

# Robust Information Retrieval

SIGIR 2024 tutorial



---

**Yu-An Liu<sup>a,b</sup>, Ruqing Zhang<sup>a,b</sup>, Jiafeng Guo<sup>a,b</sup> and Maarten de Rijke<sup>c</sup>**

<https://sigir2024-robust-information-retrieval.github.io/>

July 14, 2024

01:30 – 05:00 PM

<sup>a</sup> Institute of Computing Technology, Chinese Academy of Sciences

<sup>b</sup> University of Chinese Academy of Sciences

<sup>c</sup> University of Amsterdam

## About the presenters



**Yu-An Liu**

Phd student  
@ICT, CAS



**Ruqing Zhang**

Faculty  
@ICT, CAS



**Jiafeng Guo**

Faculty  
@ICT, CAS

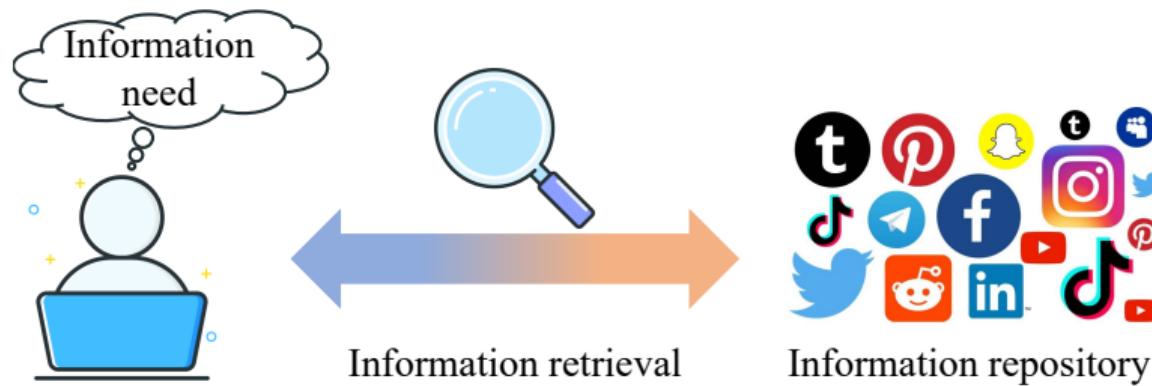


**Maarten de Rijke**

Faculty  
@UvA

## Information retrieval

Information retrieval (IR) is the activity of obtaining information resources that are relevant to an information need from a collection of those resources.



**Given:** User query (keywords, question, image, ...)

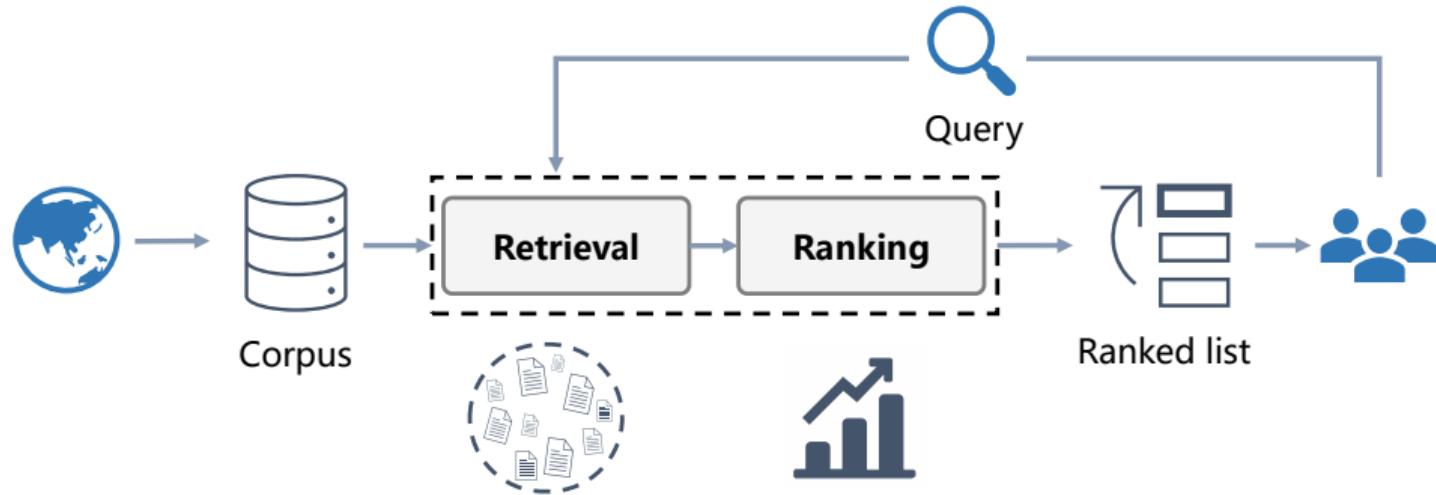
**Rank:** Information objects (passages, documents, images, products, ...)

**Ordered by:** Relevance scores

# Application of information retrieval systems

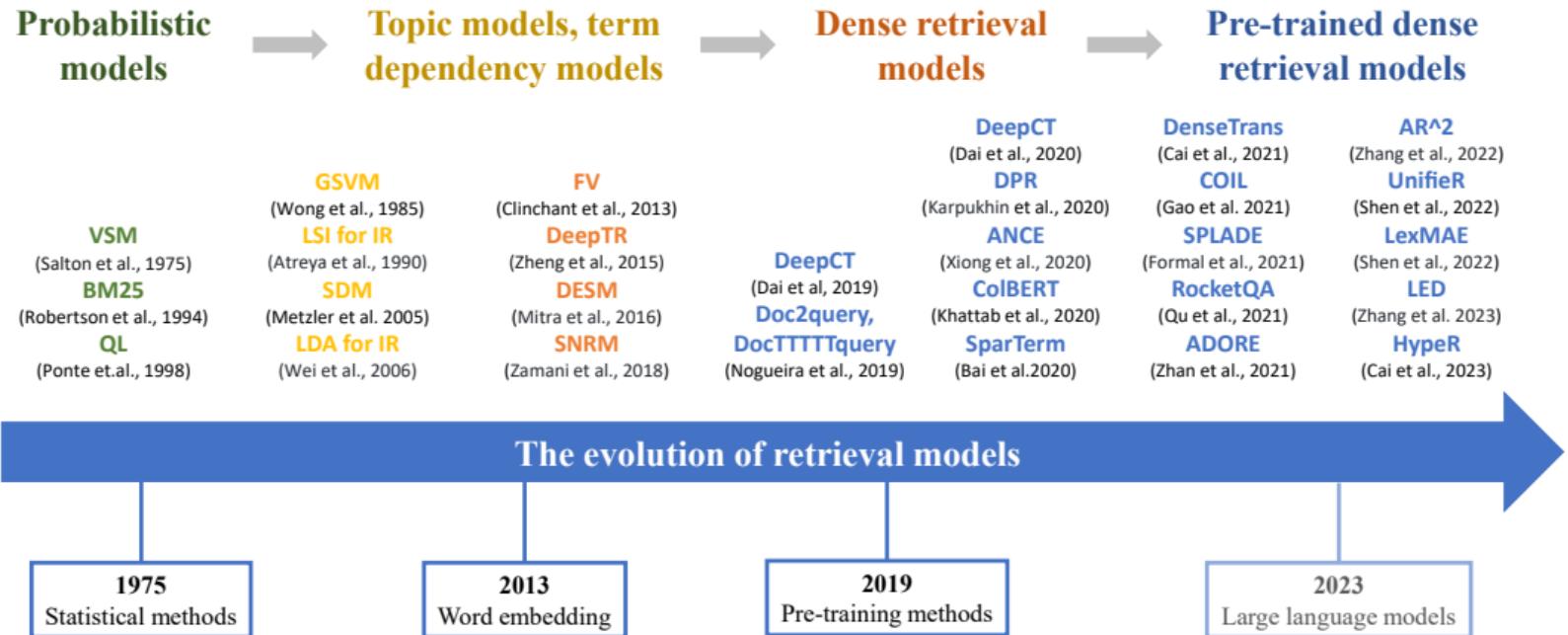


## Core pipelined paradigm: Retrieval-Ranking

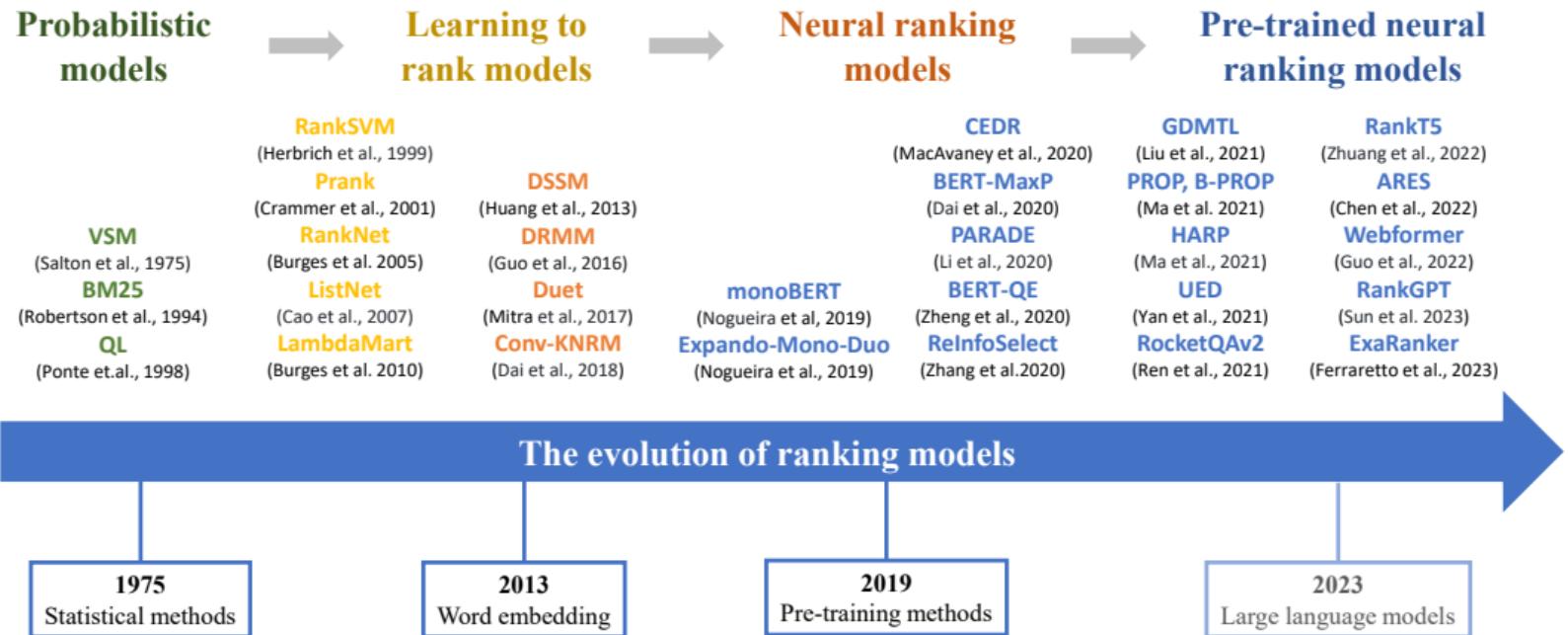


- **Retrieval:** Find an initial set of candidate documents for a query
- **Ranking:** Determine the relevance degree of each candidate

# Evolution of retrieval models



# Evolution of ranking models



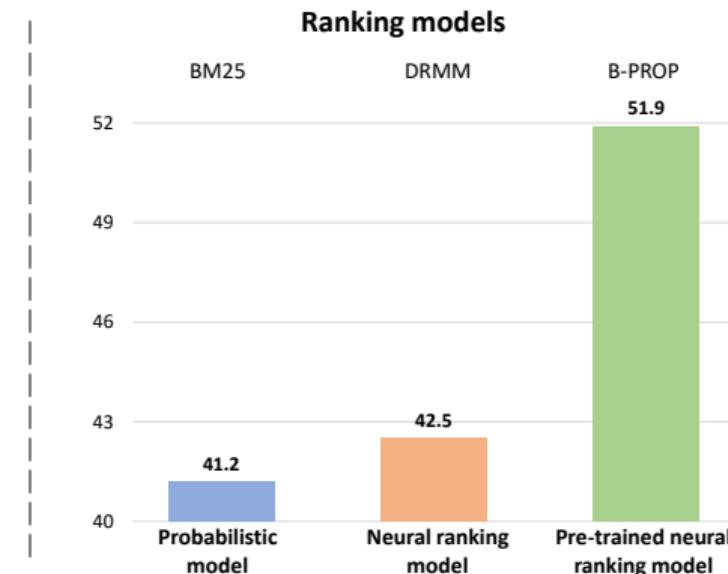
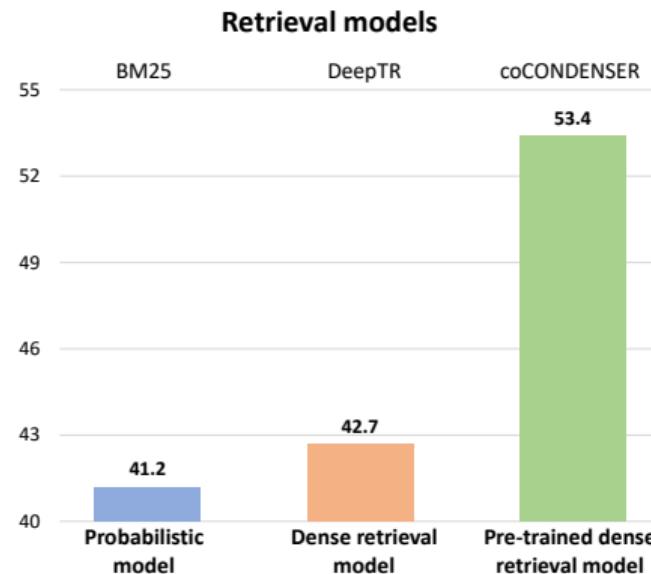
## Effectiveness of neural IR models

Neural IR models, including **dense retrieval models (DRMs)** and **neural ranking models (NRMs)**, have achieved promising ranking effectiveness

# Effectiveness of neural IR models

Neural IR models, including **dense retrieval models (DRMs)** and **neural ranking models (NRMs)**, have achieved promising ranking effectiveness

Let's take the NDCG@20 performance on TREC Robust04 as an example:



**Beyond effectiveness, what are the challenges we face when applying neural IR models in the real world?**

## Challenges 1: Performance fluctuations between queries

Major web search engine makes over **3,200 changes** to its search algorithms in a year to optimize underperforming search results for **a small number** of queries

Data: How We Keep Search Relevant and Useful; Image: [Su et al., 2019]

who invented the telegraph

All Books Images News Shopping More Settings Tools

About 9,320,000 results (0.72 seconds)

**Samuel Morse**

Developed in the 1830s and 1840s by **Samuel Morse** (1791-1872) and other inventors, the telegraph revolutionized long-distance communication. It worked by transmitting electrical signals over a wire laid between stations.

[en.wikipedia.org](https://en.wikipedia.org)

who made listerine

All Shopping Images News Videos More Settings Tools

About 6,130,000 results (0.89 seconds)

**Joseph Lister**

Listerine is a brand of antiseptic mouthwash product. It is promoted with the slogan "Kills germs that cause bad breath". Named after **Joseph Lister**, a pioneer of antiseptic surgery, Listerine was developed in 1879 by Joseph Lawrence, a chemist in St. Louis, Missouri.

  
www.listerine.co.za

(a) A **correct answer** for the query “*who invented the telegraph*”.

(b) A **wrong answer** for the query “*who made listerine*”.

## Challenges 1: Performance fluctuations between queries

Major web search engine makes over **3,200 changes** to its search algorithms in a year to optimize underperforming search results for **a small number** of queries

Data: How We Keep Search Relevant and Useful; Image: [Su et al., 2019]

who invented the telegraph

All Books Images News Shopping More Settings Tools

About 9,320,000 results (0.72 seconds)

**Samuel Morse**

Developed in the 1830s and 1840s by **Samuel Morse** (1791-1872) and other inventors, the telegraph revolutionized long-distance communication. It worked by transmitting electrical signals over a wire laid between stations.

 en.wikipedia.org

(a) A **correct answer** for the query “*who invented the telegraph*”.

who made listerine

All Shopping Images News Videos More Settings Tools

About 6,130,000 results (0.89 seconds)

**Joseph Lister**

Listerine is a brand of antiseptic mouthwash product. It is promoted with the slogan "Kills germs that cause bad breath". Named after **Joseph Lister**, a pioneer of antiseptic surgery, Listerine was developed in 1879 by Joseph Lawrence, a chemist in St. Louis, Missouri.

 www.listerine.co.za

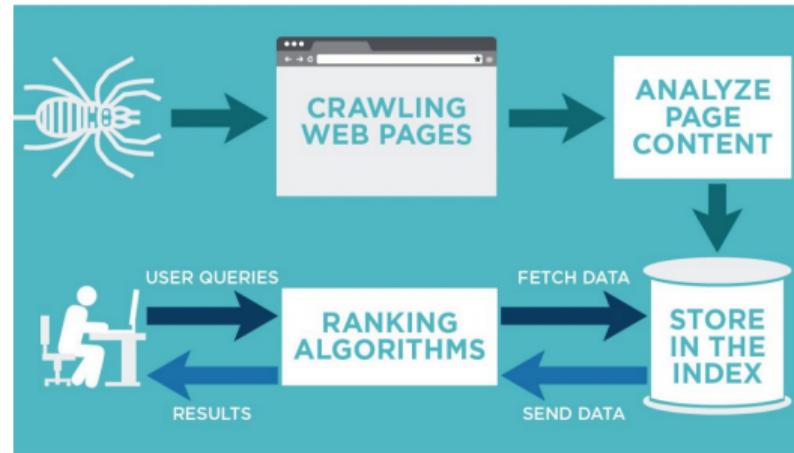
(b) A **wrong answer** for the query “*who made listerine*”.



Neural IR models need to **avoid performance fluctuations** between queries

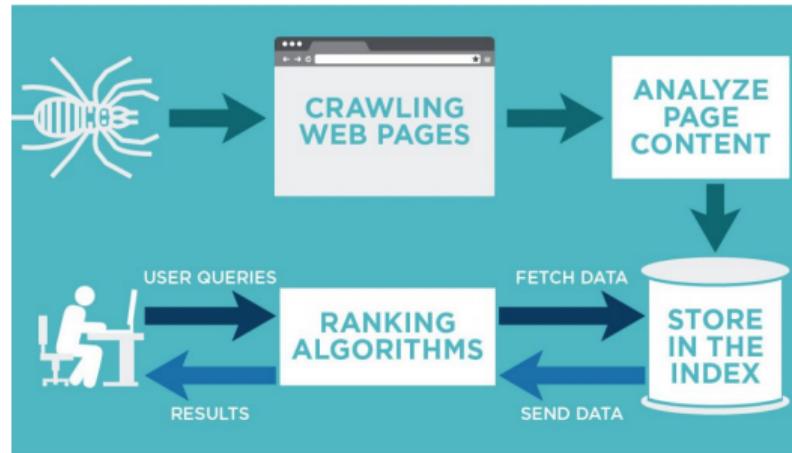
## Challenges 2: A dynamic flow of new data

Every day, billions of new web pages emerge and 15% of search queries are brand new



## Challenges 2: A dynamic flow of new data

Every day, billions of new web pages emerge and 15% of search queries are brand new



Neural IR models need to continuously **adapt to new queries and documents**

## Challenges 3: Search engine optimization (SEO)

About **60%** of marketers get quality leads by SEO, and it can drive over **1,000%** more traffic than before, with a 14.6% conversion rate



## Challenges 3: Search engine optimization (SEO)

About **60%** of marketers get quality leads by SEO, and it can drive over **1,000%** more traffic than before, with a 14.6% conversion rate

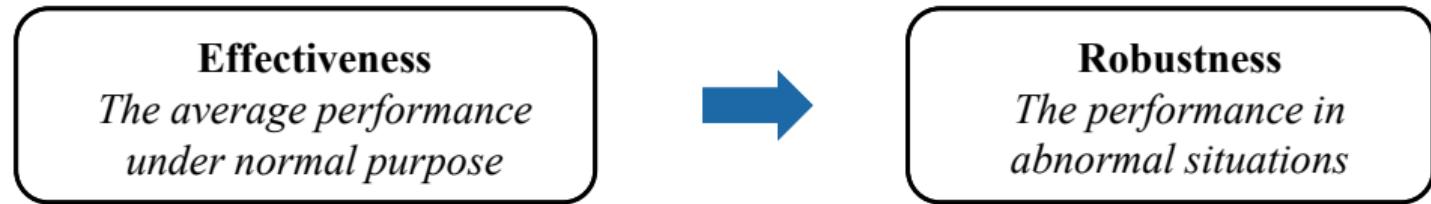


Neural IR models need to be able to **withstand potential SEO attacks**

**Distinct from effectiveness, these challenges can be characterized as robustness**

## What is robustness?

Robustness refers to the ability of a system to withstand disturbances or external factors that may cause it to malfunction or provide inaccurate results.



## What is the robustness in IR?

There is a large volume of work that covers many aspects of IR robustness, e.g.,

## What is the robustness in IR?

There is a large volume of work that covers many aspects of IR robustness, e.g.,

- **Performance variance** emphasizes the **worst-case performance** across different individual queries under the independent and identically distributed (IID) data

## What is the robustness in IR?

There is a large volume of work that covers many aspects of IR robustness, e.g.,

- **Performance variance** emphasizes the **worst-case performance** across different individual queries under the independent and identically distributed (IID) data
- **Out-of-distribution (OOD) robustness** measures the performance on unseen queries and documents from **different distributions of the training dataset**

## What is the robustness in IR?

There is a large volume of work that covers many aspects of IR robustness, e.g.,

- **Performance variance** emphasizes the **worst-case performance** across different individual queries under the independent and identically distributed (IID) data
- **Out-of-distribution (OOD) robustness** measures the performance on unseen queries and documents from **different distributions of the training dataset**
- **Adversarial robustness** focuses on the ability to **defend against malicious adversarial attacks** aimed at manipulating rankings

## Impact of poor robustness on IR systems

If we only focus on effectiveness while ignoring robustness . . .

## Impact of poor robustness on IR systems

If we only focus on effectiveness while ignoring robustness . . .

- Search engine results pages may be flooded with commercial websites that manipulate rankings

## Impact of poor robustness on IR systems

If we only focus on effectiveness while ignoring robustness . . .

- Search engine results pages may be flooded with commercial websites that manipulate rankings
- When we want to explore a new topic, it's difficult to find relevant results

## Impact of poor robustness on IR systems

If we only focus on effectiveness while ignoring robustness . . .

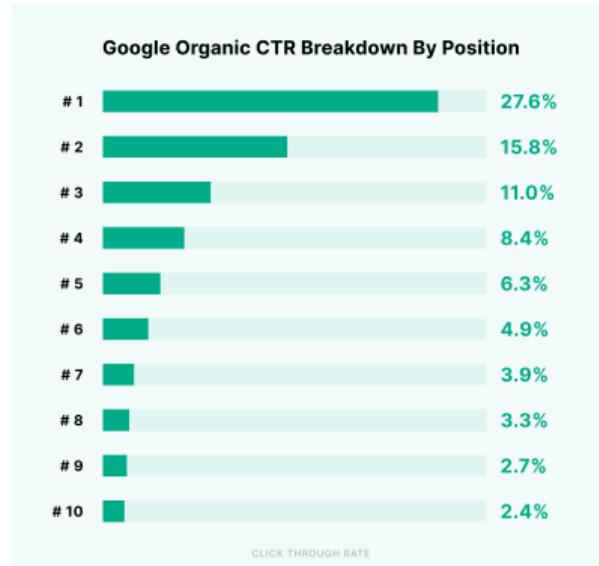
- Search engine results pages may be flooded with commercial websites that manipulate rankings
- When we want to explore a new topic, it's difficult to find relevant results

If these robustness issues are unresolved, they can directly **impact user satisfaction**, which in turn **hinder the widespread adoption** of neural IR models

**Can we follow the experience of other fields to solve the robustness issues in IR?**

# A deep look into robust IR

User attention mainly focuses on the **Top-K** results and increases with **higher rankings**



# A deep look into robust IR

The core of robust IR is to protect the stability of the **Top- $K$**  results



# Comparison with CV and NLP

	CV	NLP	IR
<b>Representative task</b>	Image classification	Text classification	Document ranking
<b>Input format</b>	Single image 😊	Single text 😊	Paired text 🙄
<b>Input space</b>	Continuous 😊	Discrete 🙄	Discrete 🙄
<b>Robustness requirement</b>	Stability of classification 🙄 (dog or cat)	Stability of classification 🙄 (pos or neg)	Stability of top- $K$ result 🙄 (permutation maintenance)

😊 normal

🙄 challenging

😴 hard

# Comparison with CV and NLP

	CV	NLP	IR
<b>Representative task</b>	Image classification	Text classification	Document ranking
<b>Input format</b>	Single image 😊	Single text 😊	Paired text 🙄
<b>Input space</b>	Continuous 😊	Discrete 😨	Discrete 😨
<b>Robustness requirement</b>	Stability of classification 😕 (dog or cat)	Stability of classification 😕 (pos or neg)	Stability of top- $K$ result 😕 (permutation maintenance)

😊 normal

😕 challenging

❗ hard

Experiences from other fields may not be as effective in IR 😞

# Comparison with CV and NLP

	CV	NLP	IR
<b>Representative task</b>	Image classification	Text classification	Document ranking
<b>Input format</b>	Single image 😊	Single text 😊	Paired text 🙄
<b>Input space</b>	Continuous 😊	Discrete 😨	Discrete 😨
<b>Robustness requirement</b>	Stability of classification 😕 (dog or cat)	Stability of classification 😕 (pos or neg)	Stability of top- $K$ result 😕 (permutation maintenance)

😊 normal

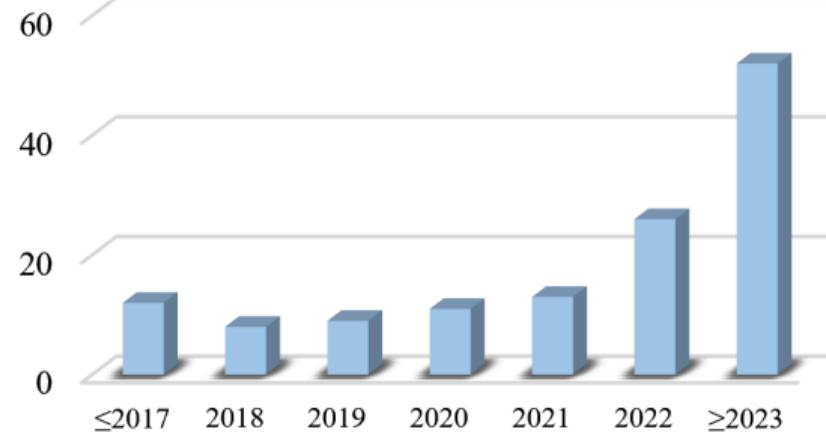
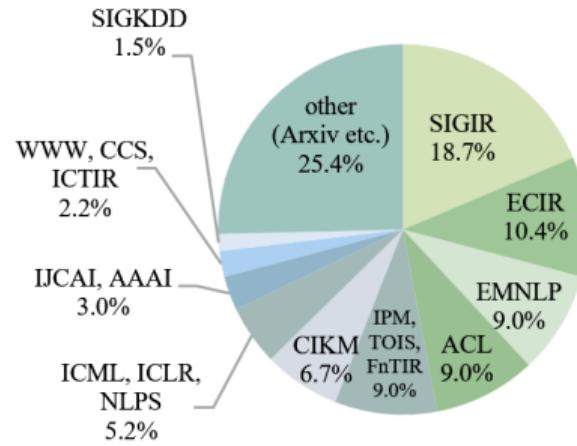
😕 challenging

❗ hard

Experiences from other fields may not be as effective in IR 😰

How can we tailor solutions for robustness issues in IR?

# Publications dedicated to addressing robustness issues in IR



The data statistics cover up to July 10, 2024.

Scan them!

## All about robust information retrieval



Our survey



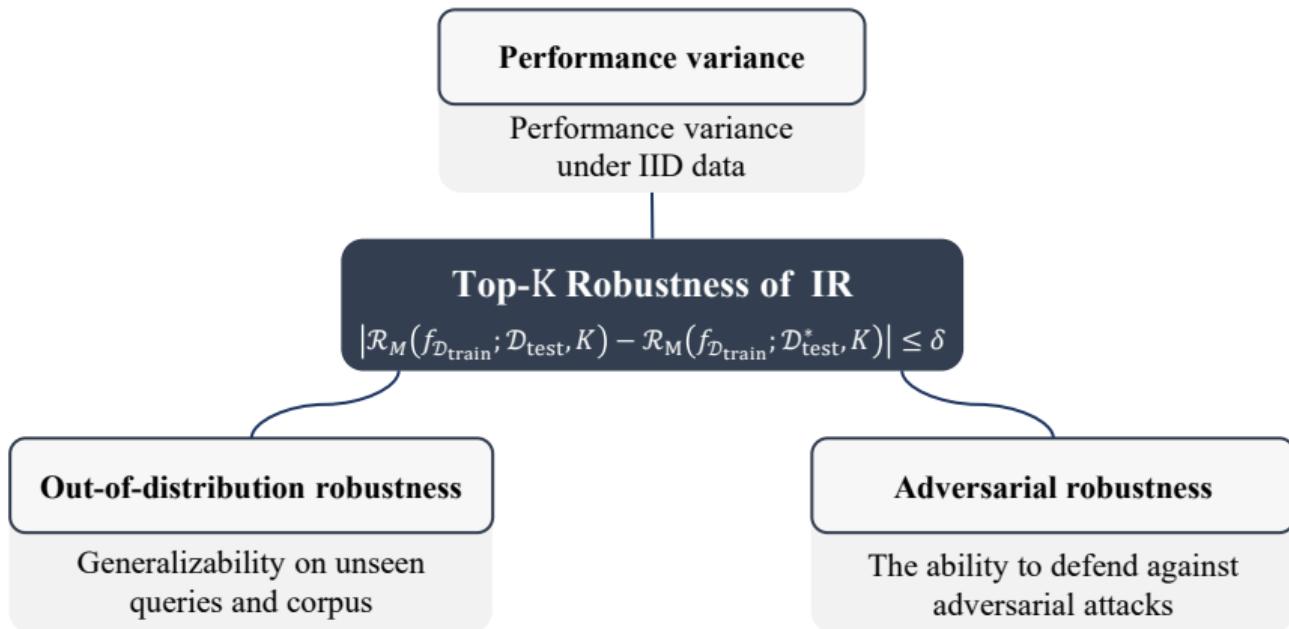
Paper list



Benchmark

# Our survey about robust IR

Our survey on robust neural information retrieval [Liu et al., 2024b], is now available!



## Scope of this tutorial

In this tutorial, we pay special attention to two frequently studied types of robustness, i.e., adversarial robustness and OOD robustness

## Goals of the tutorial

- We will cover key developments in robust information retrieval (mostly 2020–2024)
  - **Definition and taxonomy of robustness in IR**
  - **Adversarial robustness**
  - **Out-of-distribution robustness**
  - **Robust IR in the age of LLMs**

## Goals of the tutorial

- We will cover key developments in robust information retrieval (mostly 2020–2024)
  - **Definition and taxonomy of robustness in IR**
  - **Adversarial robustness**
  - **Out-of-distribution robustness**
  - **Robust IR in the age of LLMs**
- Through this tutorial, we hope to ...
  - Draw attention to the important topic of robustness in IR
  - Help interested beginners to get started and more experienced researchers to gain a systematic understanding of this field
  - Share our perspectives on **future directions**

# Schedule

Time	Section	Presenter
01:30-01:50 PM	Section 1: Introduction	Maarten
01:50-02:10 PM	Section 2: Preliminaries	Yu-An
02:10-03:00 PM	Section 3: Adversarial robustness	Yu-An



30min coffee break

03:30-04:20 PM	Section 4: Out-of-distribution robustness	Yu-An
04:20-04:30 PM	Section 5: Robust IR in the age of LLMs	Yu-An
04:30-04:50 PM	Section 6: Conclusions and future directions	Maarten
04:50-05:00 PM	Q & A	All

## **Section 2: Preliminaries**

## Information retrieval task

Given:

- A **query**  $q$ ,
- A **document**  $d$  from corpus  $D$ .

## Information retrieval task

Given:

- A **query**  $q$ ,
- A **document**  $d$  from corpus  $D$ .

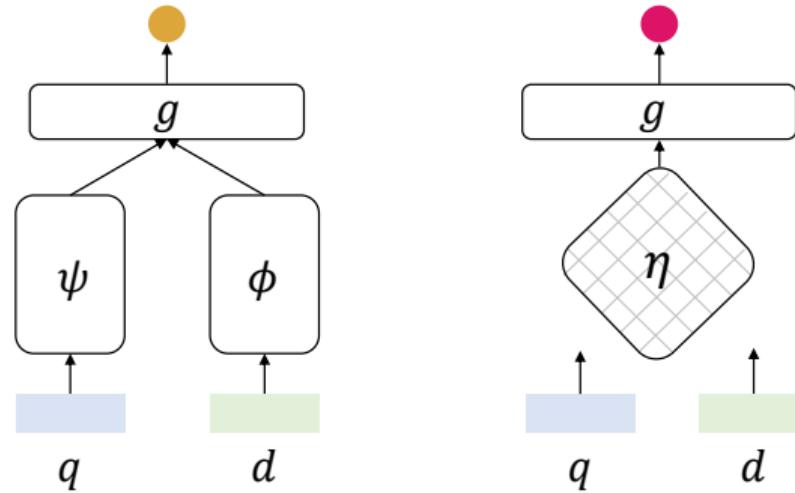
The goal of an IR system is to employ the **ranking function**  $f$  to generate a score  $f(q, d)$  for any **query-document pair**  $(q, d)$ , reflecting the relevance degree between them, and **produce a relevance permutation**  $\pi_f(q, D)$  according to the predicted score:

$$f(q, d) = g(\psi(q), \phi(d), \eta(q, d)),$$

where  $\psi$ ,  $\phi$ , and  $\eta$  return representations of  $q$ ,  $d$ , or both

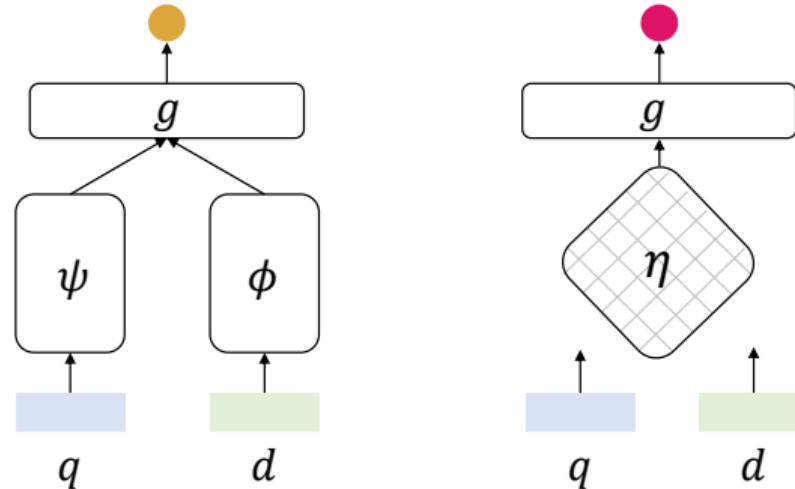
# Neural IR model

$$f(q, d) = g \left( \psi(q), \phi(d), \eta(q, d) \right)$$



# Neural IR model

$$f(q, d) = g \left( \psi(q), \phi(d), \eta(q, d) \right)$$



Dense retrieval model **efficiently** recalls document candidates with **dual-encoder**  
Neural ranking model **effectively** generates the final ranked list with **cross-encoder**

## Evaluation of IR model

In IR, we mainly focus on the top- $K$  ranking result. Given:

- A metric  $M$  focus on the top- $K$  ranking results, e.g., NDCG@ $K$  and MRR@ $K$ ;
- A test dataset  $\mathcal{D}_{\text{test}}$  with ground truth  $Y$ ;

## Evaluation of IR model

In IR, we mainly focus on the top- $K$  ranking result. Given:

- A metric  $M$  focus on the top- $K$  ranking results, e.g., NDCG@ $K$  and MRR@ $K$ ;
- A test dataset  $\mathcal{D}_{\text{test}}$  with ground truth  $Y$ ;

The ranking performance  $\mathcal{R}_M$  of the IR model is usually evaluated by

$$\mathcal{R}_M(f; \mathcal{D}_{\text{test}}, K) = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(q, D, Y) \in \mathcal{D}_{\text{test}}} M(f; (q, D, Y), K).$$

## Evaluation of IR model

In IR, we mainly focus on the top- $K$  ranking result. Given:

- A metric  $M$  focus on the top- $K$  ranking results, e.g., NDCG@ $K$  and MRR@ $K$ ;
- A test dataset  $\mathcal{D}_{\text{test}}$  with ground truth  $Y$ ;

The ranking performance  $\mathcal{R}_M$  of the IR model is usually evaluated by

$$\mathcal{R}_M(f; \mathcal{D}_{\text{test}}, K) = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(q, D, Y) \in \mathcal{D}_{\text{test}}} M(f; (q, D, Y), K).$$

$M$  includes a mapping function  $h$  related to ranking and an indicator function  $\mathbb{I}\{\cdot\}$ :

$$M(f; (q, D, Y), K) = \sum_{(d, y_d) \in (D, Y)} y_d \cdot h(\pi_f(q, d)) \cdot \mathbb{I}\{\pi_f(q, d) \leq K\}.$$

### Definition (**Top- $K$ robustness in information retrieval**)

Let  $\delta \geq 0$  denote an acceptable error threshold. Given an IR model  $f_{\mathcal{D}_{\text{train}}}$  trained on training dataset  $\mathcal{D}_{\text{train}}$  with a corresponding testing dataset  $\mathcal{D}_{\text{test}}$ , an unseen test dataset  $\mathcal{D}_{\text{test}}^*$ , for the top- $K$  ranking result, if

$$|\mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}_{\text{test}}, K) - \mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}_{\text{test}}^*, K)| \leq \delta,$$

we consider the model  $f_{\mathcal{D}_{\text{train}}}$  to be **Top- $K$ -robust** for metric  $M$ .

## Adversarial robustness in IR: Definition

To avoid the vulnerabilities of neural IR models being exploited by black hat SEO, we study adversarial robustness.

## Adversarial robustness in IR: Definition

To avoid the vulnerabilities of neural IR models being exploited by black hat SEO, we study adversarial robustness.

### **Definition (Adversarial robustness in information retrieval)**

Given an IR model  $f_{\mathcal{D}_{\text{train}}}$  trained on training dataset  $\mathcal{D}_{\text{train}}$  with a corresponding testing dataset  $\mathcal{D}_{\text{test}}$ , a new document set  $D_{\text{adv}}$  containing adversarial examples, and an acceptable error threshold  $\delta$ , for the top- $K$  ranking result, if

$$|\mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}_{\text{test}}, K) - \mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}'_{\text{test}}, K)| \leq \delta \text{ such that } \mathcal{D}'_{\text{test}} \leftarrow \mathcal{D}_{\text{test}} \cup D_{\text{adv}},$$

where  $\mathcal{D}_{\text{test}} \cup D_{\text{adv}}$  denotes injecting the set of all generated adversarial examples  $D_{\text{adv}}$  into the original test dataset, and then model  $f$  is considered  $\delta$ -robust against adversarial examples for metric  $M$ .

## Out-of-distribution robustness: Definition

OOD generalizability stands as a pivotal requirement for contemporary IR systems, given the dynamic nature of user needs and evolving data landscapes.

## Out-of-distribution robustness: Definition

OOD generalizability stands as a pivotal requirement for contemporary IR systems, given the dynamic nature of user needs and evolving data landscapes.

### **Definition (Out-of-distribution robustness of information retrieval)**

Given an IR model  $f_{\mathcal{D}_{\text{train}}}$ , an original dataset with training and test data,  $\mathcal{D}_{\text{train}}$  and  $\mathcal{D}_{\text{test}}$ , drawn from the original distribution  $\mathcal{G}$ , along with a new test dataset  $\tilde{\mathcal{D}}_{\text{test}}$  drawn from the new distribution  $\tilde{\mathcal{G}}$ , and an acceptable error threshold  $\delta$ , for the top- $K$  ranking result, if

$$|\mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}_{\text{test}}, K) - \mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \tilde{\mathcal{D}}_{\text{test}}, K)| \leq \delta \text{ where } \mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}} \sim \mathcal{G}, \tilde{\mathcal{D}}_{\text{test}} \sim \tilde{\mathcal{G}},$$

the model  $f$  is considered  $\delta$ -robust against out-of-distribution data for metric  $M$ .

## Performance variance: Definition

A robust neural IR model should not only have good performance over the entire query set, but also ensure that the performance on individual queries is not too bad.

## Performance variance: Definition

A robust neural IR model should not only have good performance over the entire query set, but also ensure that the performance on individual queries is not too bad.

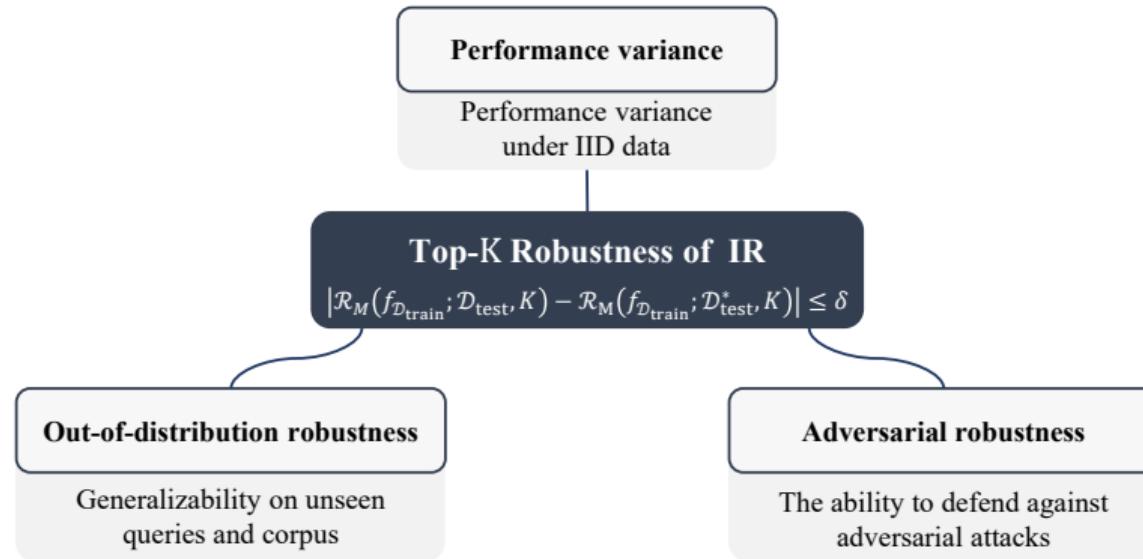
### **Definition (Performance variance of information retrieval)**

Given an IR model  $f_{\mathcal{D}_{\text{train}}}$  trained on training dataset  $\mathcal{D}_{\text{train}}$  with a corresponding testing dataset  $\mathcal{D}_{\text{test}}$ , and an acceptable error threshold  $\delta$ , for the top- $K$  ranking result, if

$$\text{Var}(\{M(f_{\mathcal{D}_{\text{train}}}; (q, D, Y), K) \mid (q, D, Y) \in \mathcal{D}_{\text{test}}\}) \leq \delta,$$

where  $\text{Var}(\cdot)$  is the variance of the ranking performance of the IR model  $f_{\mathcal{D}_{\text{train}}}$  on  $\mathcal{D}_{\text{test}}$ , then the model  $f$  is considered  $\delta$ -robust in terms of performance variance for metric  $M$ .

# Robustness in IR: Taxonomy



We will address **adversarial robustness** in Section 3 and OOD robustness in Section 4!

## **Section 3: Adversarial robustness**

## Revisit the definition of adversarial robustness

Ability of Neural IR models to maintain Top- $K$  ranking performance when subjected to adversarial attacks.

### **Definition (Adversarial robustness in information retrieval)**

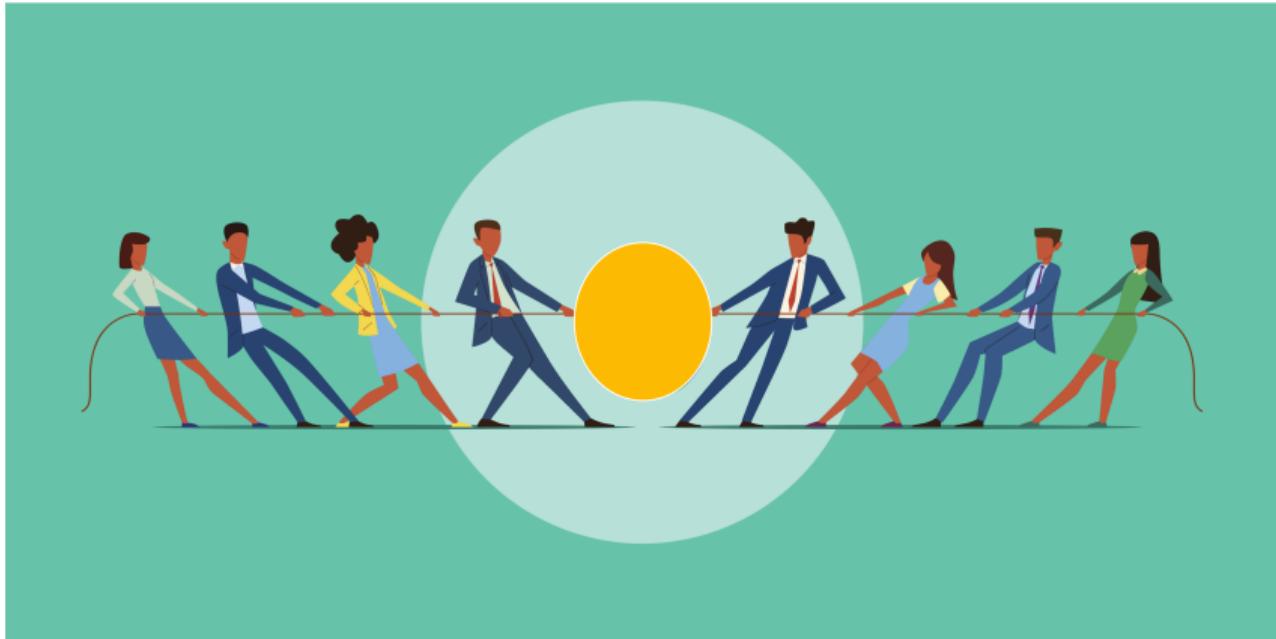
Given an IR model  $f_{\mathcal{D}_{\text{train}}}$  trained on training dataset  $\mathcal{D}_{\text{train}}$  with a corresponding testing dataset  $\mathcal{D}_{\text{test}}$ , a new document set  $D_{\text{adv}}$  containing adversarial examples, and an acceptable error threshold  $\delta$ , for the top- $K$  ranking result, if

$$|\mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}_{\text{test}}, K) - \mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}'_{\text{test}}, K)| \leq \delta \text{ such that } \mathcal{D}'_{\text{test}} \leftarrow \mathcal{D}_{\text{test}} \cup D_{\text{adv}},$$

where  $\mathcal{D}_{\text{test}} \cup D_{\text{adv}}$  denotes injecting the set of all generated adversarial examples  $D_{\text{adv}}$  into the original test dataset, and then model  $f$  is considered  $\delta$ -robust against adversarial examples for metric  $M$ .

