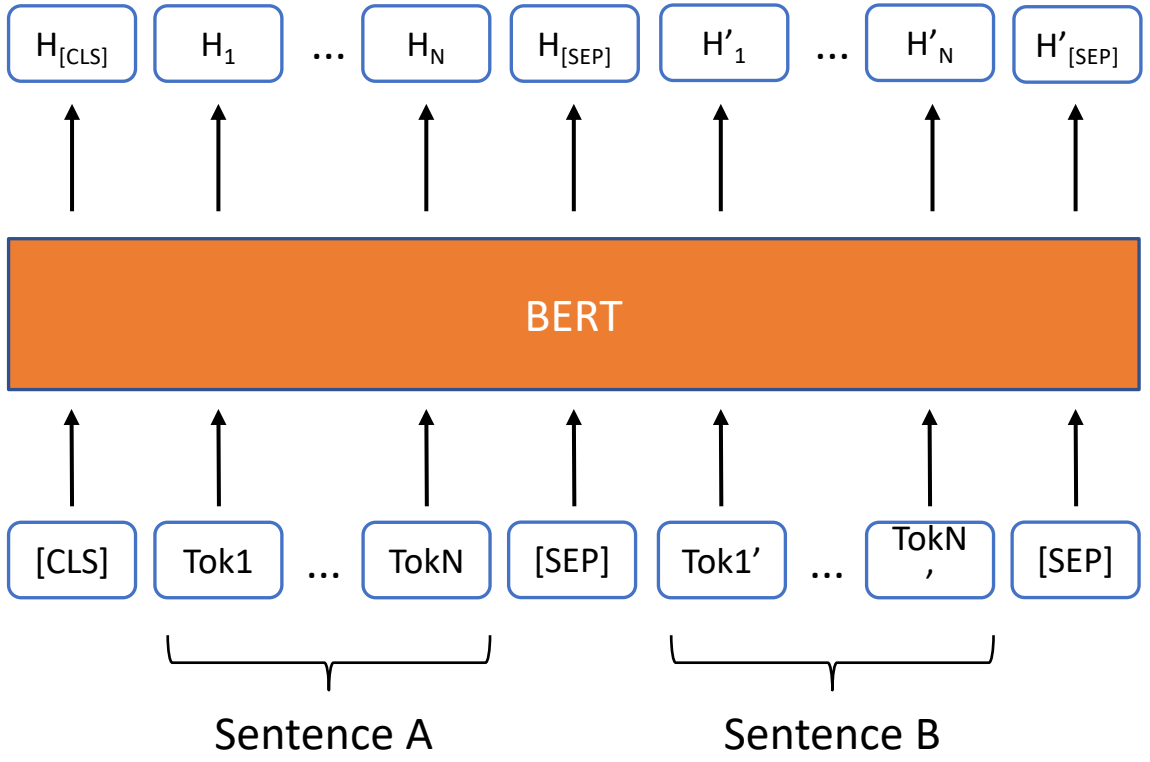
Vectors (embeddings) in this semester

	Word Embeddings	**One-hot Encodings / PMI / LSA (week 2)** **Word2vec / GloVe (week 3)**
Retriever	**Document Embeddings**	**Bag-of-words / TF-IDF (week 2)** **DPR / … (week 14)**
Retriever	**Sentence Embeddings**	**BERT (week 7)** **Sentence-BERT / SimCSE … (week 14)**

Sentence embeddings for semantic similarity tasks

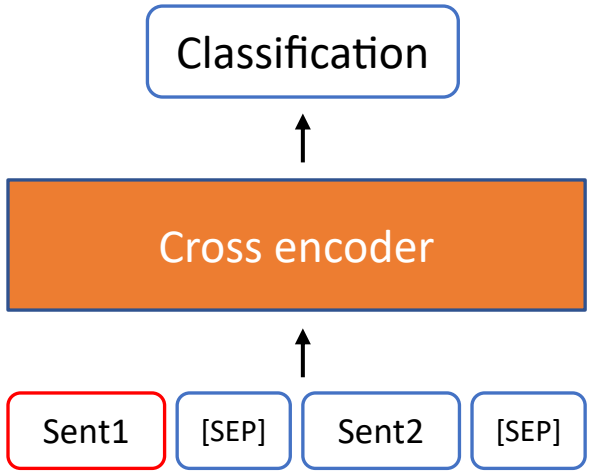
BERT as a Cross Encoder

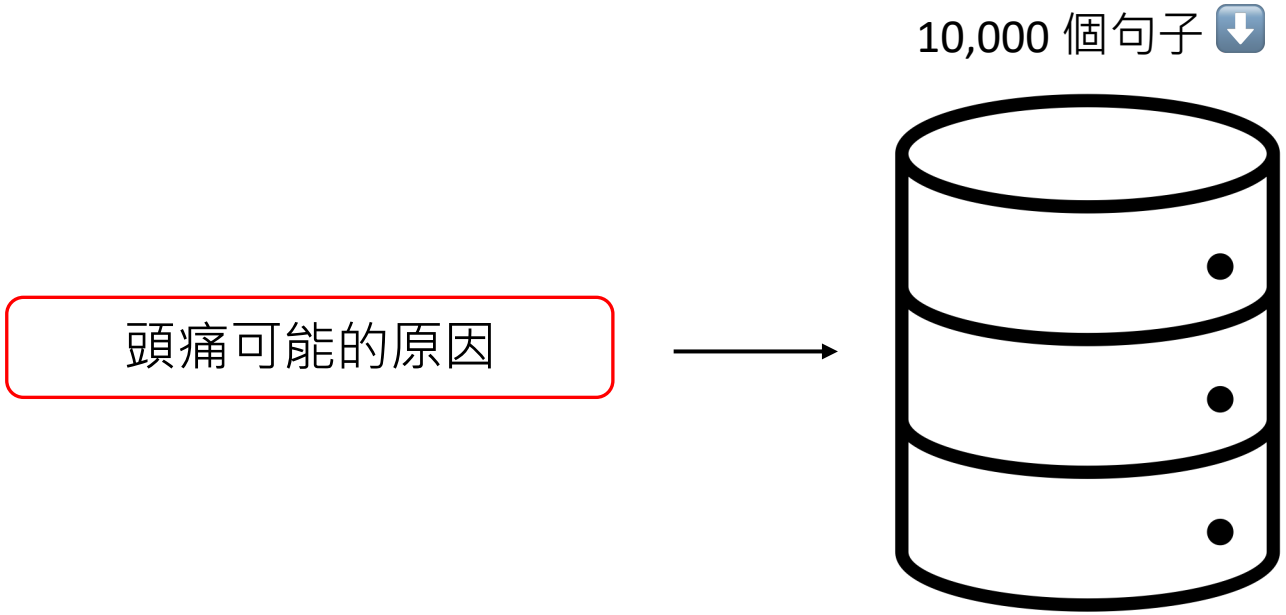
- For a cross encoder, representations of two input sentences are attended with each other.
- The hidden state of [CLS] represents the relationship between the two input sentences.



Devlin, J., Chang, M.W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL 2019.

Cross-encoder 速度問題

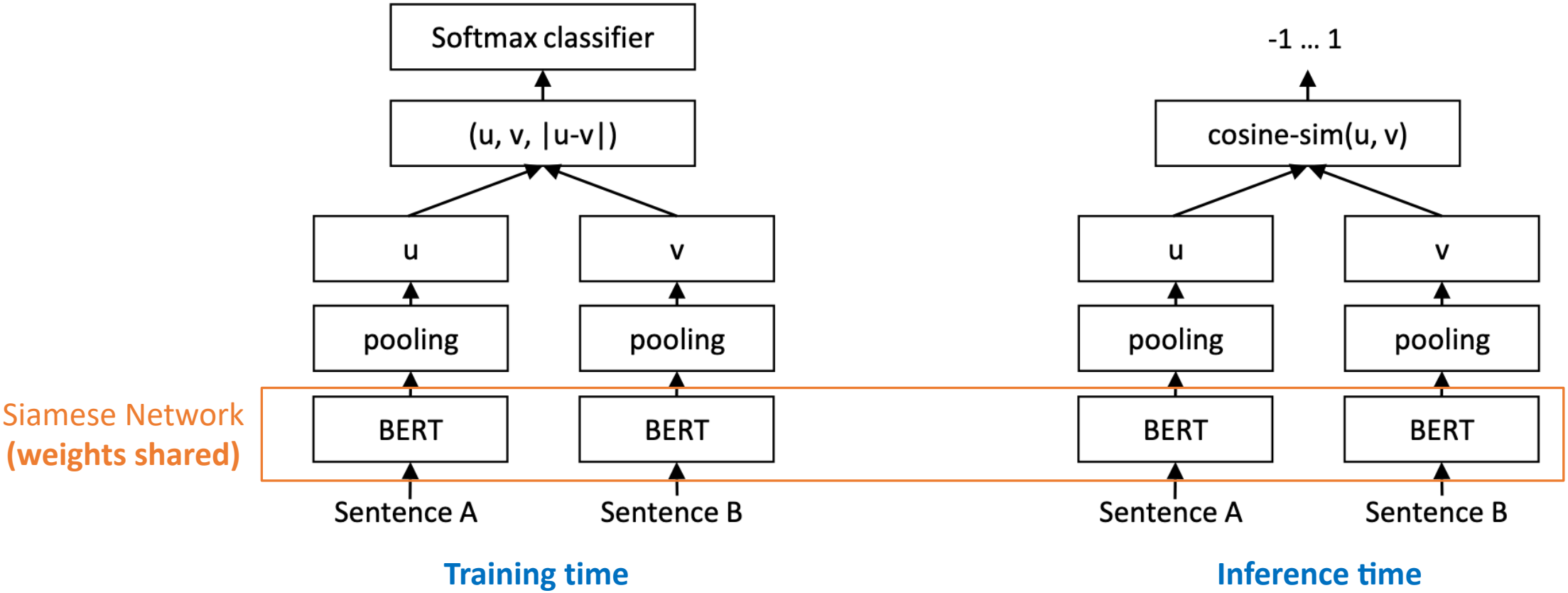




- For 10,000 sentence pairs:
  - Cross encoders: n·(n−1)/2 = 49,995,000 inference times

Sentence-BERT

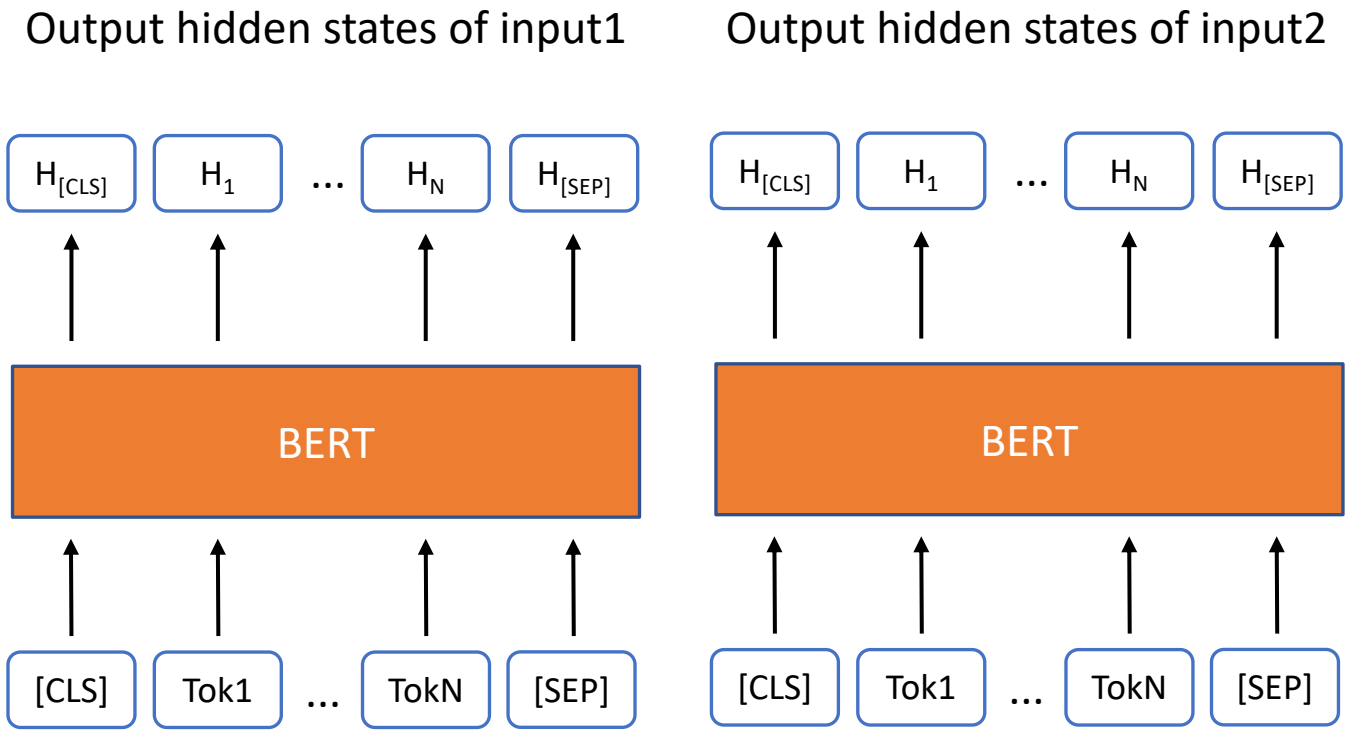
```
# Pseudo code for Bi-encoder
query_vector = encoder(query)          # [0.1, 0.2, 0.3]
document_vector = encoder(document)    # [0.2, 0.2, 0.4]
similarity = cosine_similarity(query_vector, document_vector)
```



Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *(EMNLP-IJCNLP 2019)*

Why does Sentence-BERT need pooling?

- BERT produces embeddings (hidden states from the final layer) for each token.
- We need a single fixed-size vector for the entire sentence.



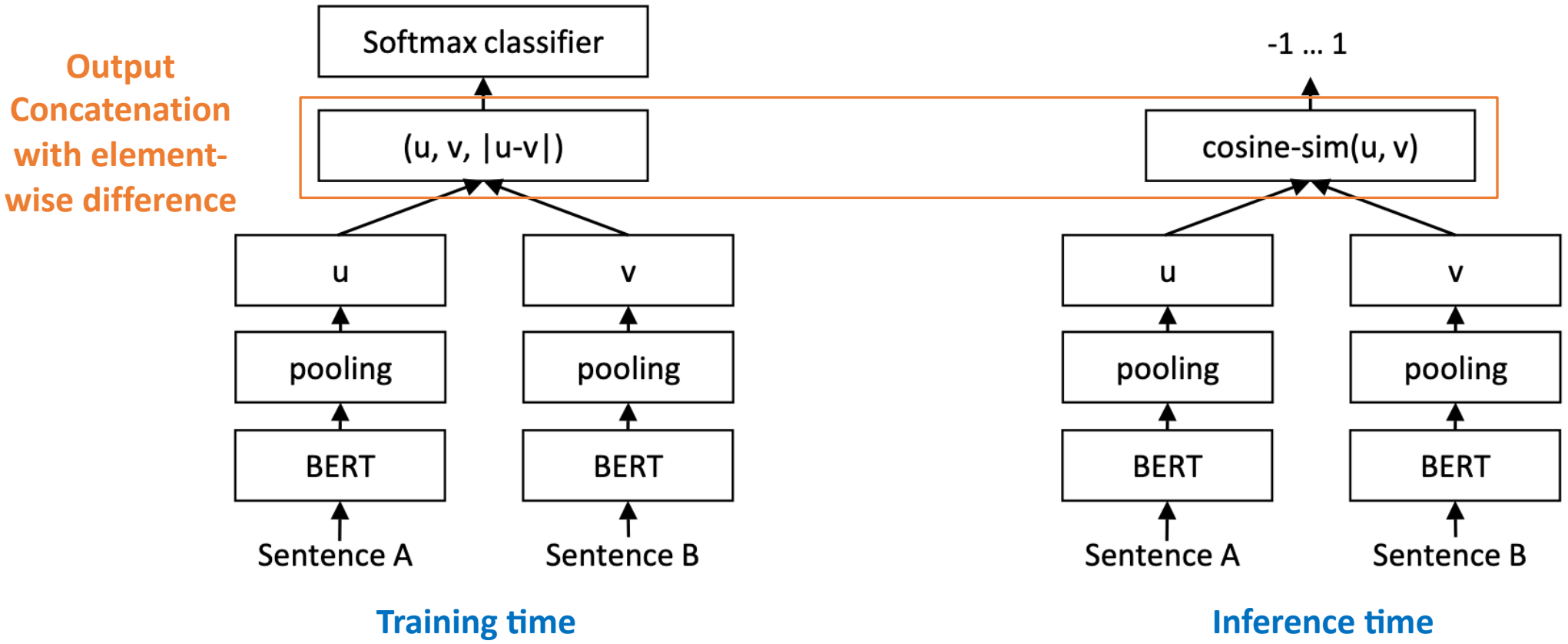
Pooling of Sentence-BERT

We need a single fixed-size vector for the entire sentence.

- **CLS: Use the [CLS] token**
  - This is the default setting in original BERT.
- **MEAN: the mean of all output vectors**
  - Averages all token embeddings.
- **MAX: max-over-time of the output vectors**
  - Takes maximum value across each dimension.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *(EMNLP-IJCNLP 2019)*

Sentence-BERT (Bi-encoder)



Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *(EMNLP-IJCNLP 2019)*

Performance comparison of Pooling and Concatenation

- Indeed, [CLS] token can directly be used for representing the entire sentence.
- But pooling may bring better performance.

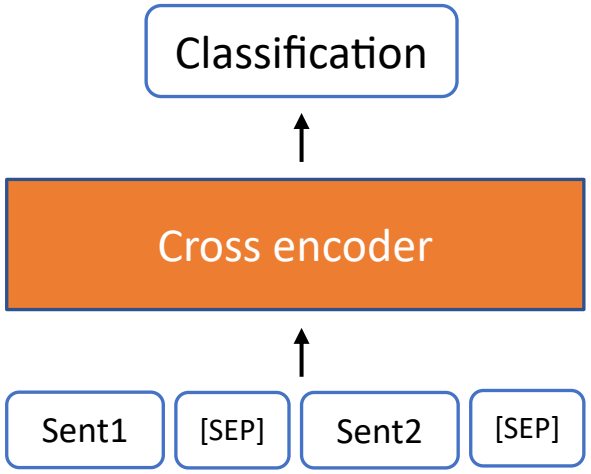
- **Experiment** shows using element-wise difference is better than the other settings.
- Note that the concatenation mode is only used for training.