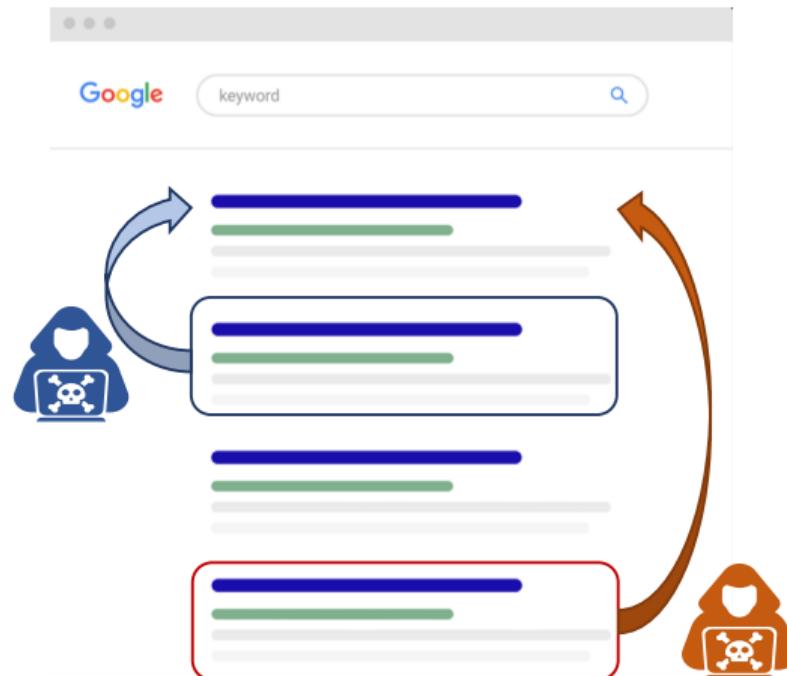
## Background: Competitive search

Search engine is a **competitive scenario**, content providers may aim to promote their products or documents in rankings for specific queries [Kurland and Tennenholz, 2022]

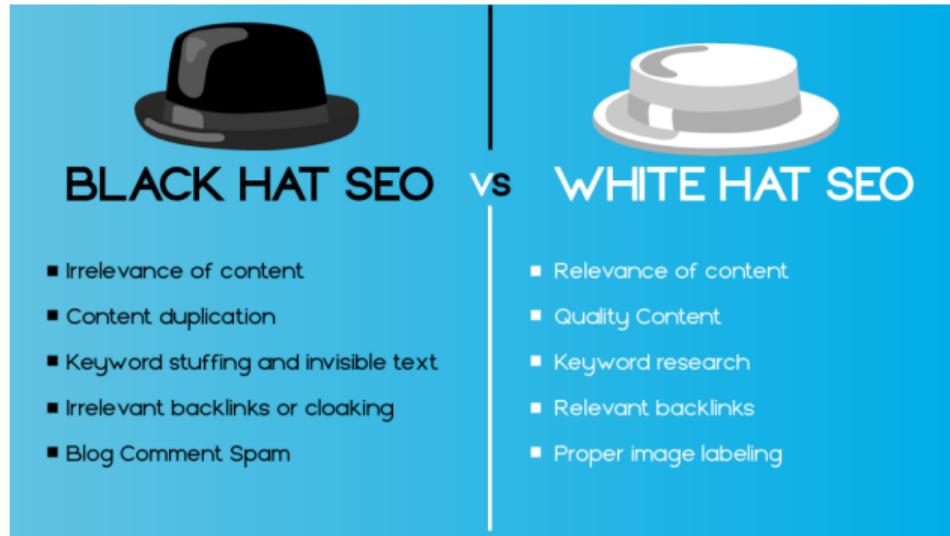


## Background: Competitive search

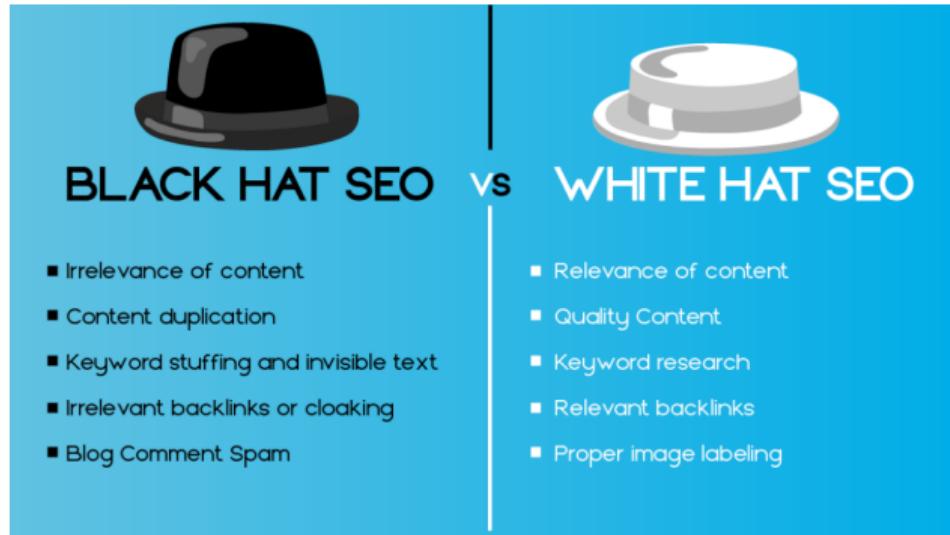
Competitive search scenario leads to the development for search engine optimization (SEO) and attack techniques against search engines [[Gyöngyi and Garcia-Molina, 2005](#)]



# Black-hat SEO vs. White-hat SEO



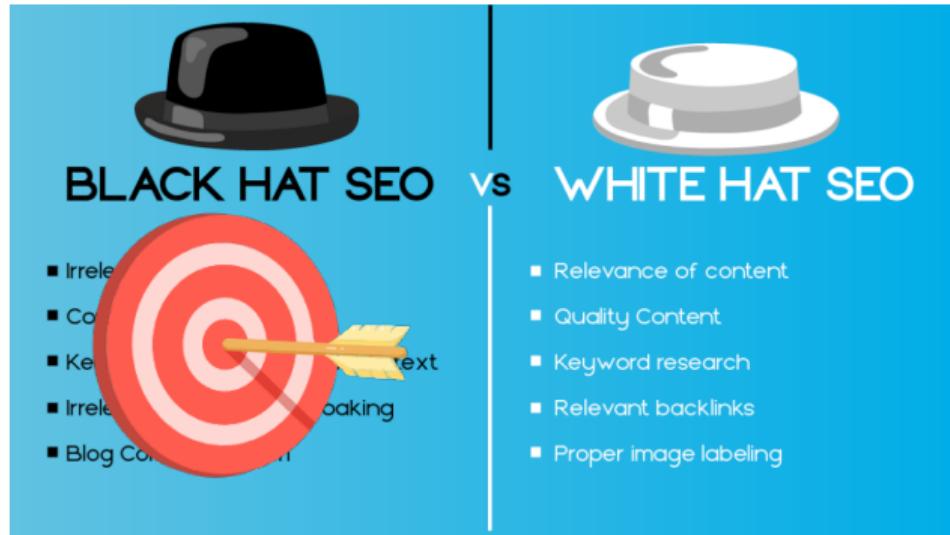
# Black-hat SEO vs. White-hat SEO



**White-hat SEO** optimizes the quality of web pages **within the rules of search engines**

**Black-hat SEO** maliciously modifies web pages by **exploiting search engine loopholes**

# Black-hat SEO vs. White-hat SEO

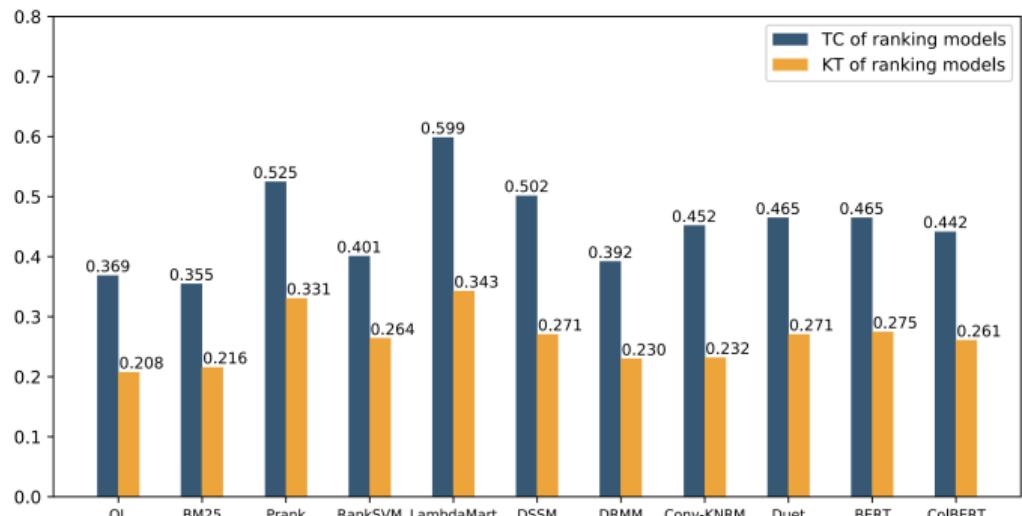


**White-hat SEO** optimizes the quality of web pages **within the rules of search engines**

**Black-hat SEO** maliciously modifies web pages by **exploiting search engine loopholes**

# The vulnerability of IR models

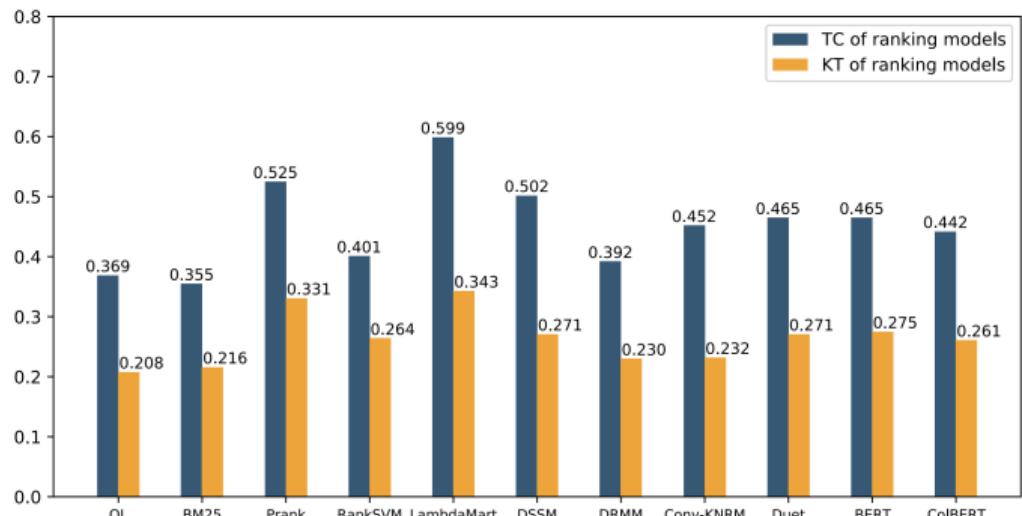
Our team found IR models are vulnerable in black-hat SEO scenarios [Wu et al., 2022b]:



- Dataset: ASRC
- Metrics:
  - TC: Change of the top-1
  - KT: Change of the ranked list

# The vulnerability of IR models

Our team found IR models are vulnerable in black-hat SEO scenarios [Wu et al., 2022b]:



- Dataset: ASRC
- Metrics:
  - TC: Change of the top-1
  - KT: Change of the ranked list

**Vulnerability** (red color indicates neural IR models):

*DSSM > BERT > Conv-KNRM > ColBERT > RankSVM > DRMM > QL > BM25*

**How to improve the adversarial robustness of neural IR models?**

## Two perspectives of adversarial robustness

Robustness is enhanced during the competition between **attacks** and **defenses**



## Two perspectives of adversarial robustness

Robustness is enhanced during the competition between **attacks** and **defenses**

- **Adversarial attacks:** Identify the vulnerability of neural IR models
- **Adversarial defenses:** Improve the adversarial robustness of neural IR models



## Outline

We will introduce the adversarial robustness through:

- **Benchmarks & settings**
- **Adversarial attacks**
- **Adversarial defenses**

## Adversarial robustness: Benchmarks

- **Basic datasets:** Original datasets in IR that are adapted for reuse by attack and defense methods, e.g., MS MARCO and Clueweb09-B

## Adversarial robustness: Benchmarks

- **Basic datasets:** Original datasets in IR that are adapted for reuse by attack and defense methods, e.g., MS MARCO and Clueweb09-B
- **Expansion of dataset:** Additional data provided by competitions, e.g., TREC DL19 and TREC DL20, are used for evaluation against the basic datasets

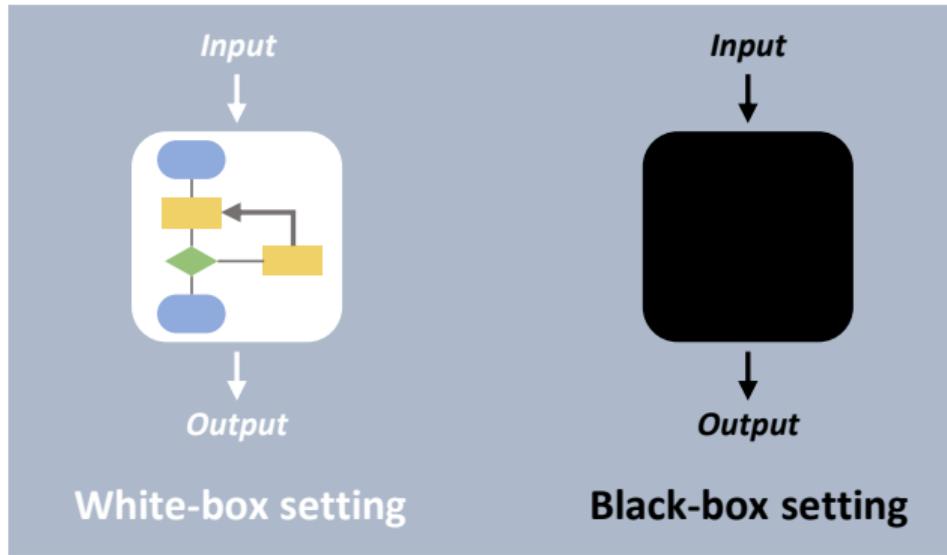
## Adversarial robustness: Benchmarks

- **Basic datasets:** Original datasets in IR that are adapted for reuse by attack and defense methods, e.g., MS MARCO and Clueweb09-B
- **Expansion of dataset:** Additional data provided by competitions, e.g., TREC DL19 and TREC DL20, are used for evaluation against the basic datasets
- **Tailored datasets:** Datasets specially tailored for adversarial attacks and defenses, e.g., ASRC and DARA

## Adversarial robustness: Benchmarks

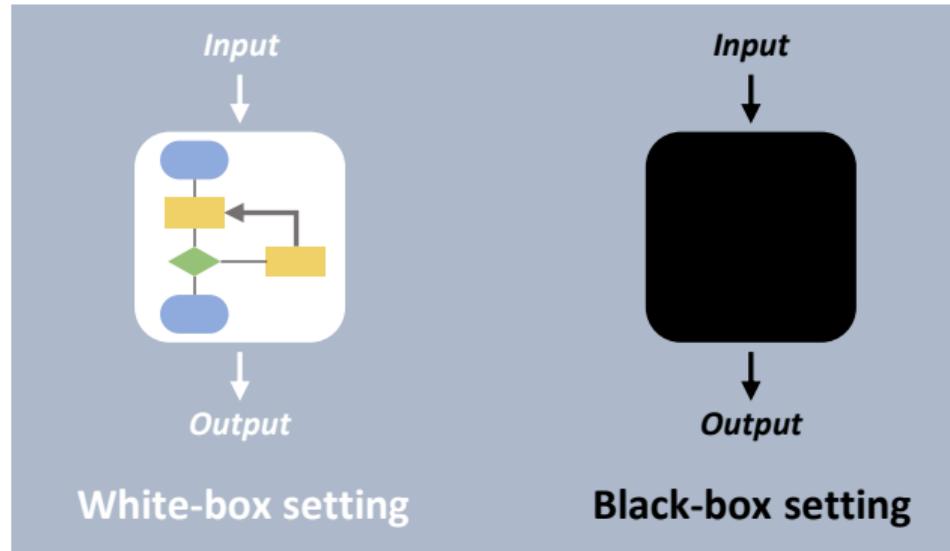
Type	Dataset	#Document	#Q <sub>train</sub>	#Q <sub>dev</sub>	#Q <sub>eval</sub>
Basic datasets	MS MARCO Doc [Nguyen et al., 2016]	3.2M	370K	5,193	5,793
	MS MARCO Pas [Nguyen et al., 2016]	8.8M	500K	6,980	6,837
	Clueweb09-B [Clarke et al., 2009]	50M	150	-	-
	NQ [Kwiatkowski et al., 2019]	21M	60K	8.8k	3.6k
	TriviaQA [Joshi et al., 2017]	21M	60K	8.8K	11.3K
Dataset expansion	TREC DL19 [Craswell et al., 2020]	-	-	43	-
	TREC DL20 [Craswell et al., 2021a]	-	-	54	-
	TREC MB14 [Lin et al., 2013]	-	-	50	-
Tailored datasets	ASRC [Raifer et al., 2017]	1,279	-	31	-
	Q-MS MARCO [Liu et al., 2023b]	-	-	4,000	-
	Q-Clueweb09 [Liu et al., 2023b]	-	-	292	-
	DARA [Chen et al., 2023c]	164k	50k	3,490	3,489

## Adversarial robustness: Settings



- **White-box setting:** attackers can fully access the model parameters and leverage the target model gradient to directly generate perturbations
- **Black-box setting:** attackers can only obtain the output by querying the target model, without having access to the internal parameters or gradients

## Adversarial robustness: Settings



Considering real-world applications, existing work pays more attention on the more practical and challenging **black-box setting**

## Traditional web spamming

**Web spamming:** any form of search engine ranking manipulation without regard to any value for the user

The main forms include:

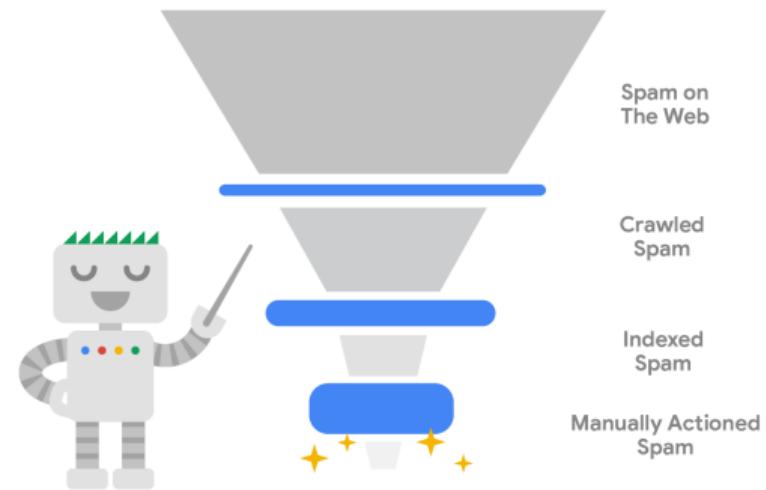
- **Keyword stuffing** →
- Excessive links
- Sneaky redirects
- Phishing
- ...

**Query:** *What's the best resort in Washington?*

**Spammy web site:** *The Capitol Grand Hotel offers a comfort, and best resort best resort best resort. Just steps away from iconic landmarks such as Washington Washington Washington, this prestigious hotel is perfect for both leisure and business travelers. The Capitol Grand features best best best resort resort resort including high-speed internet.*

## Traditional web spamming is . . .

- **Easily detected**
  - Major search engines said to automatically discover over **40 billion spammy pages** per day, which may keep more than **99% of visits completely without spam**
- **Mainly targeted at traditional IR models**
  - Spamming methods pose a **limited threat in the age of neural models**



**How to perform adversarial attacks against neural IR models to expose their vulnerabilities?**

## Requirements of adversarial attacks

Inspired by black-hat SEO, given a **low-ranked target document**, the requirements of adversarial attacks in IR include:

- Identifying **gradient vulnerabilities** of neural IR models on the target document
- Perturbing the target document **in a human-imperceptible way**
- Maximizing ranking improvement of the target document in the **Top- $K$  results**

## Definition of adversarial attacks

Given:

- a neural IR model  $f$  and a query  $q$ , and
- a top- $K$  ranked list and a low-ranked target document  $d$ .

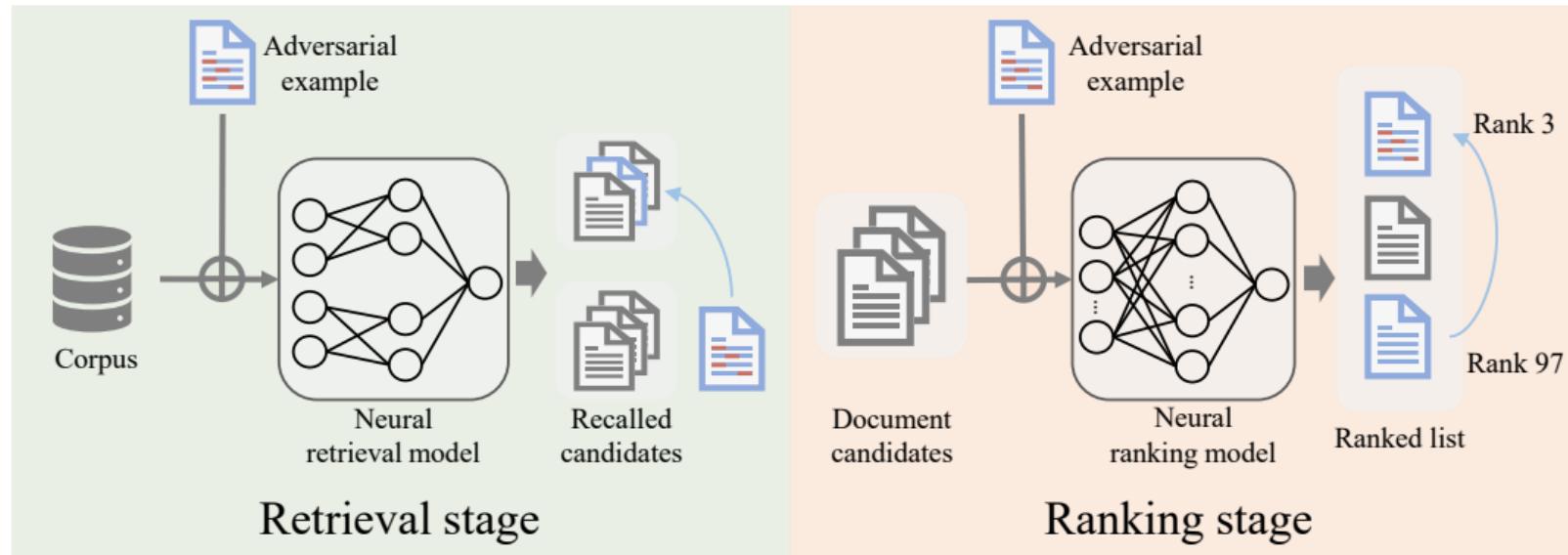
The goal is to improve the ranking of  $d$  under  $q$  with human-imperceptible perturbations  $p$ :

$$\max_p \left( K - \pi_f(q, d \oplus p) + \lambda \cdot \text{Sim}(d, d \oplus p) \right),$$

It consists of two parts:

- Minimize the ranking position of the perturbed document  $d \oplus p$
- Maximize the similarity between the perturbed  $d \oplus p$  and original document  $d$

# Classification of adversarial attacks



- **Adversarial retrieval attack** retrieves a target document **outside the top- $K$  candidates** to appear among the top- $K$  candidates in response to a query
- **Adversarial ranking attack** promotes the target document in rankings **in the top- $K$  candidates** with respect to a query

## Adversarial retrieval attack

The definition of adversarial retrieval attacks can be formalized as:

$$\max_p \left( K - \text{Recall}_f(q, d \oplus p) + \lambda \cdot \text{Sim}(d, d \oplus p) \right),$$

where  $\text{Recall}_f(q, d \oplus p)$  denotes the recalled position of the perturbed document  $d \oplus p$  generated by the dense retrieval model  $f$  with respect to query  $q$  given the entire corpus

The low-ranked target document  $d$  is out of the Top- $K$  results

## Adversarial ranking attack

The definition of **adversarial ranking attacks** can be formalized as:

$$\max_p \left( K - \text{Rank}_f(q, d \oplus p) + \lambda \cdot \text{Sim}(d, d \oplus p) \right),$$

where  $\text{Rank}_f(q, d \oplus p)$  denotes the ranking position of the perturbed document  $d \oplus p$  in the final ranked list generated by the neural retrieval model  $f$  with respect to query  $q$

The low-ranked target document  $d$  is **in the Top- $K$  results**

## Topic-oriented adversarial retrieval/ranking attack

Web page owners usually expect their content to have a general advantage in ranked lists for **for queries under the same search intent**

## Topic-oriented adversarial retrieval/ranking attack

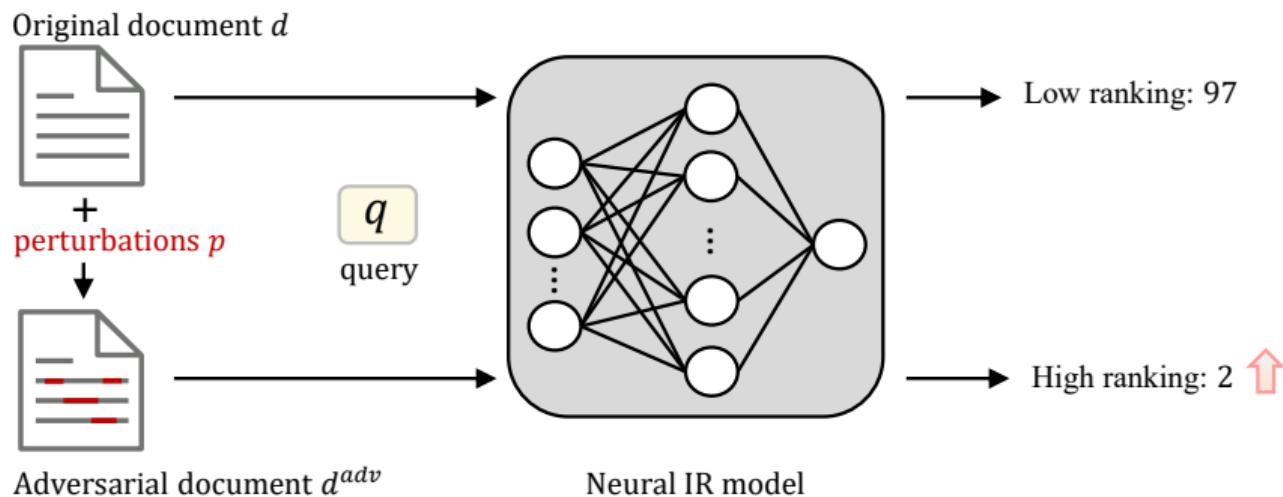
Web page owners usually expect their content to have a general advantage in ranked lists for **for queries under the same search intent**

In **paid search advertising**, when advertisers create an advertisement, they select a set of keywords for a group of target queries with the same topic:



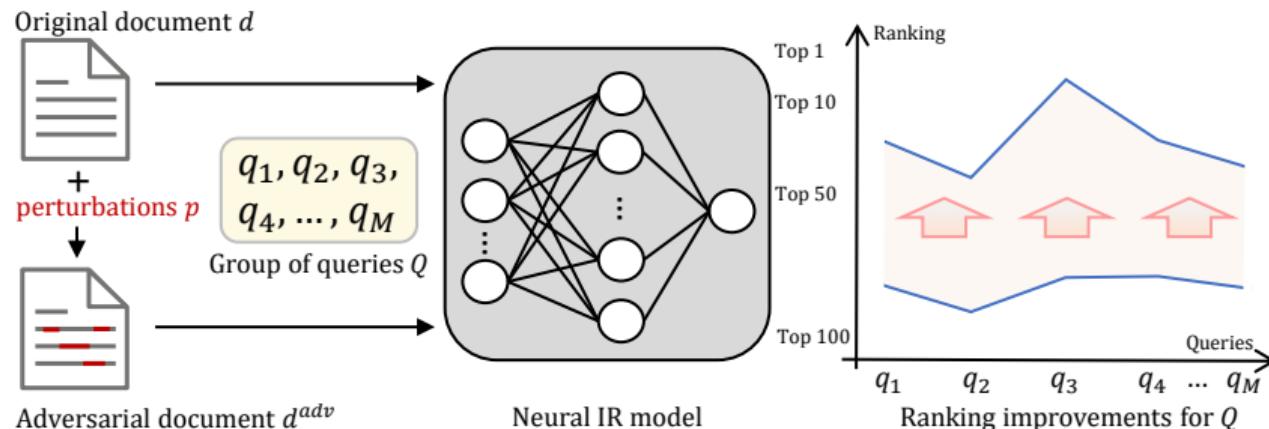
## Topic-oriented adversarial retrieval/ranking attack

**Paired attack** promotes a target document in rankings w.r.t. a specific query



# Topic-oriented adversarial retrieval/ranking attack

**Topic-oriented attack** promotes a target document in rankings on **each query in the group with the same topic**



## Topic-oriented adversarial retrieval/ranking attack



“Advantages” of topic-oriented attack:

- Meet the needs of realistic SEO
- More challenging than paired attack
- Identifying the generic vulnerability of neural IR models

## Key steps of adversarial attacks

**Steal knowledge from  
black-box models**

# Key steps of adversarial attacks

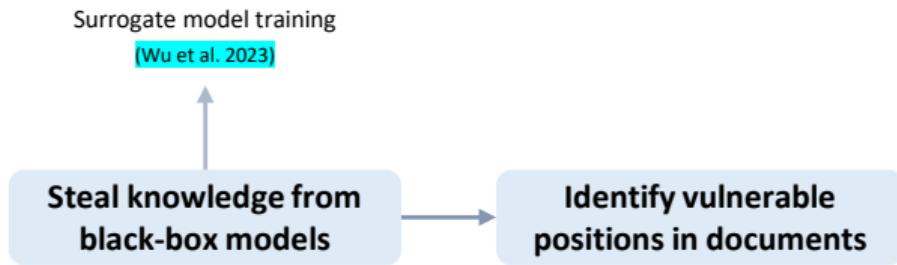
Surrogate model training

(Wu et al. 2023)

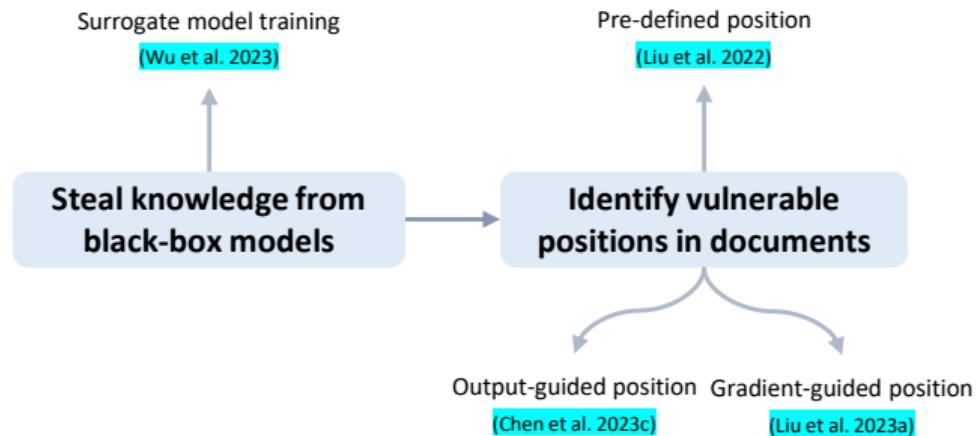
Steal knowledge from  
black-box models



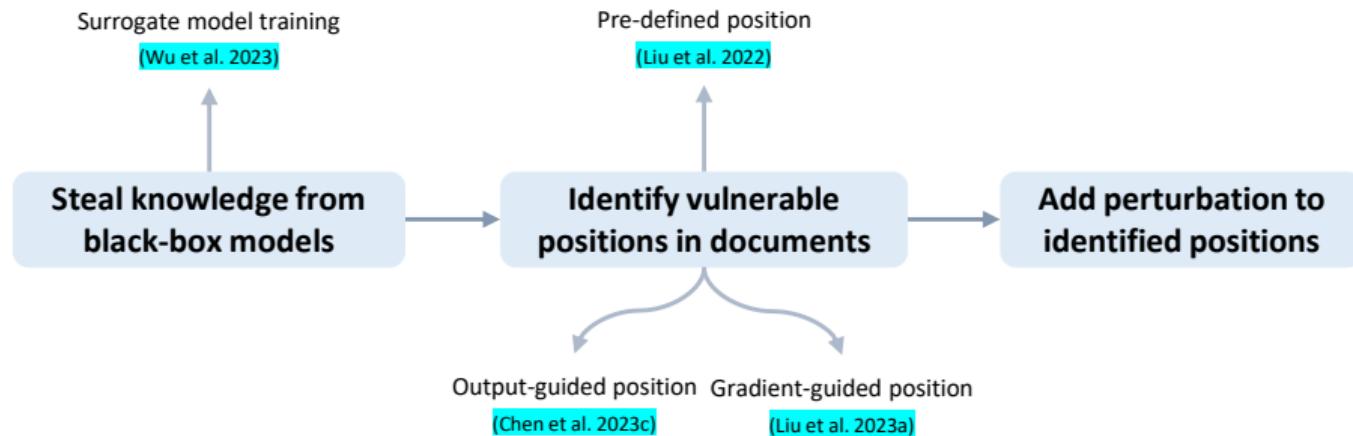
# Key steps of adversarial attacks



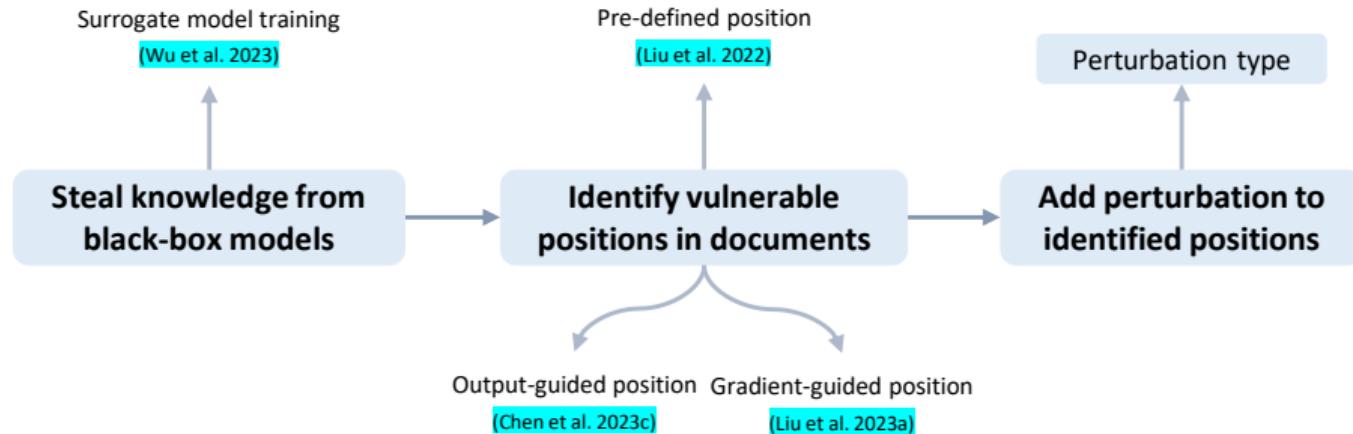
# Key steps of adversarial attacks



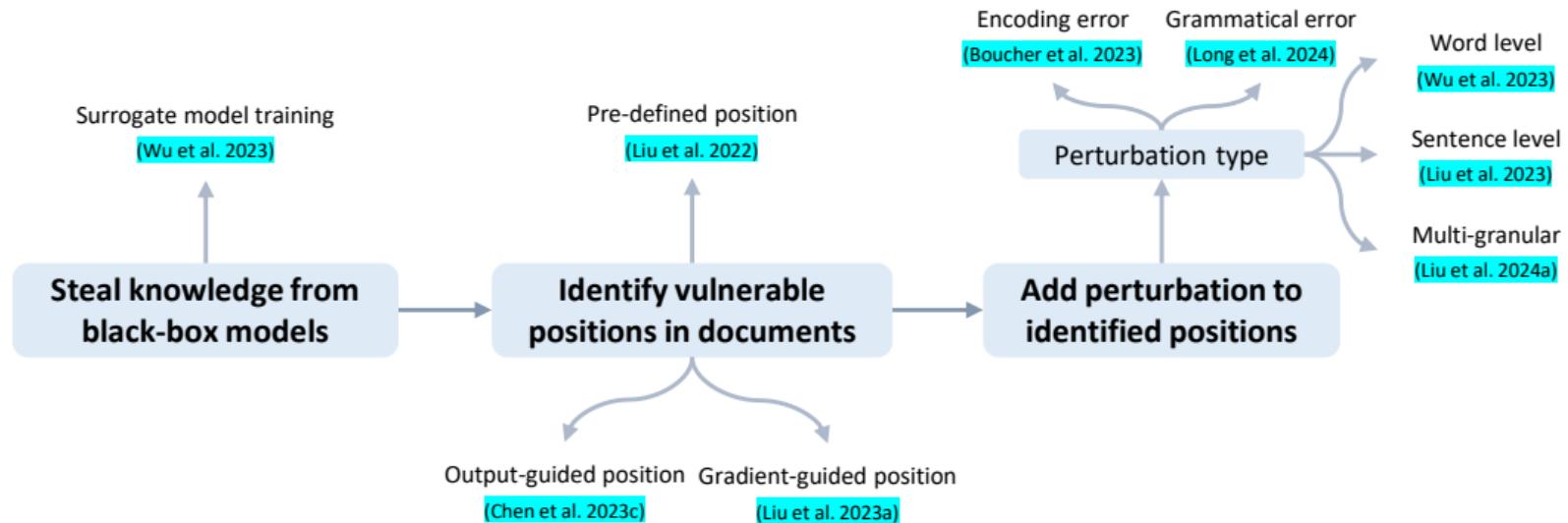
# Key steps of adversarial attacks



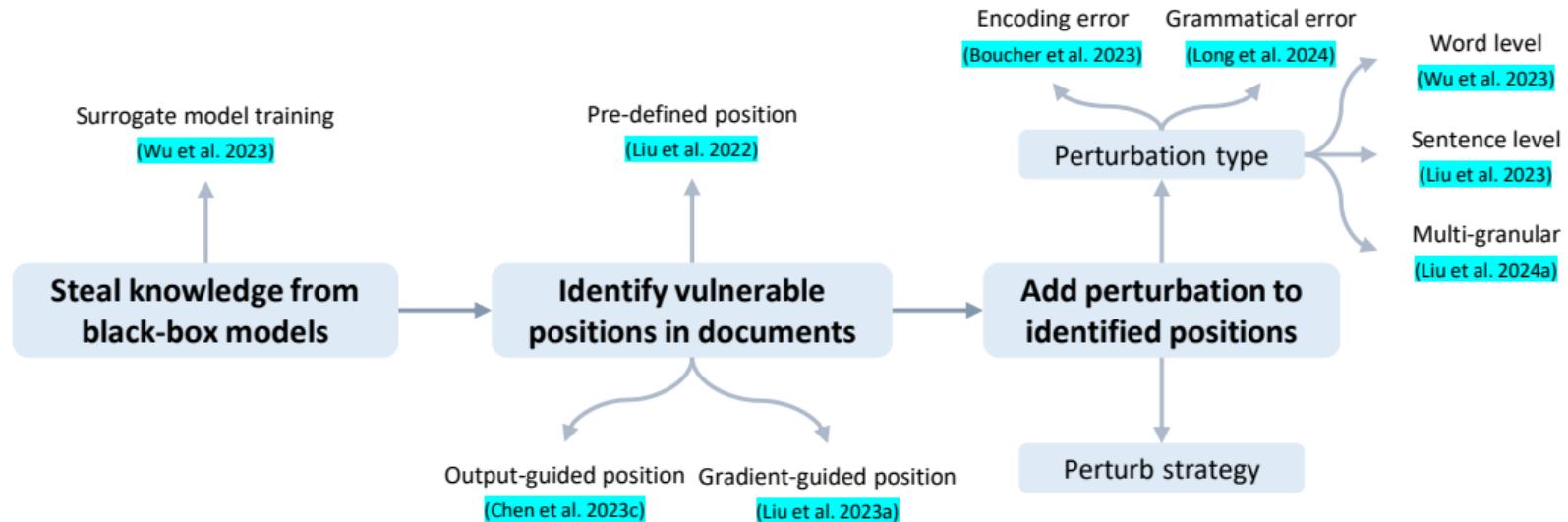
# Key steps of adversarial attacks



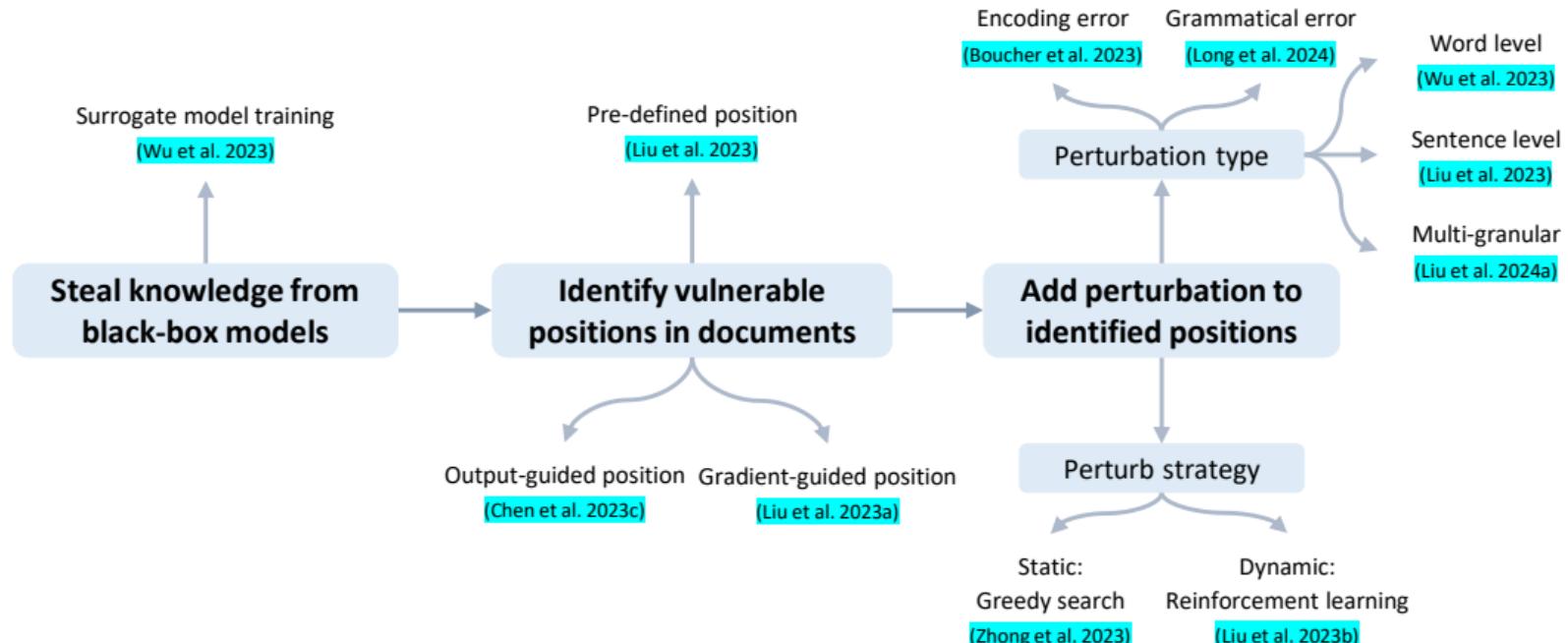
# Key steps of adversarial attacks



# Key steps of adversarial attacks



# Key steps of adversarial attacks

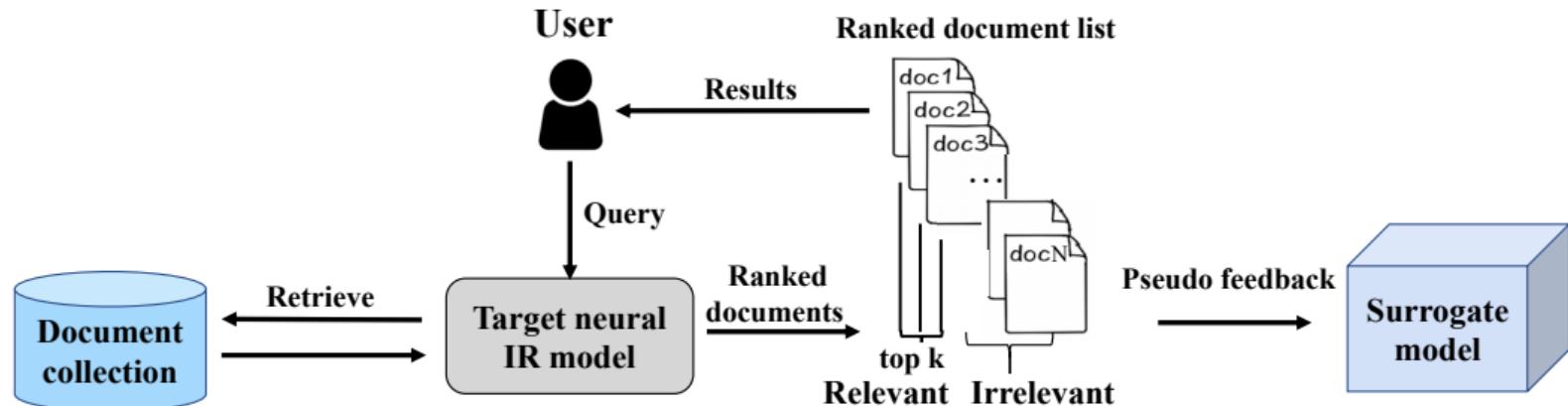


## Steal knowledge from black-box models



## Steal knowledge from black-box models: Surrogate model training

- **Objective:** Training a surrogate white-box model to steal target model knowledge
- **Approach:** Continuously querying the target model and obtaining its outputs



## Steal knowledge from black-box models: Surrogate model training

Take the idea of **pseudo-relevance feedback**:

- **Given:** a query collection  $\mathcal{Q}$ , a target model  $f$

## Steal knowledge from black-box models: Surrogate model training

Take the idea of **pseudo-relevance feedback**:

- **Given:** a query collection  $\mathcal{Q}$ , a target model  $f$
- **Get:** a rank list  $L$  returned by the target model

## Steal knowledge from black-box models: Surrogate model training

Take the idea of **pseudo-relevance feedback**:

- **Given:** a query collection  $\mathcal{Q}$ , a target model  $f$
- **Get:** a rank list  $L$  returned by the target model
- **Pseudo-labels:** take the top- $k$  ranked documents  $L[: k]$  as relevant documents and the other documents  $L[k + 1 : N]$  as irrelevant documents

## Steal knowledge from black-box models: Surrogate model training

Take the idea of **pseudo-relevance feedback**:

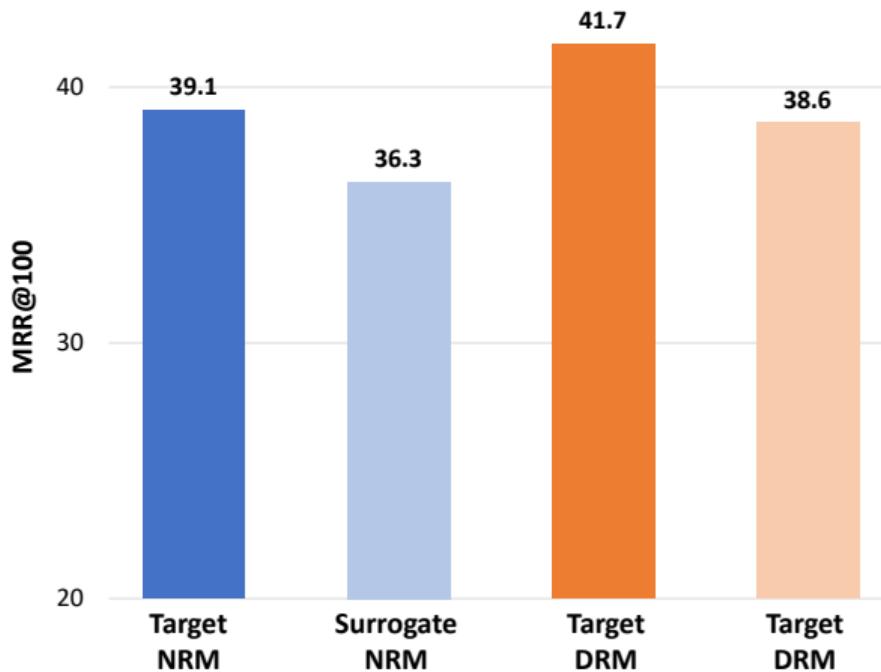
- **Given:** a query collection  $\mathcal{Q}$ , a target model  $f$
- **Get:** a rank list  $L$  returned by the target model
- **Pseudo-labels:** take the top- $k$  ranked documents  $L[: k]$  as relevant documents and the other documents  $L[k + 1 : N]$  as irrelevant documents
- **Pair-wise training:**

$$\mathcal{L} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \max(0, \eta - \tilde{f}(q, L[: k]) + \tilde{f}(q, L[k + 1 : N])),$$

Finally, we get **surrogate model  $\tilde{f}$**  that can imitate the performance of target model

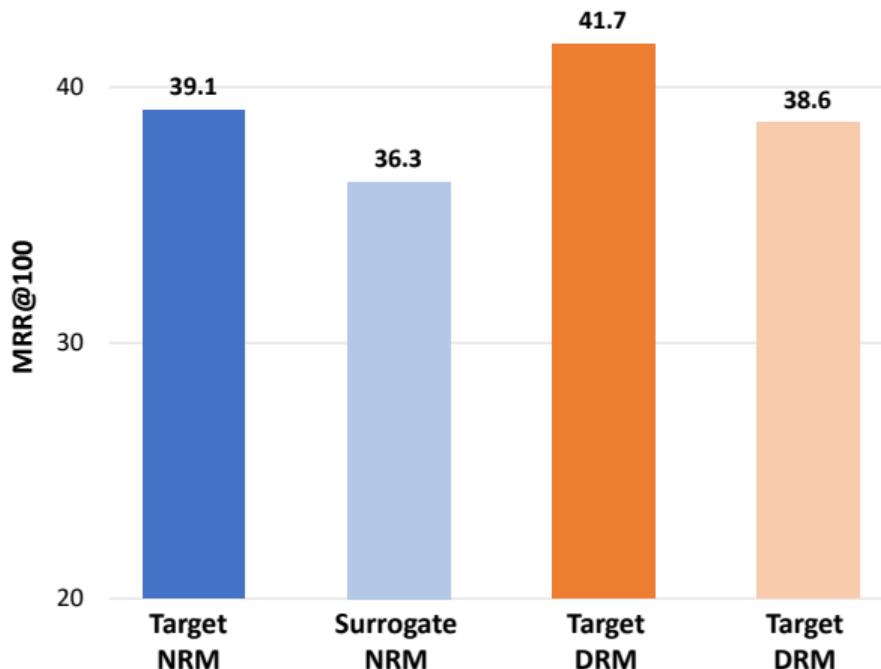
"PRADA: Practical Black-Box Adversarial Attacks Against Neural Ranking Models" [Wu et al., 2023]

## Steal knowledge from black-box models: Surrogate model training



- Dataset: MS MARCO
- Backbone:
  - Target NRM: PROP
  - Surrogate NRM: BERT-cross encoder
  - Target DRM: CoCondenser
  - Surrogate DRM: BERT-encoder

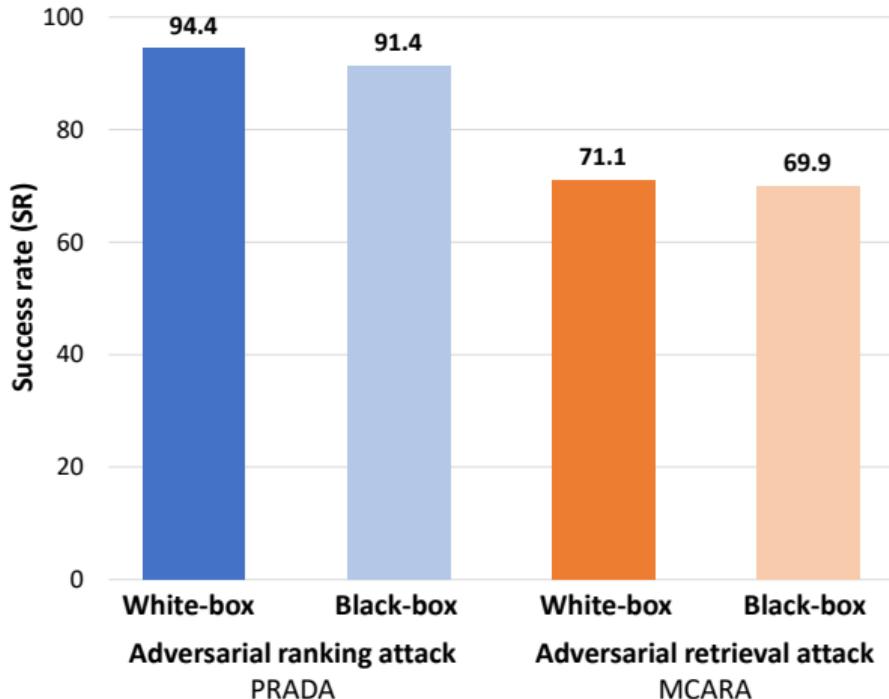
## Steal knowledge from black-box models: Surrogate model training



- Dataset: MS MARCO
- Backbone:
  - Target NRM: PROP
  - Surrogate NRM: BERT-cross encoder
  - Target DRM: CoCondenser
  - Surrogate DRM: BERT-encoder

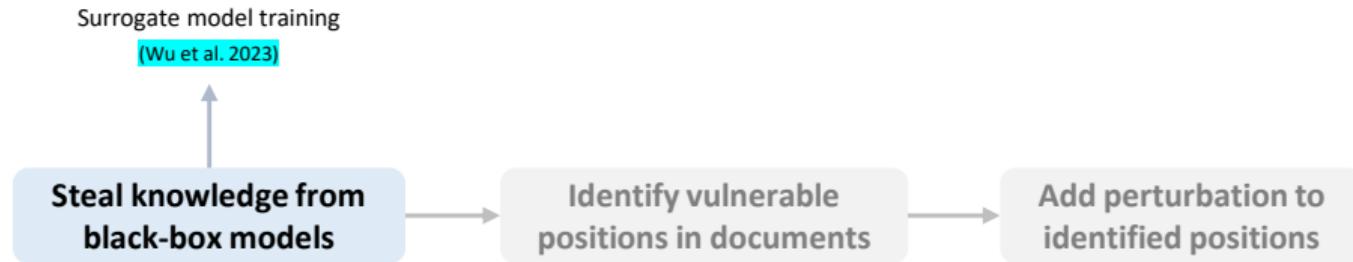
The surrogate model can **imitate the performance** of the target model

## Black-box vs. White-box setting



- Dataset: MS MARCO
- Observations: Surrogate model training can effectively transfer vulnerabilities from the target model

# Steal knowledge from black-box models



## Identify vulnerable positions



## Identify vulnerable positions

**Key idea:** Identify the positions in the low-ranked document that have greatest impact on its ranking

## Identify vulnerable positions: Pre-defined position

**Assumption:** The beginning of the document has the greatest impact on its ranking

**Pre-defined position:** Fix the perturbation position at the beginning of the document and add sentences or substitute words [Liu et al., 2022]

## Identify vulnerable positions: Pre-defined position

**Assumption:** The beginning of the document has the greatest impact on its ranking

**Pre-defined position:** Fix the perturbation position at the beginning of the document and add sentences or substitute words [[Liu et al., 2022](#)]



Simple, efficient and easy to implement

## Identify vulnerable positions: Pre-defined position

**Assumption:** The beginning of the document has the greatest impact on its ranking

**Pre-defined position:** Fix the perturbation position at the beginning of the document and add sentences or substitute words [[Liu et al., 2022](#)]



Simple, efficient and easy to implement



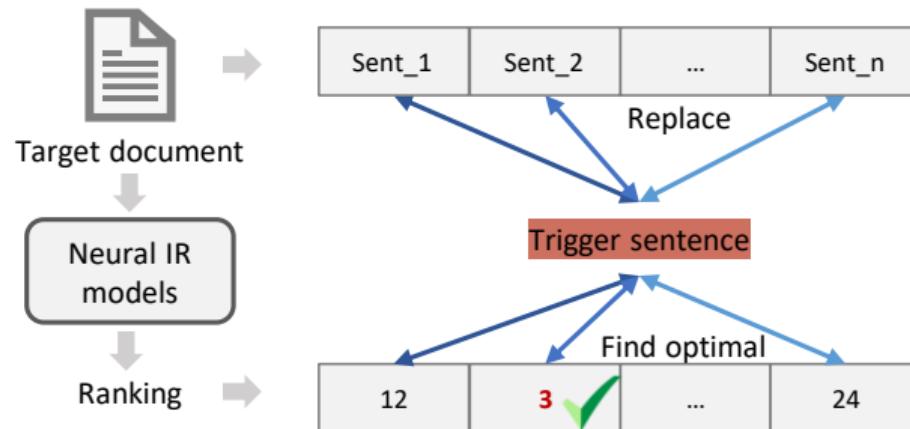
The beginning of a document is a dangerous place to be suspected



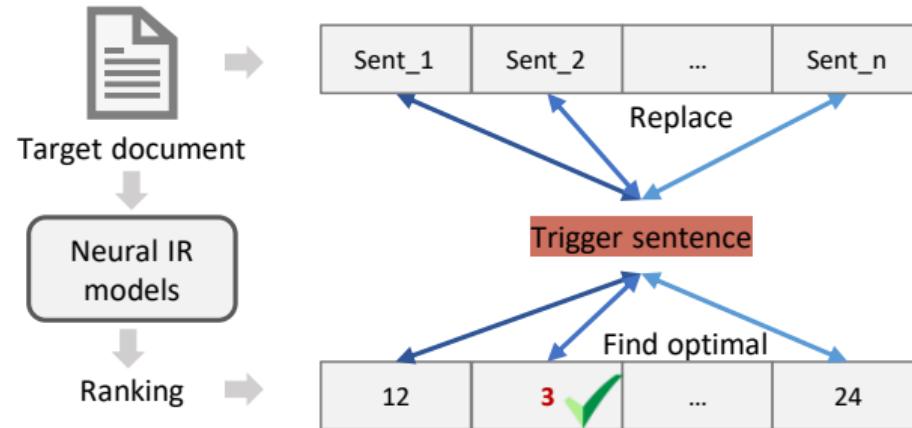
Loss of flexibility, limiting the performance of the method

## Identify vulnerable positions: Output-guided position

**Output-guided position:** Replace sentences sequentially to each position and decide the perturbation position by the relevant score of the surrogate model outputs [Chen et al., 2023d]

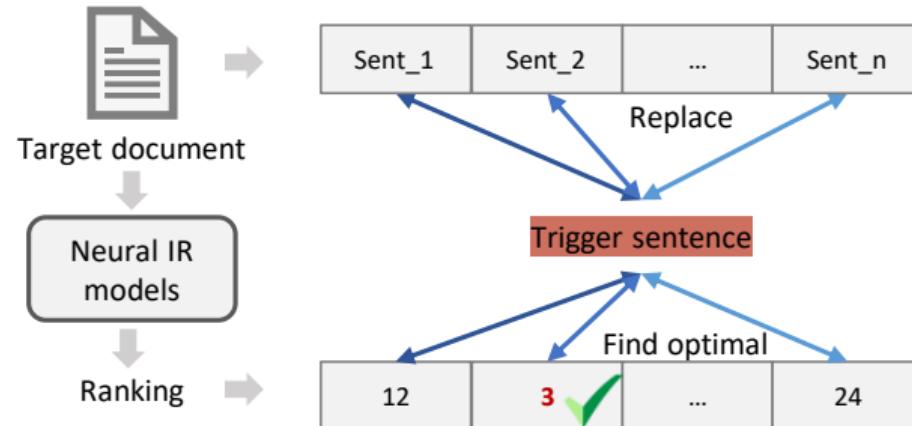


## Identify vulnerable positions: Output-guided position



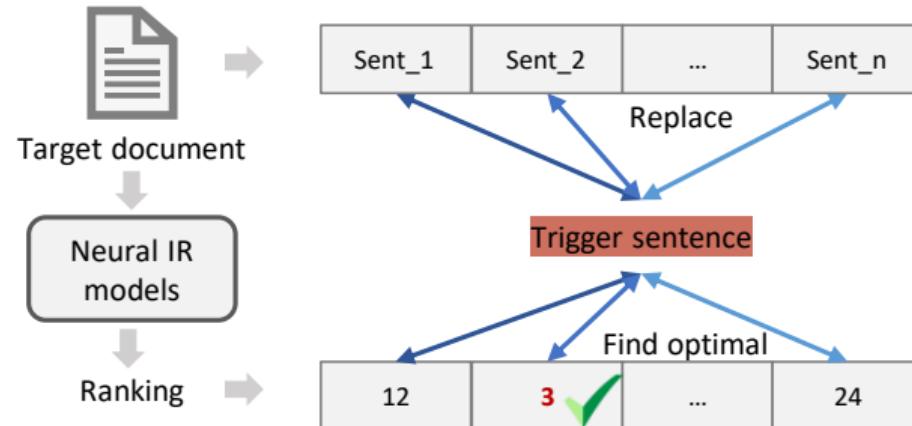
- Generate perturbations, e.g., trigger sentence

## Identify vulnerable positions: Output-guided position



- Generate perturbations, e.g., trigger sentence
- Replace original sentences one by one

## Identify vulnerable positions: Output-guided position



- Generate perturbations, e.g., trigger sentence
- Replace original sentences one by one
- Find the position that can achieve optimal ranking

## Identify vulnerable positions: Output-guided position



Straightforward: Relying on model outputs to identify positions

## Identify vulnerable positions: Output-guided position



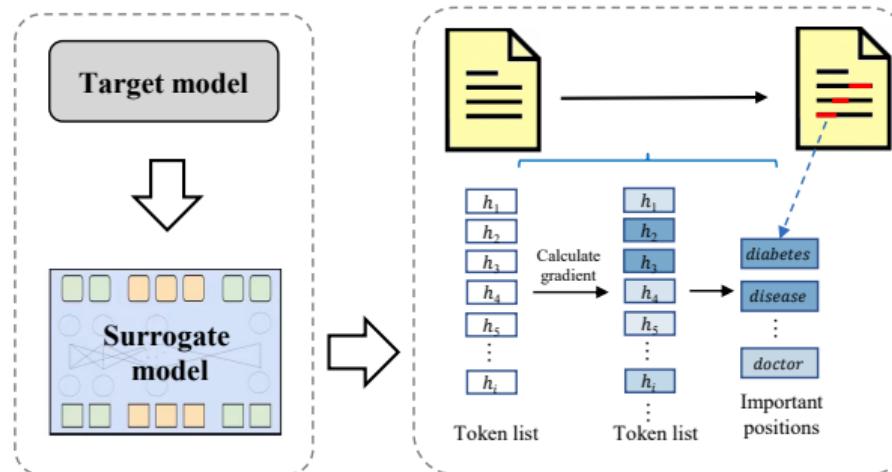
Straightforward: Relying on model outputs to identify positions



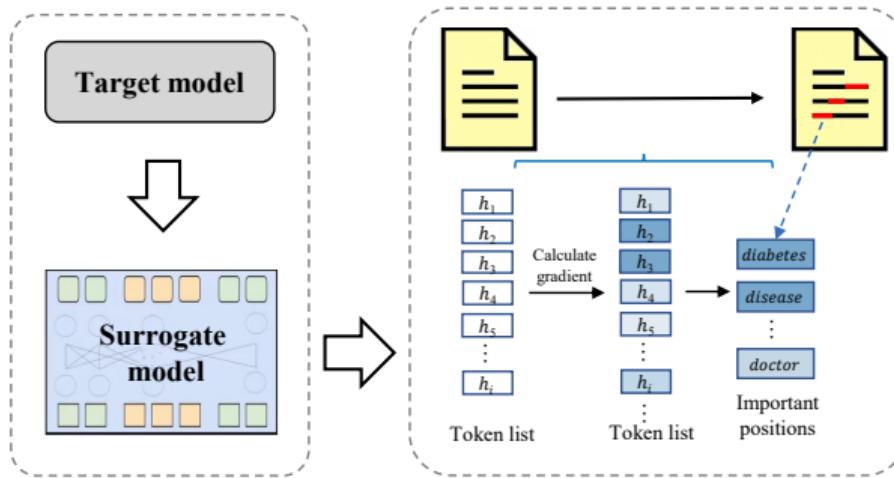
High overhead: Needing to enumerate all possible positions, only applicable to coarse-grained, e.g. sentence-level, perturbations

## Identify vulnerable positions: Gradient-guided position

**Gradient-guided position:** Calculate the gradient on the surrogate model to backpropagate to document tokens and identify important positions by large gradients [Liu et al., 2023a]

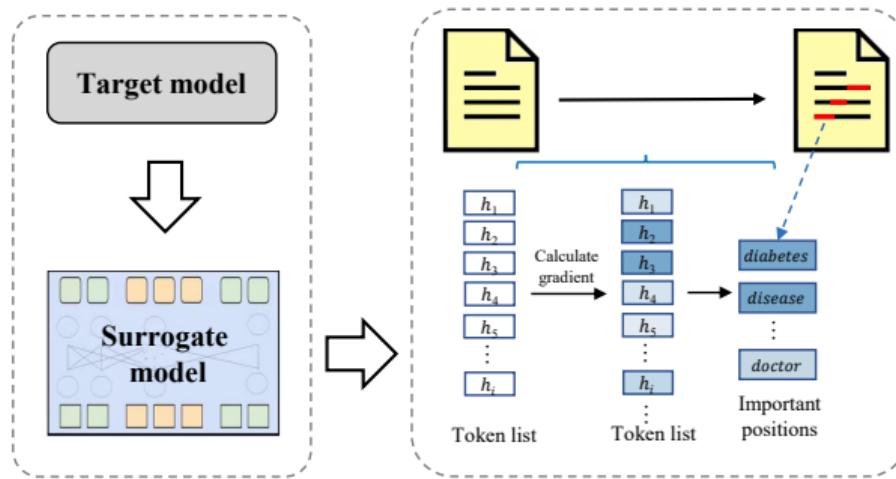


## Identify vulnerable positions: Gradient-guided position



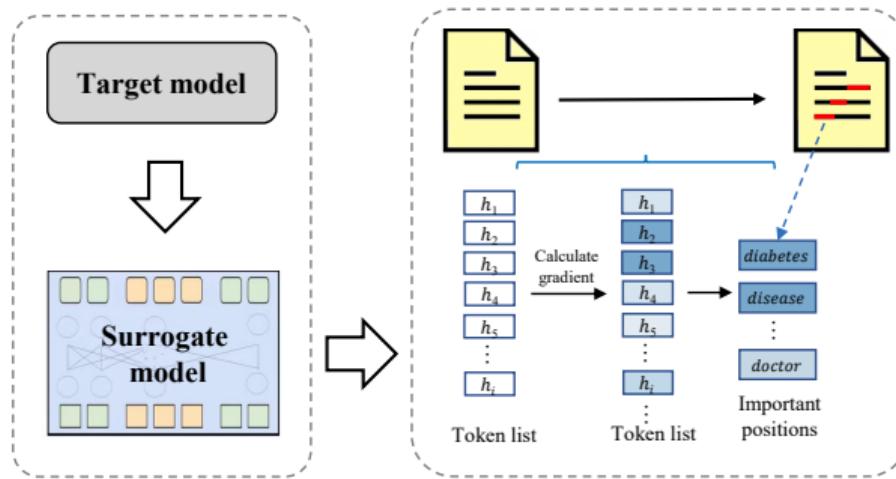
- Input the target document (with query) into the surrogate model

## Identify vulnerable positions: Gradient-guided position



- Input the target document (with query) into the surrogate model
- Calculate gradients by the loss function and back-propagate to the token embedding layer

## Identify vulnerable positions: Gradient-guided position



- Input the target document (with query) into the surrogate model
- Calculate gradients by the loss function and back-propagate to the token embedding layer
- Find tokens with large gradients as vulnerable positions in the document

## Identify vulnerable positions: Gradient-guided position



Effective: The position found is precise

## Identify vulnerable positions: Gradient-guided position

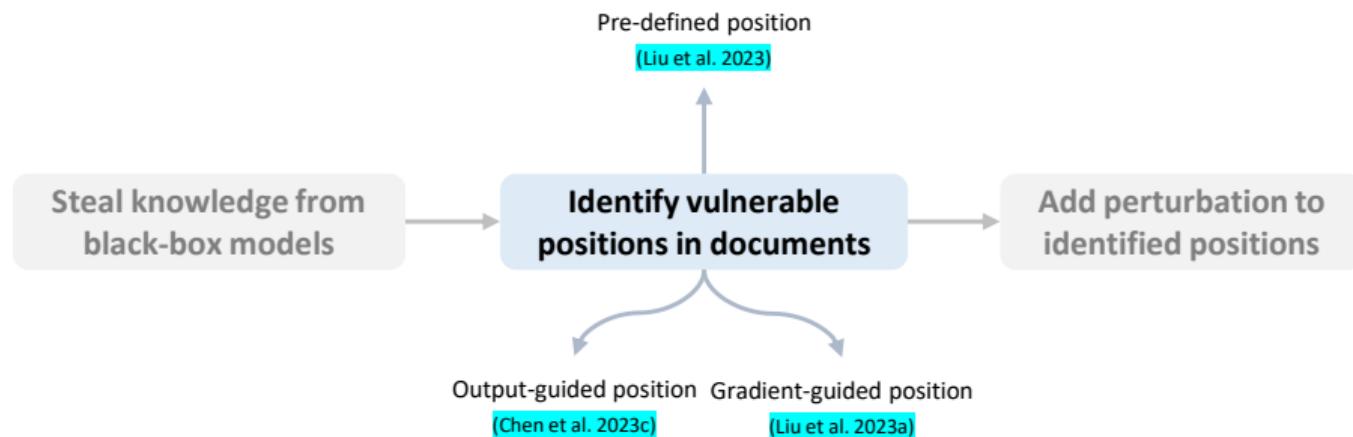


Effective: The position found is precise

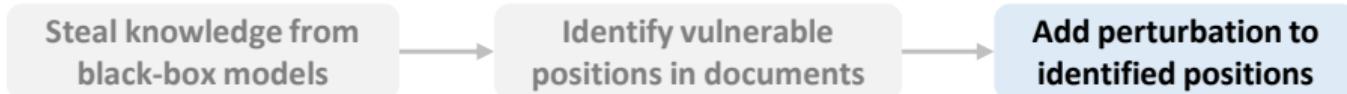


Restricted: Vulnerability position varies from document to document and may not apply to preset perturbation types

# Identify vulnerable positions



## Add perturbation to identified positions



Add perturbation to identified positions

- 1. Determine the type/types of perturbations**
- 2. Add perturbations for the identified position through a strategy**

## Add perturbation to identified positions



## Add perturbation to identified positions: Perturbation type

Selecting perturbation type is a **trade-off between attack effectiveness and naturalness**

[Query] What is the Star Wars?	Word level attack	Phase level attack	Sentence level attack
[Doc] Star Trek is a science fiction media franchise made by Gene Roddenberry, which begin with the eponymous 1960s television series. It attracted a fan cohort and emerged as an iconic symbol. More-over the franchise has expanded into various films and television series. [Rank] 98	begin ↓ began	various films ↓ several movies	It attracted a fan cohort and emerged ↓ It gained a devoted fanbase has expanded
	98→54	98→36	98→22

## Add perturbation to identified positions: Perturbation type

Selecting perturbation type is a **trade-off between attack effectiveness and naturalness**

[Query] What is the Star Wars?	Word level attack	Phase level attack	Sentence level attack
[Doc] Star Trek is a science fiction media franchise made by Gene Roddenberry, which begin with the eponymous 1960s television series. It attracted a fan cohort and emerged as an iconic symbol. More-over the franchise has expanded into various films and television series. [Rank] 98	begin ↓ began	various films ↓ several movies	It attracted a fan cohort and emerged ↓ It gained a devoted fanbase has expanded
	98→54	98→36	98→22

In general, **different scenarios** and **different query-document pairs** suit different types of perturbations

## Perturbation type based on text granularity

The regular types of perturbation are mainly based on different text granularities such as character, **word**, **sentence**, etc.

## Perturbation type based on text granularity

The regular types of perturbation are mainly based on different text granularities such as character, **word**, **sentence**, etc.

- **Word level**

- **Word substitution** [[Wu et al., 2023](#)]

- Replace words in identified positions in the document with synonyms

- Word removal, word addition ...

## Perturbation type based on text granularity

The regular types of perturbation are mainly based on different text granularities such as character, **word**, **sentence**, etc.

- **Word level**

- **Word substitution** [[Wu et al., 2023](#)]

- Replace words in identified positions in the document with synonyms

- Word removal, word addition ...

- **Sentence level**

- **Trigger injection** [[Liu et al., 2022](#)]

- Generate a sentence for a specific position in the document and inject it

- Sentence substitution, Connection sentence addition ...

## Perturbation type based on text granularity

The regular types of perturbation are mainly based on different text granularities such as character, **word**, **sentence**, etc.

- **Word level**

- **Word substitution** [Wu et al., 2023]

- Replace words in identified positions in the document with synonyms

- Word removal, word addition ...

- **Sentence level**

- **Trigger injection** [Liu et al., 2022]

- Generate a sentence for a specific position in the document and inject it

- Sentence substitution, Connection sentence addition ...

- **Multi-granular** [Liu et al., 2024a]

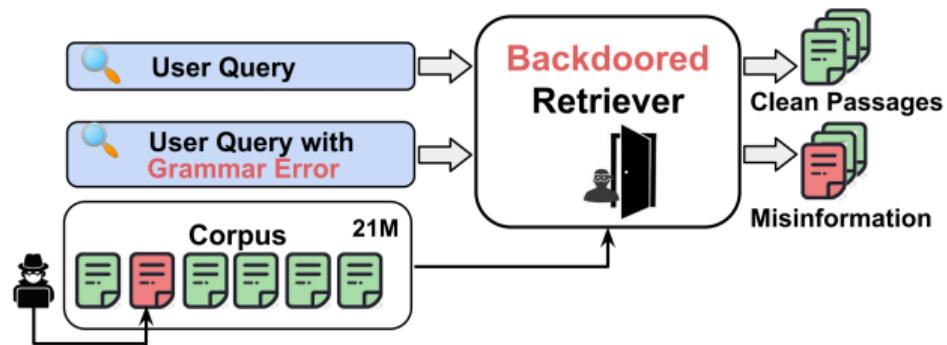
- Different types of perturbations are added according to different vulnerability positions, such as word level, phrase level, and sentence level

## Perturbation type based on special errors

Other types of perturbation are based on special errors such as [[Long et al., 2024](#)]

## Perturbation type based on special errors

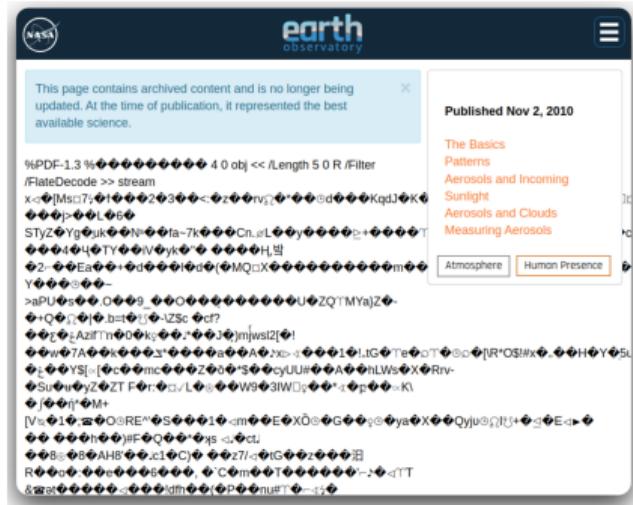
Other types of perturbation are based on special errors such as [Long et al., 2024]



**Grammatical error:** Add grammatical errors to the document so that the target document is recalled when a similar grammatical error occurs in the query

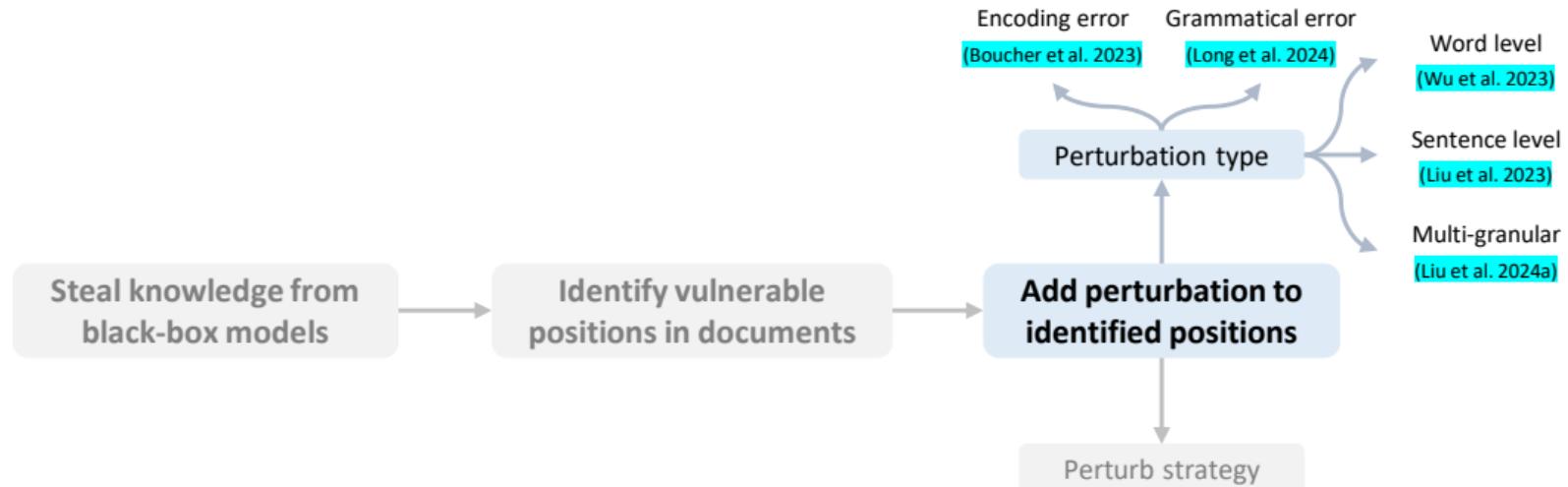
## Perturbation type based on special errors

Other types of perturbation are based on special errors such as [Boucher et al., 2023]



**Encoding error:** Use error to generate invisible perturbations, where the perturbed document appears to be unchanged, but the text encoding is different

## Add perturbation to identified positions: Perturbation type



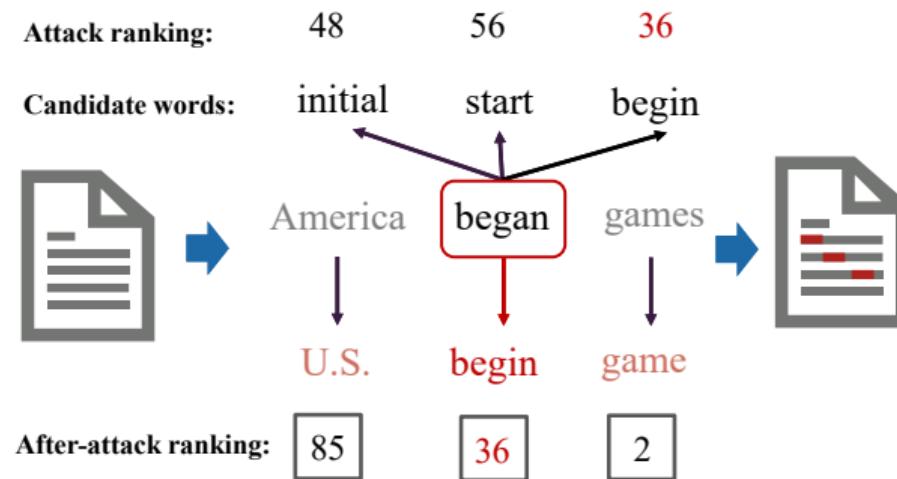
## Add Perturbation to identified positions: Perturb strategy

After determining the type of perturbation, there are two strategies, **static** and **dynamic**, for generating specific perturbations for each position:

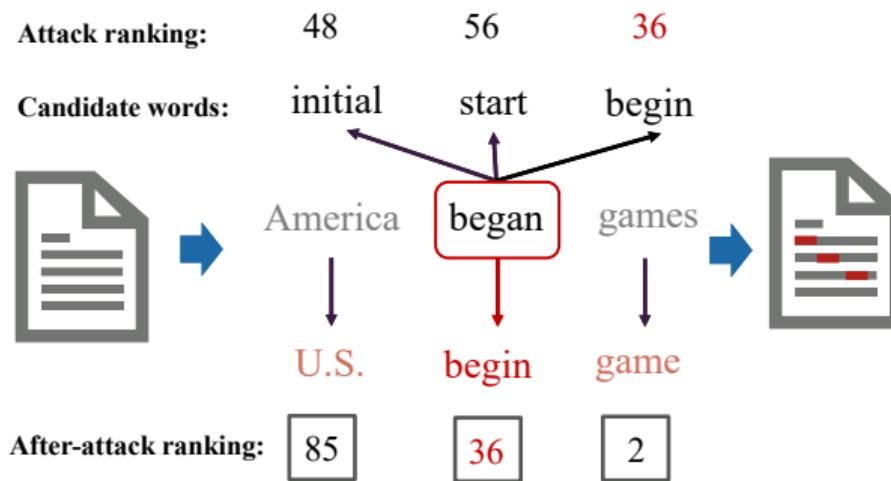
- **Static:** Greedy search
- **Dynamic:** Reinforcement learning (RL)

## Static perturb strategy: Greedy search

**Greedy-based strategy:** For each perturbation position, candidate perturbations are tried in turn, and the one with the highest rank improvement is selected as the final perturbation for the current position [Zhong et al., 2023]



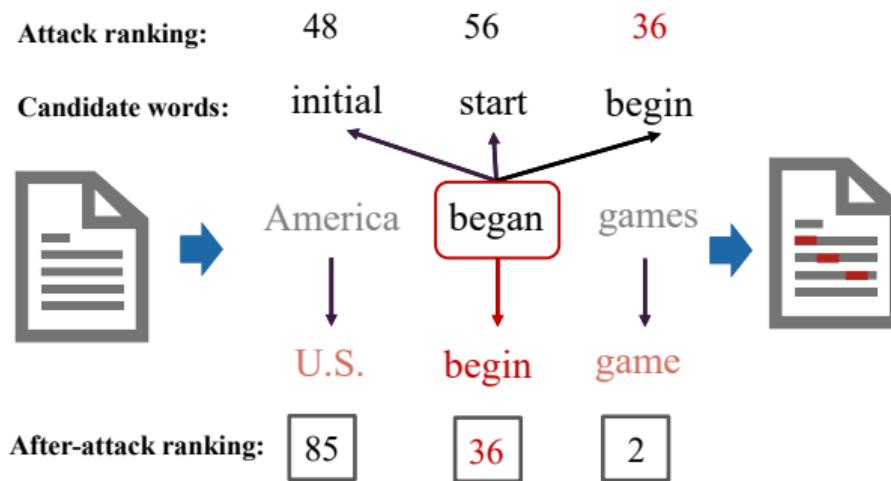
## Static perturb strategy: Greedy search



Let's take an example of word substitution. For each selected word position:

- Find synonyms in a synonym network for the current word as candidates

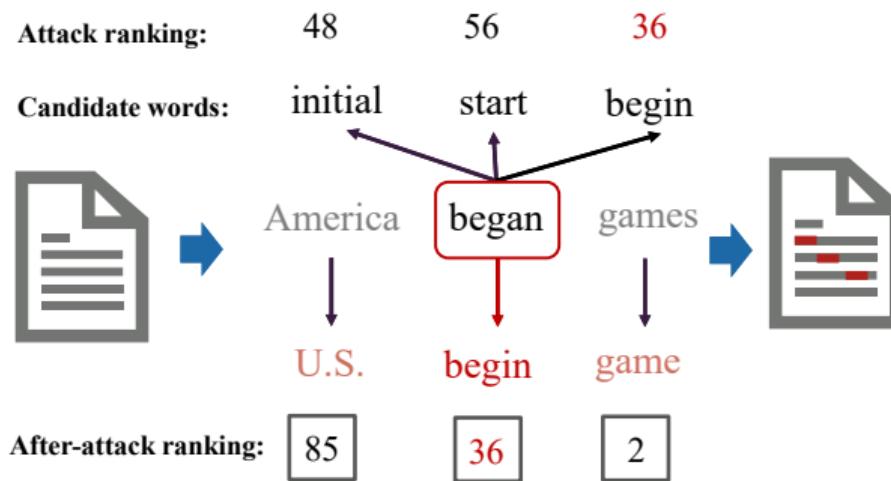
## Static perturb strategy: Greedy search



Let's take an example of word substitution. For each selected word position:

- Find synonyms in a synonym network for the current word as candidates
- Replace the words with the candidates in turn and observe the change in ranking

## Static perturb strategy: Greedy search



Let's take an example of word substitution. For each selected word position:

- Find synonyms in a synonym network for the current word as candidates
- Replace the words with the candidates in turn and observe the change in ranking
- The word that results in the largest ranking improvement as the perturbation

## Static perturb strategy: Greedy search



Simple: Easy to implement

## Static perturb strategy: Greedy search



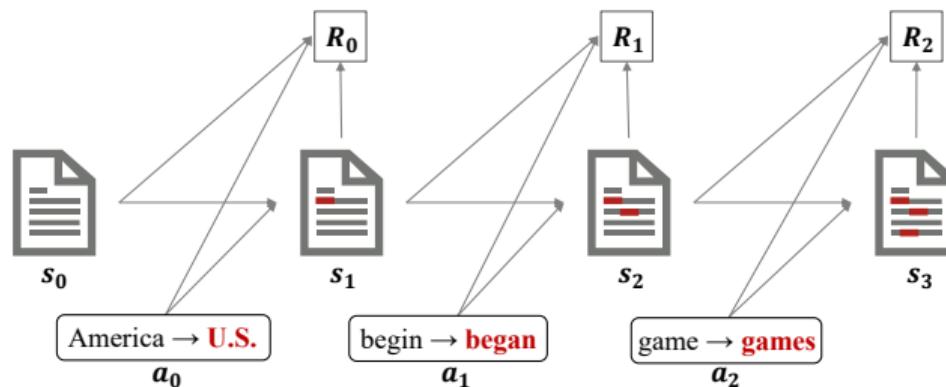
Simple: Easy to implement



Short-sighted: Ignoring the joint effect of the overall perturbation, makes it difficult to generate optimal adversarial examples

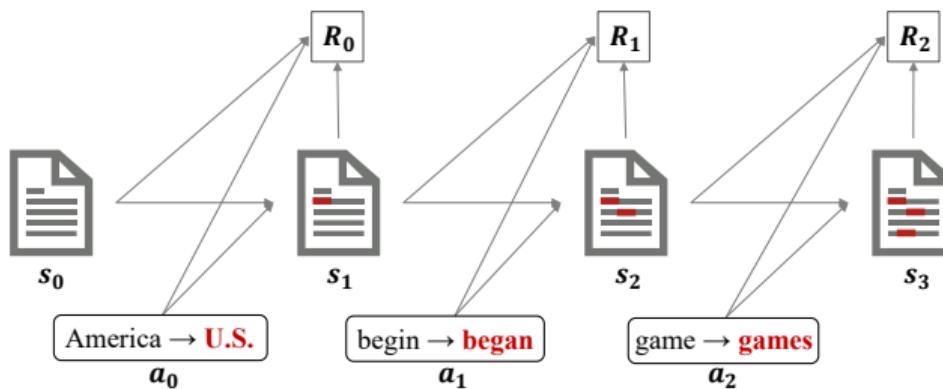
## Dynamic perturb strategy: Reinforcement learning

**RL-based strategy:** Using RL to obtain surrogate model feedback and generate appropriate perturbations based on the current ranking state [Liu et al., 2023b]



## Dynamic perturb strategy: Reinforcement learning

The attack can be modeled as a Markov decision process:



- **State:** the target document
- **Action:** adding a perturbation
- **Transition:** changes the state of the document
- **Reward:** ranking improvement

## Dynamic perturb strategy: Reinforcement learning



Reasonable: Generate the most appropriate perturbation for each state by interacting with IR models

## Dynamic perturb strategy: Reinforcement learning



Reasonable: Generate the most appropriate perturbation for each state by interacting with IR models



Complex: The implementation requires a rigorous modeling process

## Add perturbation to identified positions: Perturb strategy



# Summary

	Attack task	Vulnerable positions	Perturb strategy	Perturbation type
<b>MCARA</b> (Liu et al. 2023)	Retrieval	Gradient-guided	Greedy	Word
<b>Zhong et al. 2023</b>	Topic-oriented retrieval	Pre-defined	Greedy	Sentence
<b>Boucher et al. 2023</b>	Retrieval	Pre-defined	Greedy	Encoding error
<b>Long et al. 2024</b>	Retrieval	Pre-defined	Greedy	Grammatical error
<b>PRADA</b> (Wu et al. 2022)	Ranking	Gradient-guided	Greedy	Word
<b>PAT</b> (Liu et al. 2023)	Ranking	Pre-defined	Greedy	Sentence
<b>RELEVANT</b> (Liu et al. 2023)	Topic-oriented ranking	Gradient-guided	RL	Multi-granular
<b>IDEM</b> (Chen et al. 2023)	Ranking	Output-guided	Greedy	Sentence
<b>RL-MARA</b> (Liu et al. 2024)	Ranking	Gradient-guided	RL	Multi-granular

## Evaluation of adversarial attacks: Attack performance

Key idea: The **extent of ranking improvement** and the **impact on the top- $K$  results**

- **Attack success rate (ASR/SR)**  
Percentage of adversarial examples with improved rankings
- **Average boosted ranks (Boost/Avg.boost)**  
Average improved rankings for each adversarial examples
- **Boosted top- $K$  rate (TKR)**  
Percentage of adversarial examples that are boosted into top- $K$
- **Normalized ranking shifts rate (NRS)**  
Relative ranking improvement of adversarial examples

Key idea: The **imperceptibility**, **fluency**, and **semantic similarity**

- **Spamicity detection**

Probability of an adversarial example is spam or not

- **Grammar checkers**

Average number of grammatical errors in the adversarial examples

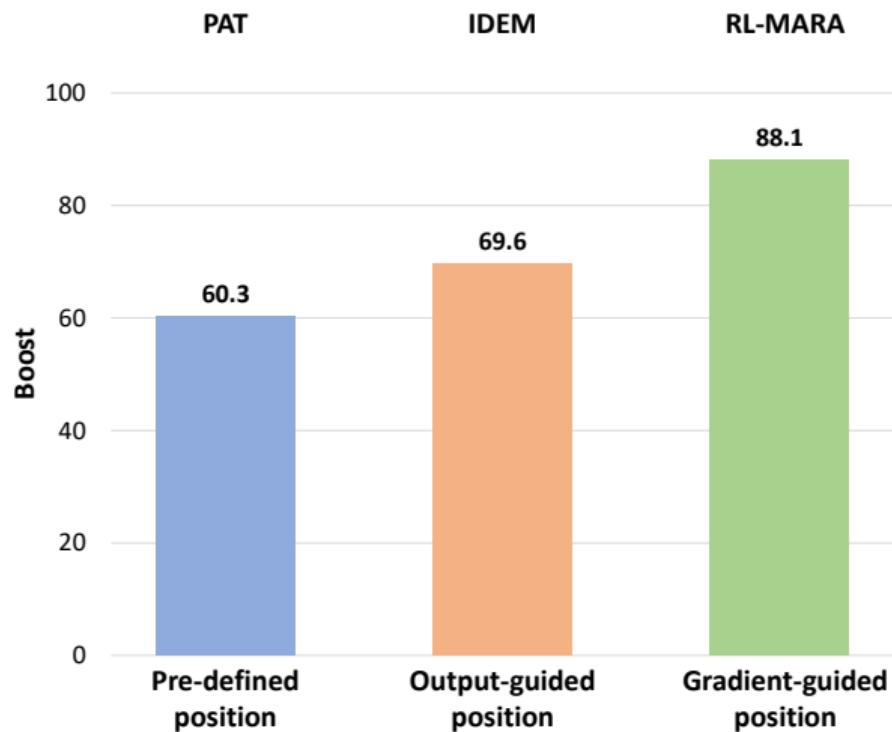
- **Language model perplexity**

Average perplexity calculated by a language model, as an indicator of fluency

- **Human evaluation**

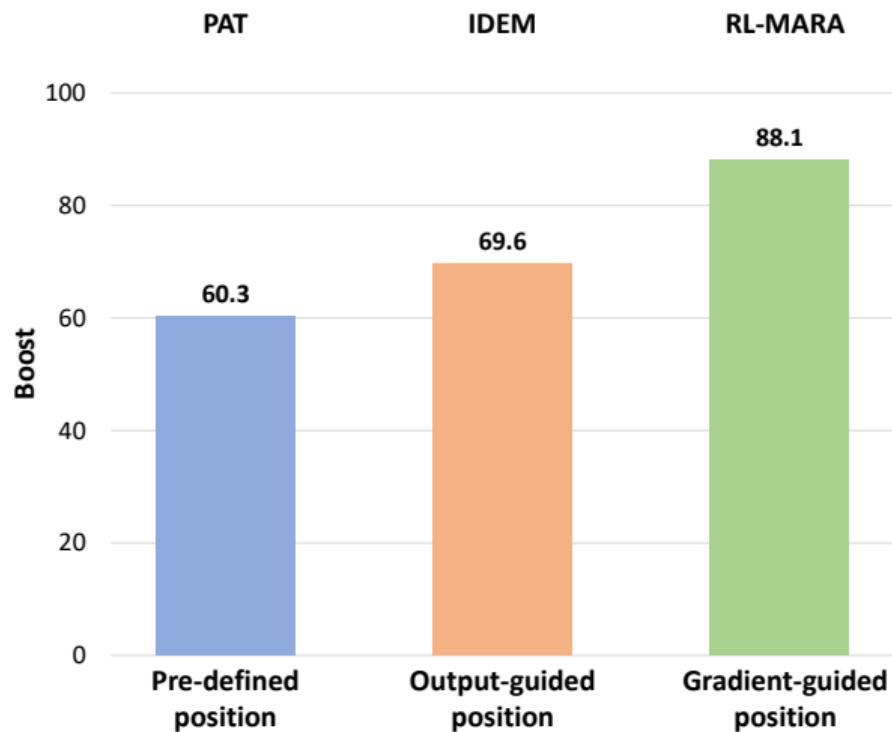
Quality of the adversarial examples w.r.t. aspects of imperceptibility, fluency, and semantic similarity