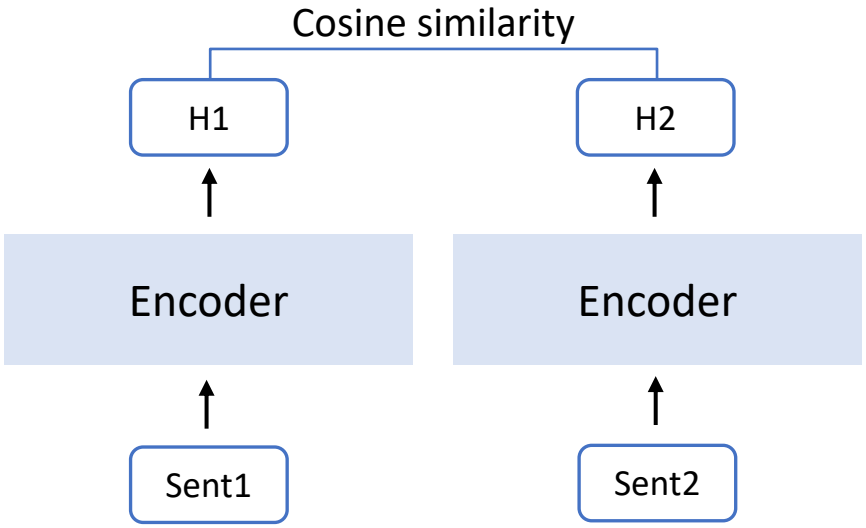
	NLI	STSb
Pooling Strategy		
MEAN	**80.78**	**87.44**
MAX	79.07	69.92
CLS	79.80	86.62
Concatenation		
(u, v)	66.04	-
$(\|u - v\|)$	69.78	-
$(u * v)$	70.54	-
$(\|u - v\|, u * v)$	78.37	-
$(u, v, u * v)$	77.44	-
$(u, v, \|u - v\|)$	**80.78**	-
$(u, v, \|u - v\|, u * v)$	80.44	-

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *(EMNLP-IJCNLP 2019)*

Computation time for Bi-encoders and Cross encoders

- For 10,000 sentence pairs:
  - Cross encoders: n·(n−1)/2 = 49,995,000 inference times
  - Bi-encoders (e.g., Sentence-BERT): 10,000 * 2 inference times (can be parallel) with cosine similarity calculation





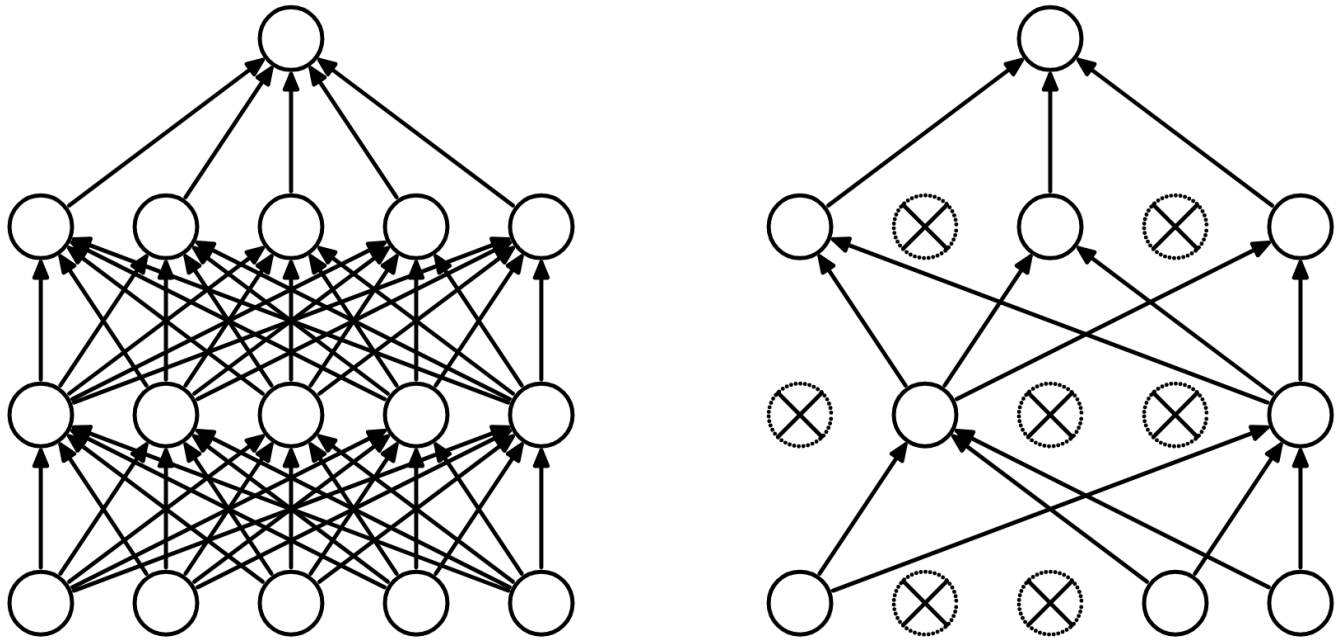
SimCSE (Bi-encoder)

- SimCSE: a simple contrastive sentence embedding framework
- Both unsupervised and supervised training approaches were proposed in SimCSE:
  - **Unsupervised** training of SimCSE
    - Relying on **Dropout**
  - **Supervised** training of SimCSE
    - Relying on **labels in a dataset**

Gao, Tianyu, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings." EMNLP 2021.

Dropout

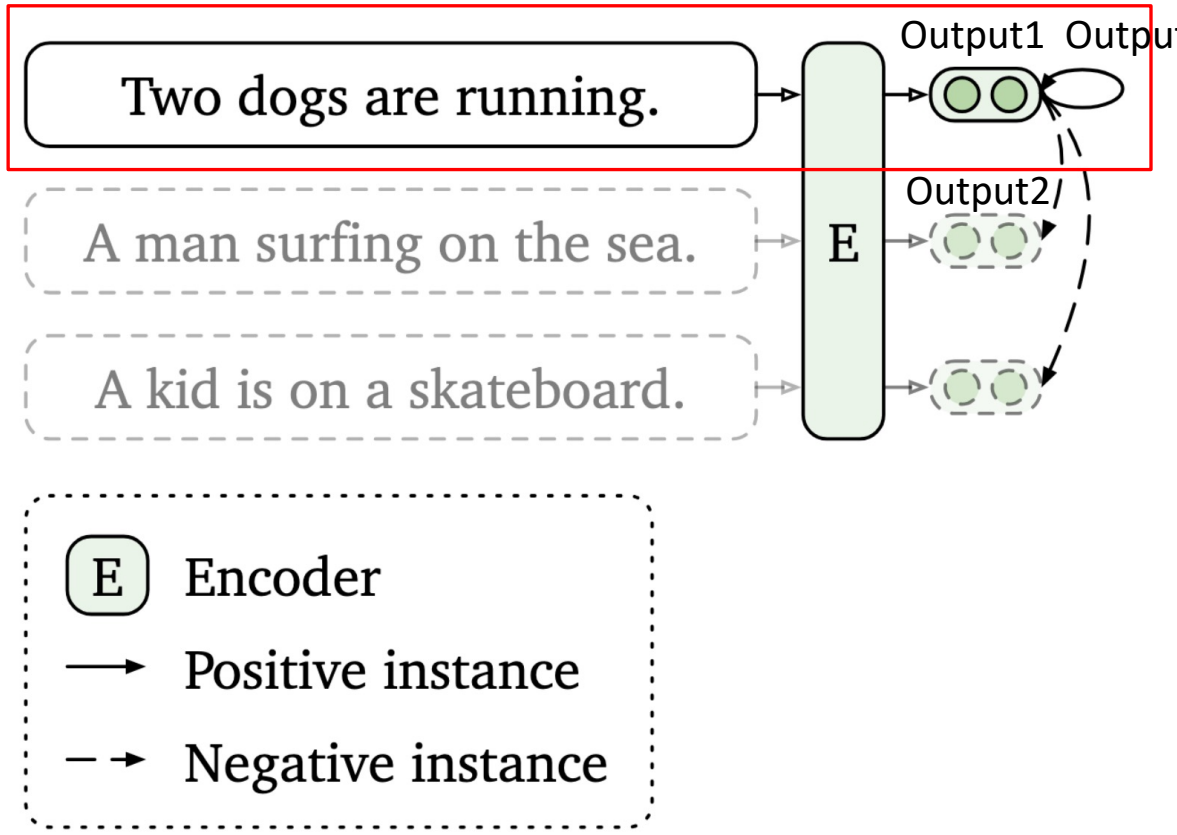
- Dropout randomly drop units (along with their connections) from the neural network during training. This approach usually brings regularization and reduces overfitting.



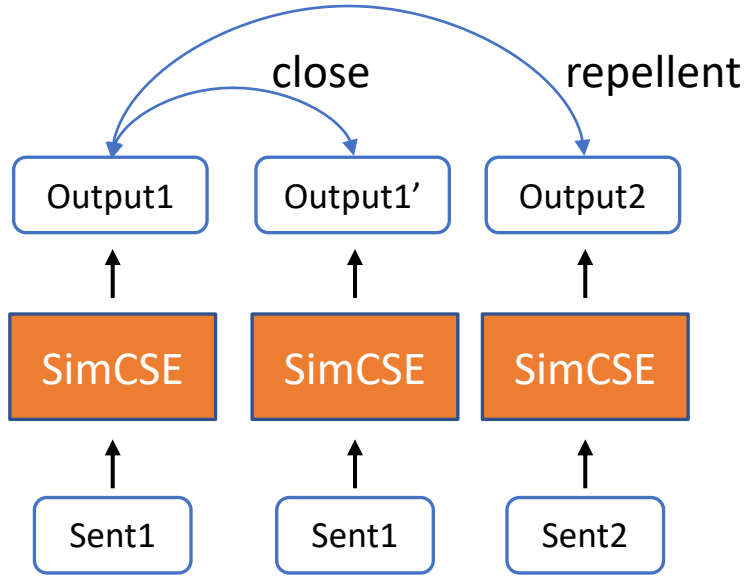
(a) Standard Neural Net (b) After applying dropout.

Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." The journal of machine learning research 15.1 (2014): 1929-1958.

Unsupervised training of SimCSE



Input a sentence twice with a dropout mask
(Dropout as data augmentation)

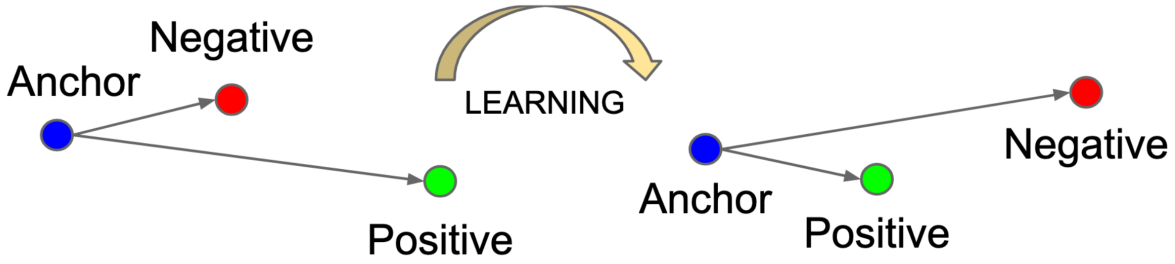


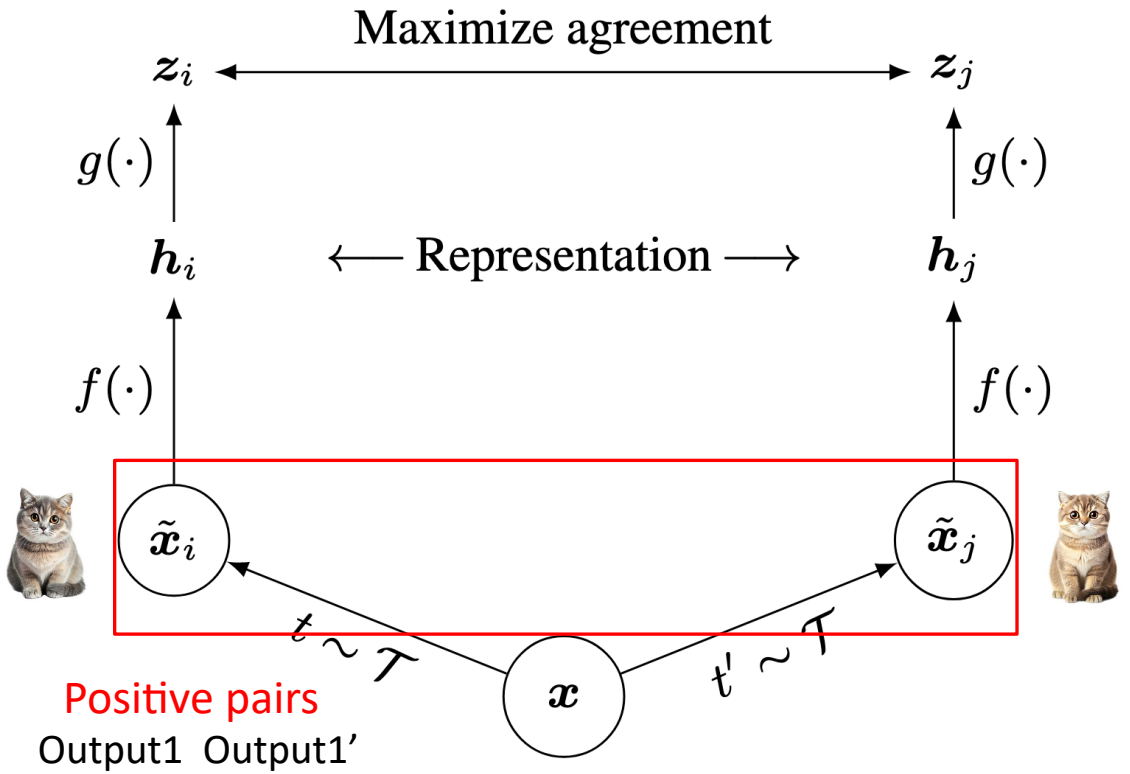
Gao, Tianyu, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings." EMNLP 2021.

Contrastive Learning

f (·): encoder network
g(·): projection head
z: output logits

- The concept of contrastive learning is to maximize the agreement of positive pairs [1][2] .
- A model will be able to learn the connections among similar data and dissimilar data.



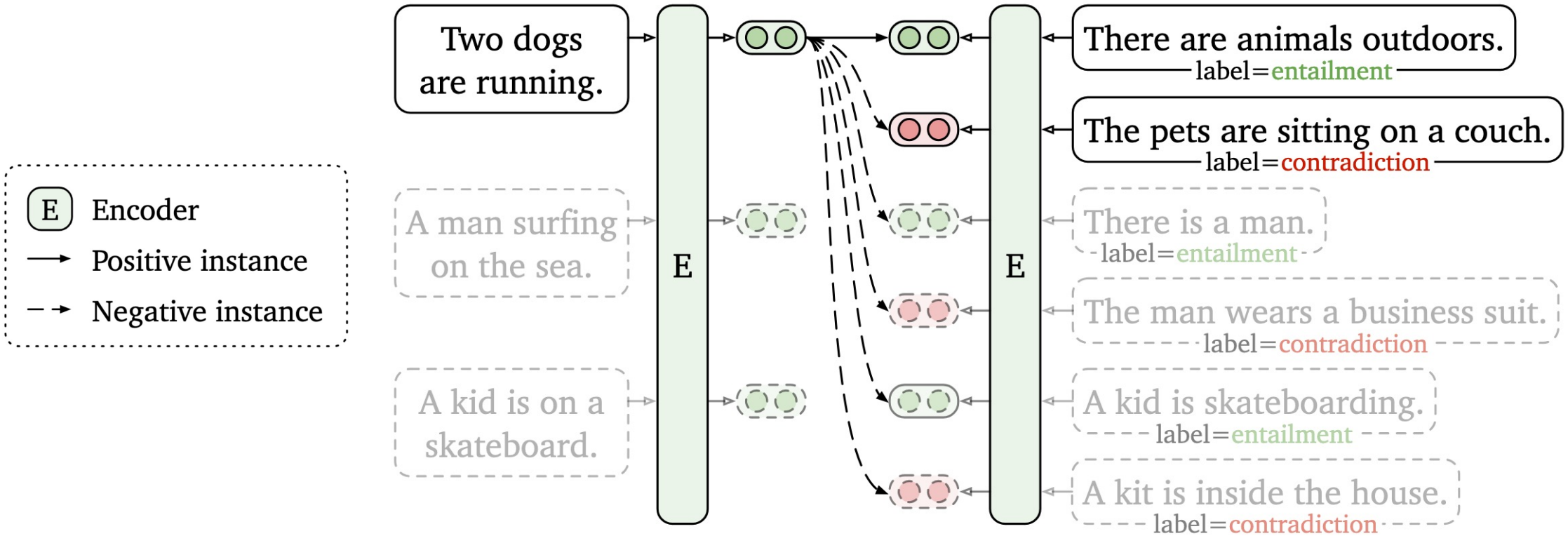


[1] Left Figure source: Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." CVPR 2015.

[2] Right Figure source: Chen, Ting, et al. "A simple framework for contrastive learning of visual representations." ICLR 2020.

Supervised training of SimCSE

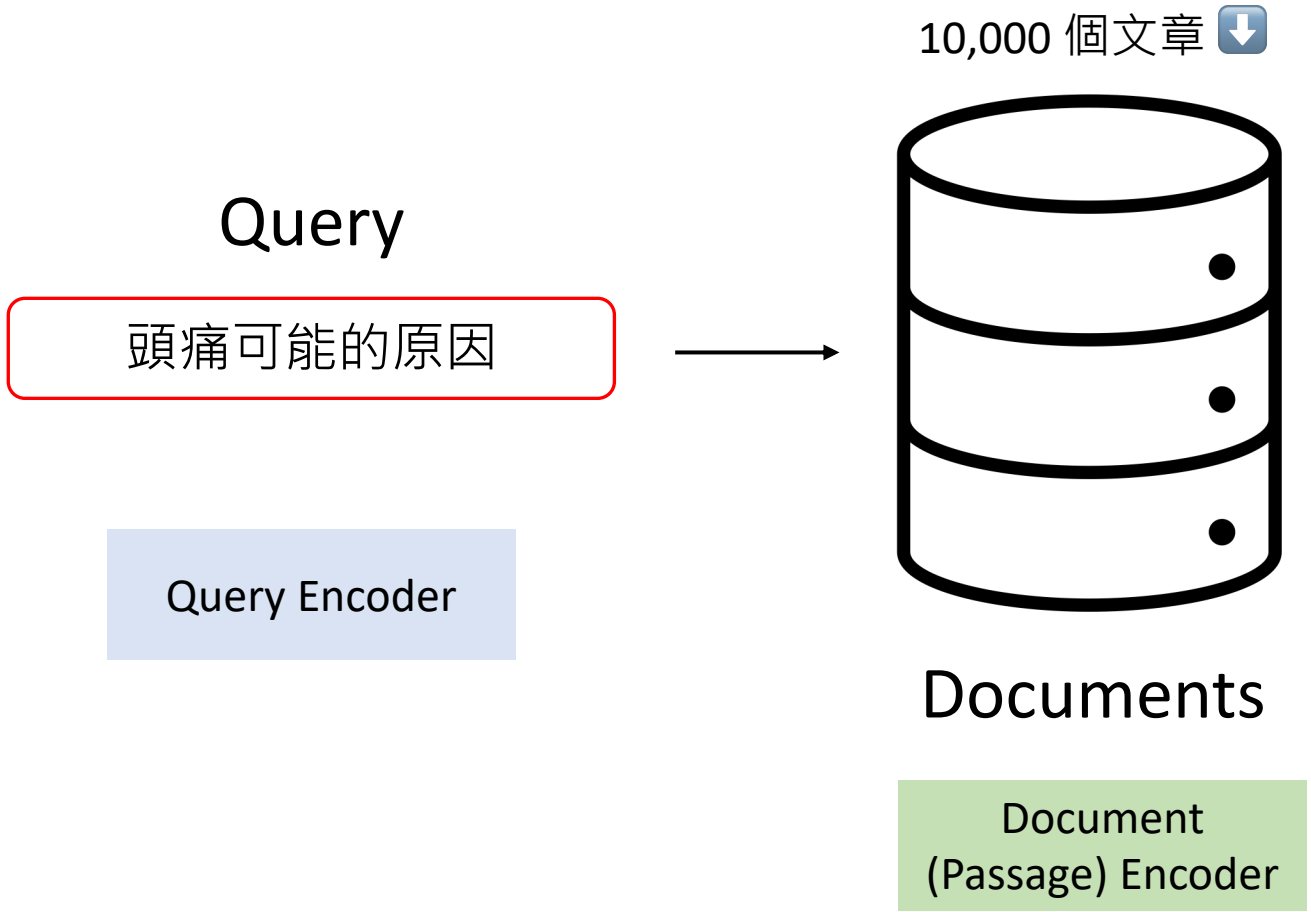
- Supervised training of SimCSE relies on labels in a dataset to define positives and negatives.