## Comparison between approaches of identifying vulnerable positions



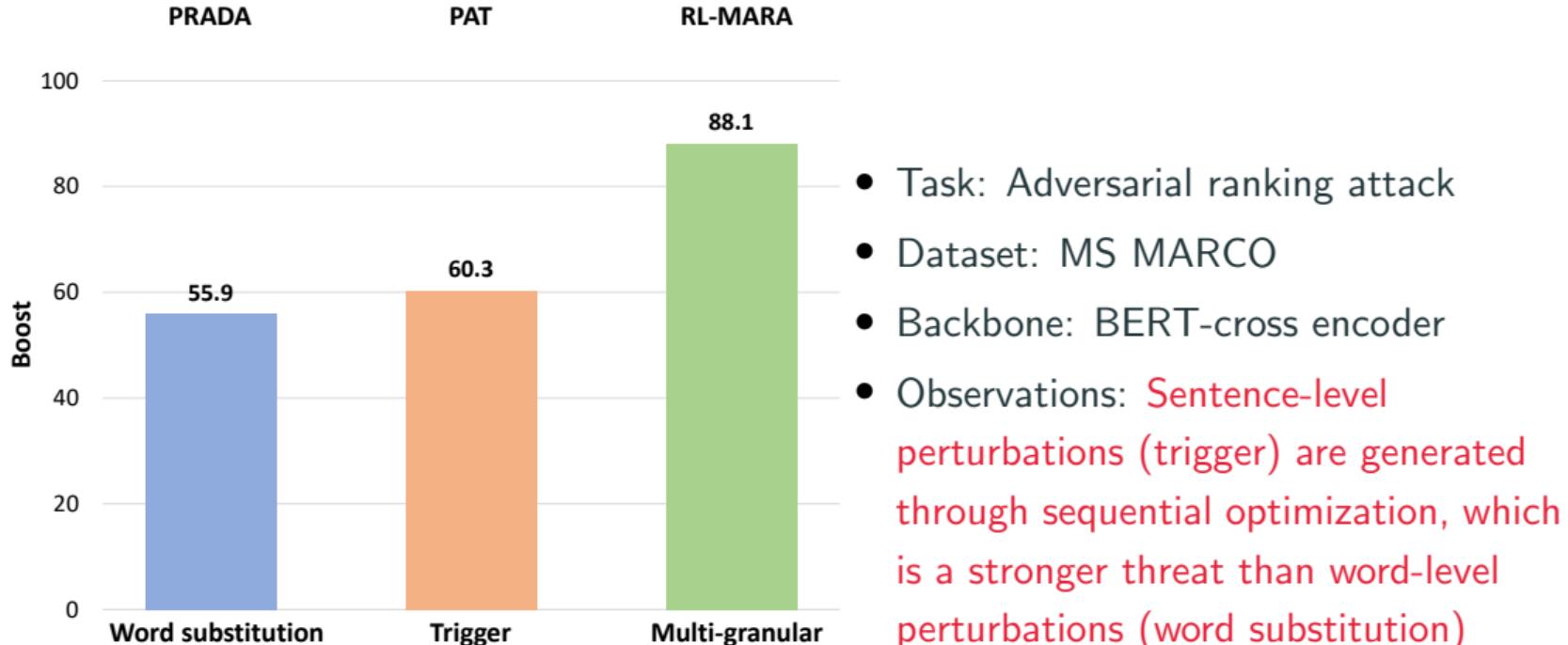
- Task: Adversarial ranking attack
- Dataset: MS MARCO
- Backbone: BERT-cross encoder
- Observations: Pre-defined positions have some effect, but flexibly identified vulnerable positions are more threatening

## Comparison between approaches of identifying vulnerable positions

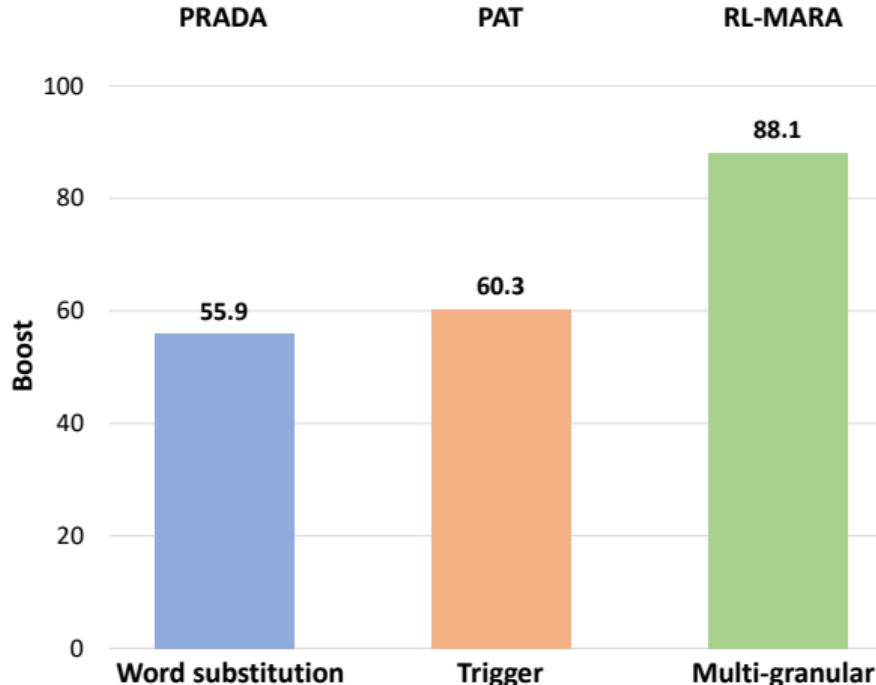


- Task: Adversarial ranking attack
- Dataset: MS MARCO
- Backbone: BERT-cross encoder
- Observations: Gradient-guided vulnerable positions directly respond to vulnerabilities inside the model, so attacks are more effective

## Comparison between perturbation types

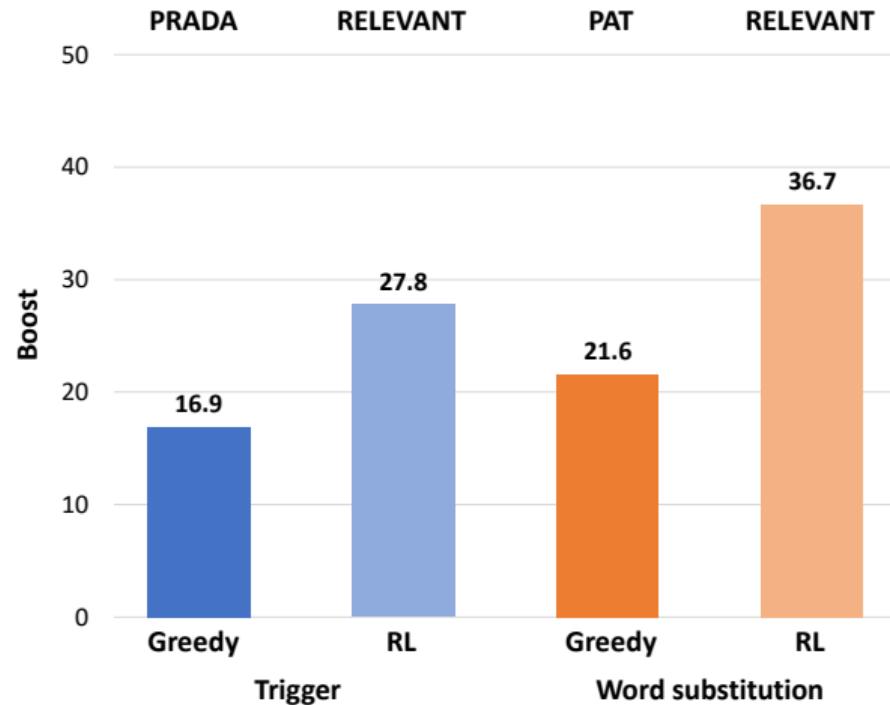


## Comparison between perturbation types



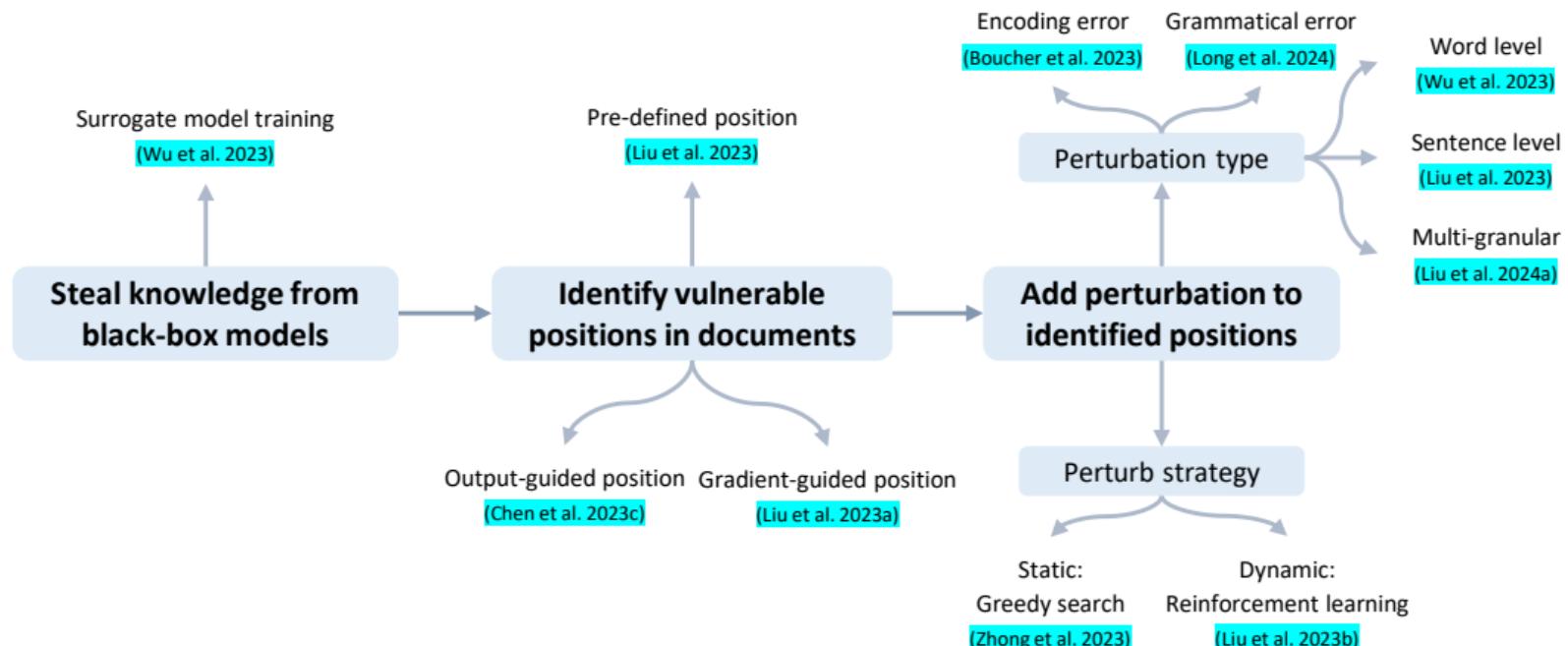
- Task: Adversarial ranking attack
- Dataset: MS MARCO
- Backbone: BERT-cross encoder
- Observations: Multi-granular perturbation allows for flexibility in adapting to a variety of vulnerable positions, and therefore more effective attacks than the first two

## Comparison between perturbing strategies



- Task: Adversarial ranking attack
- Dataset: Q-MS MARCO
- Backbone: BERT-cross encoder
- Observations: **RL-based perturbation addition strategy that dynamically adapts to the current ranking for more effective attacks**

# Key steps of adversarial attacks



## Takeaway

For adversarial attacks against neural IR models:

## Takeaway

For adversarial attacks against neural IR models:

- Restrictions make attacks simple, while flexibility makes them effective

## Takeaway

For adversarial attacks against neural IR models:

- Restrictions make attacks simple, while flexibility makes them effective
- Interaction with the target (surrogate) model is important

## Takeaway

For adversarial attacks against neural IR models:

- Restrictions make attacks simple, while flexibility makes them effective
- Interaction with the target (surrogate) model is important
- The joint combination of finding positions and adding perturbations is powerful

## Revisit two perspectives of adversarial robustness

Robustness is enhanced during the competition between **attacks** and **defenses**

- **Adversarial attacks:** Identify the vulnerability of neural IR models
- **Adversarial defenses:** Improve the adversarial robustness of neural IR models



## Requirements of adversarial defenses

When under attack, the requirements of adversarial defenses in IR including:

- Being applied during the **training or inference phase**
- **Maintaining, or even enhancing**, the performance of neural IR models
- **Guarantying stability for the top- $K$  results**

## Definition of adversarial defenses

Given:

- a neural IR model  $f$ , a metric to evaluate top- $K$  results
- an adversarial document set  $D_{\text{adv}}$  in a test set  $\mathcal{D}_{\text{test}}$
- a metric  $M$  to evaluate the ranking performance  $\mathcal{R}_M$  on top- $K$  results

The goal of adversarial defense against a neural IR model  $f$  can be formalized as:

$$\max \mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}'_{\text{test}}, K) \text{ such that } \mathcal{D}'_{\text{test}} \leftarrow \mathcal{D}_{\text{test}} \cup D_{\text{adv}}.$$

The adversarial defense task could be in the **training** or **inference** phase.

# Classification of adversarial defenses

**Training phase**

**Inference phase**

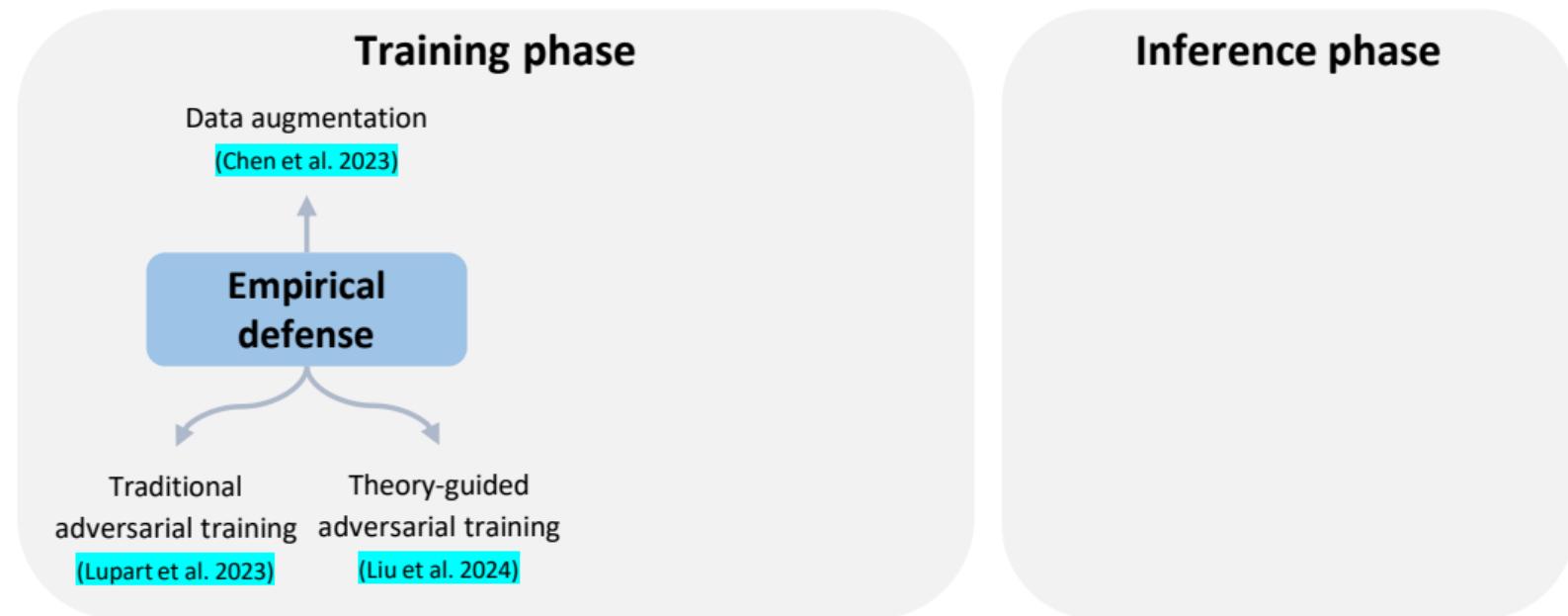
# Classification of adversarial defenses

**Training phase**

**Inference phase**

**Empirical  
defense**

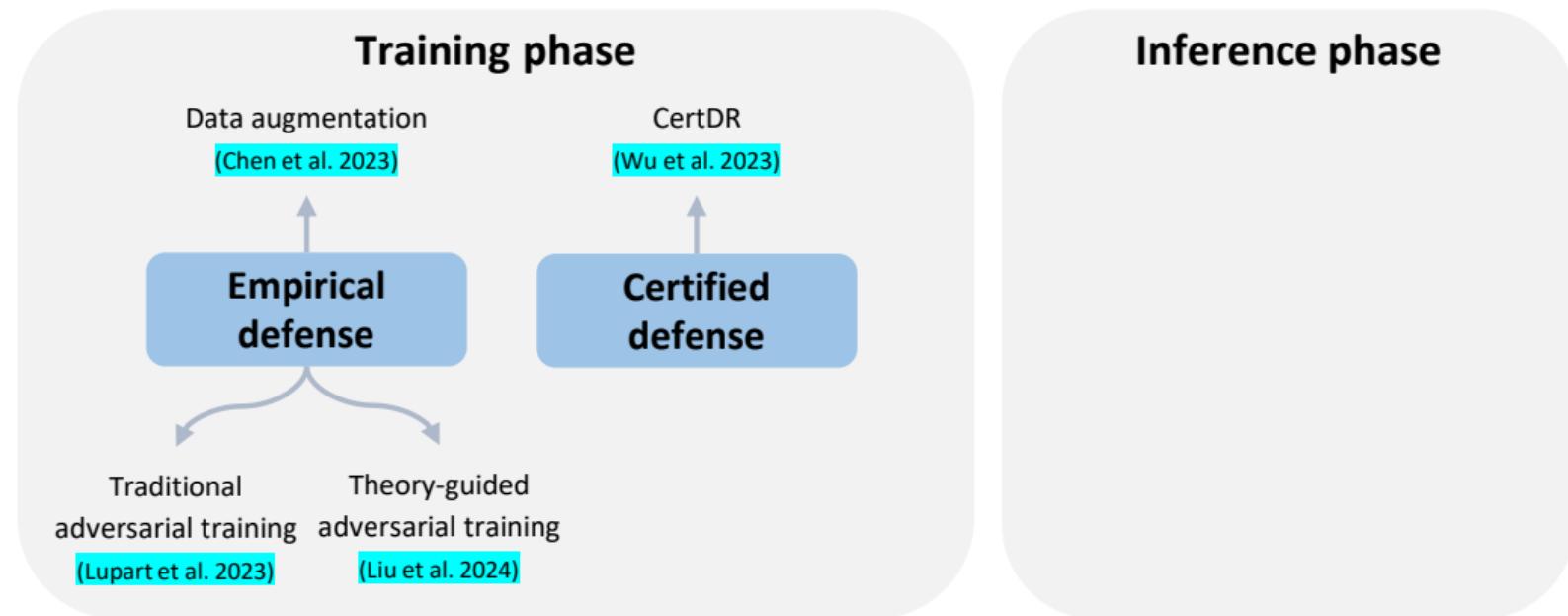
# Classification of adversarial defenses



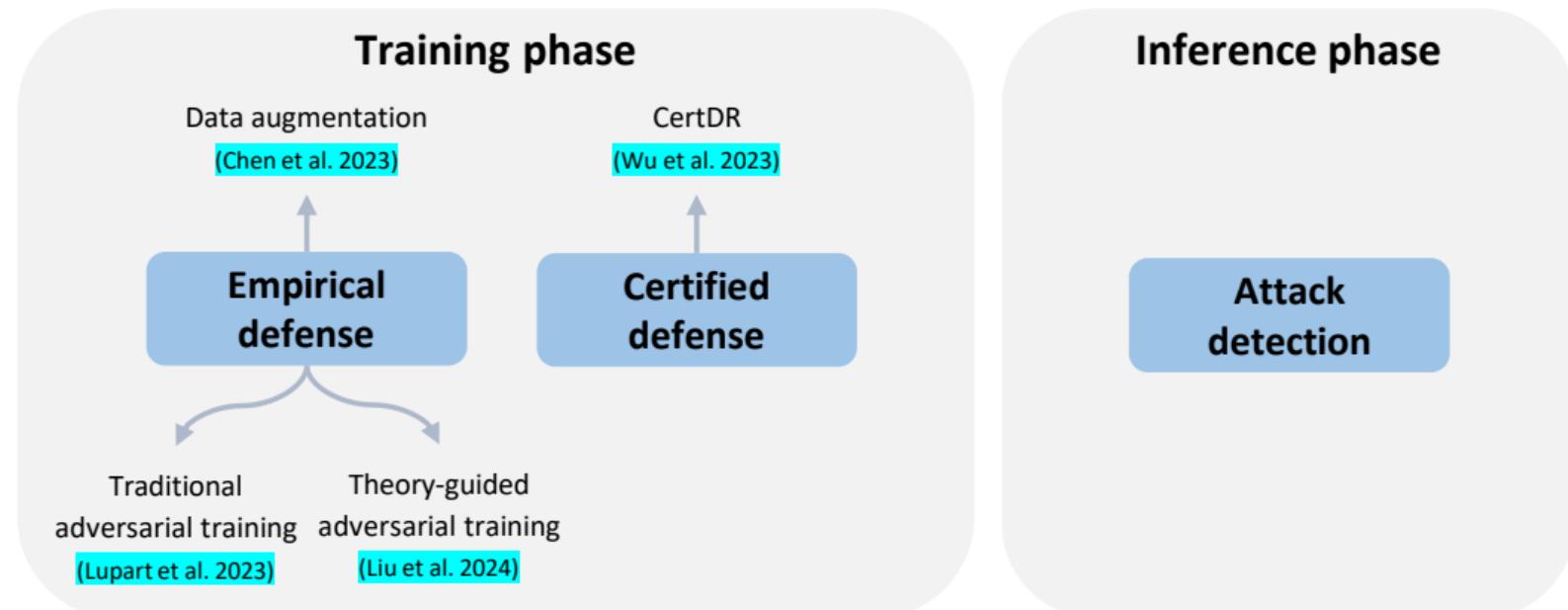
# Classification of adversarial defenses



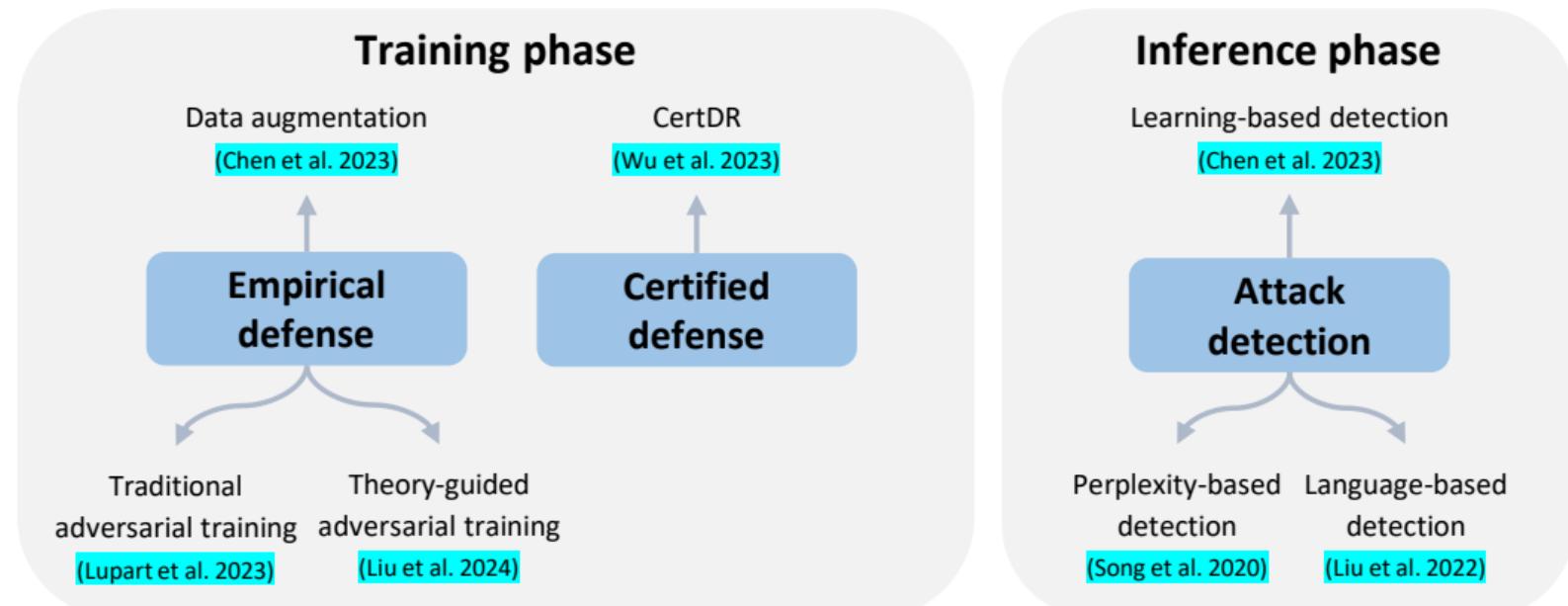
# Classification of adversarial defenses



# Classification of adversarial defenses



# Classification of adversarial defenses



# Empirical defense

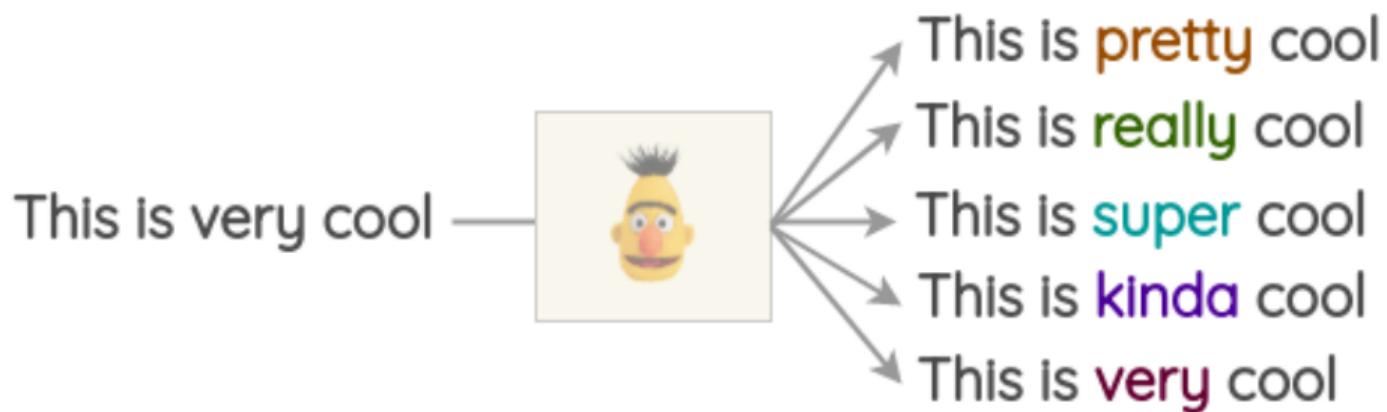


## Empirical defense

**Empirical defenses** refers to defense methods that are developed and validated through experimental data and observation. They attempt to make models empirically robust to known adversarial attacks

- Data augmentation
- Traditional adversarial training
- Theory-guided adversarial training

**Data augmentation:** For each training document, generates multiple new documents by randomly replacing words with synonyms and mixing them into the training set [Chen et al., 2023b]



## Empirical defense: Data augmentation

**Data augmentation:** For each training document, generates multiple new documents by **randomly replacing words with synonyms** and **mixing them into the training set**



Simple and low-cost: Semi-automated construction of training data

## Empirical defense: Data augmentation

**Data augmentation:** For each training document, generates multiple new documents by **randomly replacing words with synonyms** and **mixing them into the training set**



Simple and low-cost: Semi-automated construction of training data



Non-targeted: Defense is untargeted and limited in effectiveness

# Defense against: unseen attacks

### Traditional adversarial training: [Lupart and Clinchant, 2023]

- Constructs adversarial examples using existing attack methods
- Directly includes these adversarial examples into the model training along with the original examples

## Empirical defense: Traditional adversarial training

### Traditional adversarial training: [Lupart and Clinchant, 2023]

- Constructs adversarial examples using existing attack methods
- Directly includes these adversarial examples into the model training along with the original examples



Powerful: Defense is well-targeted with strong effectiveness

## Empirical defense: Traditional adversarial training

### Traditional adversarial training: [Lupart and Clinchant, 2023]

- Constructs adversarial examples using existing attack methods
- Directly includes these adversarial examples into the model training along with the original examples



Powerful: Defense is well-targeted with strong effectiveness



Costly: Constructing adversarial samples is expensive

# Defense against: seen attacks

## Empirical defense: Traditional adversarial training

The **effectiveness** and **robustness** of neural models can be odd



"Robustness May Be at Odds with Accuracy" [Tsipras et al., 2019]

## Empirical defense: Traditional adversarial training

The **effectiveness** and **robustness** of neural models can be odd



Ranking effectiveness is lost!

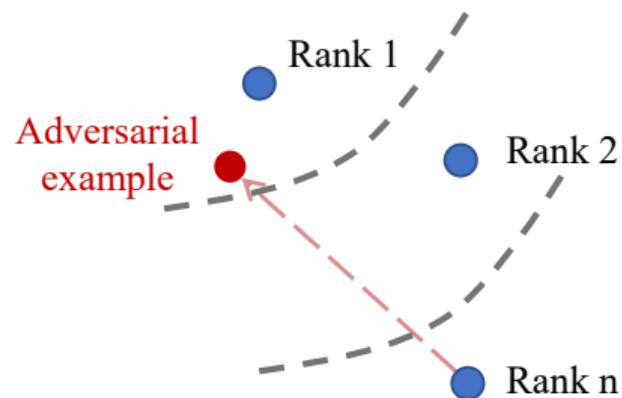
"Robustness May Be at Odds with Accuracy" [Tsipras et al., 2019]

## Empirical defense: Theory-guided adversarial training

**Theory-guided adversarial training** models the **trade-off between effectiveness and robustness** theoretically and **guides the training process through the theoretical results**

## Empirical defense: Theory-guided adversarial training

Theory-guided adversarial training models the trade-off between effectiveness and robustness theoretically and guides the training process through the theoretical results



Adversarial examples can cross the **ranking decision boundary** of the neural IR model by slight perturbations [Liu et al., 2024c]

**What causes the ranking error of neural IR models in adversarial scenarios?**

### What causes the ranking error of neural IR models in adversarial scenarios?

**Theoretically:** The robust ranking error of neural IR models can be decomposed into natural ranking error and boundary ranking error

$$\mathcal{R}_{\text{rob}}(f) = \mathcal{R}_{\text{nat}}(f) + \mathcal{R}_{\text{bdy}}(f)$$

### What causes the ranking error of neural IR models in adversarial scenarios?

**Theoretically:** The **robust ranking error** of neural IR models can be decomposed into **natural ranking error** and **boundary ranking error**

$$\mathcal{R}_{\text{rob}}(f) = \mathcal{R}_{\text{nat}}(f) + \mathcal{R}_{\text{bdy}}(f)$$

- Natural ranking error: Ranking performance on natural documents
- Boundary ranking error: Ranking performance on adversarial examples

## Empirical defense: Theory-guided adversarial training

$$\mathcal{R}_{\text{rob}}(f) = \boxed{\mathcal{R}_{\text{nat}}(f)} + \boxed{\mathcal{R}_{\text{bdy}}(f)}$$

- Natural ranking error is proven to be optimizable

## Empirical defense: Theory-guided adversarial training

$$\mathcal{R}_{\text{rob}}(f) = \mathcal{R}_{\text{nat}}(f) + \mathcal{R}_{\text{bdy}}(f)$$

- Natural ranking error is proven to be optimizable
- Boundary ranking error has a theoretical upper bound that can be indirectly optimized, that is, the perturbation invariance

## Empirical defense: Theory-guided adversarial training

$$\mathcal{R}_{\text{rob}}(f) = \mathcal{R}_{\text{nat}}(f) + \mathcal{R}_{\text{bdy}}(f)$$

- Natural ranking error is proven to be optimizable
- Boundary ranking error has a theoretical upper bound that can be indirectly optimized, that is, the **perturbation invariance**

**Perturbation invariance:** Any perturbation to the inputted documents does not change the output ranking of neural IR models

## Empirical defense: Theory-guided adversarial training

**Perturbation-invariant adversarial training:** Using the natural and adversarial ranking loss to improve the **trade-off between effectiveness and robustness**

$$\mathcal{L} = \lambda \mathcal{L}_{\text{nat}} + (1 - \lambda) \mathcal{L}_{\text{adv}}$$

**Perturbation-invariant adversarial training:** Using the natural and adversarial ranking loss to improve the **trade-off between effectiveness and robustness**

$$\mathcal{L} = \lambda \mathcal{L}_{\text{nat}} + (1 - \lambda) \mathcal{L}_{\text{adv}}$$

- Natural ranking loss is a pair-wise loss that optimize natural ranking error

## Empirical defense: Theory-guided adversarial training

**Perturbation-invariant adversarial training:** Using the natural and adversarial ranking loss to improve the **trade-off between effectiveness and robustness**

$$\mathcal{L} = \lambda \mathcal{L}_{\text{nat}} + (1 - \lambda) \mathcal{L}_{\text{adv}}$$

- Natural ranking loss is a pair-wise loss that optimize natural ranking error
- Adversarial ranking loss is a list-wise loss that optimize perturbation invariance

## Empirical defense: Theory-guided adversarial training



Balanced: A good trade-off between effectiveness and robustness can be achieved

## Empirical defense: Theory-guided adversarial training



Balanced: A good trade-off between effectiveness and robustness can be achieved



Limited: Still only against seen attacks

# Defense against: seen attacks

# Review empirical defense



## Review empirical defense



Strong defense, suitable for targeting specific attack methods

## Review empirical defense



Strong defense, suitable for targeting specific attack methods



Poor performance against unseen attacks, partly lacking theoretical guarantees

## Training phase

Empirical  
defense

**Certified  
defense**

## Inference phase

Attack  
detection

## Certified defense

Empirical defenses usually only protect against seen attacks and perform poorly against  
unseen attacks

Empirical defenses usually only protect against seen attacks and perform poorly against  
unseen attacks

In the real world, new types of attacks are popping up all over the place



## Certified defense

Relying solely on empirical defenses to counter attacks turns model deployment into a never-ending game of cat and mouse



Tom Chasing Jerry

## Certified defense

**Certified defense** refers to methods that are primarily based on mathematical theories to protect against various types of attacks.

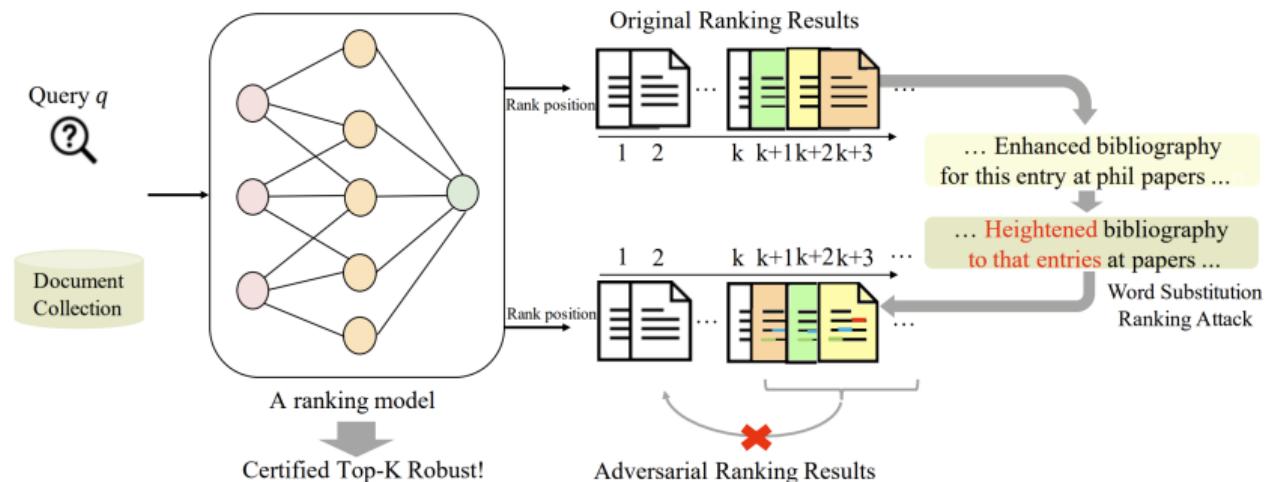
Unlike empirical defenses, which rely on experimental data, certified defenses are developed through analytical reasoning and mathematical proofs.

## Certified defense: Certified robustness

A model is said to be certified robust if an attack is theoretically guaranteed to fail, no matter how the attacker manipulates the input [Wu et al., 2022a]

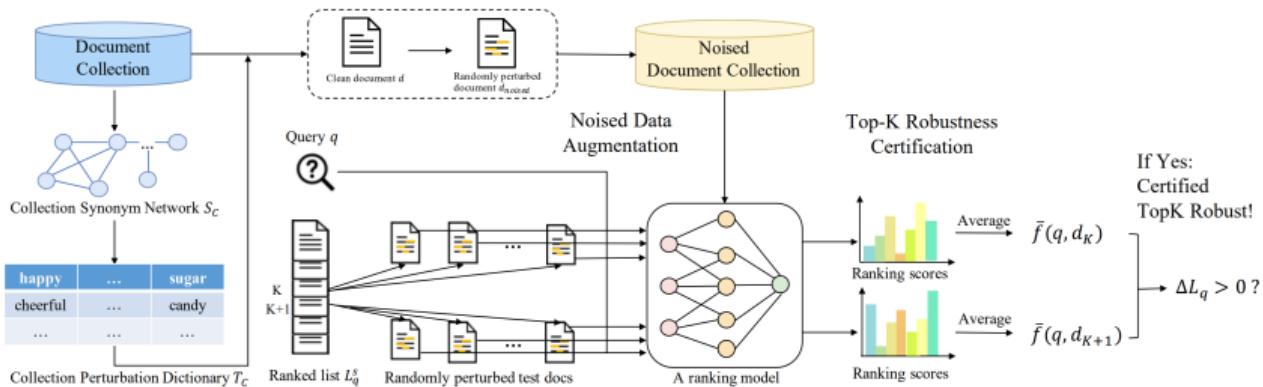
## Certified defense: Certified robustness

A model is said to be certified robust if an attack is theoretically guaranteed to fail, no matter how the attacker manipulates the input [Wu et al., 2022a]



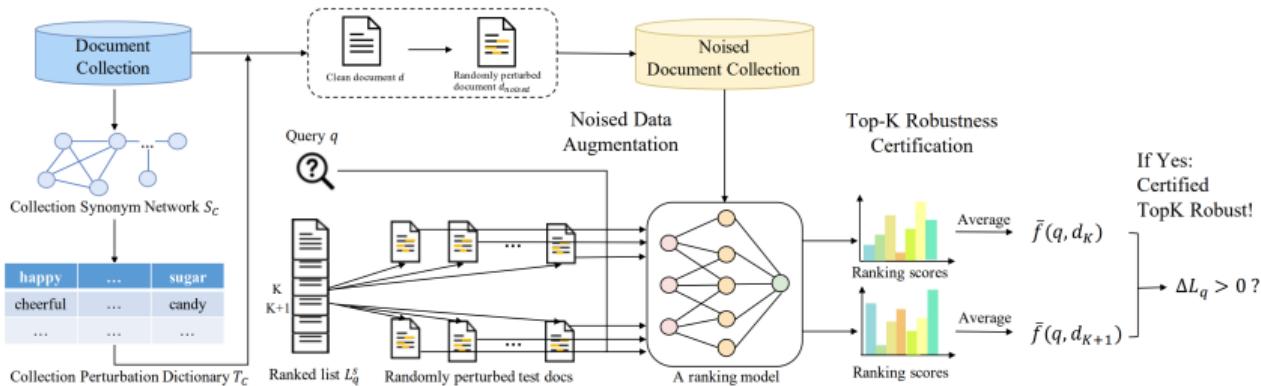
**Certified Top- $K$  Robustness:** A ranking model can keep all the adversarial examples away from the top- $K$  results under any attack

# Certified defense: Method



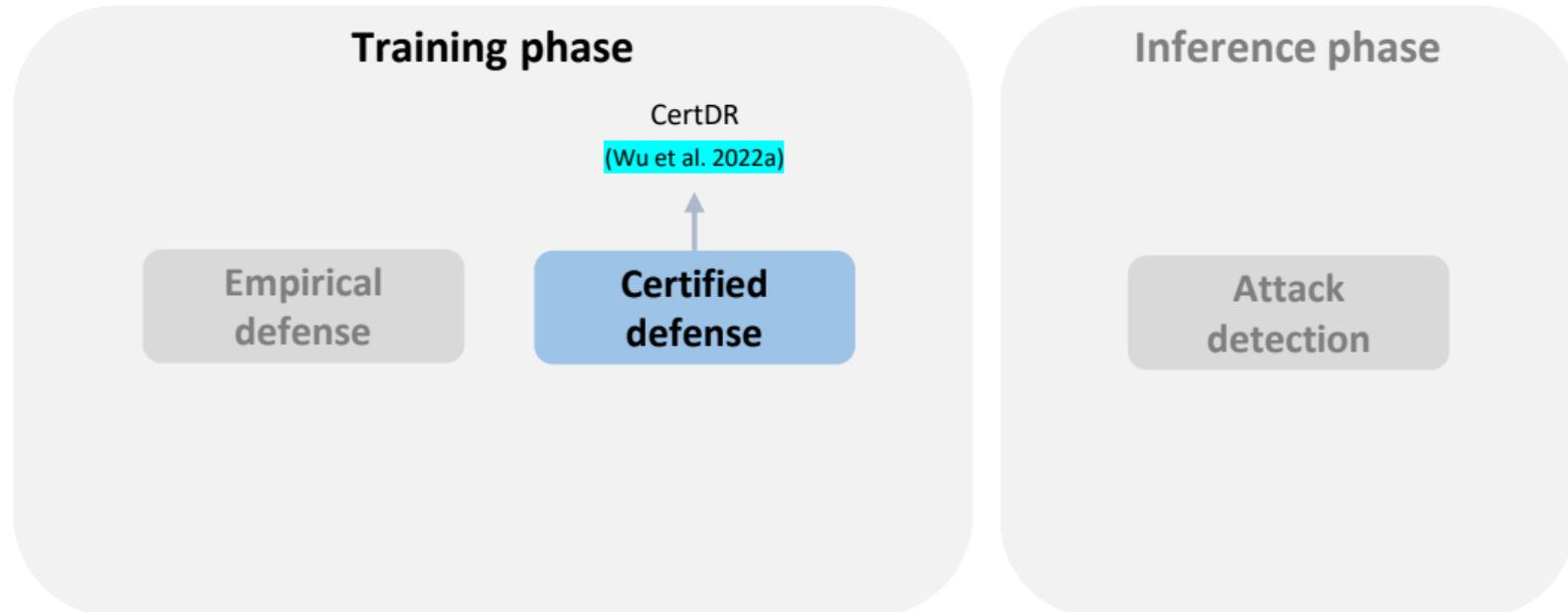
- Train a randomized smoothed ranker by voting of randomly perturbed samples derived from the original input

## Certified defense: Method



- Train a randomized smoothed ranker by voting of randomly perturbed samples derived from the original input
- Leverage the ranking property jointly with the statistical property of the ensemble to provably certify top- $L$  robustness

## Review certified defense



## Review certified defense



Reliable: Defend against any attacks within a limited range

## Review certified defense



Reliable: Defend against any attacks within a limited range



Significant: Make it possible to end the arms race between attack and defense

## Review certified defense



Reliable: Defend against any attacks within a limited range



Significant: Make it possible to end the arms race between attack and defense



Lossy: Cause decline in ranking performance

# Defense against: unseen attacks

# Attack detection



## Attack detection

**Attack detection** acts in the inference phase of the model, where different detectors determine whether a candidate document contains adversarial samples or not

Format:

- Point-wise detection
- List-wise detection

Method:

- Perplexity-based detection
- Language-based detection
- Learning-based detection

## Attack detection: Format



- **Point-wise detection** primarily emphasizes the overall **accuracy** of the detection
- **List-wise detection** further considers the **ranking quality** (e.g., MRR metric) of the final ranking list [Chen et al., 2023c]

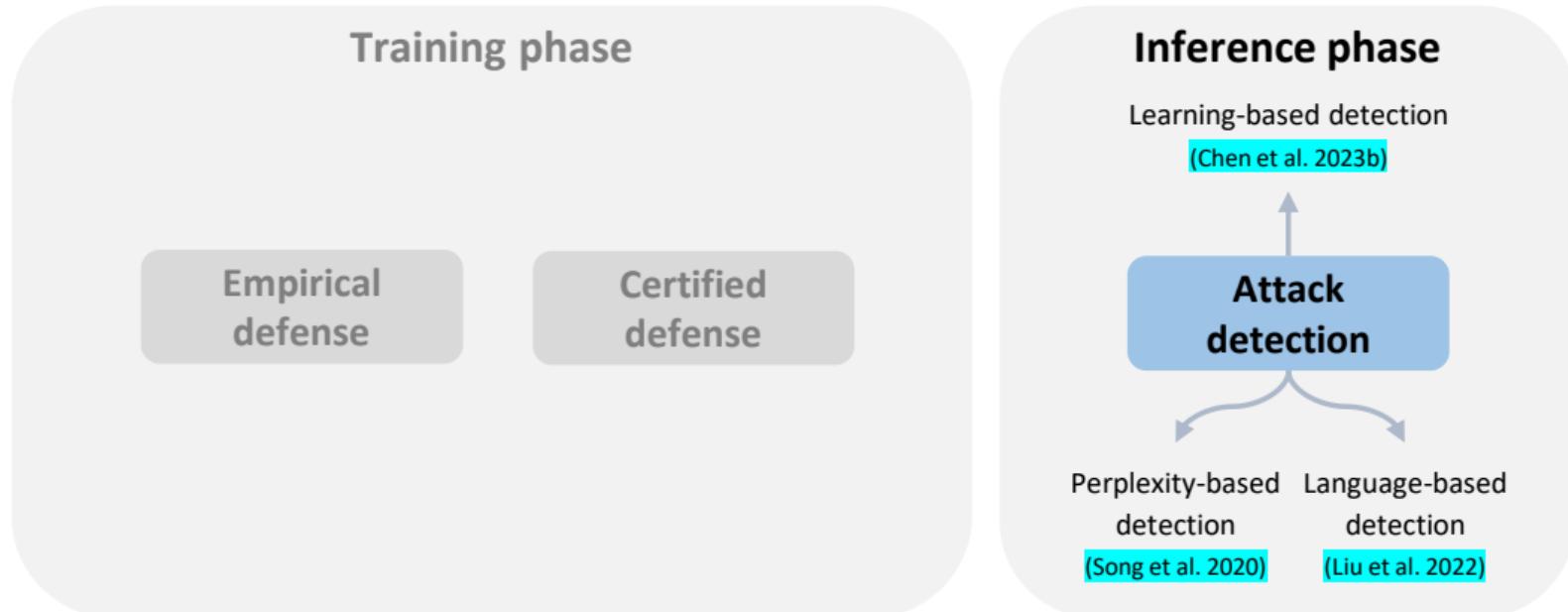
## Attack detection: Method

**Perplexity-based detection** (unseen attacks) mainly uses the difference in the distribution of perplexity (PPL) between the adversarial samples and the original document under the language model [Song et al., 2020]

**Language-based detection** (unseen attacks) employs a classification model pre-trained on the Linguistic Acceptability dataset to determine the grammaticality of the document text [Liu et al., 2022]

**Learning-based detection** (seen attacks) opts to fine-tune a classification model using the original and adversarial document pairs present in the dataset of generated adversarial examples [Chen et al., 2023c]

# Review attack detection



## Review attack detection



Lightweight: Easy to deploy, reducing the cost of defense in the training process of neural IR models