Gao, Tianyu, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings." EMNLP 2021.

Retrieval for open-domain question answering (ODQA)

Dense Passage Retrieval (DPR)



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

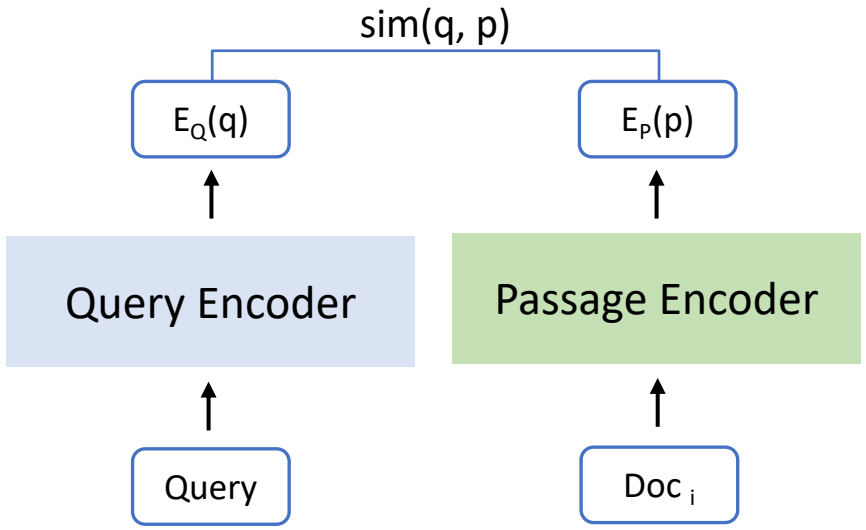
Dense Passage Retrieval (DPR)

- After training, two **BERT-based encoders** can **independently** encode question (**q**) and passage (**p**) into dense vectors.

- **Similarity** between question and passage = **dot product** between their embeddings

$$\text{sim}(q, p) = E_Q(q)^\intercal E_P(p).$$

q : question text
p : passage text
E_Q : BERT model that outputs question representation
E_p : BERT model that outputs passage representation



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

Dense Passage Retrieval (DPR)

Training the encoders

- Goal: **Relevant** pairs of questions and passages will have **smaller distance** than the irrelevant ones

- Training data

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

m training instances

Question

Relevant Passage

n Irrelevant Passages

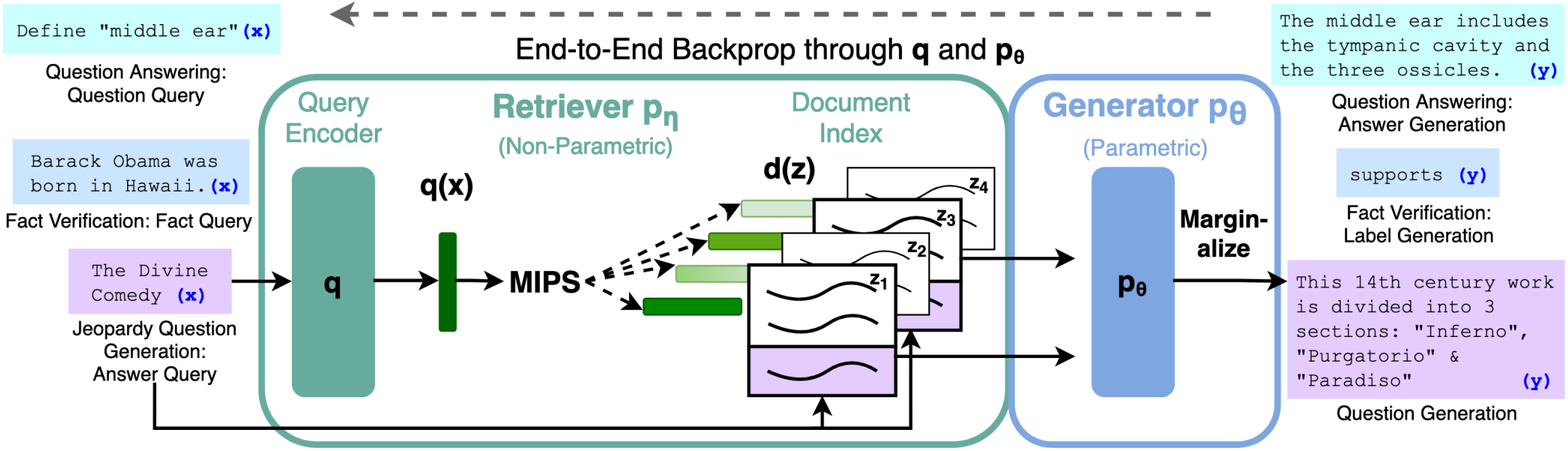
Karpukhin et al., 2020. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Summary Parameter Sharing

	BERT (Devlin et al., NAACL 2019)	Sentence-BERT (Reimers et al., EMNLP 2019)	SimCSE (Gao et al., EMNLP 2021)	DPR (Karpukhin et al., EMNLP 2020)
Encoder Type	Cross-encoder	Bi-encoder	Bi-encoder	Dual encoder
Weight Sharing	Nan	Yes	Yes	No (Separate Query Encoder and Passage Encoder)

Complete RAG

Retrieval-Augmented Generation (RAG)

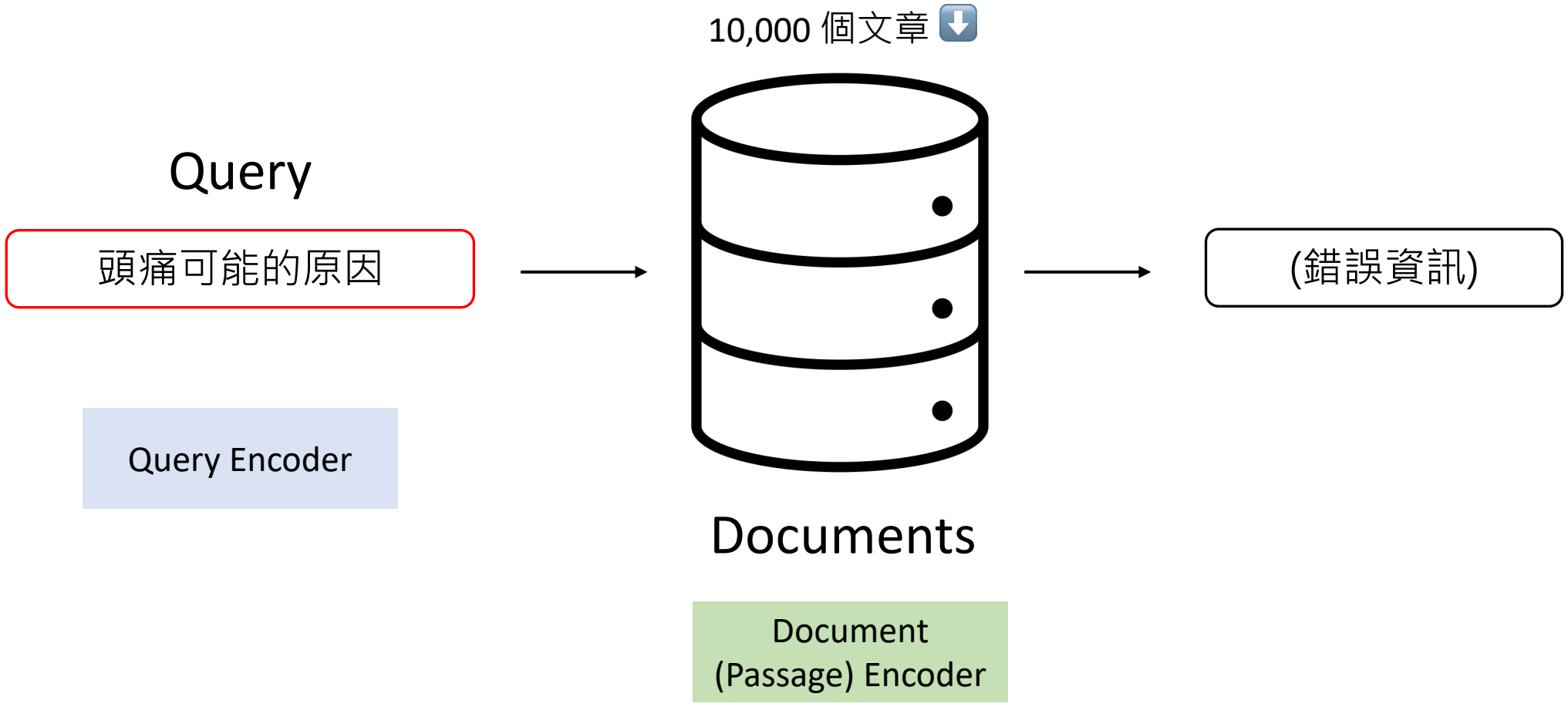


- This approach is the first RAG paper.
- No pre-training is involved. Only fine-tuning on open-domain QA.
- MIPS: Maximum Inner-Product Search (speed-up approach, supported by indexing packages like FAISS.)

Lewis, Patrick, et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks." NeurIPS 2020.

Challenges in Modern RAG

- A significant amount of noise information even fake news in the content available on the Internet.



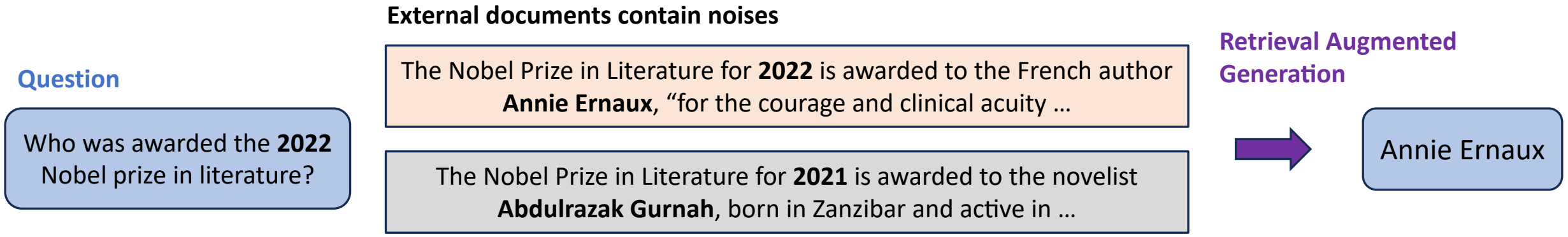
Type of Noises

- Relevant (semantically similar) but not contain the answer
- Counterfactual information
- Irrelevant information
- …

Capabilities that LLMs Should Have in RAG

Noise Robustness

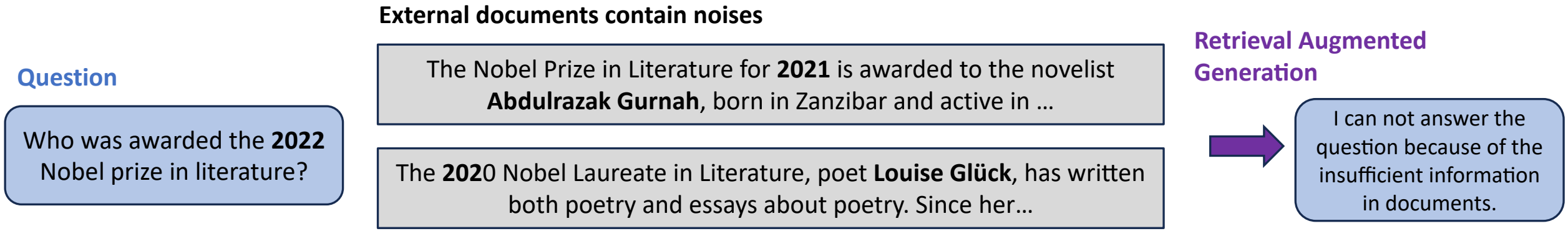
- LLMs must be able to extract the necessary information from documents despite there are noisy documents.



Capabilities that LLMs Should Have in RAG

Noise Rejection

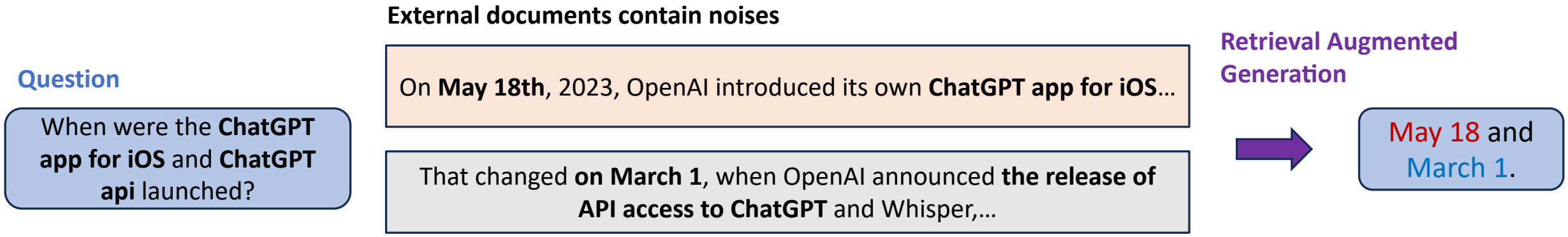
- In real-world situations, the search engine often fails to retrieve documents containing the answers.
- It is important for the model to have the capability to reject recognition and avoid generating misleading content.



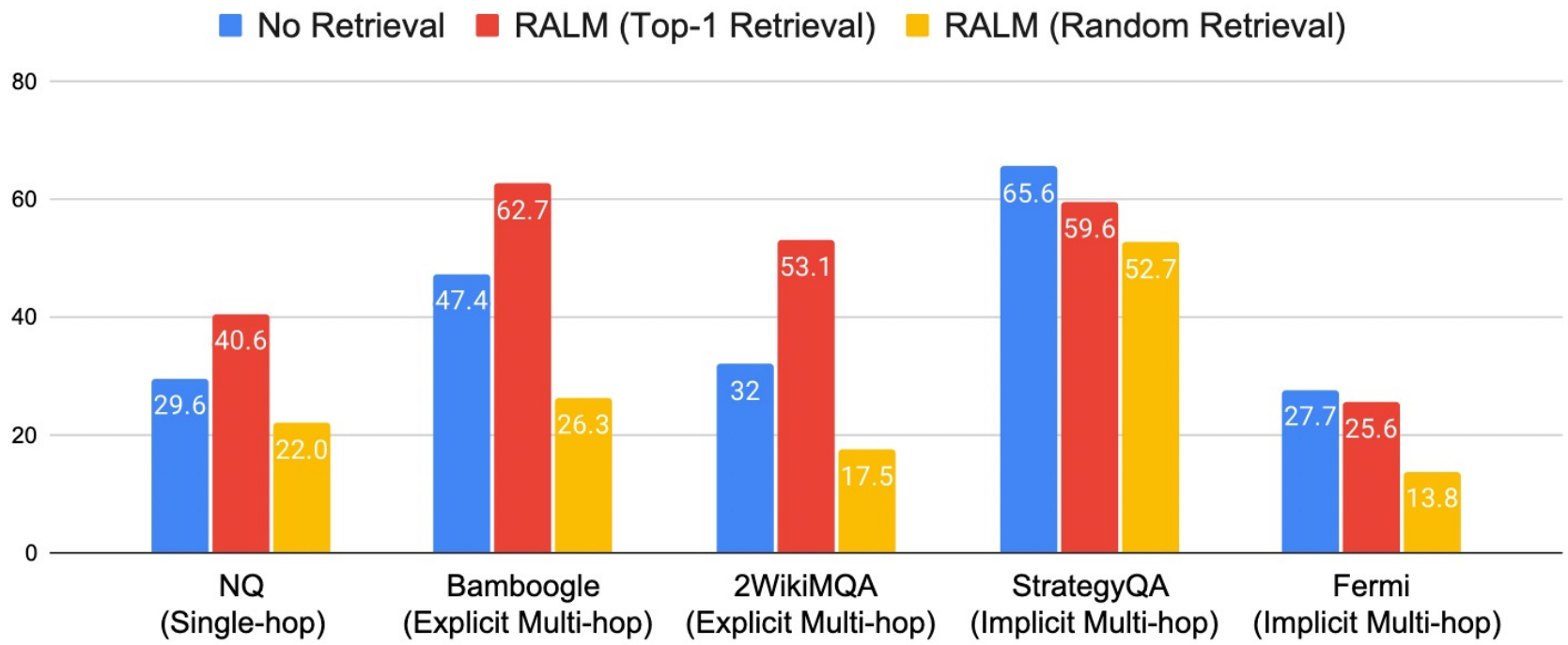
Capabilities that LLMs Should Have in RAG

Information Integration

- In many cases, the answer to a question may be contained in multiple documents.
- To provide better answers to complex questions, it is necessary for LLMs to have the ability to integrate information.



RetRobust: Making Retrieval-Augmented Language Models Robust to Irrelevant Context

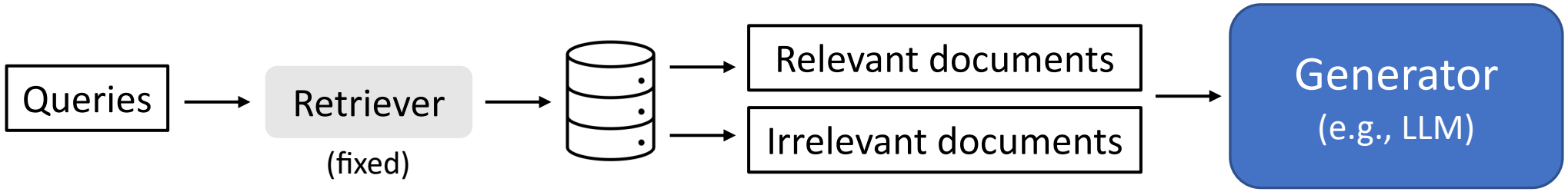


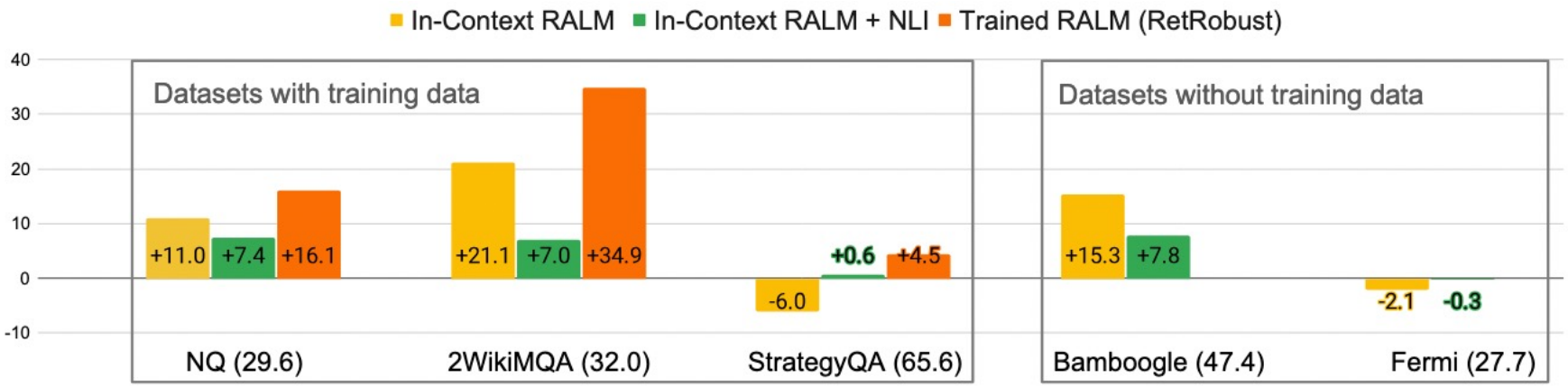
Retrieval augmentation can boost performance, but it also hurts performance on StrategyQA and Fermi, and random contexts reduce performance dramatically.