## Review attack detection

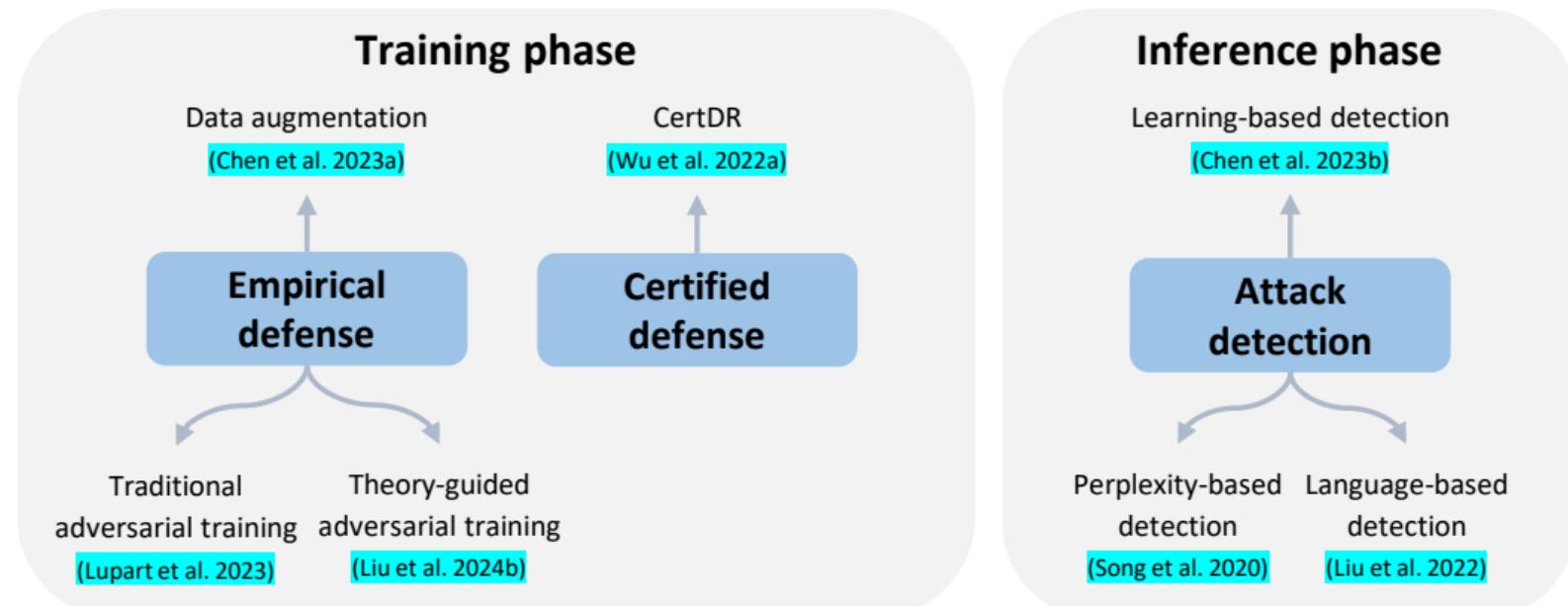


Lightweight: Easy to deploy, reducing the cost of defense in the training process of neural IR models



Error-prone: High false positive rates

# Summary



# Summary

Type of defense	Method	Phase	Attacks resisted	Nature of defense
Attack detection	<b>Perplexity-based detection</b> (Song et al. 2020)	Inference	Unseen attacks	Empirical
	<b>Language-based detection</b> (Shen et al. 2023)	Inference	Unseen attacks	Empirical
	<b>Learning-based detection</b> (Chen et al. 2023)	Inference	Seen attacks	Empirical
Empirical defense	<b>DA</b> (Wu et al. 2023)	Training	Unseen attacks	Empirical
	<b>Lupart et al. 2023</b>	Training	Seen attacks	Empirical
	<b>PIAT</b> (Liu et al. 2024)	Training	Seen attacks	Theoretical
Certified defense	<b>CertDR</b> (Wu et al. 2023)	Training	Unseen attacks	Theoretical

## Evaluation of adversarial defenses: Training phase

- **CleanMRR@ $K$**   
Top- $K$  ranking performance on a clean dataset
- **RobustMRR@ $K$**   
Top- $K$  ranking performance on the attacked test dataset
- **Attack success rate (ASR)**  
Percentage of the after-attack documents that are ranked higher than before
- **Location square deviation (LSD)**  
Consistency between the original and perturbed ranked list

## Evaluation of adversarial defenses: Inference phase

- **Point-wise detection accuracy**

Accuracy of the detection of whether a single document has been perturbed or not

- **#DD**

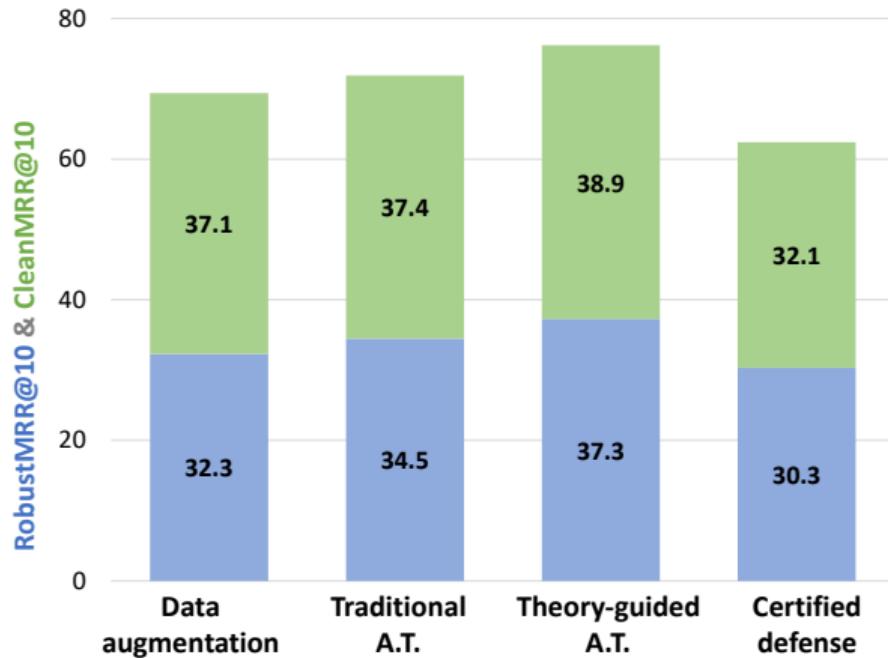
Average number of discarded documents ranked before the relevant document

- **#DR**

Average number of discarded relevant documents

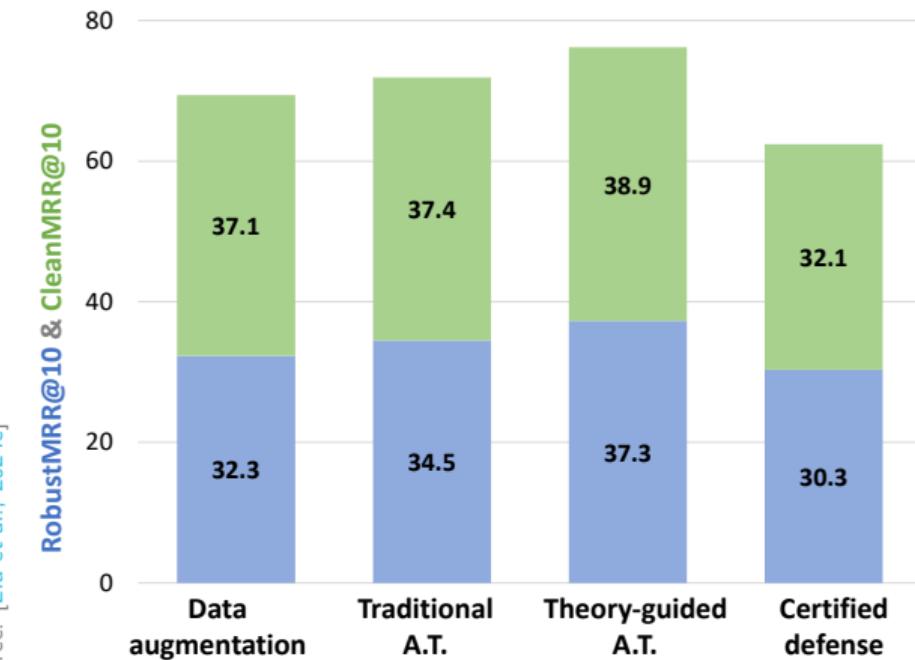
## Comparison between empirical and theoretical defenses

Data source: [Liu et al., 2024c]



- Dataset: MS MARCO
- Backbone: BERT-cross encoder
- Observations: Traditional adversarial training performs better than data augmentation because it is more specific to the adversarial example

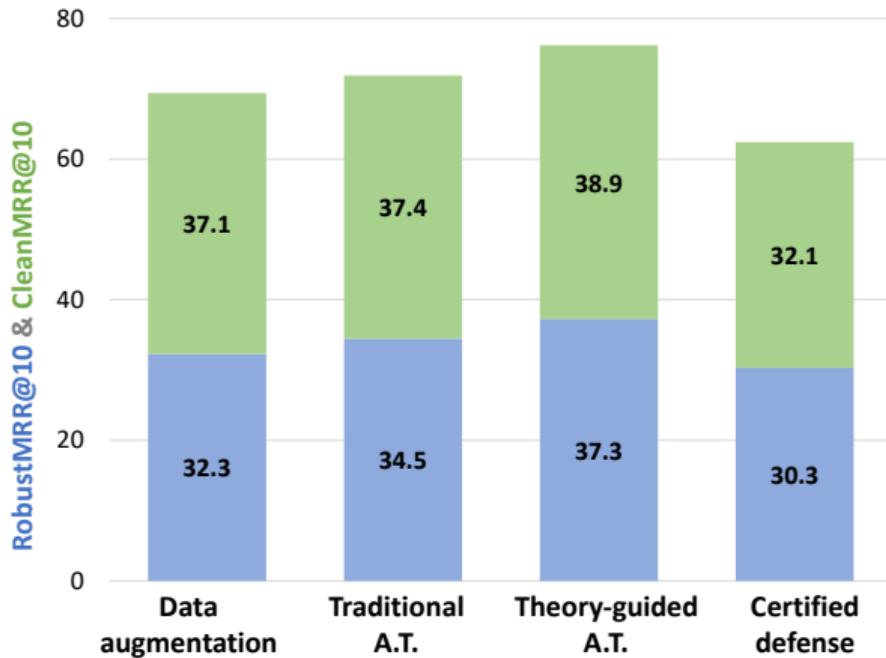
## Comparison between empirical and theoretical defenses



- Dataset: MS MARCO
- Backbone: BERT-cross encoder
- Observations: Theory-guided adversarial training can balance the trade-off between model effectiveness and robustness

## Comparison between empirical and theoretical defenses

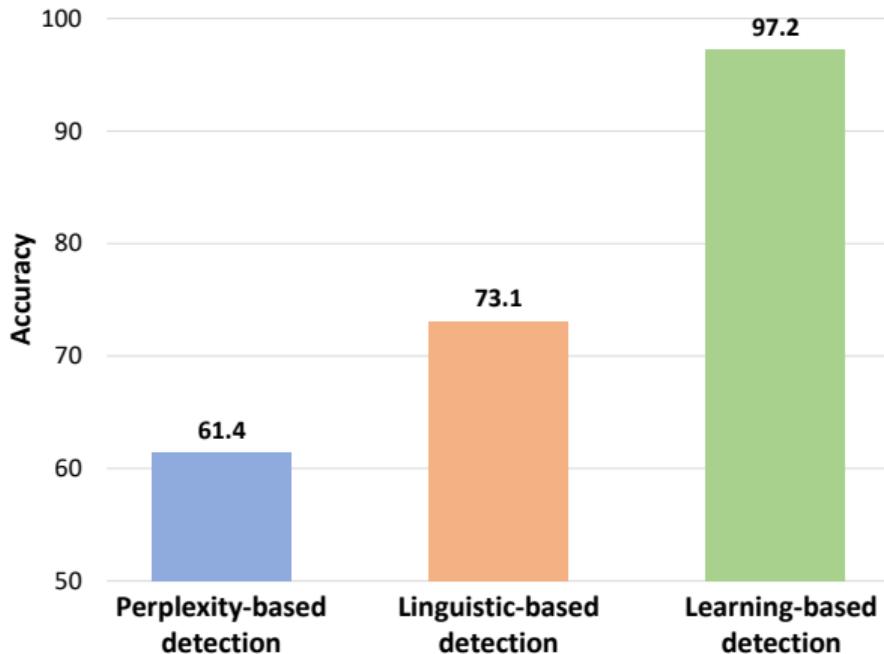
Data source: [Liu et al., 2024c]



- Dataset: MS MARCO
- Backbone: BERT-cross encoder
- Observations: Certified defense maximizes the assurance that Top- $K$  results are not contaminated at the expense of ranking performance

## Comparison between attack detections

Data source: [Chen et al., 2023c]



- Dataset: DARA
- Observations: PPL-based and linguistic-based detectors show limited effectiveness while learning-based detectors demonstrate greater reliability in identifying adversarial documents

## Takeaway

For adversarial defenses against neural IR models:

## Takeaway

For adversarial defenses against neural IR models:

- A good defense should balance effectiveness and robustness

## Takeaway

For adversarial defenses against neural IR models:

- A good defense should balance effectiveness and robustness
- Theoretical guidance helps produce reliable defense methods

## Takeaway

For adversarial defenses against neural IR models:

- A good defense should balance effectiveness and robustness
- Theoretical guidance helps produce reliable defense methods
- Accurately identifying the characteristics of adversarial samples helps to achieve the least costly defense

**Coffee break**

## **Section 4: Out-of-distribution robustness**

## Revisit the definition of out-of-distribution robustness

Ability of Neural IR models to maintain Top- $K$  ranking performance when exposed to queries and documents that deviate from the distribution seen during training

### **Definition (Out-of-distribution robustness of information retrieval)**

Given an IR model  $f_{\mathcal{D}_{\text{train}}}$ , an original dataset with training and test data,  $\mathcal{D}_{\text{train}}$  and  $\mathcal{D}_{\text{test}}$ , drawn from the original distribution  $\mathcal{G}$ , along with a new test dataset  $\tilde{\mathcal{D}}_{\text{test}}$  drawn from the new distribution  $\tilde{\mathcal{G}}$ , and an acceptable error threshold  $\delta$ , for the top- $K$  ranking result, if

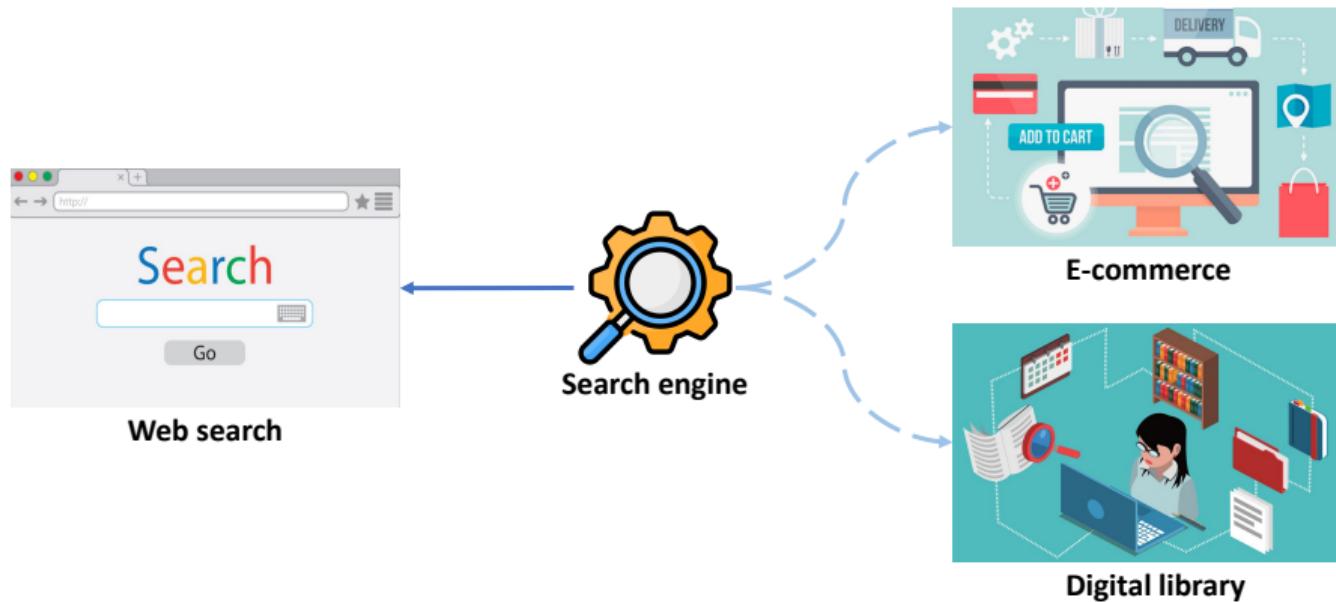
$$|\mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \mathcal{D}_{\text{test}}, K) - \mathcal{R}_M(f_{\mathcal{D}_{\text{train}}}; \tilde{\mathcal{D}}_{\text{test}}, K)| \leq \delta \text{ where } \mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}} \sim \mathcal{G}, \tilde{\mathcal{D}}_{\text{test}} \sim \tilde{\mathcal{G}},$$

the model  $f$  is considered  $\delta$ -robust against out-of-distribution data for metric  $M$ .

## Background: Migration scenarios for search engines

A good search engine can be migrated to **various scenarios** at a low cost. Difficulty:

- Documents from different domains
- Queries with different types



## Background: Dynamic scenarios for search engines

A good search engine should **keep up with the trends** at a low cost. Difficulty:

- Documents on new hotspots
- Queries with new expressions



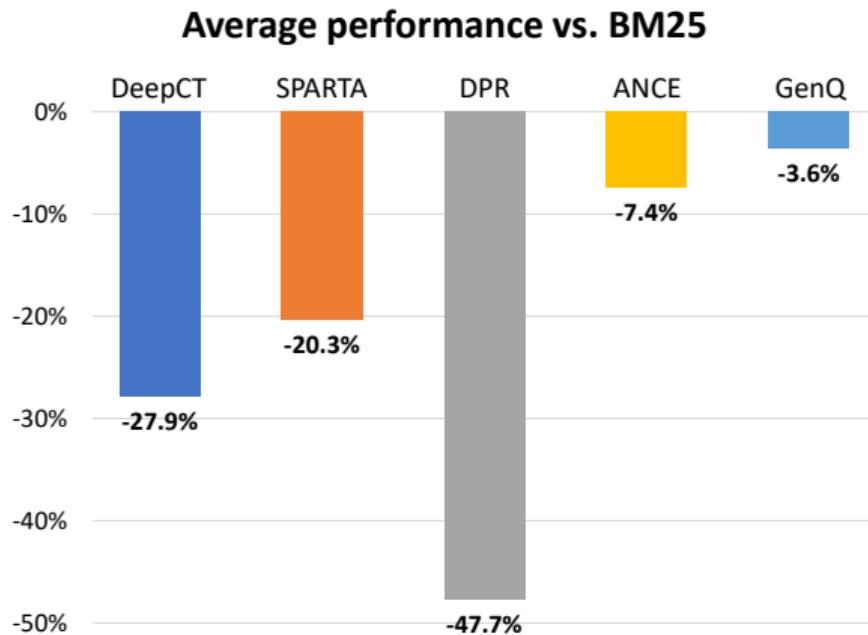
The above are uniformly described as out-of-distribution (OOD) scenarios

## Dilemma: Neural IR models struggle with OOD scenarios

Without retraining, the performance of the neural IR model decreases significantly when faced with OOD data

## Dilemma: Neural IR models struggle with OOD scenarios

Without retraining, the performance of the neural IR model decreases significantly when faced with OOD data



- Dataset: BEIR
- Scenario: OOD corpus
- Observations: The zero-shot performance of neural IR models is worse than traditional IR models

A straightforward solution

**“Let’s just retrain the neural IR models dynamically in response to OOD data. Problem solved.”**

However, neural IR models are data-hungry

Training an effective neural IR model is **very costly**:

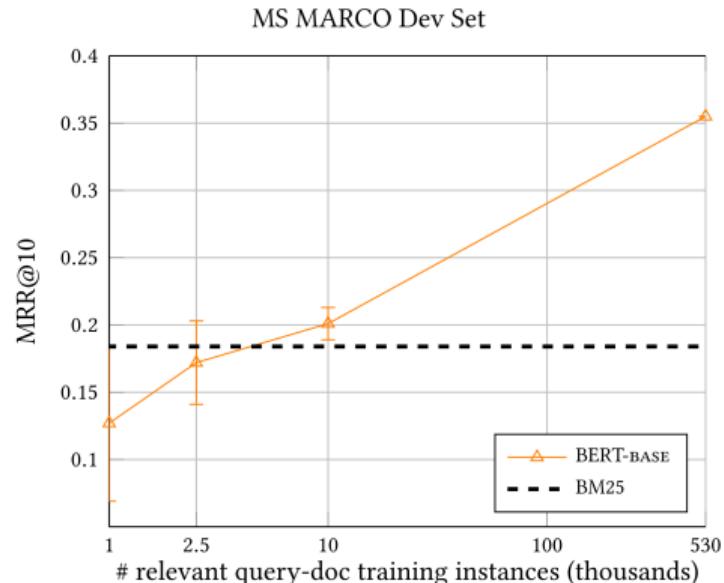
- **Quantity:** Large-scale queries and documents
- **Quality:** Relevance labels provided by experts

However, neural IR models are data-hungry

Training an effective neural IR model is **very costly**:

- **Quantity:** Large-scale queries and documents
- **Quality:** Relevance labels provided by experts

Dataset	Year	Query	Corpus
Robust04	2004	250	0.5M
MQ2007	2007	1.7k	25M
Clueweb09-B	2009	150	<b>50M</b>
MS MARCO	2017	<b>367k</b>	3.3M



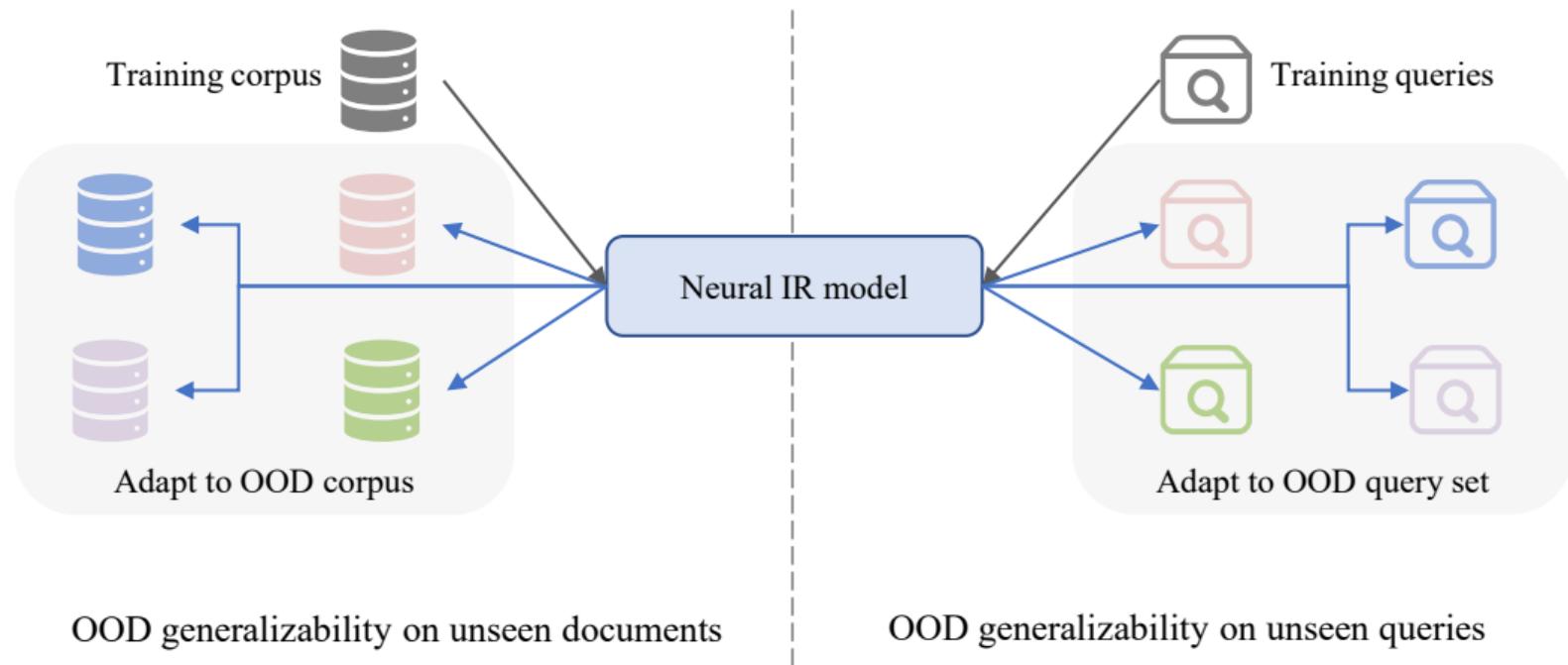
**How can we flexibly enhance the OOD robustness of neural IR models?**

**How can we flexibly enhance the OOD robustness of neural IR models?**

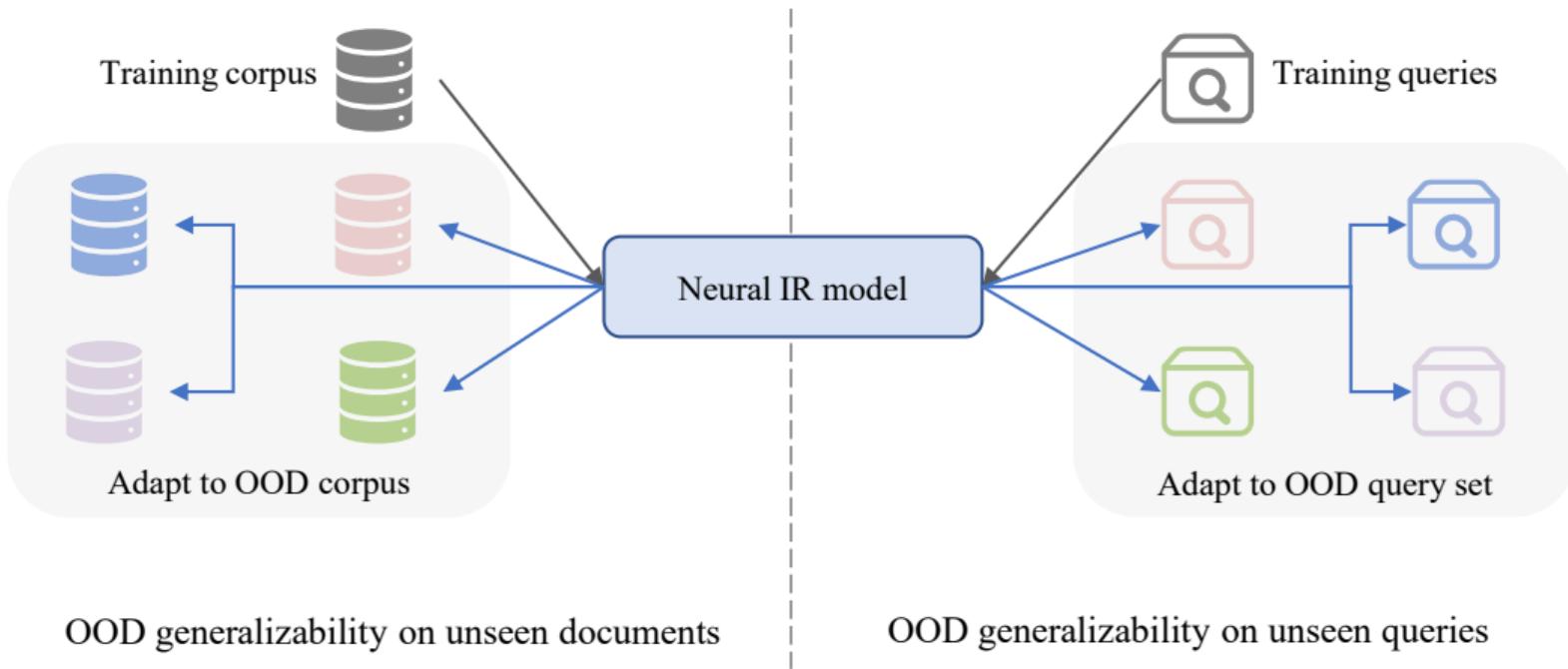
There are two perspectives...

## Two perspectives of OOD robustness

The OOD robustness of neural IR models can be categorized into the generalizability on **unseen documents** and **unseen queries**



## Two perspectives of OOD robustness



- **Unseen documents:** Corpus of new domains, corpus incrementation
- **Unseen queries:** Query variation (typos, etc.), new query types

## Outline

We will introduce the OOD robustness through:

- **OOD generalizability on unseen documents**
  - Benchmarks
  - Adaptation to new corpus
  - Updates to a corpus
- **OOD generalizability on unseen queries**
  - Benchmarks
  - Query variation
  - Unseen query type

## OOD generalizability on unseen documents

IR systems need to adapt to different environments and variations in the corpus

## OOD generalizability on unseen documents

IR systems need to adapt to different environments and variations in the corpus

There are two scenarios:

- **Adaptation to new corpus:** Neural IR models trained on the original corpus are migrated to the new domain corpus

## OOD generalizability on unseen documents

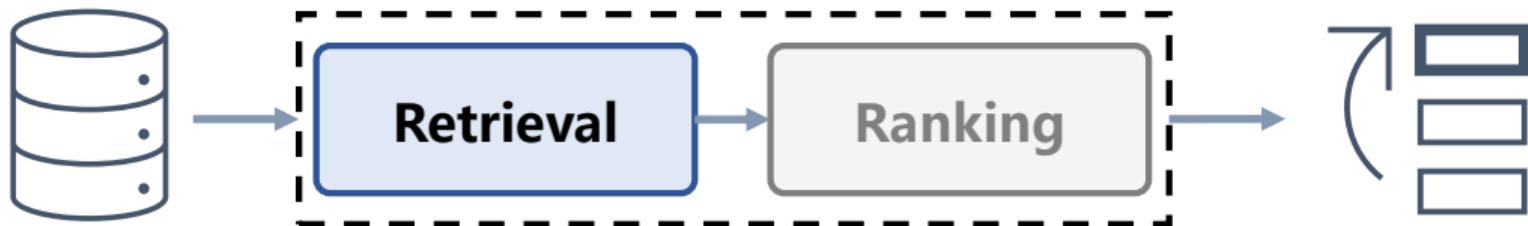
IR systems need to adapt to different environments and variations in the corpus

There are two scenarios:

- **Adaptation to new corpus:** Neural IR models trained on the original corpus are migrated to the new domain corpus
- **Updates to a corpus:** Neural IR models trained on the original corpus, adapted to the continuous growth of documents in the corpus

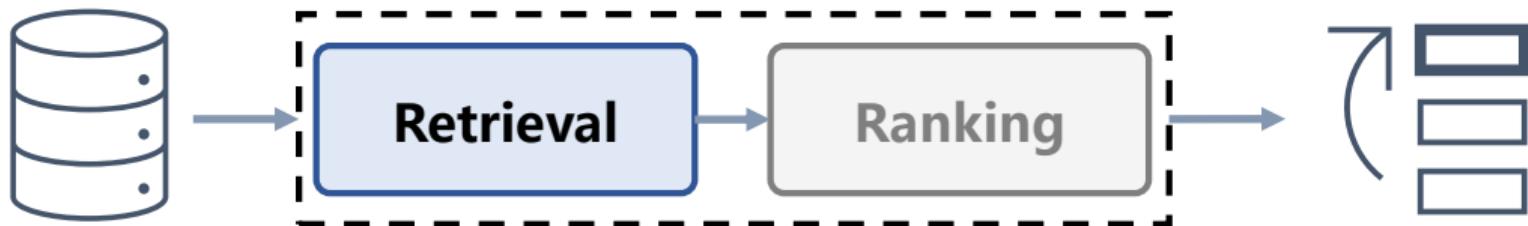
## OOD generalizability on unseen documents

The above scenarios have a **direct impact on the performance of the retrieval stage**



## OOD generalizability on unseen documents

The above scenarios have a **direct impact on the performance of the retrieval stage**



Existing work mainly focuses on **neural retrieval models**, i.e., dense retrieval models (DRMs) and generative retrieval models (GRMs)

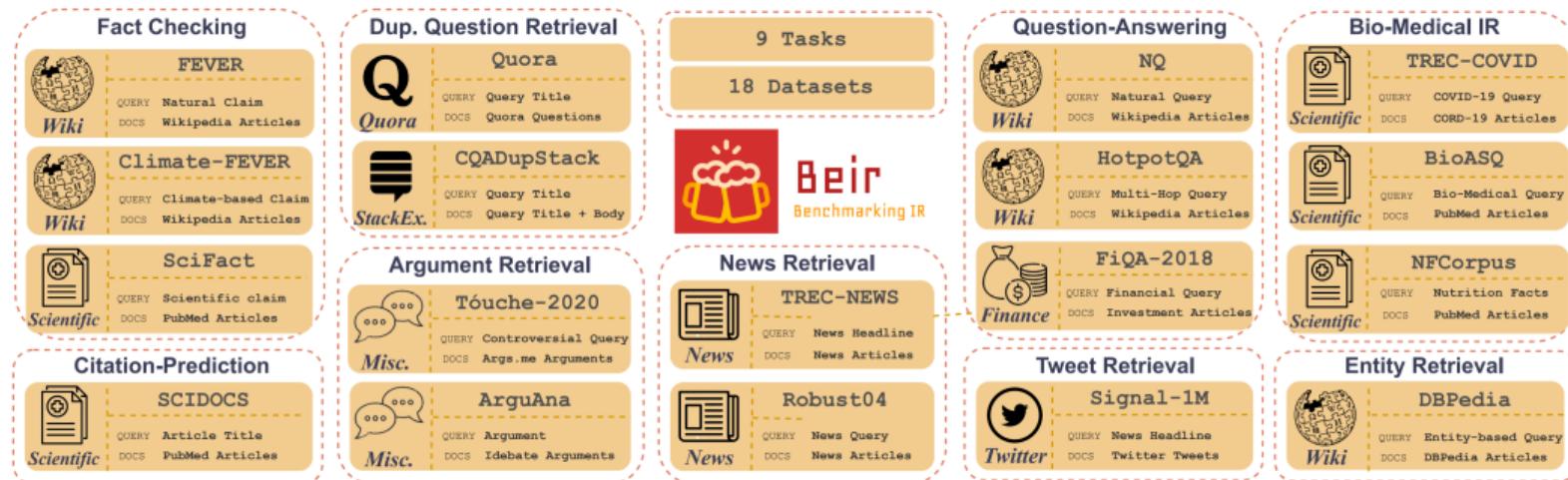
## OOD generalizability on unseen documents: Benchmarks

**Adaptation to new corpus** typically aggregates multiple existing domain IR datasets.

# OOD generalizability on unseen documents: Benchmarks

**Adaptation to new corpus** typically aggregates multiple existing domain IR datasets.

BEIR is the most typical, it includes **18 datasets** from **9 different retrieval tasks**, such as news retrieval, entity retrieval.



## OOD generalizability on unseen documents: Benchmarks

**Updates to a corpus** mainly slices or expands the existing dataset

## OOD generalizability on unseen documents: Benchmarks

**Updates to a corpus** mainly slices or expands the existing dataset

For example, CDI-MS first randomly sampled 60% documents from the whole corpus as the base documents

Then, it randomly samples 10% documents from the remaining corpus as the new document set, and repeated 4 times



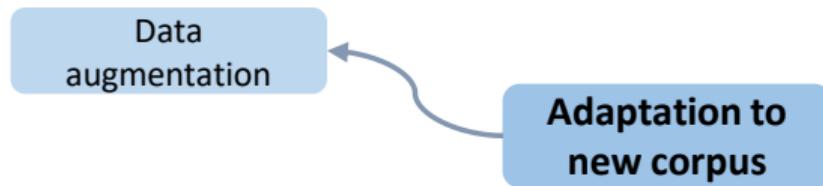
## OOD generalizability on unseen documents: Benchmarks

Type	Dataset		#Retrieval task	#Corpus	
Adaptation new corpus	to BEIR [Thakur et al., 2021]	9		18	
Type	Dataset	#D	#Q <sub>train</sub>	#Q <sub>dev</sub>	#Q <sub>eval</sub>
Updates to original corpus	CDI-MS [Chen et al., 2023a]	3.2M	370K	5,193	5,793
	CDI-NQ [Chen et al., 2023a]	8.8M	500K	6,980	6,837
	LL-LoTTE [Cai et al., 2023]	5.5M	16K	8.5k	8.6k
	LL-MultiCPR [Cai et al., 2023]	3.0M	136K	15k	15k

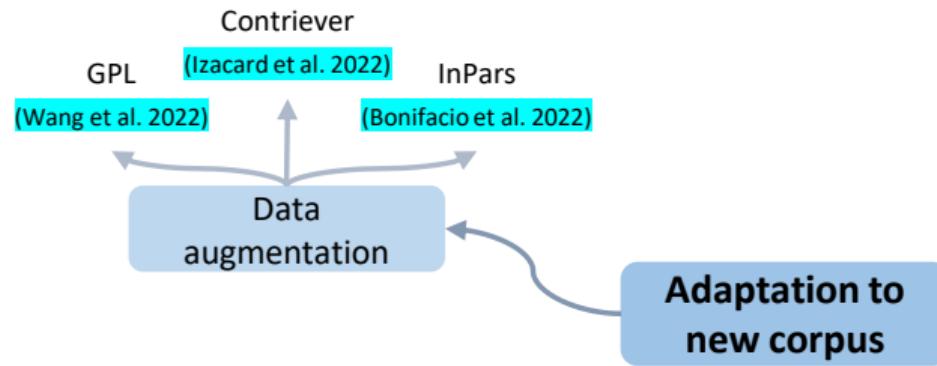
# Classification of adaptation to new corpus

**Adaptation to  
new corpus**

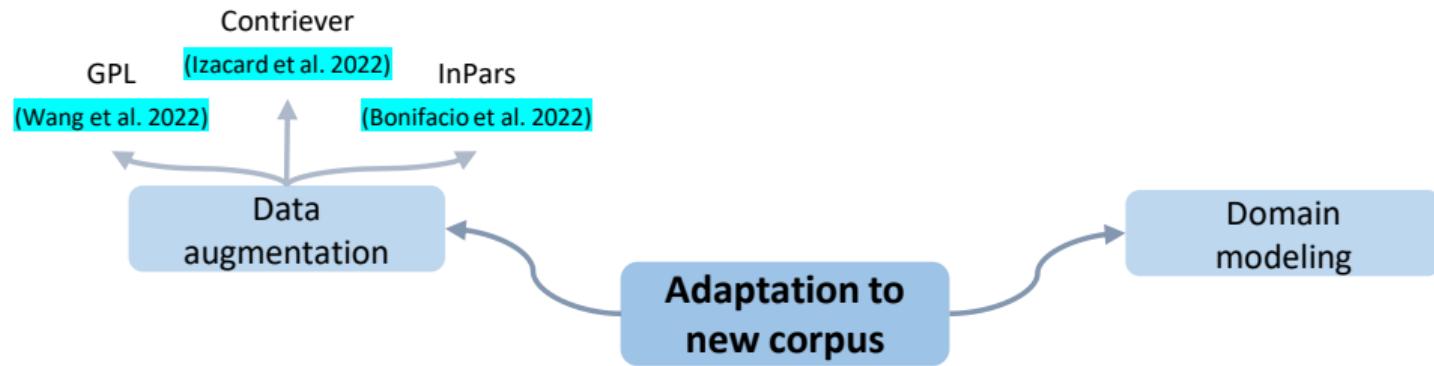
## Classification of adaptation to new corpus



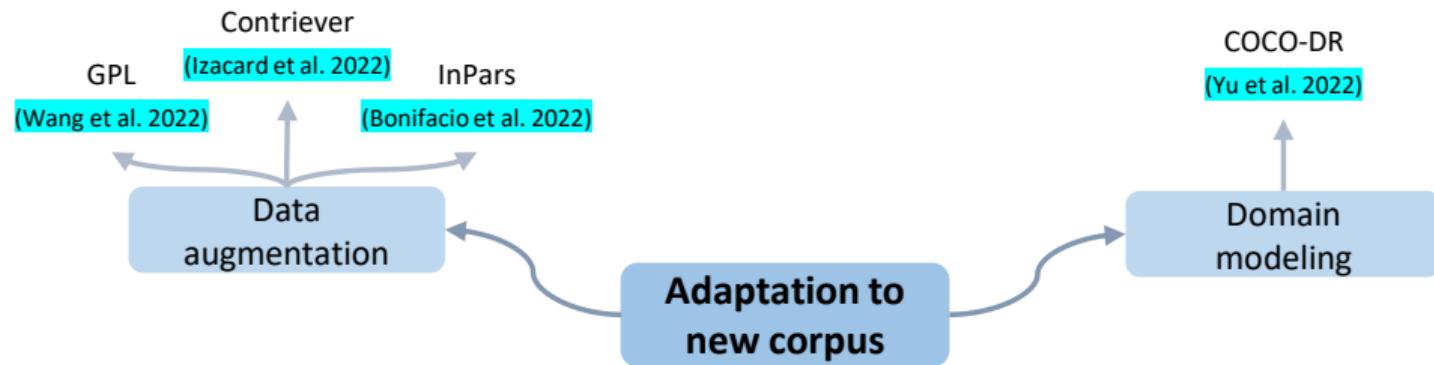
# Classification of adaptation to new corpus



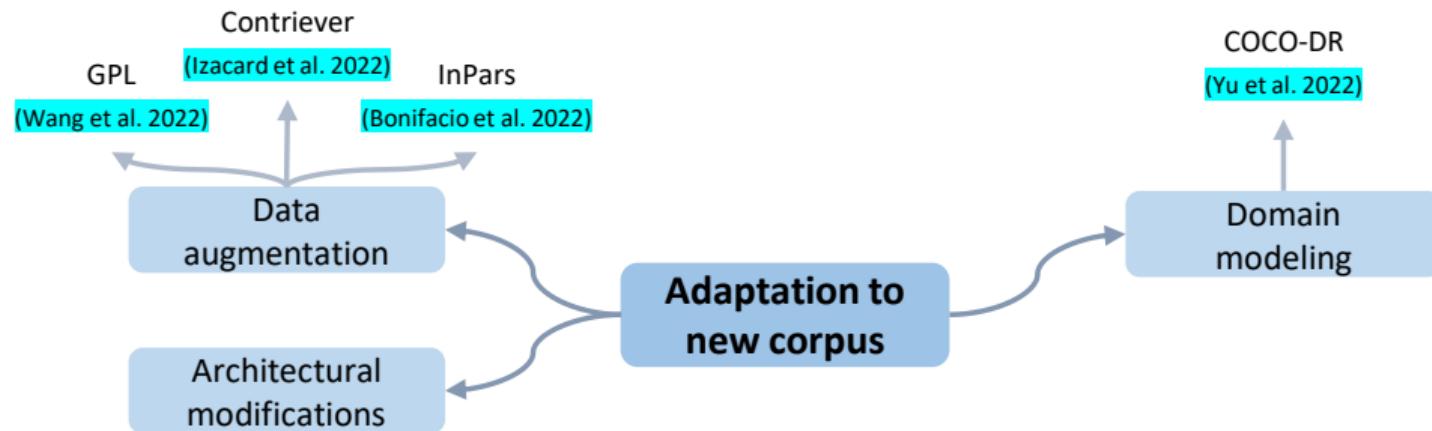
# Classification of adaptation to new corpus



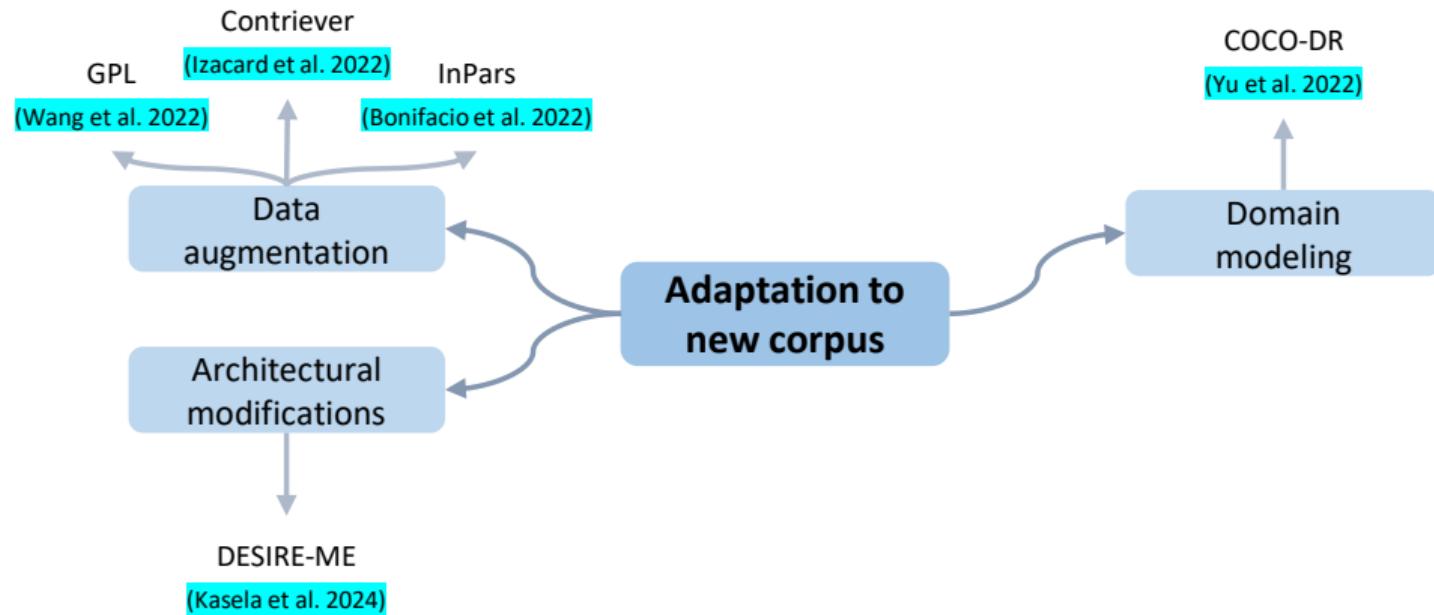
# Classification of adaptation to new corpus



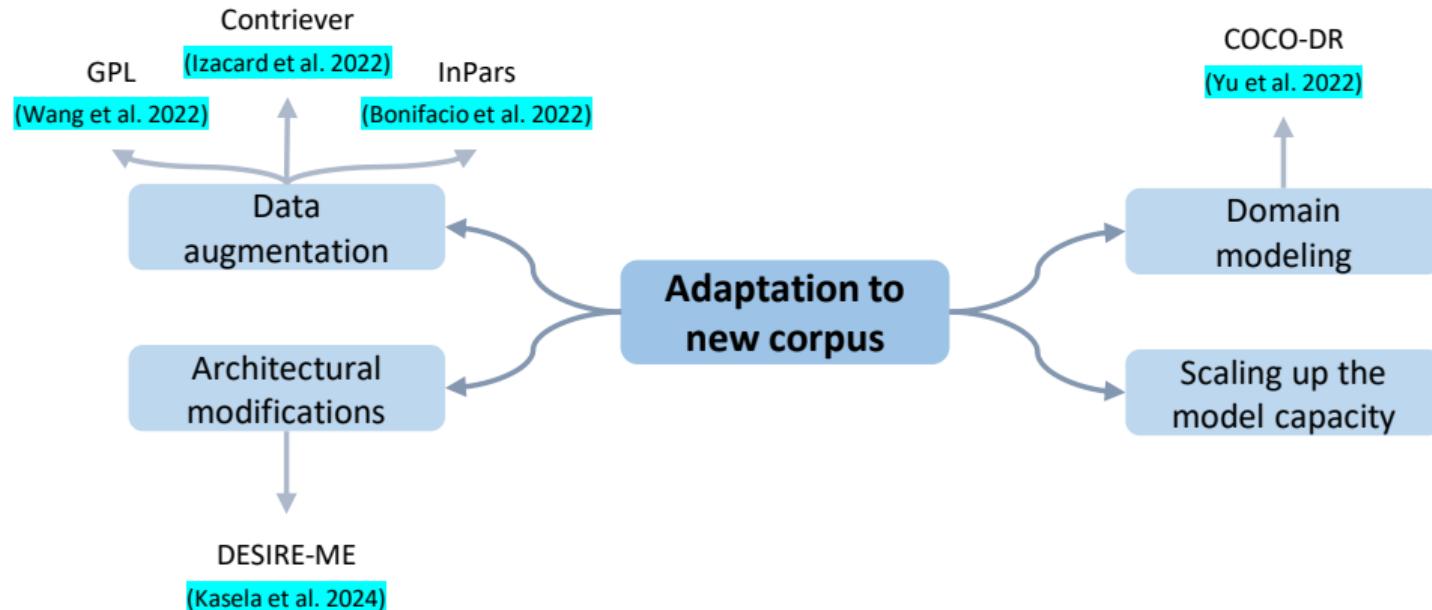
# Classification of adaptation to new corpus