Yoran, Ori, et al. "Making Retrieval-Augmented Language Models Robust to Irrelevant Context." ICLR 2024.

RetRobust: Making Retrieval-Augmented Language Models Robust to Irrelevant Context

Train the generator on QA tasks with both relevant and irrelevant retrieved documents.

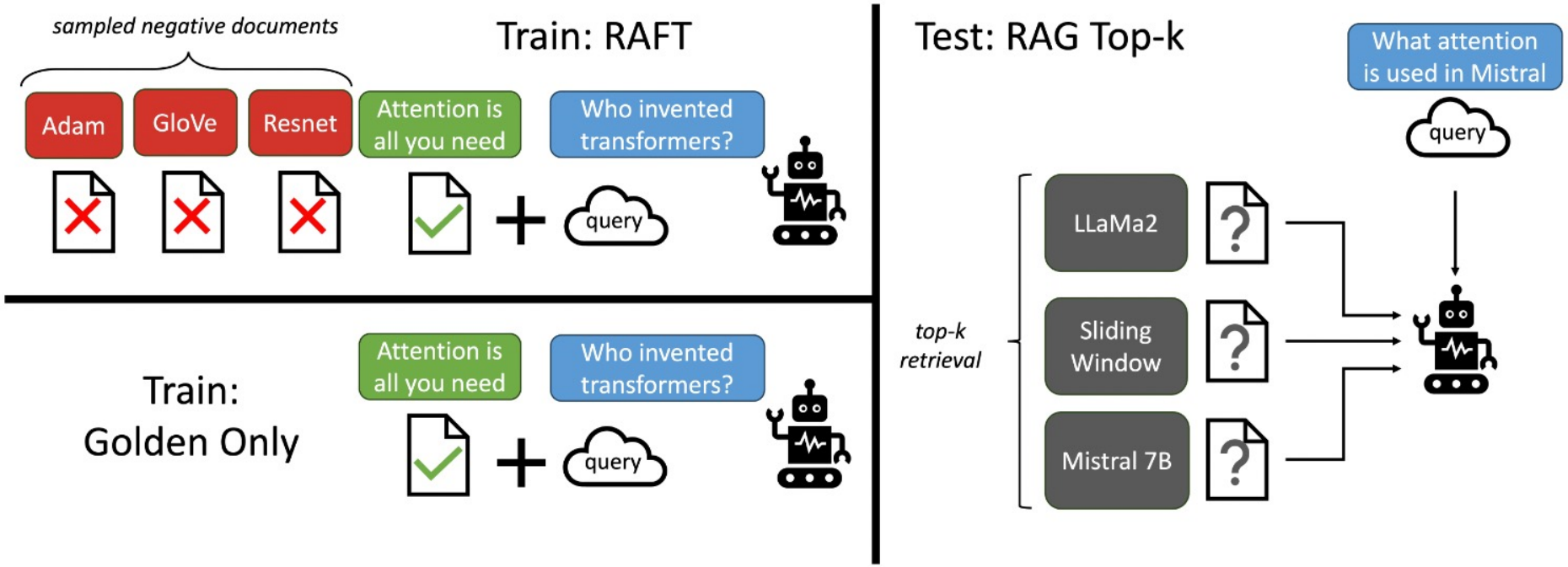




Yoran, Ori, et al. "Making Retrieval-Augmented Language Models Robust to Irrelevant Context." ICLR 2024.

RAFT: Adapting LLM to RAG

- In this work, the retriever is fixed. An LLM is trainable.
- Both correct and incorrect documents are included during training.



MateiZaharia, TianjunZhang ShishirG Patil NamanJain ShengShen, and IonStoica JosephE Gonzalez. "RAFT: Adapting Language Model to Domain Specific RAG." COLM 2024.

Difference between RetRobust and RAFT

Context 包含錯誤資訊

RAFT uses **CoT Answer (answer with rationales).**

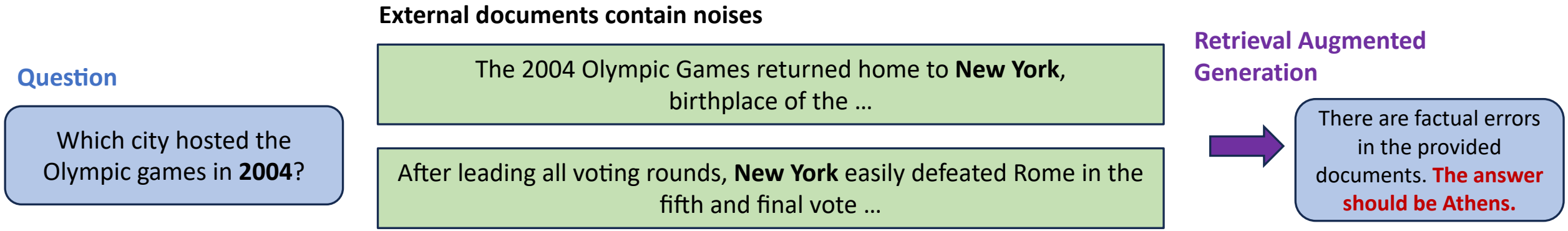
```
Question: The Oberoi family is part of a hotel company that has a head office
in what city?

context: [The Oberoi family is an Indian family that is famous for its
involvement in hotels, namely through The Oberoi Group]...[It is located in
city center of Jakarta, near Mega Kuningan, adjacent to the sister JW Marriott
Hotel. It is operated by The Ritz-Carlton Hotel Company. The complex has two
towers that comprises a hotel and the Airlangga Apartment respectively]...[The
Oberoi Group is a hotel company with its head office in Delhi.]

Instruction: Given the question, context and answer above, provide a logical
reasoning for that answer. Please use the format of: ##Reason: {reason}
##Answer: {answer}.

--------------------------------------------------------------------------------

CoT Answer: ##Reason: The document ##begin_quote## The Oberoi family is an
Indian family that is famous for its involvement in hotels, namely through The
Oberoi Group. ##end_quote## establishes that the Oberoi family is involved in
the Oberoi group, and the document ##begin_quote## The Oberoi Group is a hotel
company with its head office in Delhi. ##end_quote## establishes the head
office of The Oberoi Group. Therefore, the Oberoi family is part of a hotel
company whose head office is in Delhi. ##Answer: Delhi
```


MateiZaharia, TianjunZhang ShishirG Patil NamanJain ShengShen, and IonStoica JosephE Gonzalez. "RAFT: Adapting Language Model to Domain Specific RAG." COLM 2024.

Capabilities that LLMs Should Have in RAG

Counterfactual Robustness

- In the real world, there is an abundance of false information on the internet.
- LLMs should identify risks of known factual errors in the retrieved documents when the LLMs are given warnings about potential risks in the retrieved information through instruction.