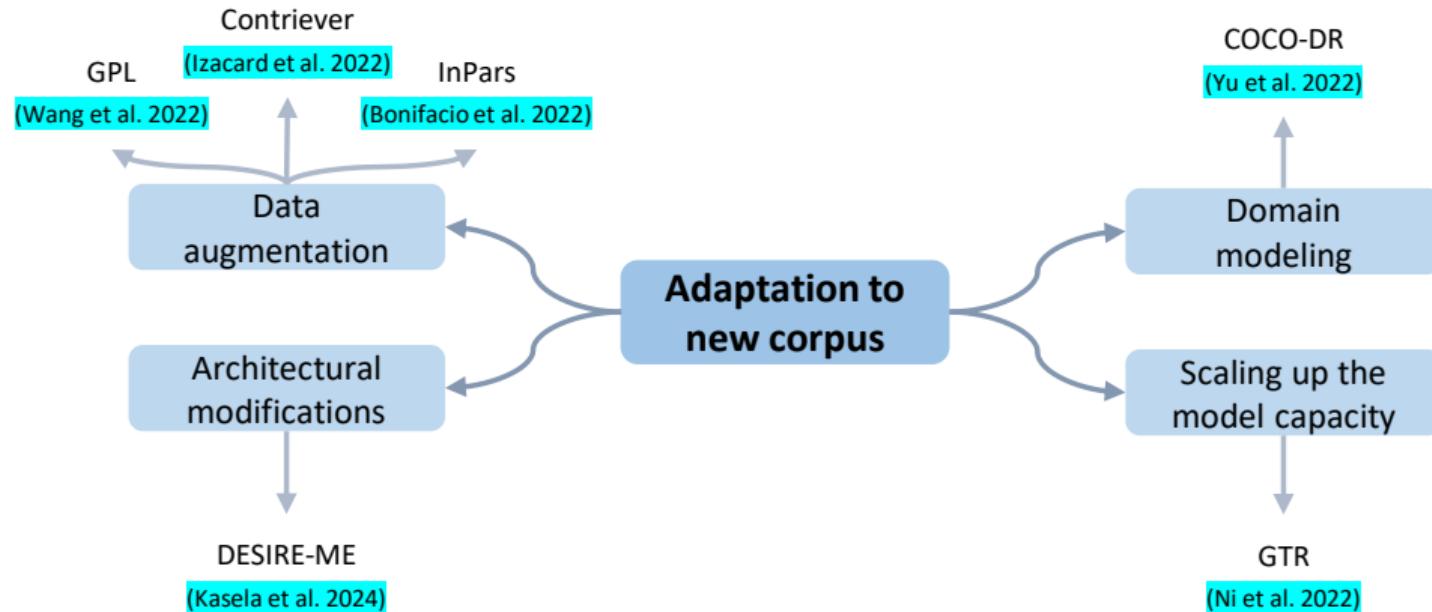
# Classification of adaptation to new corpus



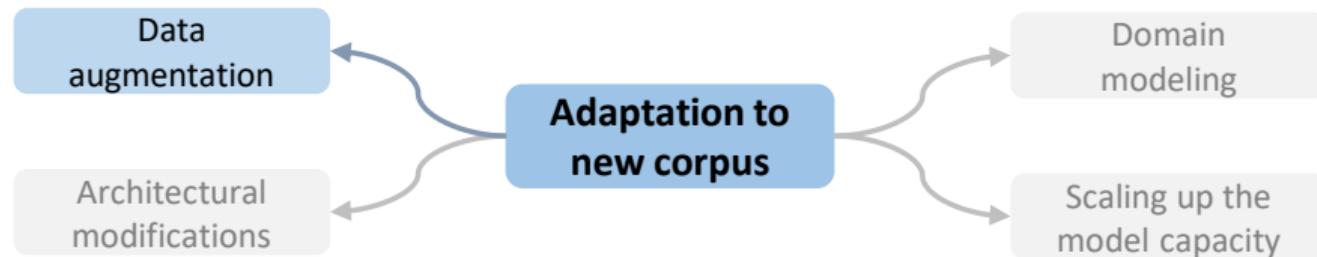
# Classification of adaptation to new corpus



# Classification of adaptation to new corpus

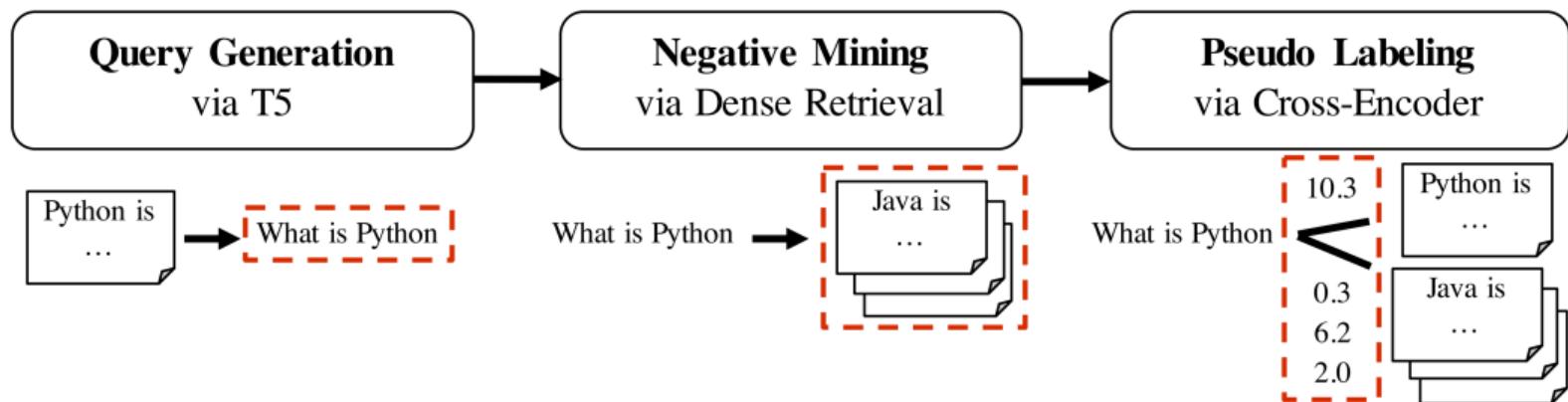


## Adaptation to new corpus: Data augmentation

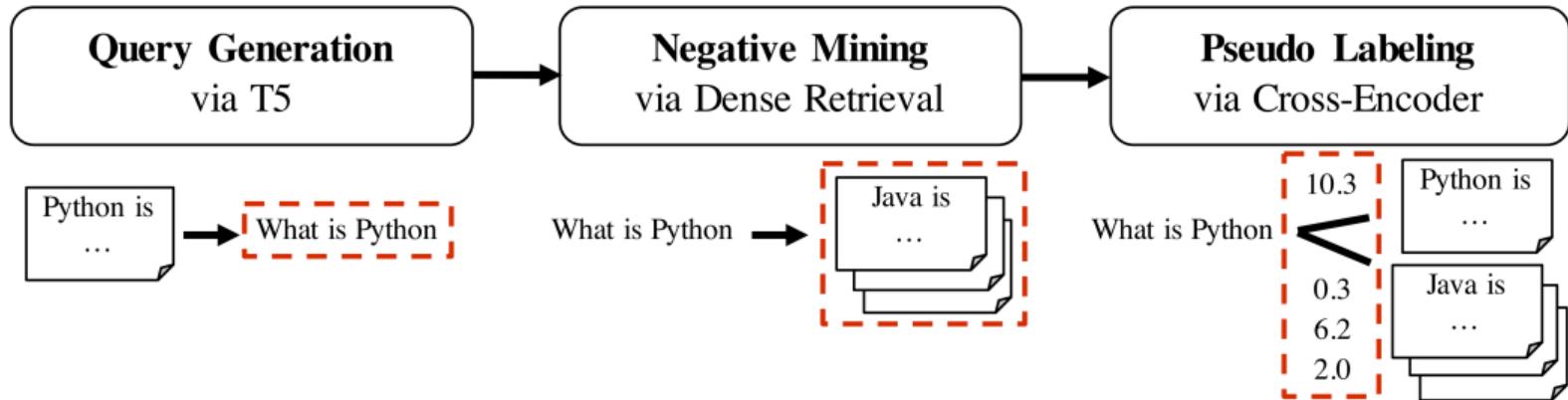


## Adaptation to new corpus: Data augmentation

**Generative pseudo labeling (GPL)** combines a **query generator** with **pseudo labeling** from a cross-encoder to generate additional training data [Wang et al., 2022]

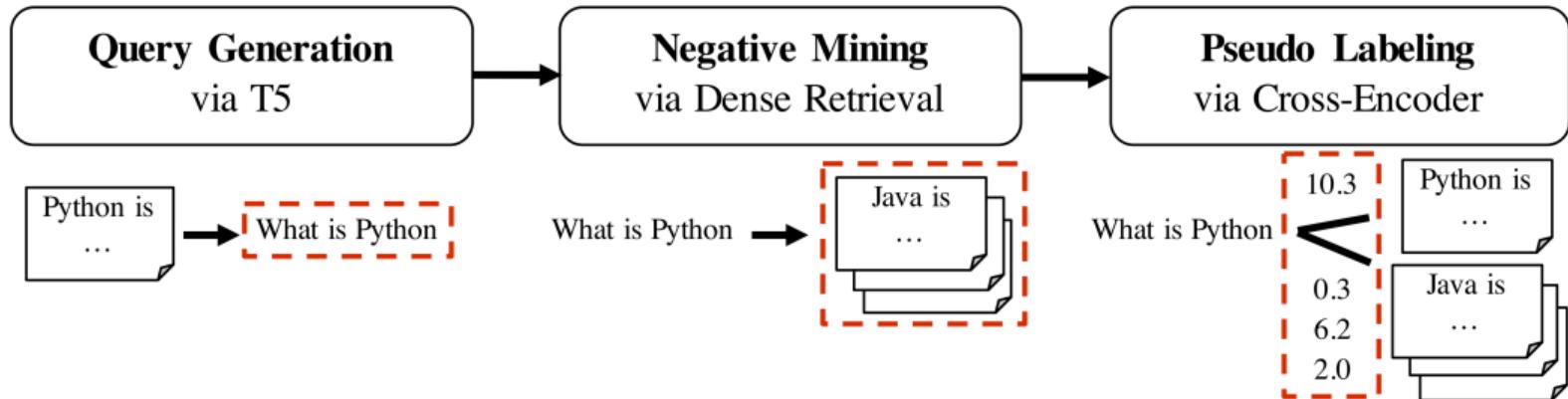


## Adaptation to new corpus: Data augmentation



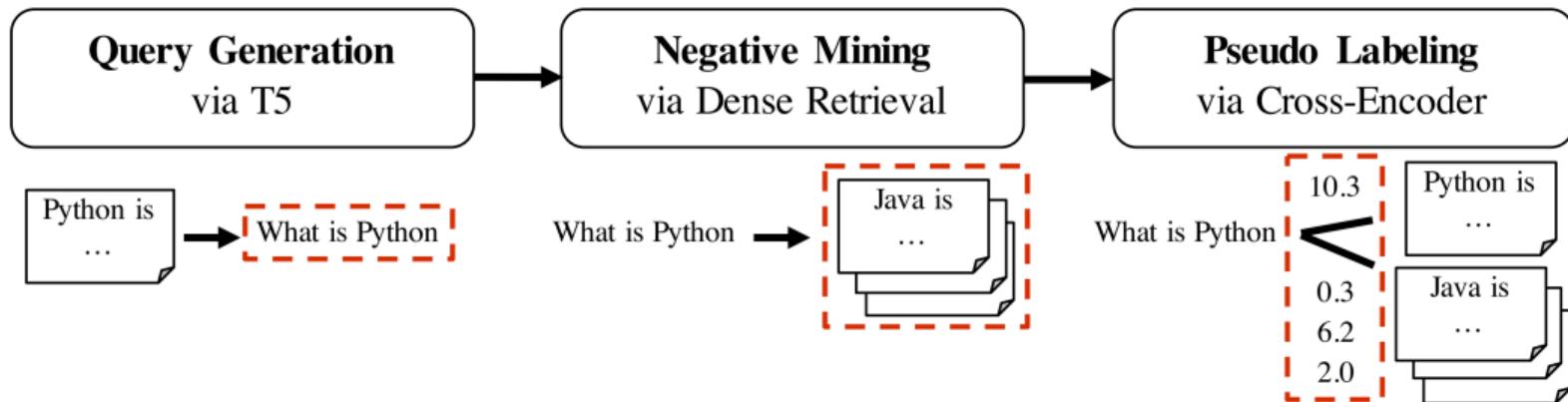
- Synthetic queries are generated for each passage from the target corpus

## Adaptation to new corpus: Data augmentation



- Synthetic queries are generated for each passage from the target corpus
- The generated queries are used for mining negative passages

## Adaptation to new corpus: Data augmentation



- Synthetic queries are generated for each passage from the target corpus
- The generated queries are used for mining negative passages
- The query-passage pairs are labeled by a cross-encoder and used to train the domain-adapted dense retriever

## Data augmentation: GPL

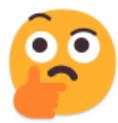


Straightforward: Access to large amounts of pseudo labeled data

## Data augmentation: GPL



Straightforward: Access to large amounts of pseudo labeled data



Unstable: Not all generated queries are of high quality

## Data augmentation: GPL



Straightforward: Access to large amounts of pseudo labeled data



Unstable: Not all generated queries are of high quality



Dependent: Over-reliance on cross-coder performance

## Adaptation to new corpus: Data augmentation

**Contriever** explores the limits of contrastive learning as a way to pre-train in an unsupervised way a dense retriever [Izacard et al., 2021]

## Adaptation to new corpus: Data augmentation

**Contriever** explores the limits of contrastive learning as a way to pre-train in an unsupervised way a dense retriever [Izacard et al., 2021]

- Build positive pairs from a single document through the inverse Cloze task

## Adaptation to new corpus: Data augmentation

**Contriever** explores the limits of contrastive learning as a way to pre-train in an unsupervised way a dense retriever [Izacard et al., 2021]

- Build positive pairs from a single document through the inverse Cloze task
- Build a large set of negative pairs, including in-batch negatives and cross-batch negatives

## Adaptation to new corpus: Data augmentation

**Contriever** explores the limits of contrastive learning as a way to **pre-train in an unsupervised way** a dense retriever [Izacard et al., 2021]

- Build positive pairs from a single document through the inverse Cloze task
- Build a large set of negative pairs, including in-batch negatives and cross-batch negatives
- Perform contrastive learning on the whole constructed training data

## Data augmentation: Contriever



Low data costs: Unsupervised construction of a large amount of pre-training data

## Data augmentation: Contriever



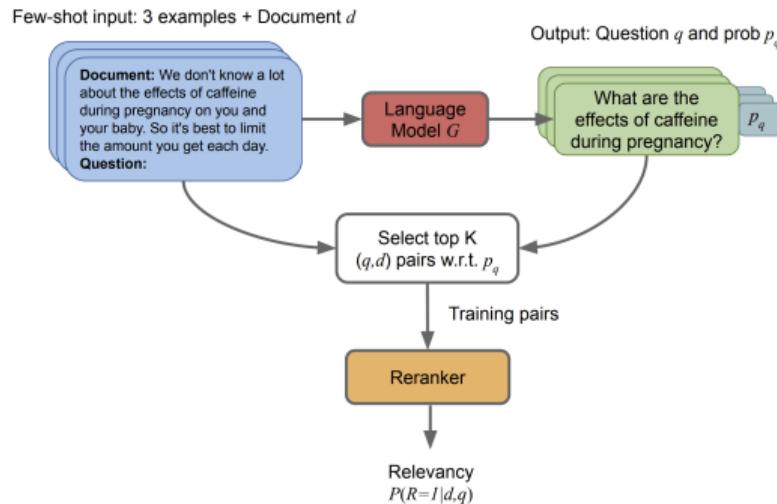
Low data costs: Unsupervised construction of a large amount of pre-training data



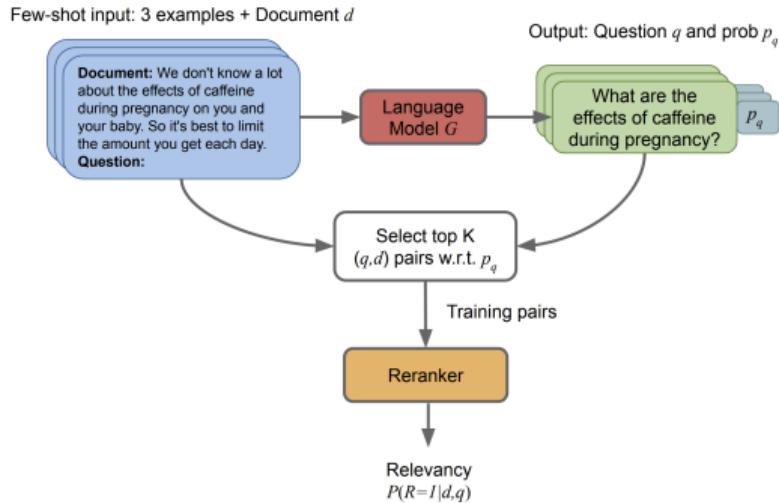
High training costs: High cost of pre-training

## Adaptation to new corpus: Data augmentation

**InPars** harnesses the **few-shot capabilities of large language models** as synthetic data generators for IR task [Bonifacio et al., 2022]

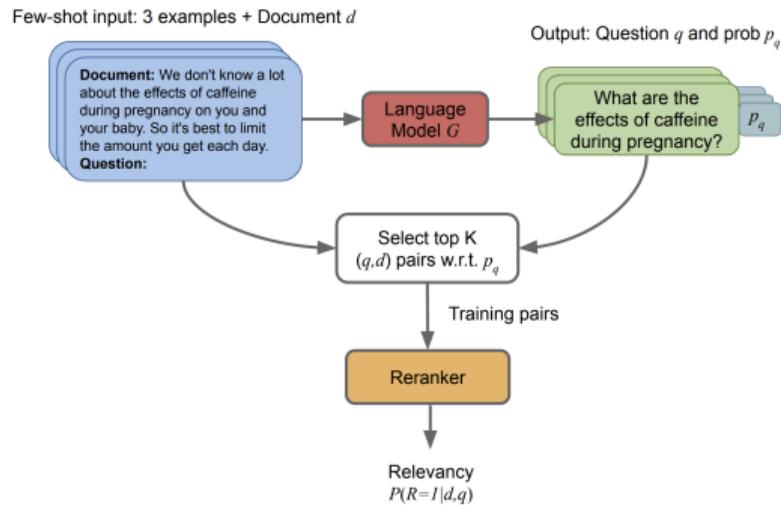


## Adaptation to new corpus: Data augmentation



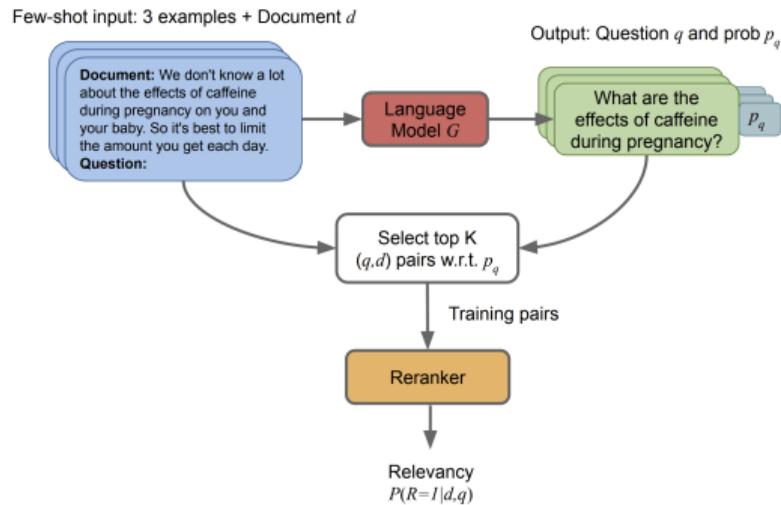
- For a document, 3 sets of q-d pairs are constructed as the instruction

## Adaptation to new corpus: Data augmentation



- For a document, 3 sets of q-d pairs are constructed as the instruction
- Generate query with LLM and get the corresponding generation probability

## Adaptation to new corpus: Data augmentation



- For a document, 3 sets of q-d pairs are constructed as the instruction
- Generate query with LLM and get the corresponding generation probability
- Based on this, the corresponding query is generated for each randomly sampled document, constituting a positive sample for training

## Data augmentation: InPars



Effective: Constructing positive samples using LLMs

## Data augmentation: InPars

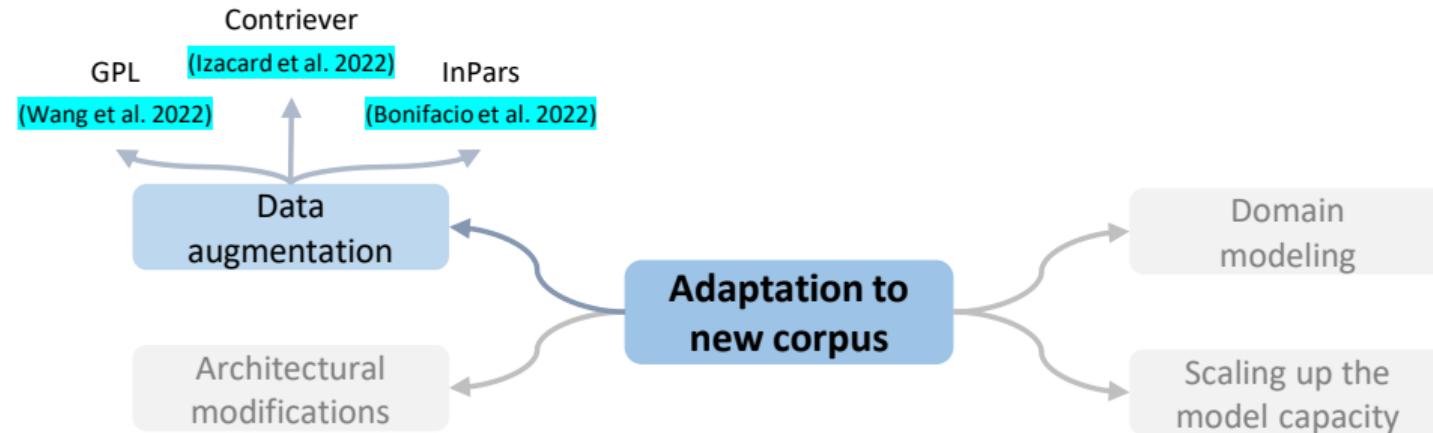


Effective: Constructing positive samples using LLMs



Risky: Low-quality generated queries may occur

# Review data augmentation



## Review data augmentation



Effective: Simple way to improve model training

## Review data augmentation



Effective: Simple way to improve model training



Diverse: There are various ways to synthesize data

## Review data augmentation



Effective: Simple way to improve model training



Diverse: There are various ways to synthesize data



Risky: Low-quality data is hard to avoid

## Adaptation to new corpus: Domain modeling



## Adaptation to new corpus: Domain modeling

**COCO-DR** uses **implicit distributionally robust optimization (iDRO)** to reweight samples from different source query clusters for improving model robustness over rare queries during fine-tuning [Yu et al., 2022]

A model trained to be **more robust on the source domain** is likely to better generalize to unseen data

## Adaptation to new corpus: Domain modeling

**COCO-DR** uses **implicit distributionally robust optimization (iDRO)** to reweight samples from different source query clusters for improving model robustness over rare queries during fine-tuning [Yu et al., 2022]

A model trained to be **more robust on the source domain** is likely to better generalize to unseen data

- Cluster source queries using K-Means and then optimize the iDRO loss

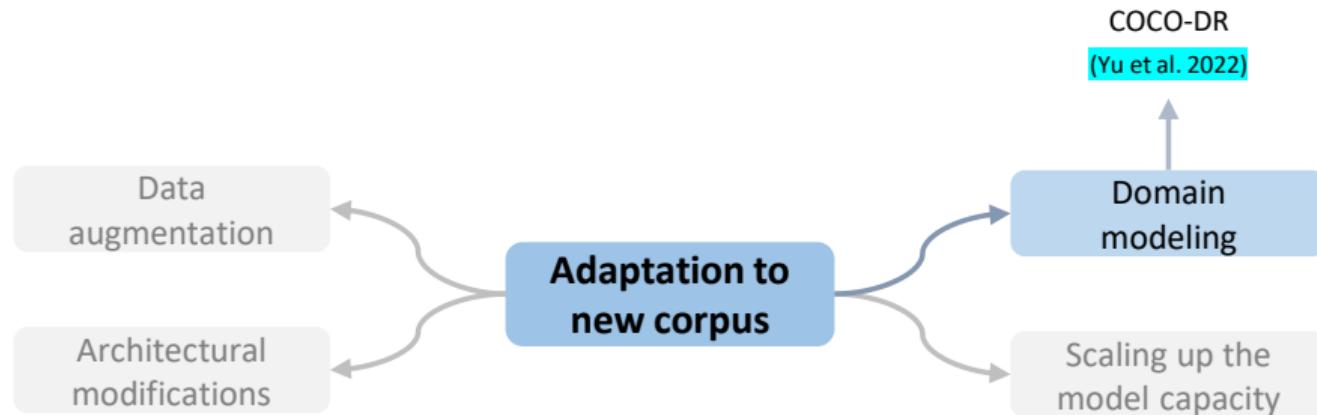
## Adaptation to new corpus: Domain modeling

**COCO-DR** uses **implicit distributionally robust optimization (iDRO)** to reweight samples from different source query clusters for improving model robustness over rare queries during fine-tuning [Yu et al., 2022]

A model trained to be **more robust on the source domain** is likely to better generalize to unseen data

- Cluster source queries using K-Means and then optimize the iDRO loss
- Dynamic weight of each cluster during fine-tuning

## Review domain modeling



## Review domain modeling



Reliable: Theoretically guaranteed generalization from existing domains to unseen domains

## Review domain modeling



Reliable: Theoretically guaranteed generalization from existing domains to unseen domains



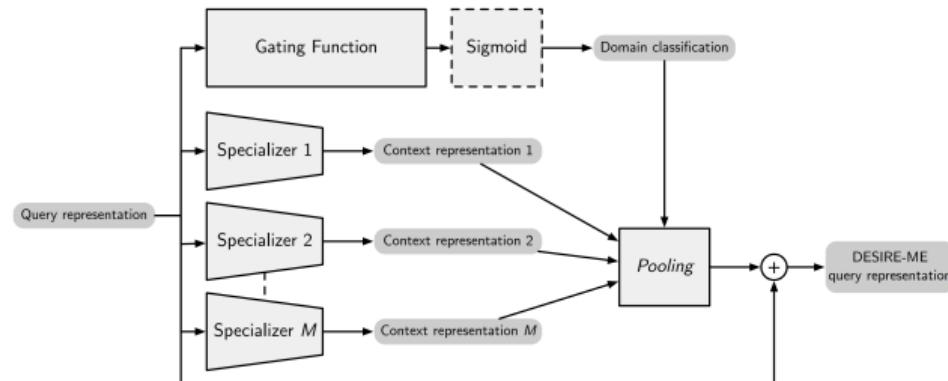
Complex: Complexity of realization and training process

## Adaptation to new corpus: architectural modifications

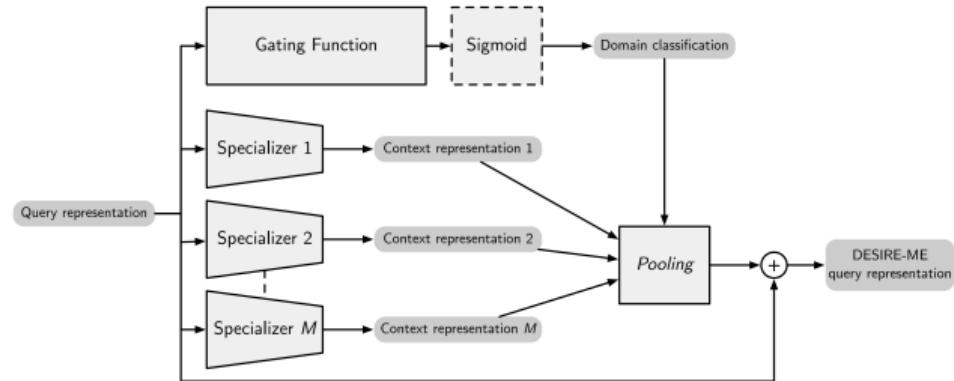


## Adaptation to new corpus: architectural modifications

**DESIRE-ME** uses the **mixture-of-experts framework** to combine multiple specialized neural models [Kasela et al., 2024]

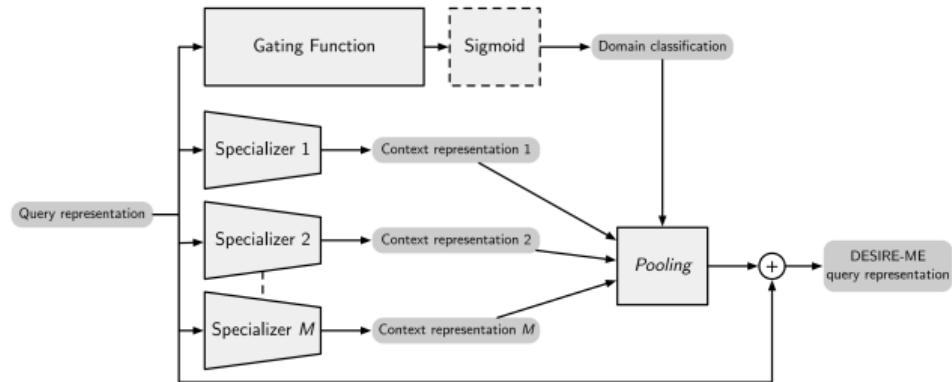


## Adaptation to new corpus: architectural modifications



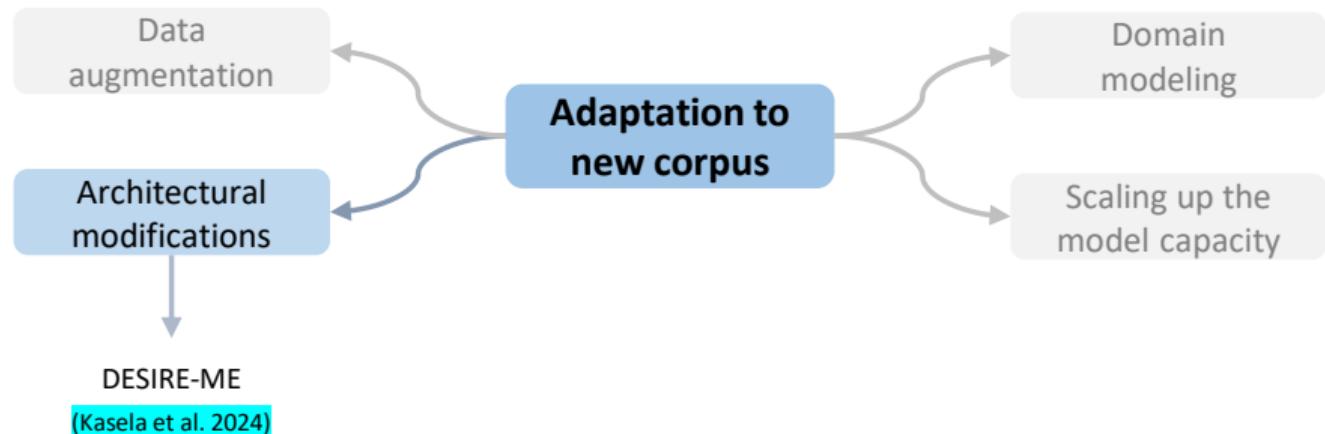
- **Specializers** focus on tuning query representation for the corresponding domain

## Adaptation to new corpus: architectural modifications



- **Specializers** focus on tuning query representation for the corresponding domain
- **Pooling module** merges the domain context representations computed by the specializers on the basis of the domain likelihood estimated by the gating function

## Review architectural modifications



## Review architectural modifications



Explainable: Explicit modeling domain information

## Review architectural modifications

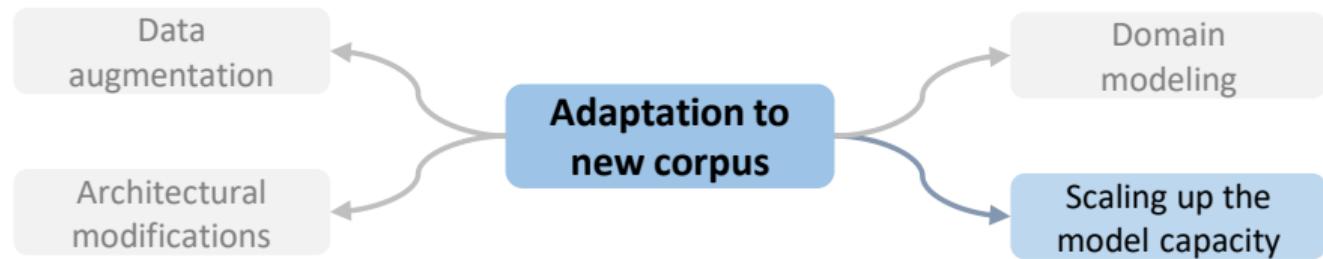


Explainable: Explicit modeling domain information



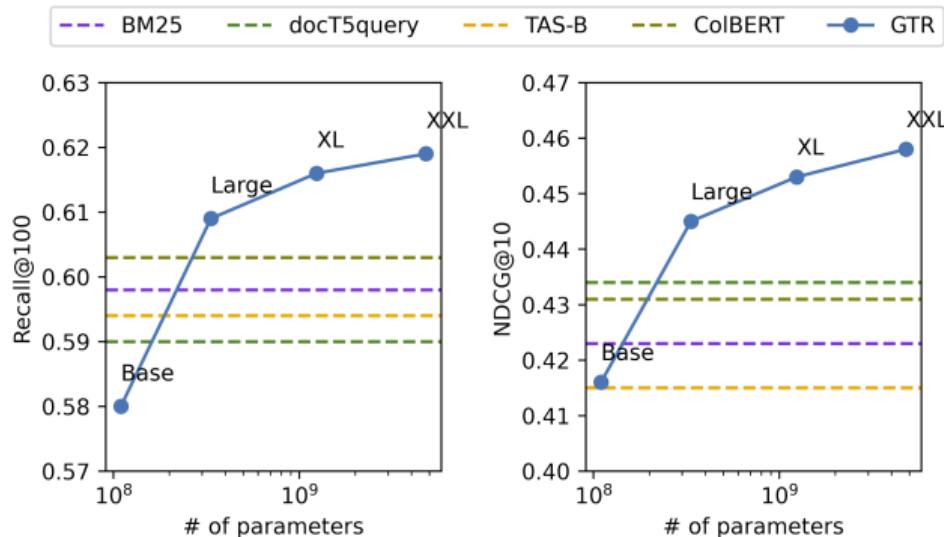
Restricted: Assumption of having query domain information

## Adaptation to new corpus: Scaling up the model capacity

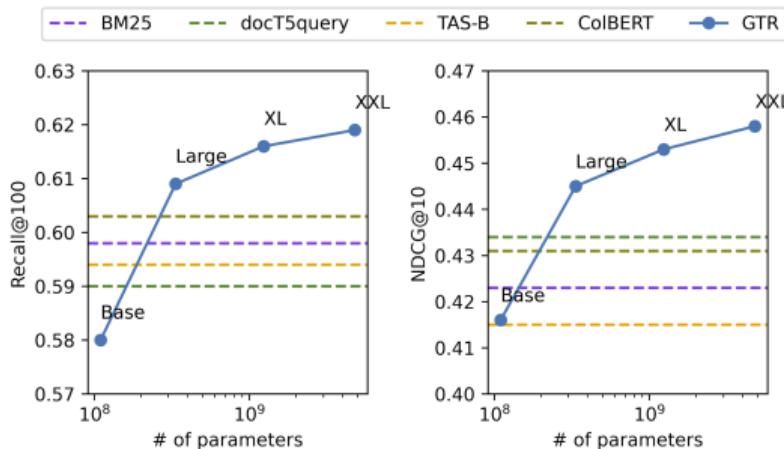


## Adaptation to new corpus: Scaling up the model capacity

**GTR** scales up the dual encoder model size while keeping the bottleneck embedding size fixed [Ni et al., 2022]

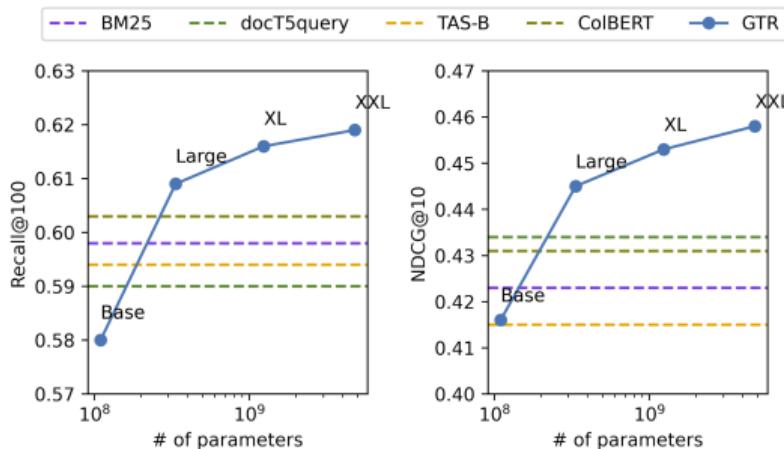


## Adaptation to new corpus: Scaling up the model capacity



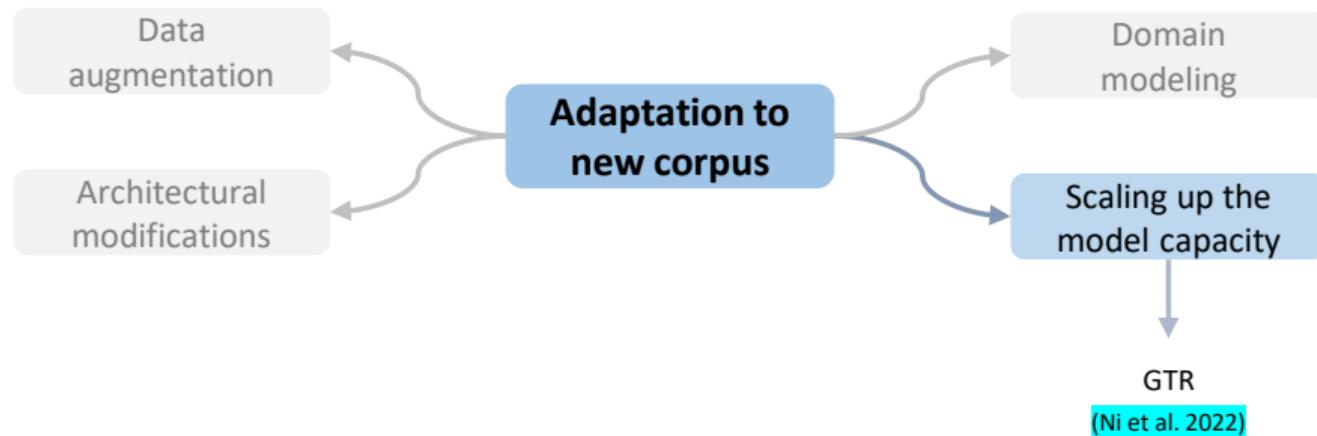
- For pre-training, the dual encoder is initialized from the T5 models and train on question-answer pairs collected from the Web

## Adaptation to new corpus: Scaling up the model capacity



- For pre-training, the dual encoder is initialized from the T5 models and train on question-answer pairs collected from the Web
- For fine-tuning, the aim is to adapt the model to retrieval using a high-quality search corpus

## Review scaling up the model capacity



## Review scaling up the model capacity



Simple: Straightforward to improve OOD robustness

## Review scaling up the model capacity

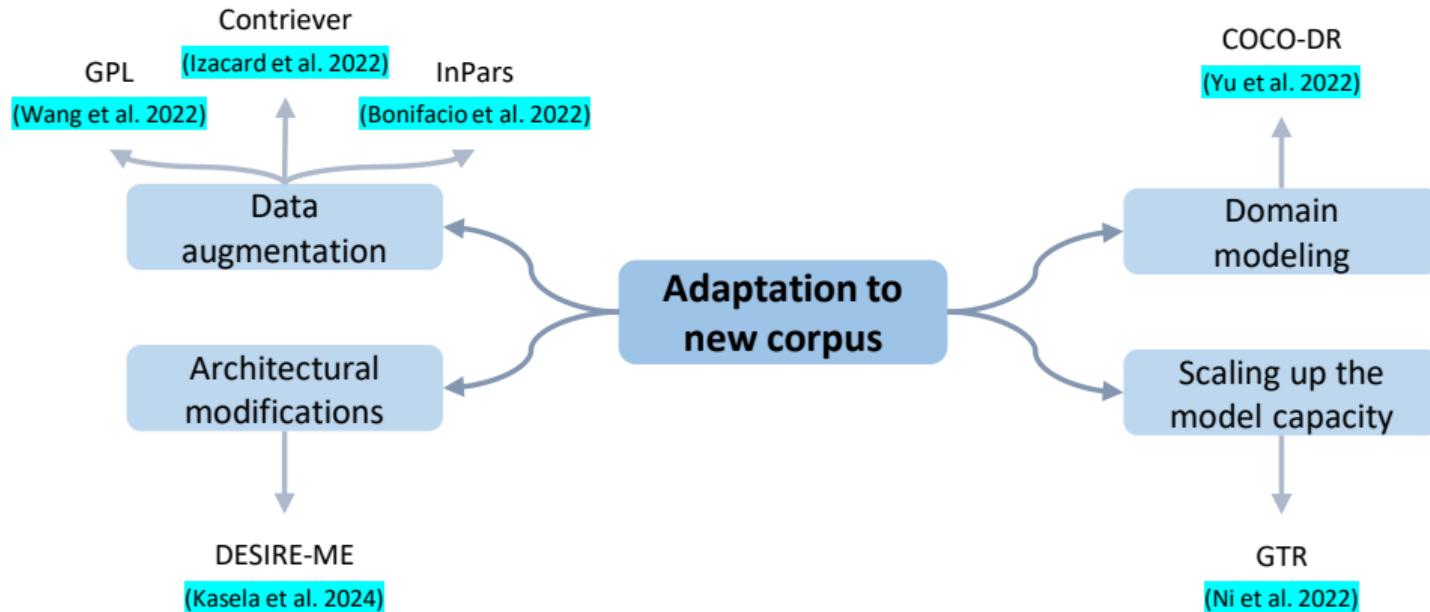


Simple: Straightforward to improve OOD robustness



Costly: High training overhead and requires more training data than before

# Adaptation to new corpus

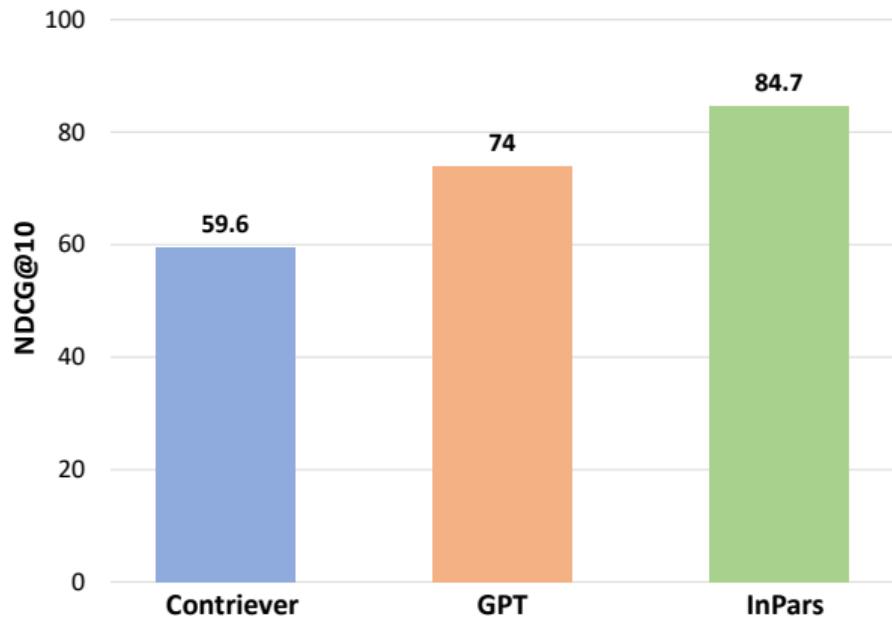


Key idea: Evaluate the **average ranking performance** across different domains

- **NDCG** evaluates the quality of ranking results by measuring the gain of a document based on its position in the ranked list
- **MRR** evaluates the performance of a ranking result by calculating the average of the reciprocal ranks of the first relevant document answer
- **HIT** evaluates the proportion of times a relevant document is found within a set of top-N ranking results
- **AP** evaluates the average performance of the ranking performance metrics, overall new domains

## Comparison between data augmentation methods

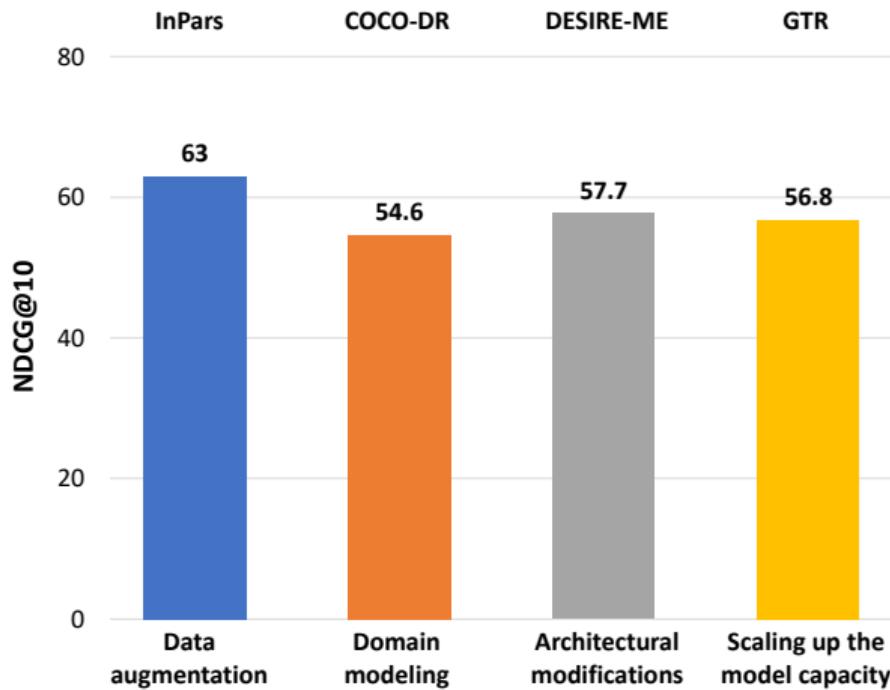
Data source: [Bonifacio et al., 2022, Izacard et al., 2021]



- Original corpus: MS MARCO
- New corpus: TREC-COVID
- Observations: Effectiveness of relevance supervised signals: heuristic < cross-coder judgment < LLMs generation

## Comparison between adaptation to new corpus methods

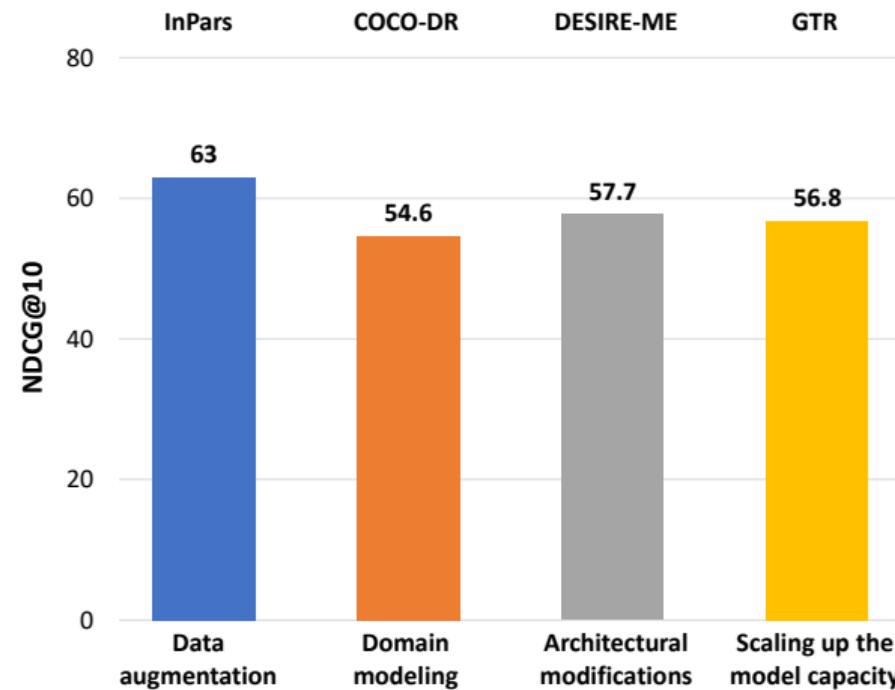
Data source: [Bonifacio et al., 2022, Kasela et al., 2024, Ni et al., 2022]



- Original corpus: MS MARCO
- New corpus: NQ
- Observations: With the help of LLMs, data augmentation becomes the most effective method

## Comparison between adaptation to new corpus methods

Data source: [Bonifacio et al., 2022, Kasela et al., 2024, Ni et al., 2022]



- Original corpus: MS MARCO
- New corpus: NQ
- Observations: Improvements from increasing model capacity or extending the model structure may be limited

## Takeaway

For adaptation to new corpus:

## Takeaway

For adaptation to new corpus:

- High-quality data and an appropriate modeling approach are key to the problem

## Takeaway

For adaptation to new corpus:

- High-quality data and an appropriate modeling approach are key to the problem
- LLMs can play a variety of roles in it

## Takeaway

For adaptation to new corpus:

- High-quality data and an appropriate modeling approach are key to the problem
- LLMs can play a variety of roles in it
- There is a trade-off between efficiency and effectiveness

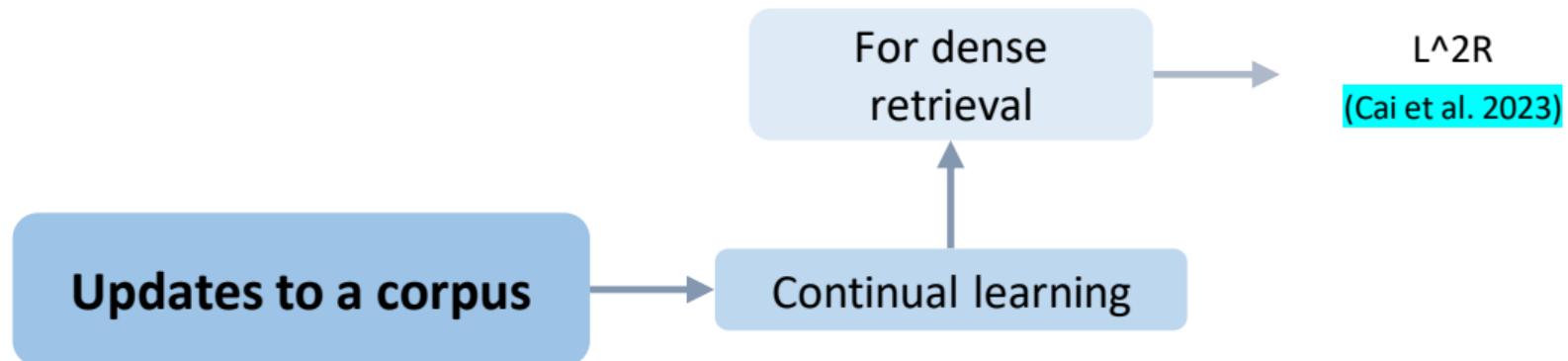
## Classification of updates to a corpus

**Updates to a corpus**

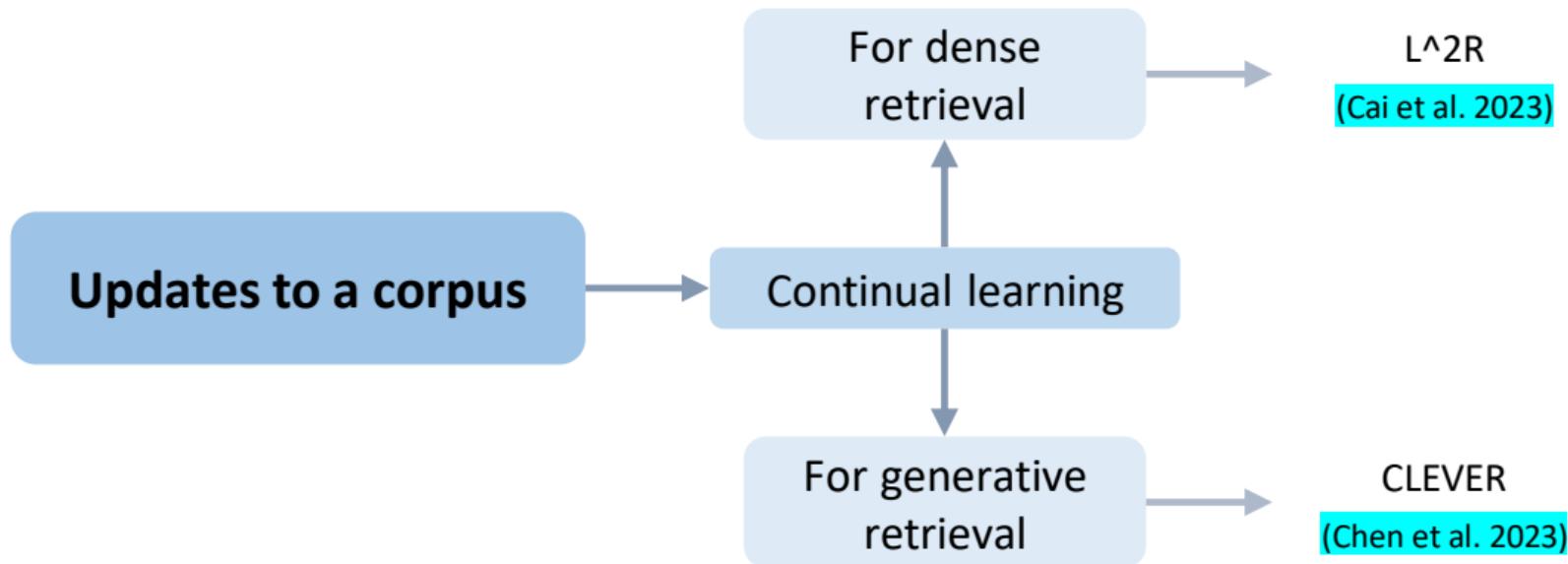
## Classification of updates to a corpus



## Classification of updates to a corpus

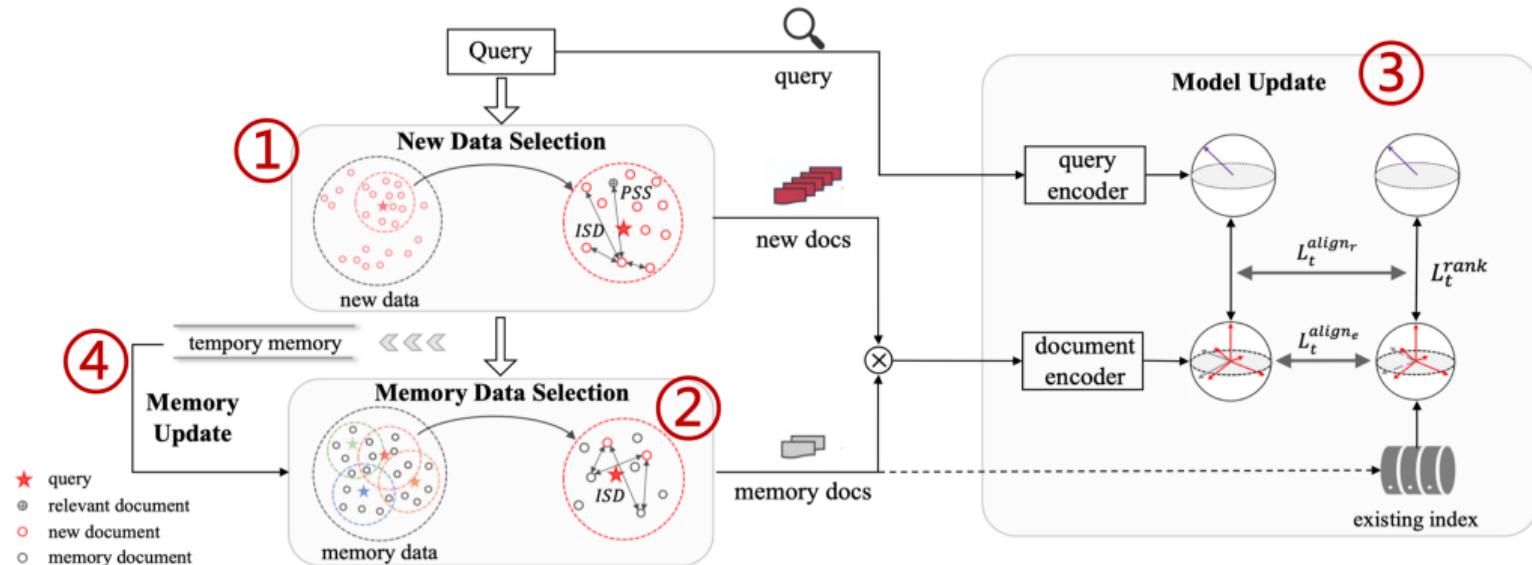


## Classification of updates to a corpus

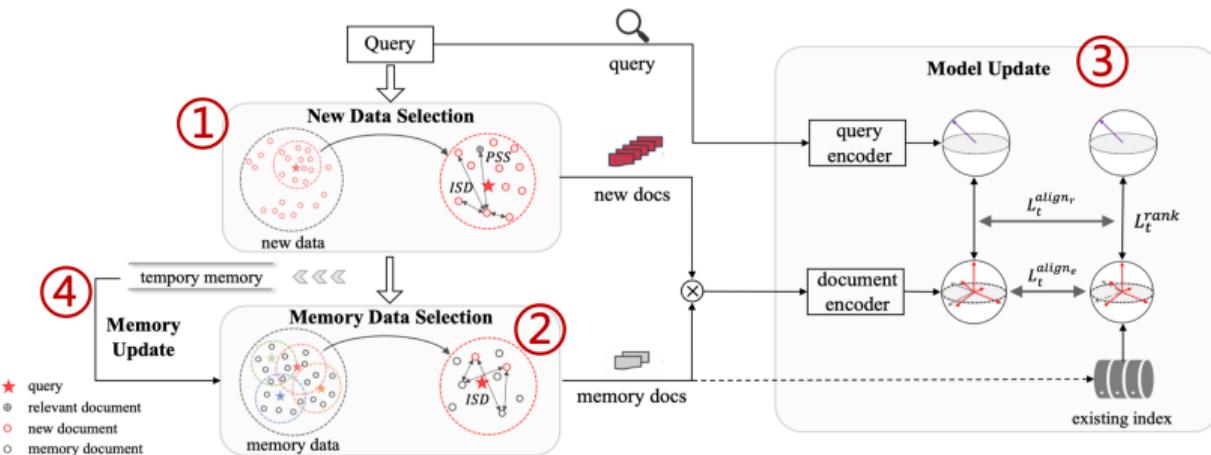


## Updates to a corpus: Dense retrieval

**L<sup>2</sup>R** employs a **replay mechanism** that maintains an external memory for storing a subset of historical documents for replay [Cai et al., 2023]



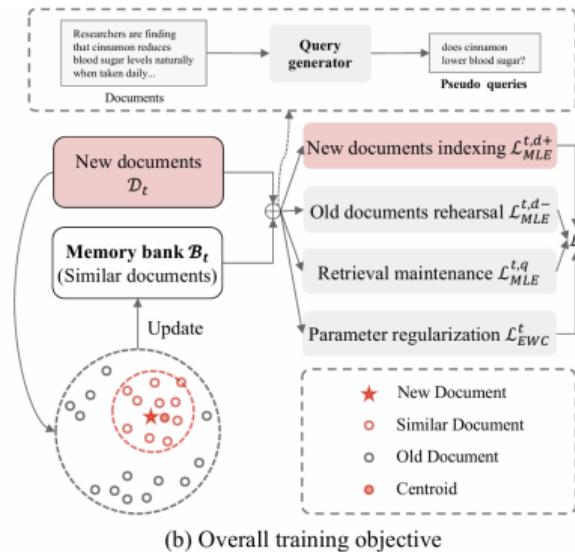
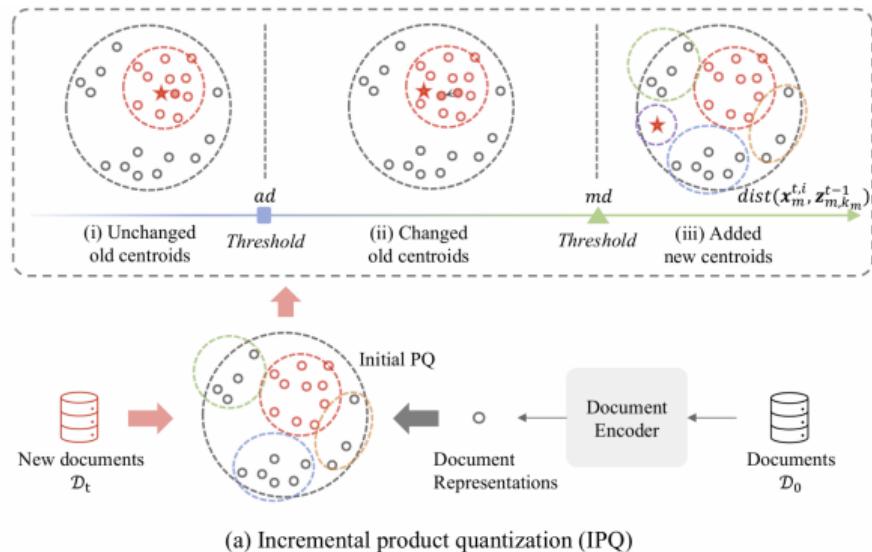
# Updates to a corpus: Dense retrieval



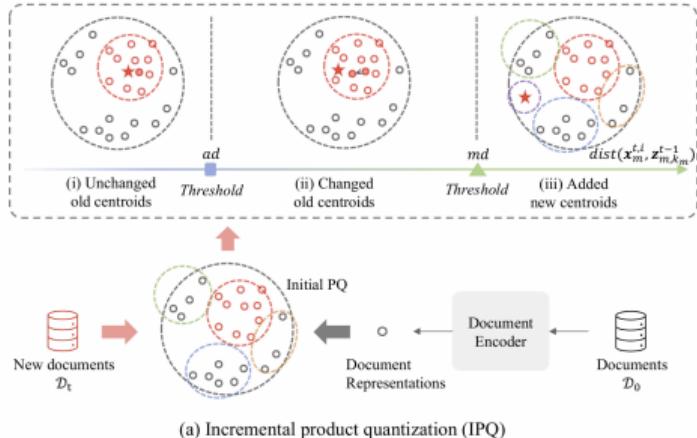
- Expanding new knowledge
- Resolving catastrophic forgetting
- Updating the model based on selected new-old samples
- Updating memory based on new data

## Updates to a corpus: Generative retrieval

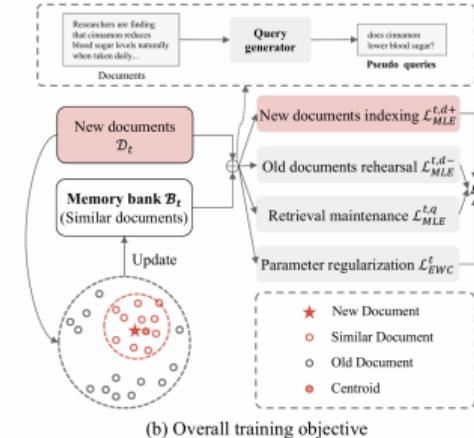
**CLEVER** incrementally indexes new documents while supporting the ability to query both newly encountered documents and previously learned documents [Chen et al., 2023a]



# Updates to a corpus: Generative retrieval



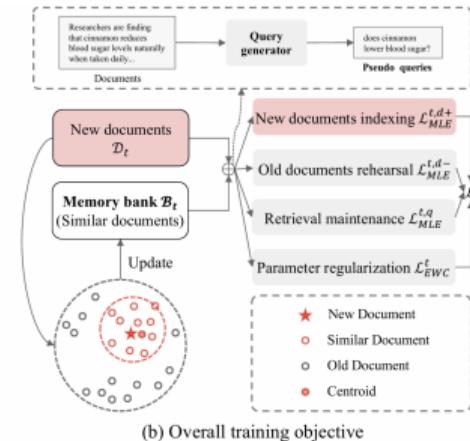
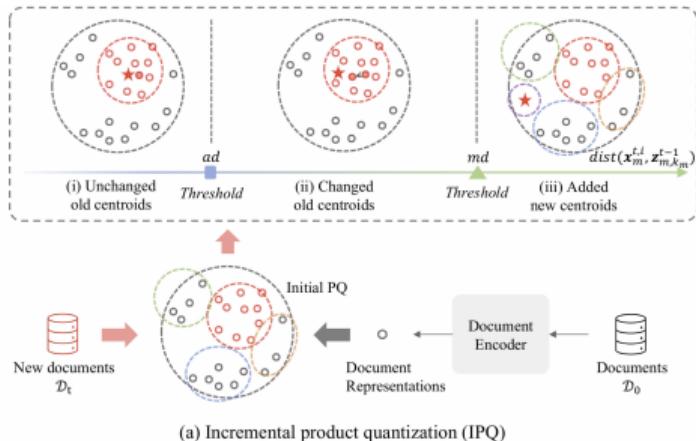
(a) Incremental product quantization (IPQ)



(b) Overall training objective

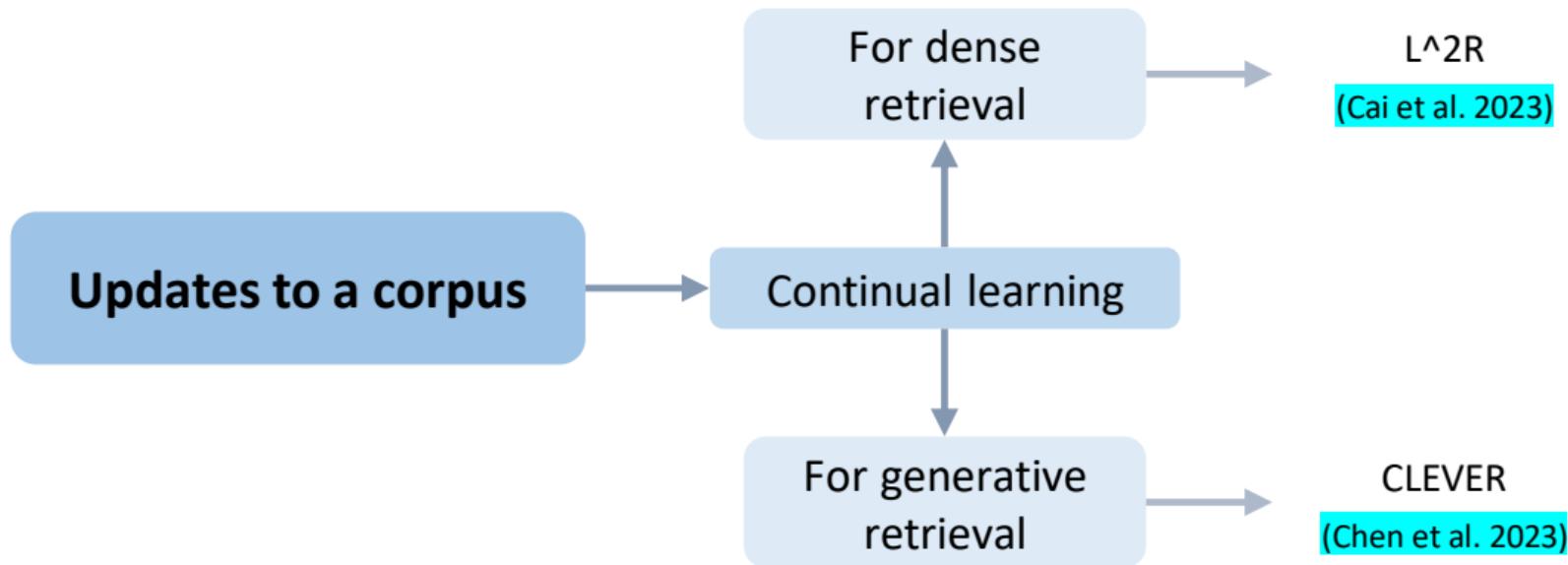
- Encoding new documents into docids by updating a subset of quantization centroids

# Updates to a corpus: Generative retrieval



- Encoding new documents into docids by updating a subset of quantization centroids
- Overall training objective for continual indexing while alleviating forgetting of the retrieval ability

## Updates to a corpus



## Review updates to a corpus methods



Sustainable: Making neural IR models understand new documents as well as not forget old documents in dynamic scenarios

## Review updates to a corpus methods



Sustainable: Making neural IR models understand new documents as well as not forget old documents in dynamic scenarios



Complex: Realization and fine-tuning requires experience

## Specific evaluation for updates to a corpus

Key idea: Besides ranking metrics, we focus on the **forgetting degree of the old corpus**

- **AP** evaluates the average performance over all sessions
- **Training time** evaluates the total time to learn new data while recalling old data
- **Forget<sub>t</sub>** evaluates how much the model forgets at session  $t$ :

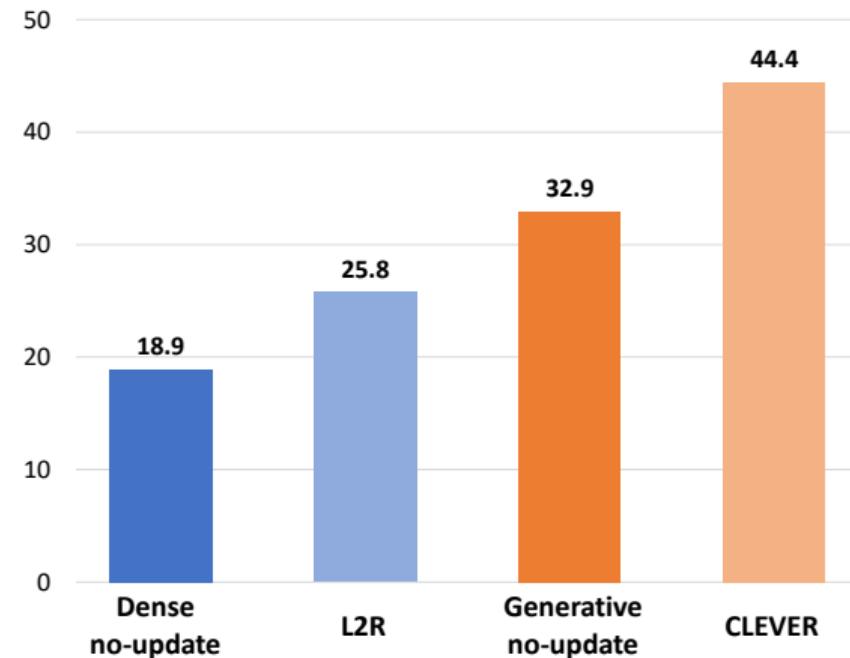
$$\text{Forget}_t = \frac{1}{t} \sum_{j=0}^{t-1} \max_{l \in \{0, \dots, t-1\}} (p_{l,j} - p_{t,j}).$$

- **FWT** evaluates how well the model transfers knowledge from one session to future sessions:

$$\text{FWT} = \frac{\sum_{i=1}^{j-1} \sum_{j=2}^T p_{i,j}}{\frac{T(T-1)}{2}}.$$

## Comparison between updates to a corpus methods

Data source: [Cai et al., 2023, Chen et al., 2023a]



- Dataset of dense retrieval: LL-MultiCPR
- Dataset of generative retrieval: CDI-MS
- Ranking metric: MRR@10
- Observations: Continual learning can effectively improve the performance of dense retrieval and generative retrieval in dynamic scenario

## Takeaway

For updates to a corpus:

## Takeaway

For updates to a corpus:

- Understanding of new data and recall of old data need to be balanced

## Takeaway

For updates to a corpus:

- Understanding of new data and recall of old data need to be balanced
- Effective selection of old data can help understand new data

## Takeaway

For updates to a corpus:

- Understanding of new data and recall of old data need to be balanced
- Effective selection of old data can help understand new data
- Maintaining a well-structured memory is important

## OOD generalizability on unseen queries: Benchmarks

**Query variation datasets** are designed to contain sets of queries that aim for the same information need but are expressed in various ways

## OOD generalizability on unseen queries: Benchmarks

**Query variation datasets** are designed to contain sets of queries that aim for the **same information need** but are expressed in various ways

They can include paraphrased queries, queries with typos, order-swapped queries, and queries without stop words

---

Original query	who wrote most of the declaration of independence
Misspelling	who <b>wreit</b> most of the declaration of independence
Naturality	<b>who</b> wrote most <b>of the</b> declaration <b>of</b> independence
Order	who <b>declaration</b> most of the <b>wrote</b> of independence
Paraphrasing	who <b>authored</b> most of the declaration of independence

---

**Unseen query type datasets** consist of queries that are not represented in the training data, either by virtue of their topic or the **nature of the information being sought**

## OOD generalizability on unseen queries: Benchmarks

**Unseen query type datasets** consist of queries that are not represented in the training data, either by virtue of their topic or the **nature of the information being sought**

For example, the MS MARCO dataset contains 5 types of queries, i.e., location, numeric, person, description, and entity:

Query type	Percentage
Description	53.12%
Numeric	26.12%
Entity	8.81%
Location	6.17%
Person	5.78%

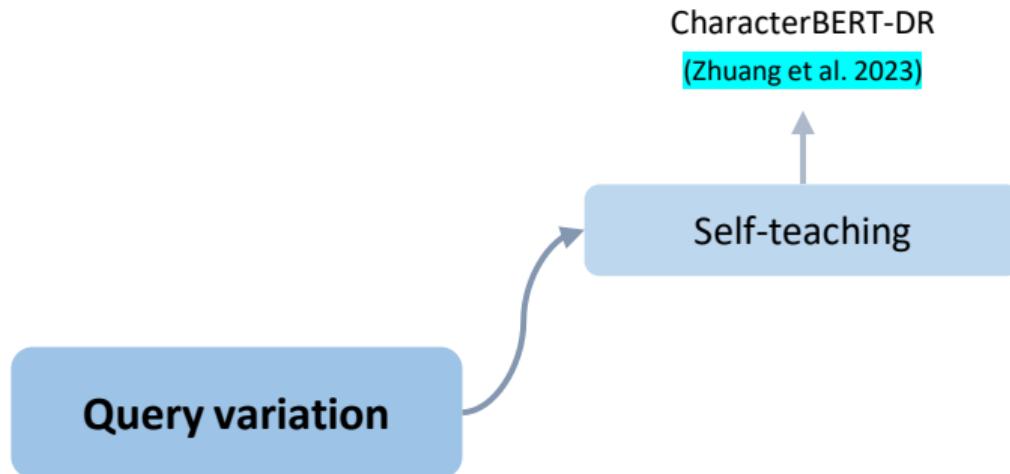
## OOD generalizability on unseen queries: Benchmarks

Type	Dataset	#Q <sub>eval</sub>
Query variation	DL-Typo [Zhuang and Zuccon, 2022]	60
	noisy-MS MARCO [Campos et al., 2023]	5.6k
	rewrite-MS MARCO [Campos et al., 2023]	5.6k
	noisy-NQ [Campos et al., 2023]	2k
	noisy-TQA [Campos et al., 2023]	3k
	noisy-ORCAS [Campos et al., 2023]	20k
	variations-ANTIQUE [Penha et al., 2022]	2k
	variations-TREC19 [Penha et al., 2022]	430
	[Zhuang and Zuccon, 2021]	41k
Unseen query type	MS MARCO [Nguyen et al., 2016]	15k
	L4 [Surdeanu et al., 2008]	10k

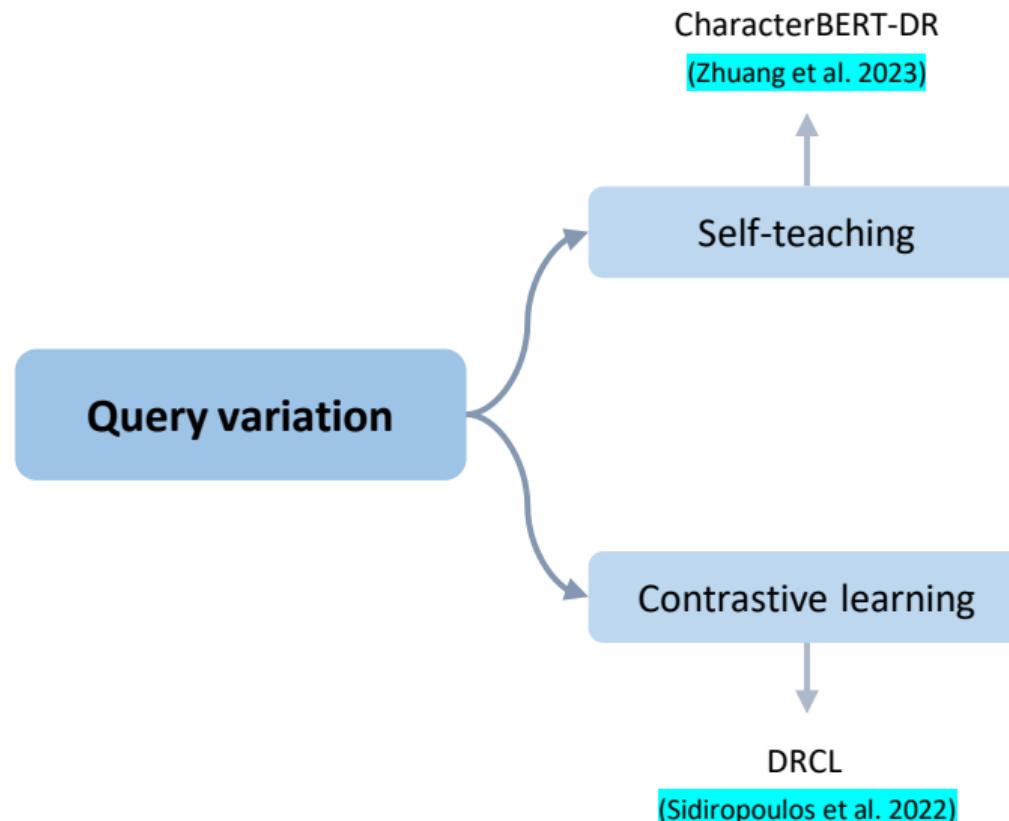
## Classification of query variation

**Query variation**

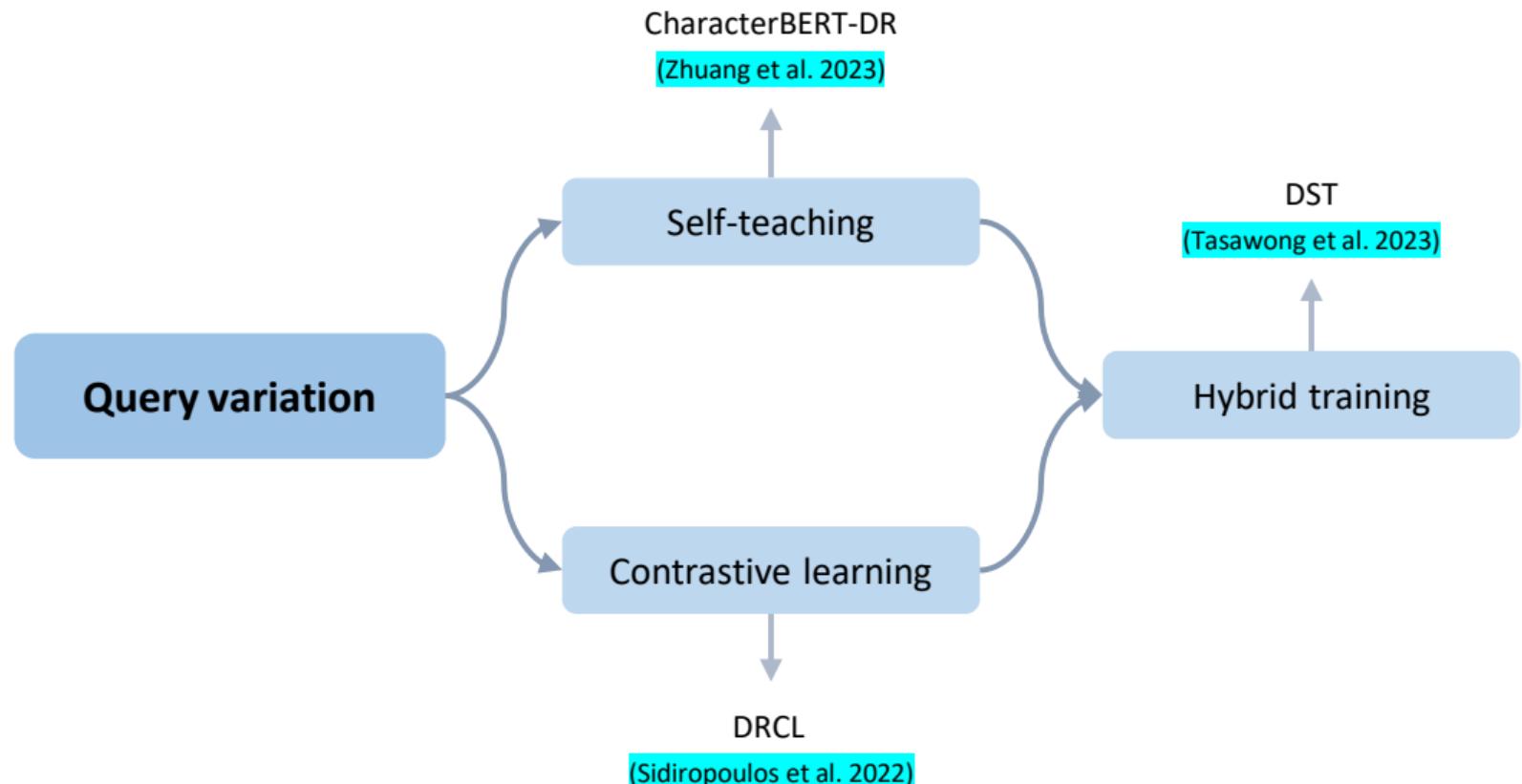
## Classification of query variation



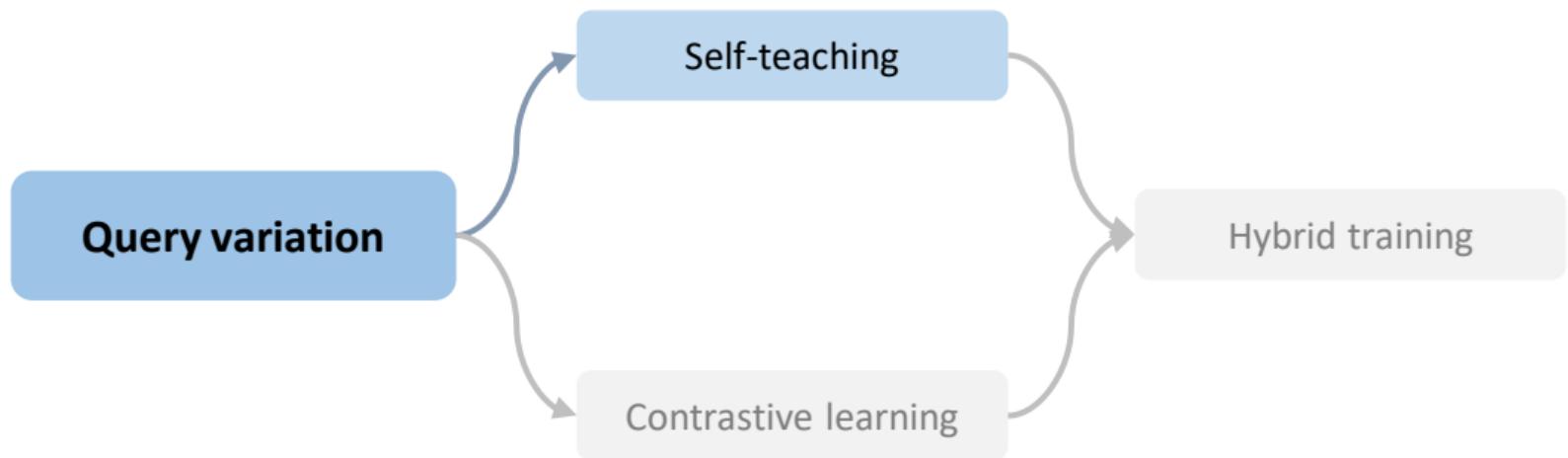
## Classification of query variation



## Classification of query variation

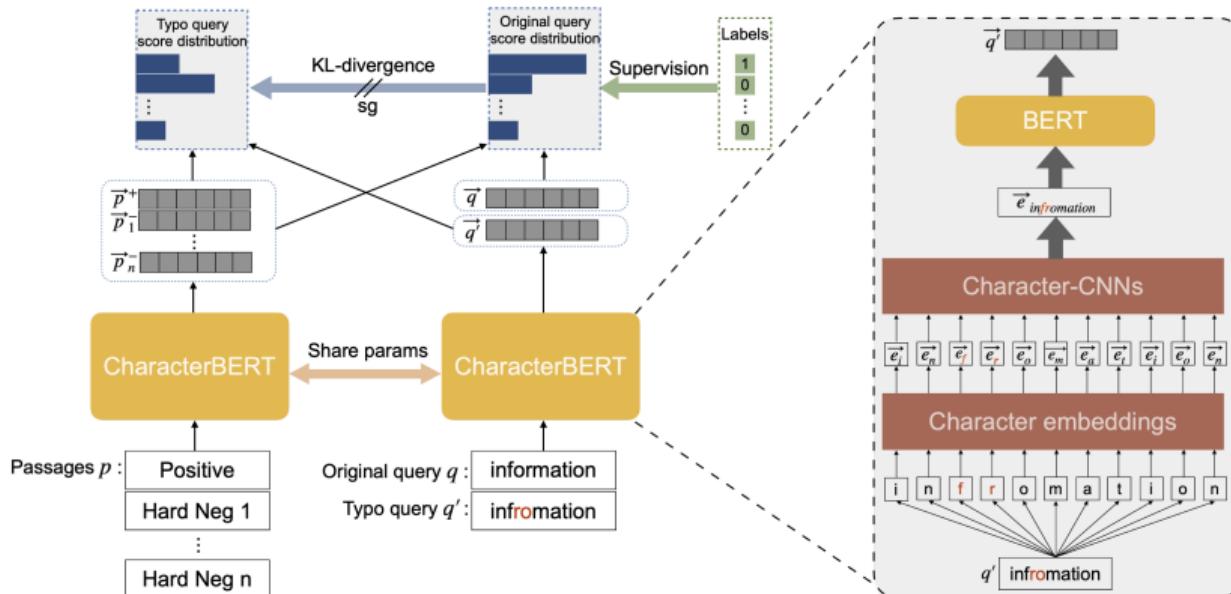


## Query variation: Self-teaching

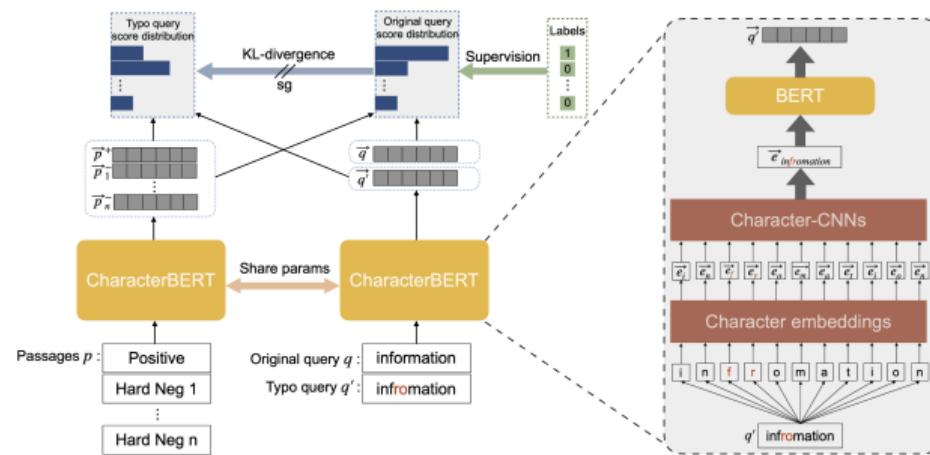


## Query variation: Self-teaching

**CharacterBERT-DR** uses CharacterBERT with a self-teaching training method, that distills knowledge from queries without typos into queries with typos [Zhuang and Zuccon, 2022]

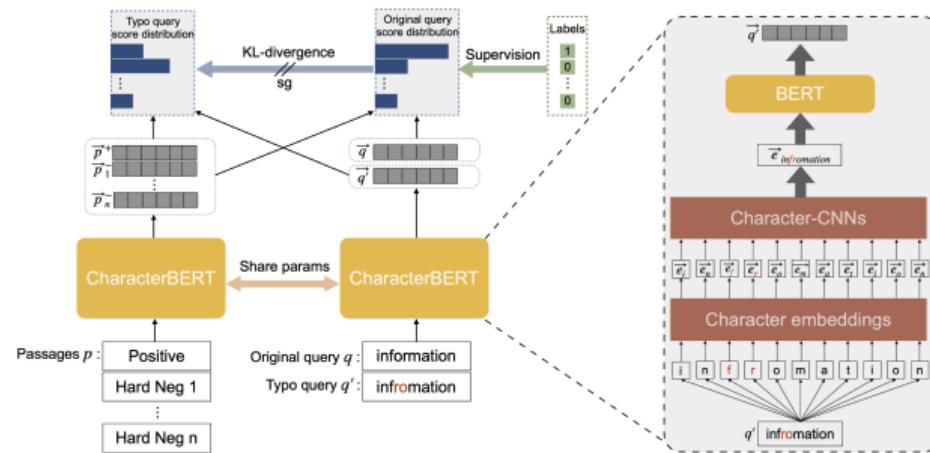


## Query variation: Self-teaching



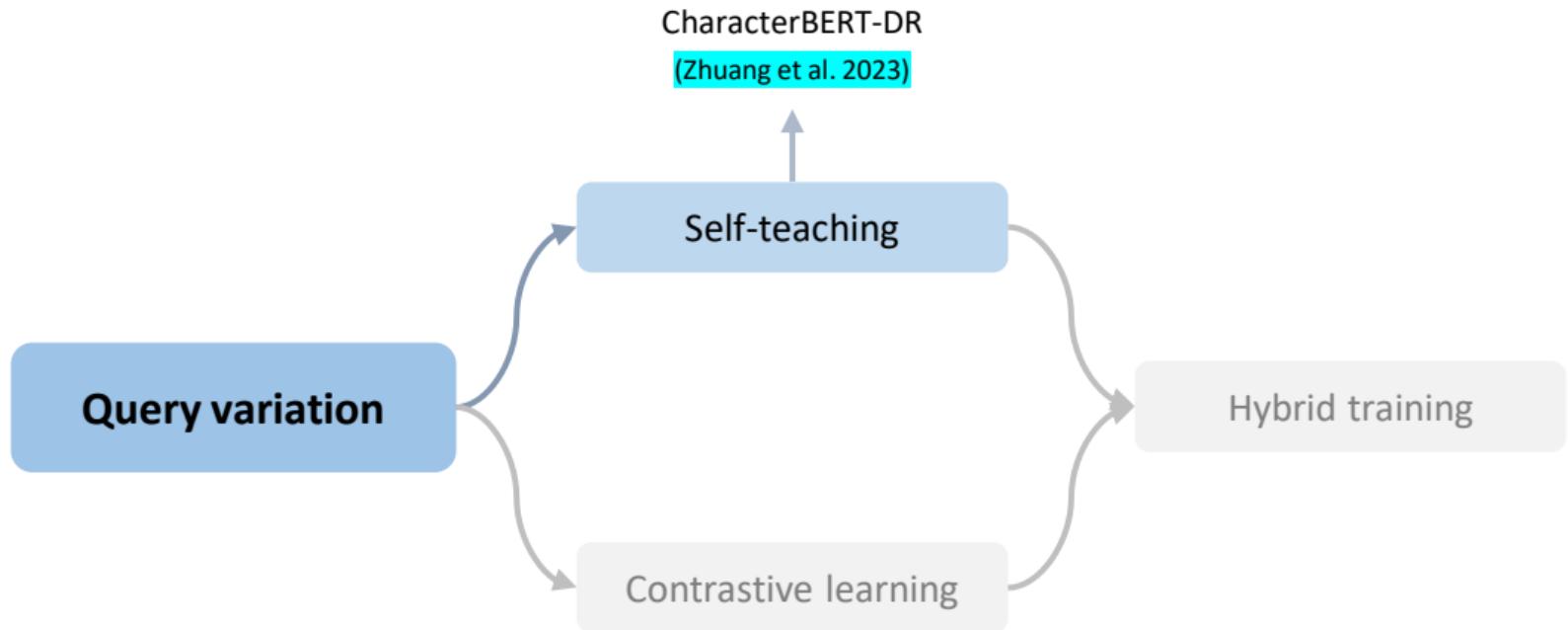
- Modify the [CLS] token embedding output from CharacterBERT to encode both queries and passages

## Query variation: Self-teaching



- Modify the [CLS] token embedding output from CharacterBERT to encode both queries and passages
- Use self-teaching to minimise the difference between the score distribution obtained from the query with typos and the corresponding clean query