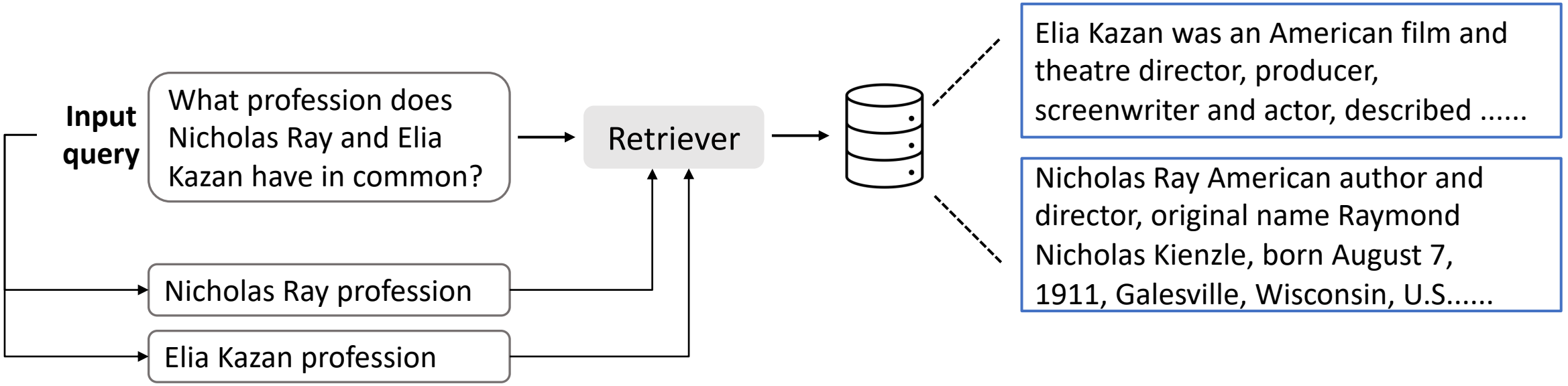


Other Techniques for Boosting RAG Performance

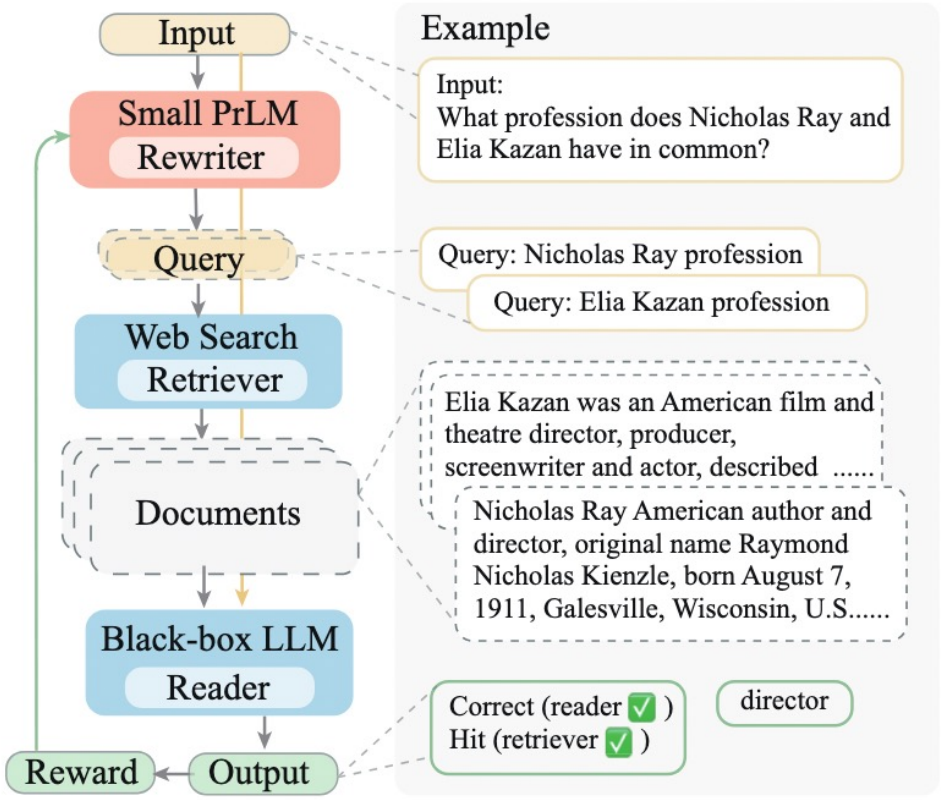
Query Rewriting

Ma, Xinbei, et al. "Query Rewriting in Retrieval-Augmented Large Language Models." EMNLP 2023.

- Motivation:
  - There is inevitably a gap between the input text and the knowledge that is really needed to query.



Query Rewriting – Approach



Ma, Xinbei, et al. "Query Rewriting in Retrieval-Augmented Large Language Models." EMNLP 2023.

Query Rewriting – Prompt for the LLMs

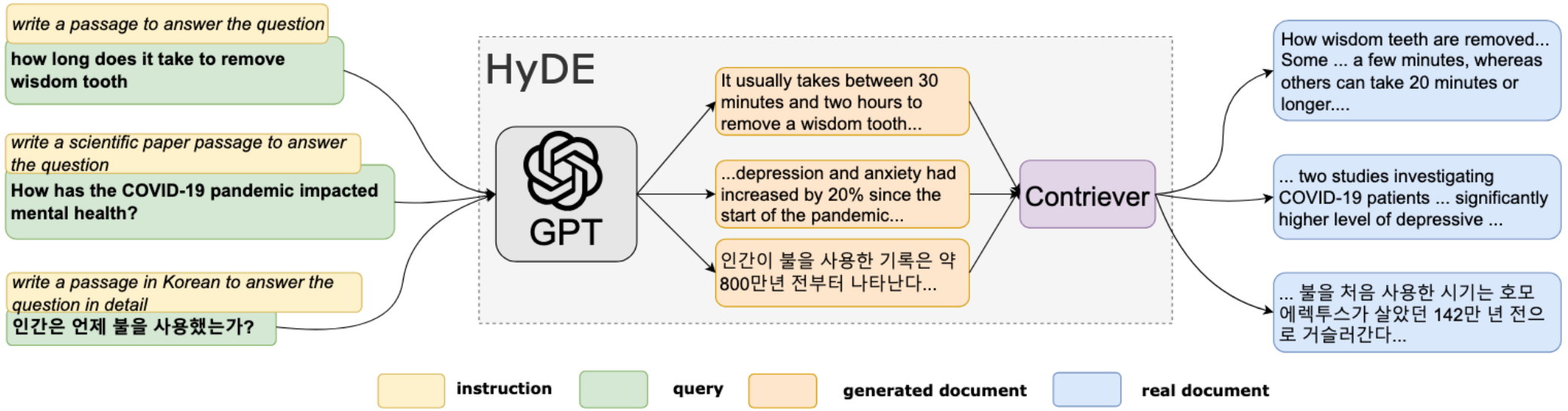
Direct prompt
Answer the question in the following format, end the answer with '**'. {demonstration} Question: {x} Answer:
Reader prompt in retrieval-augment pipelines
Answer the question in the following format, end the answer with '**'. {demonstration} Question: {doc} {x} Answer:
Prompts for LLM as a frozen rewriter
Open-domain QA: Think step by step to answer this question, and provide search engine queries for knowledge that you need. Split the queries with ';' and end the queries with '**'. {demonstration} Question: {x} Answer: *Multiple choice QA:* Provide a better search query for web search engine to answer the given question, end the queries with '**'. {demonstration} Question: {x} Answer:

Table 1: Prompt lines used for the LLMs.

Ma, Xinbei, et al. "Query Rewriting in Retrieval-Augmented Large Language Models." EMNLP 2023.

Hypothetical Document Embeddings (HyDE)

- Use an LLM (InstructGPT) to perform query transformations by asking the LLM to write a passage (**similar to Query Rewriting**).



Gao, Luyu, et al. "Precise Zero-Shot Dense Retrieval without Relevance Labels." ACL 2023.

HyDE can be better than Query Rewriting for retrieval tasks

Method	TREC DL19					TREC DL20				
	mAP	nDCG@10	R@50	R@1k	Latency	mAP	nDCG@10	R@50	R@1k	Latency
unsupervised										
BM25	30.13	50.58	38.32	75.01	**0.07**	28.56	47.96	46.18	78.63	**0.29**
Contriever	23.99	44.54	37.54	74.59	3.06	23.98	42.13	43.81	75.39	0.98
supervised										
LLM-Embedder	44.66	70.20	49.06	84.48	2.61	45.60	68.76	61.36	84.41	0.71
+ Query Rewriting	44.56	67.89	51.45	85.35	7.80	45.16	65.62	59.63	83.45	2.06
+ Query Decomposition	41.93	66.10	48.66	82.62	14.98	43.30	64.95	57.74	84.18	2.01
+ HyDE	50.87	**75.44**	54.93	88.76	7.21	50.94	**73.94**	63.80	88.03	2.14
+ Hybrid Search	47.14	72.50	51.13	89.08	3.20	47.72	69.80	64.32	88.04	0.77
+ HyDE + Hybrid Search	**52.13**	73.34	**55.38**	**90.42**	11.16	**53.13**	72.72	**66.14**	**90.67**	2.95

Table 7: Results for different retrieval methods on TREC DL19/20. The best result for each method is made bold and the second is underlined.

Wang, Xiaohua, et al. "Searching for best practices in retrieval-augmented generation." EMNLP 2024.

Recent RAG Developments

Type	Method	Paper	Venue Year
Enhancing Retrieval	Query Rewriting	Query Rewriting in Retrieval-Augmented Large Language Models	EMNLP 2023
	HyDE	Precise Zero-Shot Dense Retrieval without Relevance Labels	ACL 2023
Enhancing RAG	RetRobust	Making Retrieval-Augmented Language Models Robust to Irrelevant Context	ICLR 2024
	RAFT	RAFT: Adapting Language Model to Domain Specific RAG	COLM 2024
	Self-RAG	Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection	ICLR 2024
	RAAT	Enhancing Noise Robustness of Retrieval-Augmented Language Models with Adaptive Adversarial Training	ACL 2024
Enhancing RAG with Continual Retrieval	FLARE	Active Retrieval Augmented Generation	EMNLP 2023

作業繳交時程

項目	一般截止日期	畢業生截止日期
Homework 4	2025/06/06 23:59 (W16)	2025/05/28 23:59 (W15)
Checkpoint3 簡報檔案 (5/26報告組)	2025/05/25 23:59 (W15)	同左
Checkpoint3 簡報檔案 (6/02報告組)	2025/06/01 23:59 (W16)	-
Final project 程式碼與書面報告	2025/06/06 23:59 (W16)	2025/05/28 23:59 (W15)

補交規範

- 所有作業都能補交，分數打七折
  - (如有特殊原因，請寄信與老師說明)
  - 總補交期限為 2025/06/06 23:59 (W16)
  - 畢業生總補交期限為 2025/05/28 23:59 (W15)
- 小考不能補交
- Project checkpoints 不能補交

Checkpoint 3 (for W15 / W16 oral)

- 一組 10-15 分鐘，老師QA 5分鐘
- Week 14: Retrieval-augmented Generation (RAG)
- Week 15: 6組 (共約 120 分鐘)
  - (Presentations first)
  - Learning-based NLG evaluations
- Week 16: 4組 (共約 80 分鐘)
  - (Presentations first)
  - DeepSeek, mixture of experts (MoE)

Week 15 / Week 16 之前要繳交什麼？

- 一組繳交一份，請上傳至 Teams
- 檔名：NLP_teamN_checkpoint3.pdf 或 NLP_teamN_checkpoint3.pptx
- 前10頁：Checkpoint1+2 原始簡報內容 (如有需要，可修改)
- 後5頁 (或更多)：新進度補充
  1. 實作的方法介紹 (代表各組需完成初步實作)，可以包含：
     - 資料前處理、模型介紹、訓練策略 (如 loss function、optimizer、scheduler 等) 等…
  2. 與上次 (Checkpoint2) 的差異
  3. 實驗結果比較 (含實驗設定說明)，可比較上次結果
  4. Kaggle Leaderboard 名次或分數 (請截圖貼到pptx中)
  5. 時程規劃 (再來還要簡單測試什麼？用表格列出未來 1週內的可能測試與安排)
  6. 針對 Checkpoint3 之前的小組分工細節

互評機制

- **每人**要為與自己相同題目的**組別**打分數
- 打分數表單將於 Week 15 上課前公布

Thank you!

Instructor: 林英嘉

✉ yjlin@cgu.edu.tw

TA: 吳宣毅

✉ m1161007@cgu.edu.tw