## Review self-teaching



## Review self-teaching



Simple: Easy to implement

## Review self-teaching

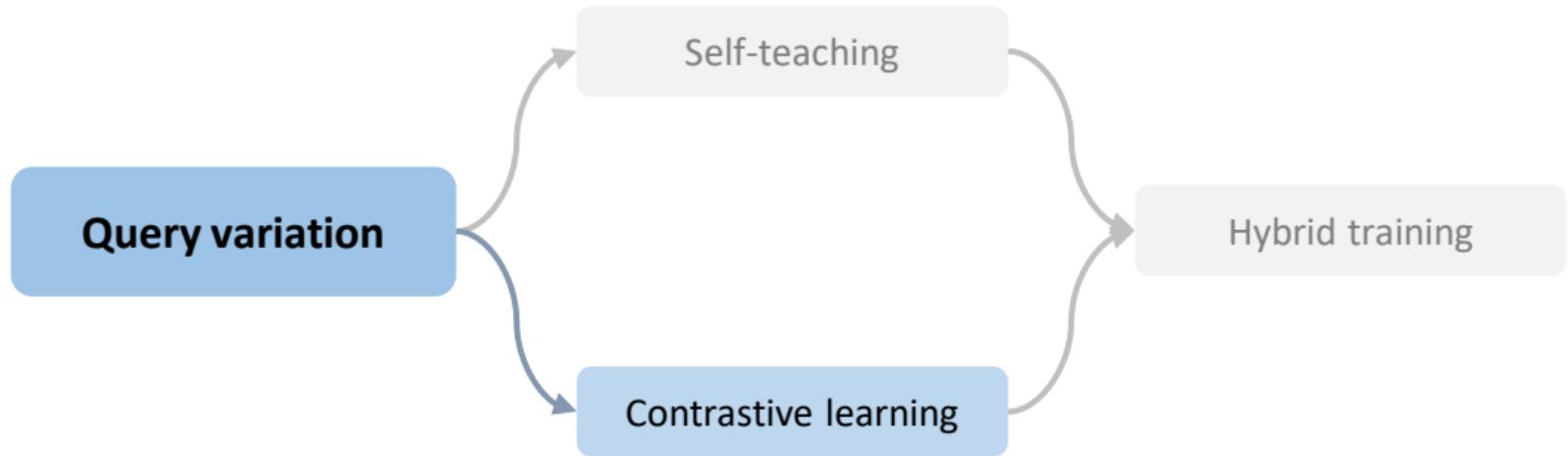


Simple: Easy to implement



Data-starved: Models may not be adequately trained when typo data is limited

## Query variation: Contrastive learning



## Query variation: Contrastive learning

**DRCL** improves robustness under query variations by combining data augmentation with contrastive learning [[Sidiropoulos and Kanoulas, 2022](#)]

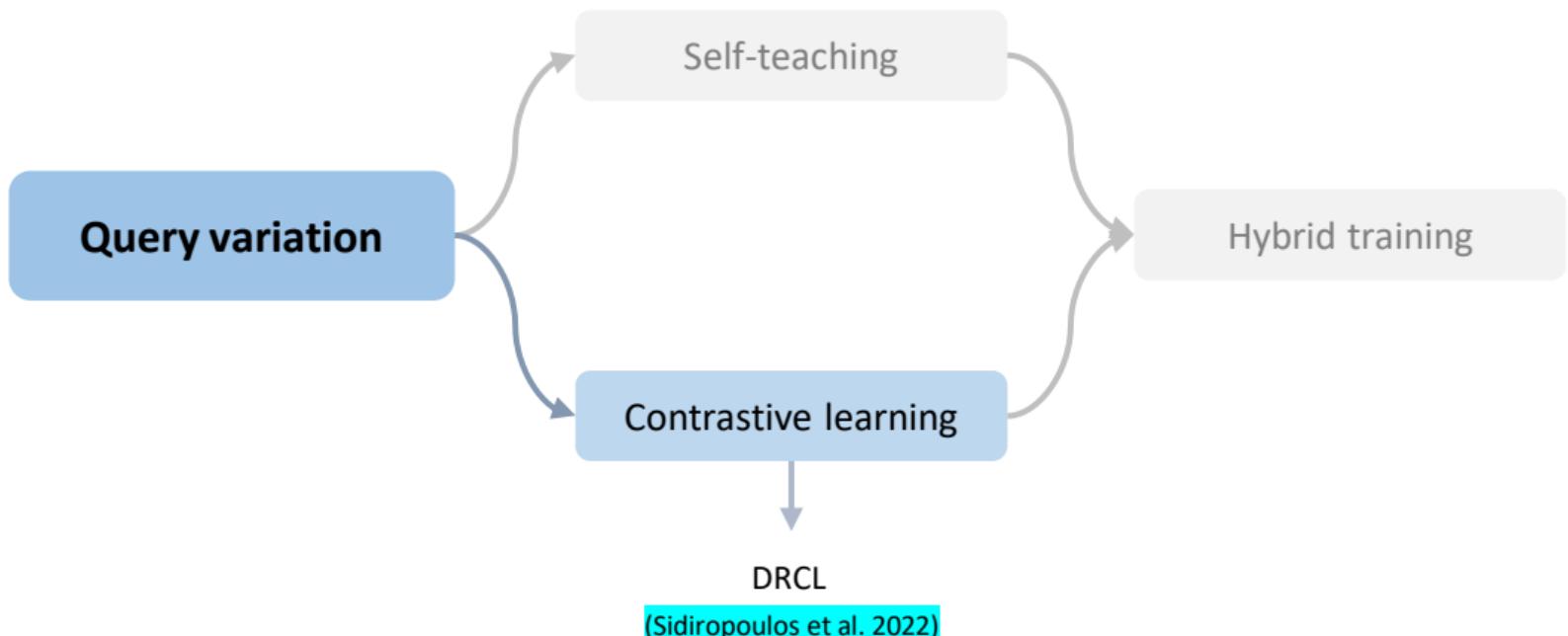
- **Data augmentation:** On training time, each original correctly query is randomly used itself or variations

## Query variation: Contrastive learning

**DRCL** improves robustness under query variations by combining data augmentation with contrastive learning [[Sidiropoulos and Kanoulas, 2022](#)]

- **Data augmentation:** On training time, each original correctly query is randomly used itself or variations
- **Contrastive learning:** Comparing the similarity between a query and its typoed variations and other distinct queries

## Review contrastive learning



## Review contrastive learning



Data-rich: Models can be fully trained

## Review contrastive learning



Data-rich: Models can be fully trained



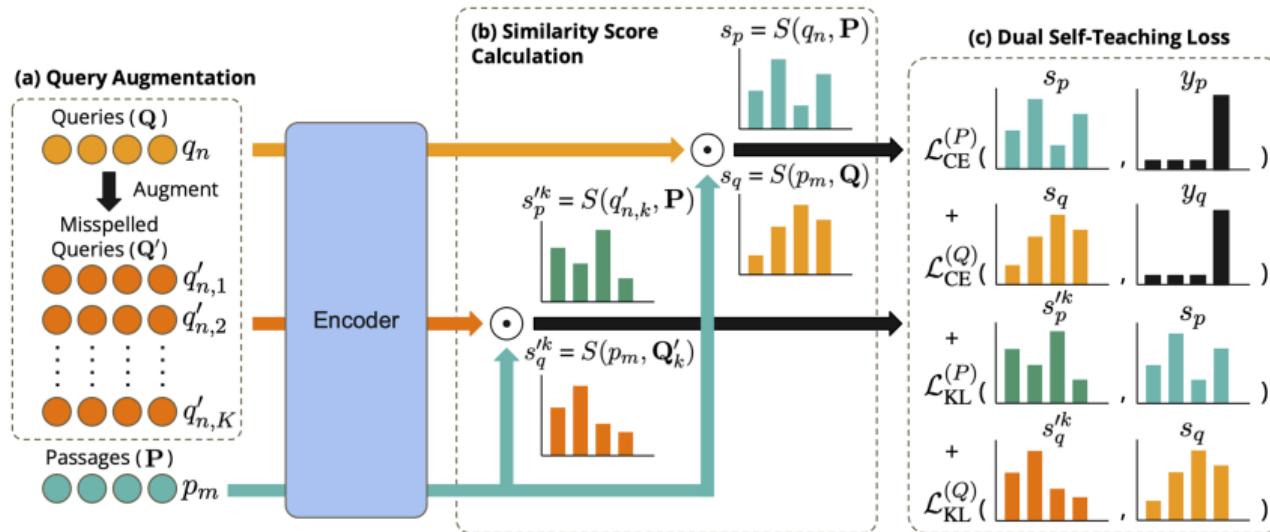
Costly: Need to construct large amounts of training data

## Query variation: Hybrid training

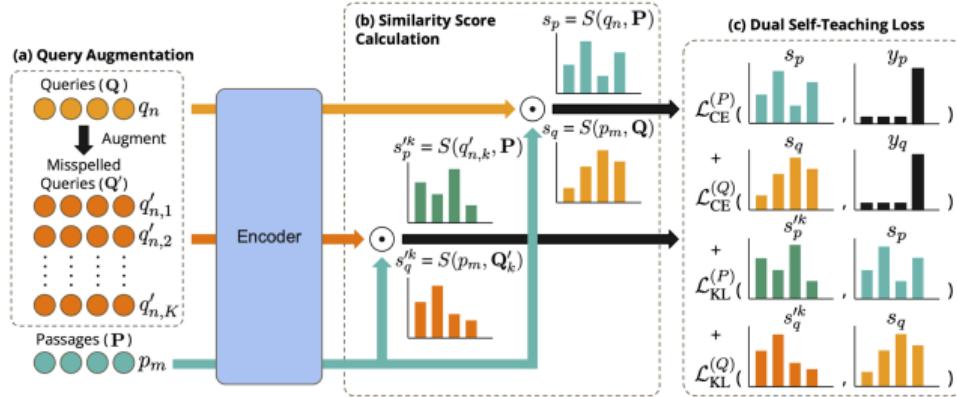


# Query variation: Hybrid training

**DST** adopts the idea of contrastive learning and self-teaching to learn robust representations [Tasawong et al., 2023]

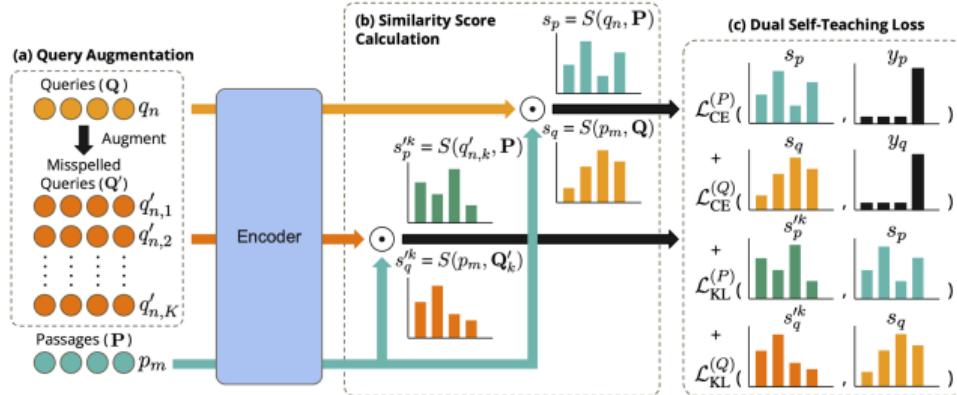


## Query variation: Hybrid training



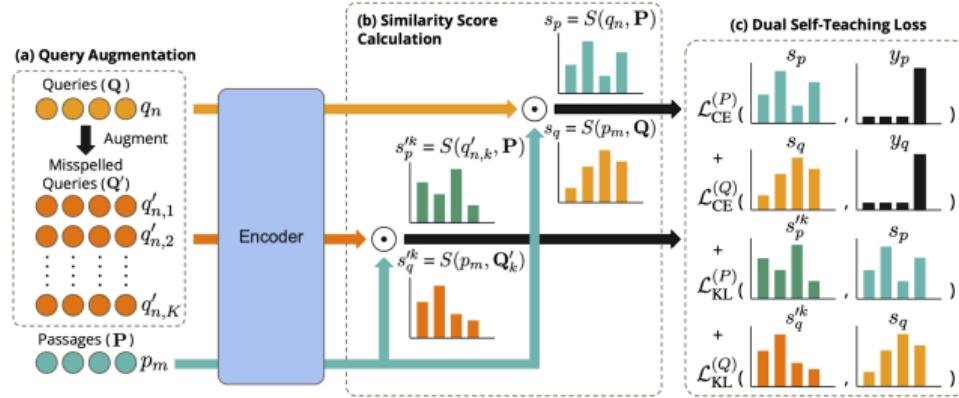
- **Alignment:** align queries with their corresponding passages

## Query variation: Hybrid training



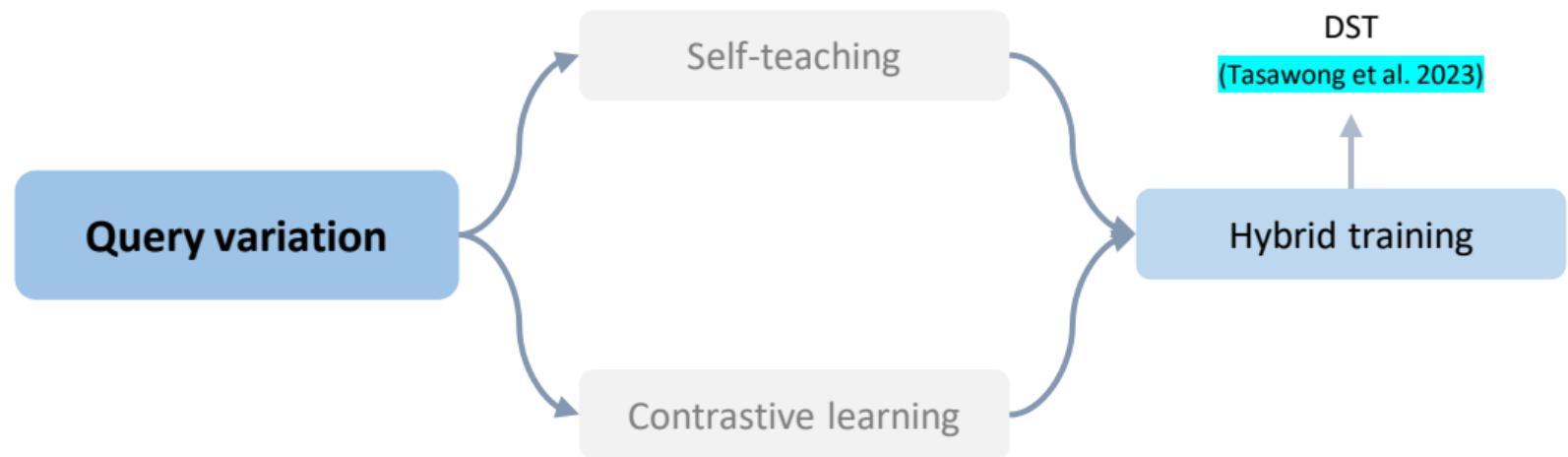
- **Alignment:** align queries with their corresponding passages
- **Robustness:** align misspelled queries with their pristine queries

## Query variation: Hybrid training



- **Alignment:** align queries with their corresponding passages
- **Robustness:** align misspelled queries with their pristine queries
- **Contrast:** separate queries that refer to different passages and passages that correspond to different queries

## Review hybrid training



## Review hybrid training



Sufficient: Multiple training objectives guarantee model robustness to query variants

## Review hybrid training

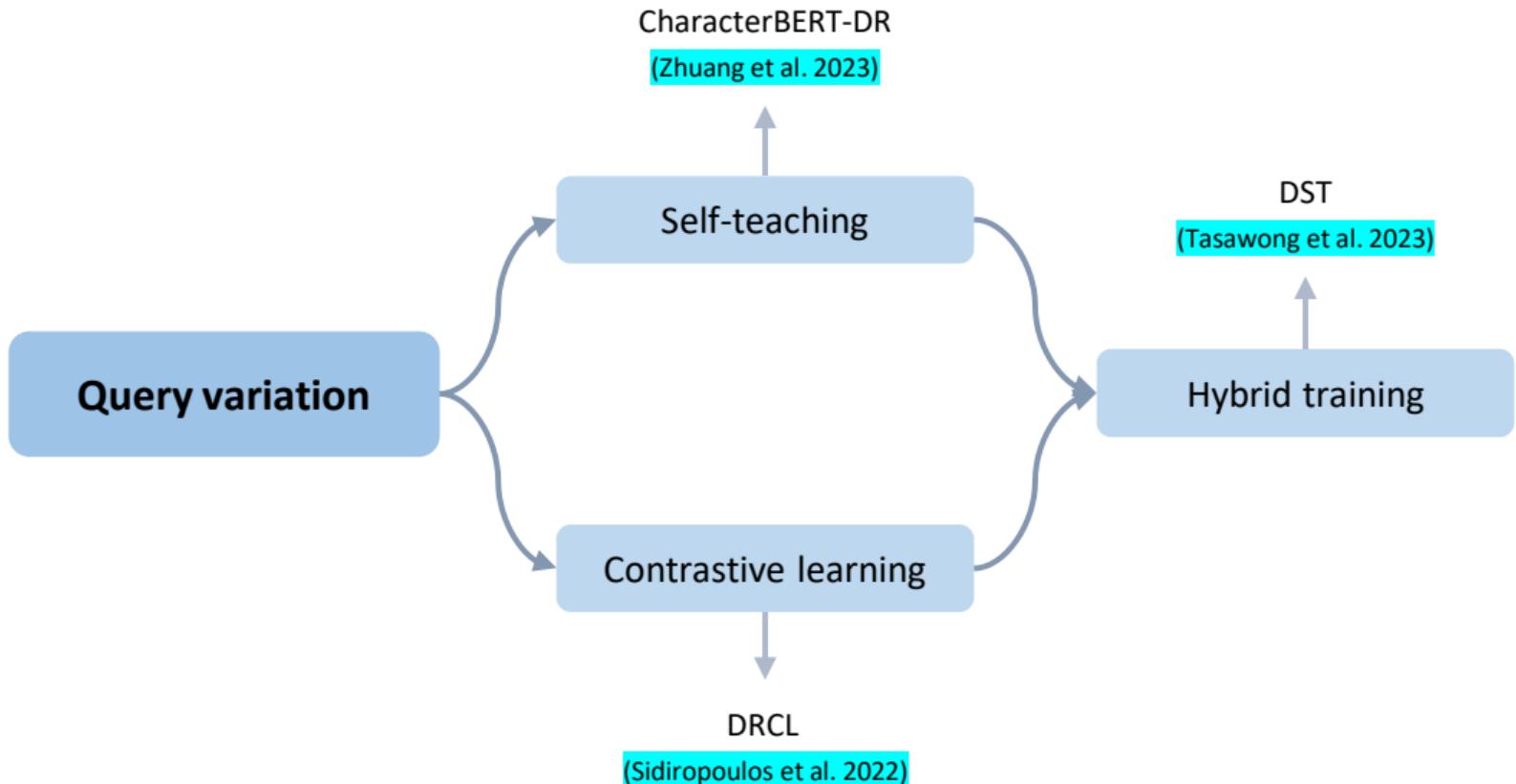


Sufficient: Multiple training objectives guarantee model robustness to query variants



Empirical: The need to balance between different training objectives

## Query variation

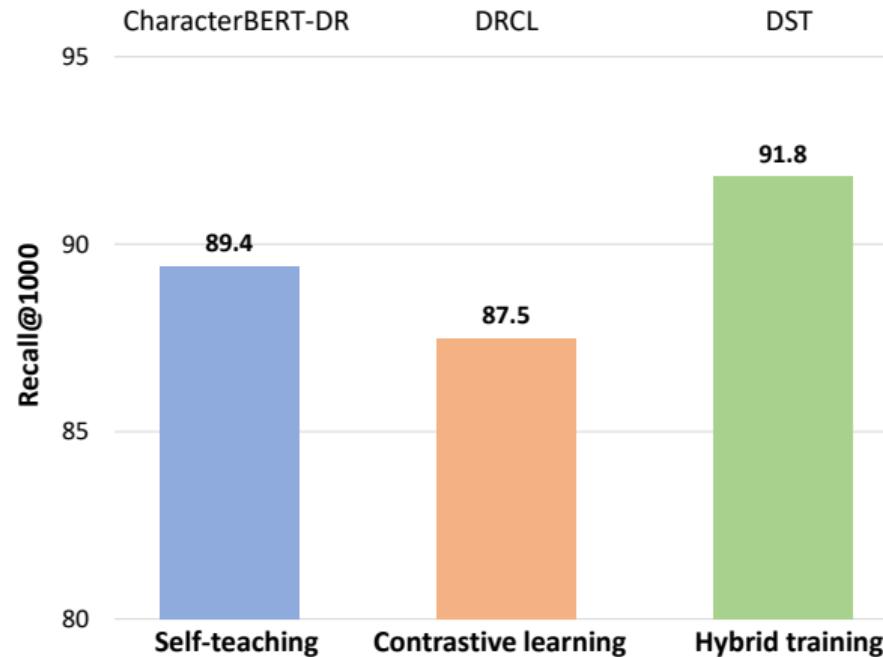


## Evaluation

In addition to MRR and NDCG, the ranking performance under unseen queries is evaluated by other common metrics for query variation and unseen query type

- **Recall** measures the proportion of relevant documents that are successfully retrieved from the total amount of relevant documents available
- **MAP** quantifies the average precision of retrieval across different recall levels, effectively summarizing the precision at each point where a relevant document is retrieved

## Comparison between query variation methods



- Dataset: MS MARCO (with typo)
- Observations: *Self-teaching is made more effective by contrastive learning, and combining these two training methods allows for further model robustness improvements*

## Takeaway

For query variation:

## Takeaway

For query variation:

- An appropriate backbone is the foundation

## Takeaway

For query variation:

- An appropriate backbone is the foundation
- Alignment and contrast are key

## Takeaway

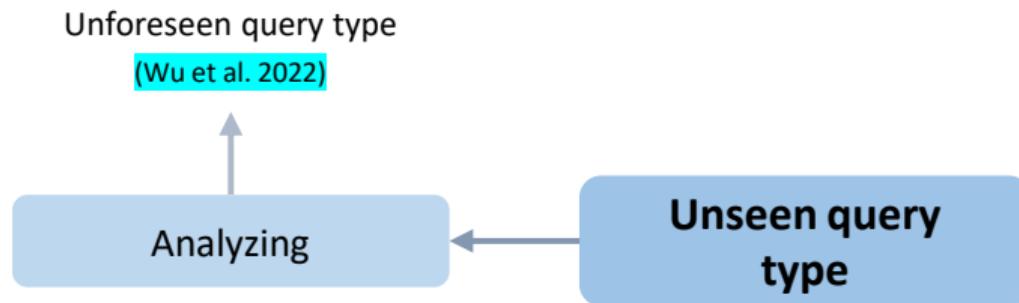
For query variation:

- An appropriate backbone is the foundation
- Alignment and contrast are key
- Integration of various training objectives is the icing on the cake

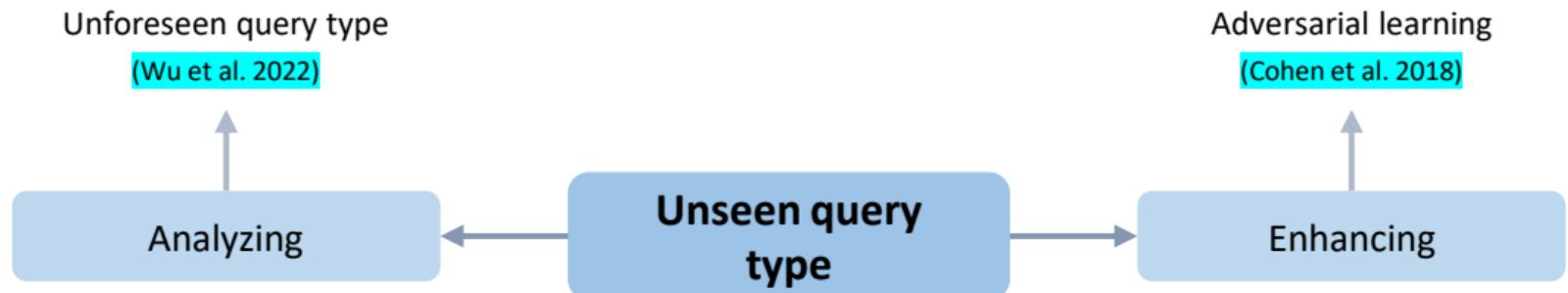
## Classification of unseen query type

**Unseen query  
type**

## Classification of unseen query type



## Classification of unseen query type



## Specific metrics for unseen query type

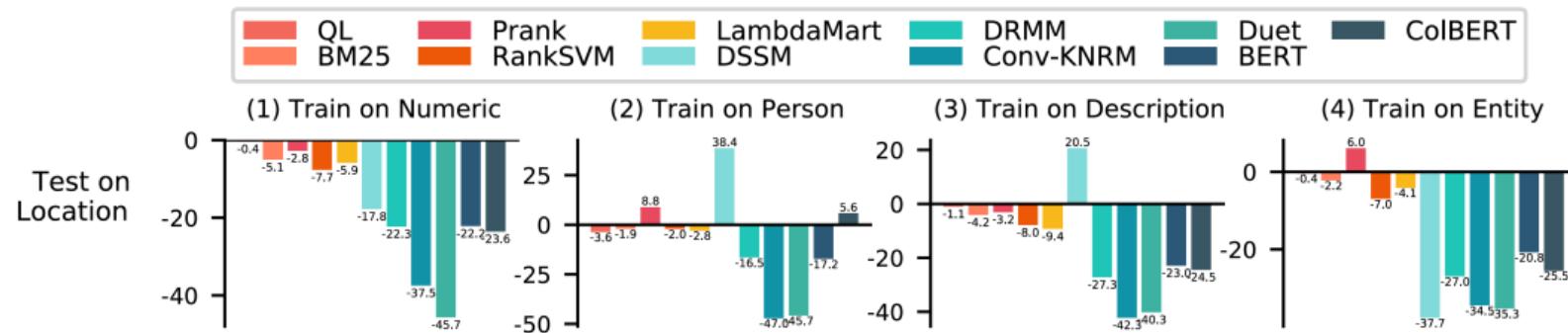
$DR_{OOD}$  evaluates the drop rate between the ranking performance on the original type of queries and the ranking performance on the unseen type of queries:

$$DR_{OOD} = \frac{p_{OOD} - p_{IID}}{p_{IID}},$$

where  $p_{IID}$  is the ranking performance on original type of queries and  $p_{OOD}$  is the ranking performance on unseen type of queries

# Unseen query type: Analyzing

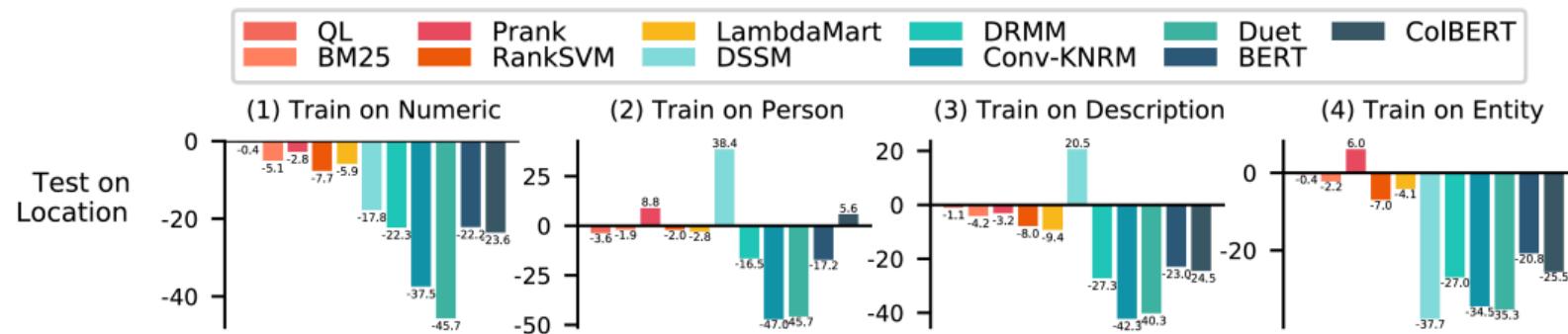
NRMs have poor performance on unseen query types



- NRMs with deep networks can fit seen query types well, at the cost of further loss in performance on the held-out OOD query types

# Unseen query type: Analyzing

NRMs have poor performance on unseen query types



- NRMs with deep networks can fit seen query types well, at the cost of further loss in performance on the held-out OOD query types
- Pre-trained models have shown good robustness to OOD query types

## Unseen query type: Enhancing

Cohen et al. study the effectiveness of **adversarial learning as a cross-domain regularizer** to deal with unseen query type [Cohen et al., 2018]

- Force the NRMs to learn domain-independent features that are useful to estimate relevance

## Unseen query type: Enhancing

Cohen et al. study the effectiveness of **adversarial learning as a cross-domain regularizer** to deal with unseen query type [Cohen et al., 2018]

- Force the NRMs to learn domain-independent features that are useful to estimate relevance
- Shift the model parameters in the opposite direction to the domain specific spaces on the manifold

## Unseen query type: Enhancing

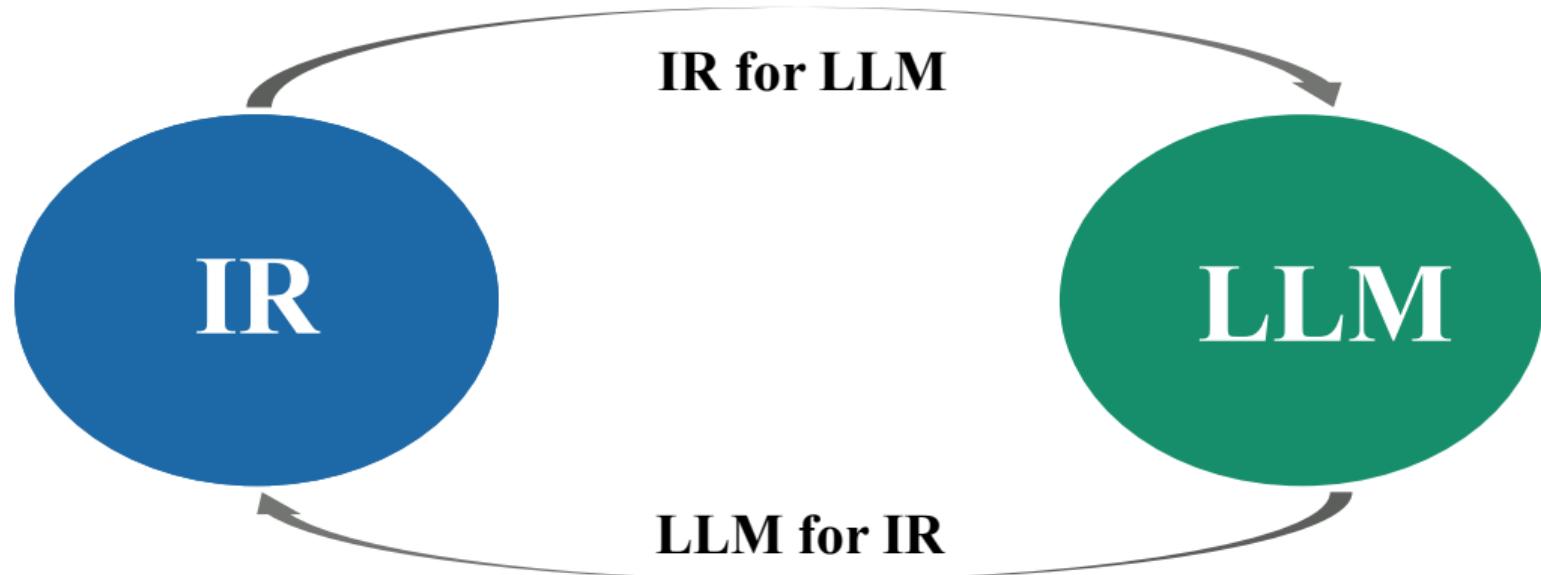
Cohen et al. study the effectiveness of **adversarial learning as a cross-domain regularizer** to deal with unseen query type [Cohen et al., 2018]

- Force the NRMs to learn domain-independent features that are useful to estimate relevance
- Shift the model parameters in the opposite direction to the domain specific spaces on the manifold

Further work in this field is waiting to be explored ...

## **Section 5:**

# **Robust IR in the age of LLMs**



- **IR for LLM:** Retrieval-augmented generation
- **LLM for IR:** A double-edged sword

**New opportunities for IR robustness via LLMs**

## New opportunities to adversarial robustness

LLMs hold promise for improving the adversarial robustness of IR systems through their ability to generate and identify adversarial examples:

## New opportunities to adversarial robustness

LLMs hold promise for improving the adversarial robustness of IR systems through their ability to generate and identify adversarial examples:

- **Generating adversarial examples with LLMs**
  - AIGC scenario
  - Superior capabilities in language generation and interaction
  - Hardening the IR system with generated adversarial samples

## New opportunities to adversarial robustness

LLMs hold promise for improving the adversarial robustness of IR systems through their ability to generate and identify adversarial examples:

- **Generating adversarial examples with LLMs**
  - AIGC scenario
  - Superior capabilities in language generation and interaction
  - Hardening the IR system with generated adversarial samples
- **Adversarial defense assisted with LLMs**
  - Identifying adversarial samples
  - Enhancing existing defense strategies

## New opportunities to OOD robustness

The powerful generation and language understanding capability of LLMs can help to improve the OOD robustness of IR systems:

The powerful generation and language understanding capability of LLMs can help to improve the OOD robustness of IR systems:

- **Synthesizing OOD training data with LLMs**
  - LLMs can generate diverse and complex datasets that **mirror OOD scenarios**
  - **Synthetic data** can help improve the generalizability and robustness of IR models against OOD inputs

The powerful generation and language understanding capability of LLMs can help to improve the OOD robustness of IR systems:

- **Synthesizing OOD training data with LLMs**

- LLMs can generate diverse and complex datasets that **mirror OOD scenarios**
- **Synthetic data** can help improve the generalizability and robustness of IR models against OOD inputs

- **LLMs for OOD detection**

- With capabilities of language understanding, LLMs can **detect OOD queries**
- Neural IR models may perform worse on these OOD queries that deviate from the training distribution

**New challenges for IR robustness via LLMs**

## New challenges to adversarial robustness

When applied to IR systems, the adversarial vulnerability of the LLMs themselves is introduced at the same time, as demonstrated by:

## New challenges to adversarial robustness

When applied to IR systems, the adversarial vulnerability of the LLMs themselves is introduced at the same time, as demonstrated by:

- **The vulnerability caused by hallucinations of LLMs**
- **Defense costs associated with the scale and opacity of LLMs**

## New challenges to adversarial robustness

### The vulnerability caused by hallucinations of LLMs

- With **hallucination**, LLMs can generate plausible yet factually incorrect information
- Such reliance can undermine the **trustworthiness and reliability** of the IR system



## New challenges to adversarial robustness

Defense costs associated with the scale and opacity of LLMs

- LLMs operate as **black boxes** with limited transparency into how decisions are made
- This **opacity** complicates efforts to diagnose and mitigate vulnerabilities



## New challenges to OOD robustness

LLMs have shown biases and input sensitivities in existing work, and these will affect the OOD robustness of IR systems:

LLMs have shown biases and input sensitivities in existing work, and these will affect the OOD robustness of IR systems:

- **Bias in the corpus domain of LLMs**
  - The training process of LLMs leads to a **bias towards the domain characteristics**
  - This can degrade performance when the model encounters OOD queries or documents

LLMs have shown biases and input sensitivities in existing work, and these will affect the OOD robustness of IR systems:

- **Bias in the corpus domain of LLMs**

- The training process of LLMs leads to a **bias towards the domain characteristics**
- This can degrade performance when the model encounters OOD queries or documents

- **Sensitivity of LLMs to query inputs**

- LLMs can exhibit **high sensitivity** to slight variations in input
- This potentially leads to significantly different IR outcomes

So much to do ...

**Making robustness one of the hallmarks of IR in the age of LLMs!**

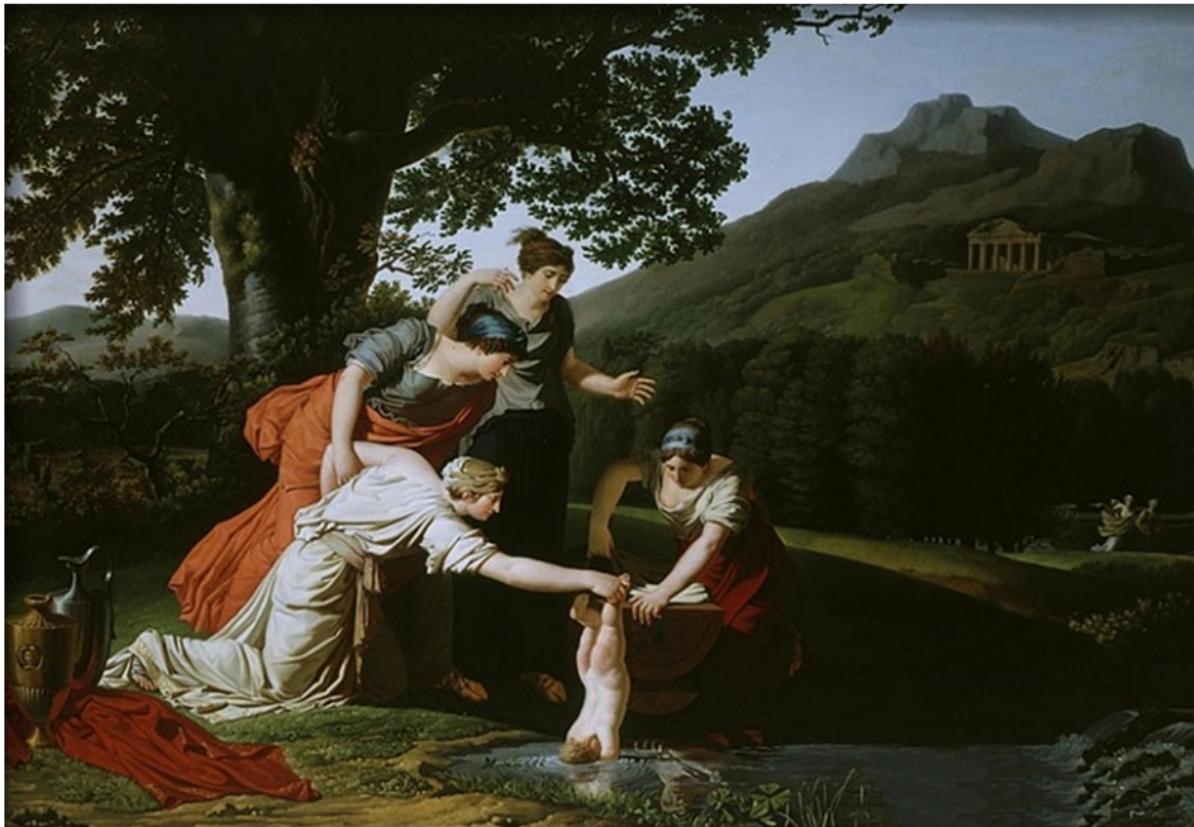
## **Section 6:**

# **Conclusions and future directions**

# Tutorial summary

- **Introduction**
- **Preliminaries**
  - Definition of robustness in IR
  - Taxonomy of robustness in IR
- **Adversarial robustness**
  - Benchmark, settings, task definition and evaluations
  - Adversarial attacks: steal black-box knowledge → identify vulnerable positions → add adversarial perturbations
  - Adversarial defenses: empirical defense, certified defense and attack detection
- **Out-of-distribution robustness**
  - OOD generalizability on unseen documents: new corpus and incrementation of original corpus
  - OOD generalizability on unseen queries: query variation and unseen query type
- **Robust IR in the age of LLMs**

## Robustness: The Achilles' heel of neural IR models



Thetis Dipping Achilles into the River Styx - Antoine Borel (1743-1810)

If robustness is so hard, what can we do with our neural IR systems today?

## Mitigating robustness

- **Before going-to-production:** Optimizing training objectives, introducing perturbations in advance
- **While in production:** Customizing analysis tools, monitoring of operational status regularly
- **Post-hoc correction:** Improving system interpretability, optimizing for weaknesses

What about tomorrow?

What about tomorrow?

**Much done, much left to do**

## Future directions for adversarial robustness

There are currently some dilemmas of adversarial robustness in IR that are worthy of future attention in the endeavor:

## Future directions for adversarial robustness

There are currently some dilemmas of adversarial robustness in IR that are worthy of future attention in the endeavor:

- **Game theory:** Modeling the market behavior of real SEO
- **Toolkit:** A systematic platform for integrating attack and defense methods
- **Industrial practice:** Considering the deployment in specific operations

## Future directions for adversarial robustness: Game theory

**Background:** In real search engine SEO scenarios, there are **multiple attackers**, working individually or in groups, with consistent or not-exactly-consistent goals.

## Future directions for adversarial robustness: Game theory

**Background:** In real search engine SEO scenarios, there are **multiple attackers**, working individually or in groups, with consistent or not-exactly-consistent goals.

**Dilemma:** It is difficult to analyze the **impact of this scaled SEO behavior** on search engines, let alone counter them.

## Future directions for adversarial robustness: Game theory

**Background:** In real search engine SEO scenarios, there are **multiple attackers**, working individually or in groups, with consistent or not-exactly-consistent goals.

**Dilemma:** It is difficult to analyze the **impact of this scaled SEO behavior** on search engines, let alone counter them.

**Promising way:** Game theory

- Multiple attackers seeking to profit is essentially a gaming problem
- Game theory can be used to find an equilibrium in this scenario
- SEO can be curbed by adjusting search engine rules to tilt the balance in favor of the user

**Background:** With the development of adversarial robustness, various attacks, defense methods and experimental datasets have emerged.

**Background:** With the development of adversarial robustness, **various attacks, defense methods and experimental datasets** have emerged.

**Dilemma:** The lack of a unified specification leads to poor direct comparability of methods, which in turn affects the accurate understanding of model robustness.

**Background:** With the development of adversarial robustness, **various attacks, defense methods and experimental datasets** have emerged.

**Dilemma:** The lack of a unified specification leads to poor direct comparability of methods, which in turn affects the accurate understanding of model robustness.

**Promising way:** Toolkit

- A high-quality codebase for robust IR research
- A unified data processing pipeline, simplified model configuration and automatic hyper-parameters tuning features equipped

## Future directions for adversarial robustness: Industrial practice

**Background:** Current adversarial attacks and defenses are mainly studied in relatively plain and contained experimental scenarios

**Background:** Current adversarial attacks and defenses are mainly studied in relatively plain and contained experimental scenarios

**Dilemma:** In real search engines, the situation that may be faced is much more complex, which may make it difficult to apply existing methods on the ground

**Background:** Current adversarial attacks and defenses are mainly studied in relatively plain and contained experimental scenarios

**Dilemma:** In real search engines, the situation that may be faced is much more complex, which may make it difficult to apply existing methods on the ground

**Promising way:** Industrial practice

- Foster academic-industrial collaborations on the topic
- Designing appropriate defense mechanisms for realistic and specific SEO scenarios

## Future directions for OOD robustness

There are also currently some dilemmas of OOD robustness in IR that are worthy of future attention in the endeavor:

## Future directions for OOD robustness

There are also currently some dilemmas of OOD robustness in IR that are worthy of future attention in the endeavor:

- **Causality modeling:** Identifying spurious correlation factors between documents and queries
- **Toolkit:** A systematic platform for integrating OOD documents and queries
- **Industrial practice:** Considering the deployment in specific operations

## Future directions for OOD robustness: Causality modeling

**Background:** Some neural IR models focus on **spurious correlations** within the domain, leading to poor out-of-distribution performance

## Future directions for OOD robustness: Causality modeling

**Background:** Some neural IR models focus on **spurious correlations** within the domain, leading to poor out-of-distribution performance

**Dilemma:** To address this problem, we currently rely on constructing large amounts of new domain data, which has significant overhead.

## Future directions for OOD robustness: Causality modeling

**Background:** Some neural IR models focus on **spurious correlations** within the domain, leading to poor out-of-distribution performance

**Dilemma:** To address this problem, we currently rely on constructing large amounts of new domain data, which has significant overhead.

**Promising way:** Causality modeling

- Causal modeling can effectively identify the key factors in a document that determine the relevance of a query
- When the domain changes, these key factors remain the same

## Future directions for OOD robustness: Toolkit

**Background:** In current approaches, OOD solutions for queries and documents are relatively separate, yet in search engines, these two problems often arise simultaneously

## Future directions for OOD robustness: Toolkit

**Background:** In current approaches, OOD solutions for queries and documents are relatively separate, yet in search engines, these two problems often arise simultaneously

**Dilemma:** It is currently difficult to analyze the full performance of specific methods under various OOD issues

## Future directions for OOD robustness: Toolkit

**Background:** In current approaches, OOD solutions for queries and documents are relatively separate, yet in search engines, these two problems often arise simultaneously

**Dilemma:** It is currently difficult to analyze the full performance of specific methods under various OOD issues

**Promising way:** Toolkit

- A unified experimental platform is needed to accommodate possible OOD problems
- A good solution should perform consistently in a variety of OOD scenarios

## Future directions for OOD robustness: Industrial practice

**Background:** In real search engines, there are more specific requirements for the fitness of neural IR models.

## Future directions for OOD robustness: Industrial practice

**Background:** In real search engines, there are more specific requirements for the fitness of neural IR models.

**Dilemma:** Current research on OOD is still based on a combination of existing experimental datasets.

**Background:** In real search engines, there are more specific requirements for the fitness of neural IR models.

**Dilemma:** Current research on OOD is still based on a combination of existing experimental datasets.

**Promising way:** Industrial practice

- Conduct experiments on real data from industrial scenarios, such as corpus increments over time
- Designing the appropriate OOD solutions for realistic and specific search engine scenarios

So much to do ...

- Developing new techniques for early detection and mitigation of adversarial attacks

So much to do ...

- Developing new techniques for early detection and mitigation of adversarial attacks
- Exploring synergies between different aspects of robustness, such as adversarial and OOD

So much to do ...

- Developing new techniques for early detection and mitigation of adversarial attacks
- Exploring synergies between different aspects of robustness, such as adversarial and OOD
- Enhancing model agility to quickly adapt to new data without extensive retraining

So much to do ...

- Developing new techniques for early detection and mitigation of adversarial attacks
- Exploring synergies between different aspects of robustness, such as adversarial and OOD
- Enhancing model agility to quickly adapt to new data without extensive retraining
- Resources and sharing

**There is still a long way to go . . .**

What do we talk about when we talk about IR robustness?

What do we talk about when we talk about IR robustness?

“Oh, you mean adversarial robustness? OOD robustness? data distribution? model architecture?”

What do we talk about when we talk about IR robustness?

“Oh, you mean adversarial robustness? OOD robustness? data distribution? model architecture?”

“Actually, I mean this deployed model will not fail next month.”

Ultimate goal for robust IR ...

**Built to withstand, designed to last!**

## **Q & A**

**Thank you for joining us today!**

**All materials are available at**

[https:](https://sigir2024-robust-information-retrieval.github.io/)

[//sigir2024-robust-information-retrieval.github.io/](https://sigir2024-robust-information-retrieval.github.io/)

## References

- L. Bonifacio, H. Abonizio, M. Fadaee, and R. Nogueira. Inpars: Data augmentation for information retrieval using large language models. *arXiv preprint arXiv:2202.05144*, 2022.
- N. Boucher, L. Pajola, I. Shumailov, R. Anderson, and M. Conti. Boosting big brother: Attacking search engines with encodings. *arXiv preprint arXiv:2304.14031*, 2023.
- Y. Cai, K. Bi, Y. Fan, J. Guo, W. Chen, and X. Cheng. L2r: Lifelong learning for first-stage retrieval with backward-compatible representations. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 183–192, 2023.
- D. Campos, C. Zhai, and A. Magnani. Noise-robust dense retrieval via contrastive alignment post training. *arXiv preprints arXiv:2304.03401*, 2023.
- J. Chen, R. Zhang, J. Guo, M. de Rijke, W. Chen, Y. Fan, and X. Cheng. Continual learning for generative retrieval over dynamic corpora. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 306–315, 2023a.
- X. Chen, B. He, K. Hui, L. Sun, and Y. Sun. Dealing with textual noise for robust and effective bert re-ranking. *Information Processing & Management*, 60(1):103135, 2023b.

- X. Chen, B. He, L. Sun, and Y. Sun. Defense of adversarial ranking attack in text retrieval: Benchmark and baseline via detection. *arXiv preprint arXiv:2307.16816*, 2023c.
- X. Chen, B. He, Z. Ye, L. Sun, and Y. Sun. Towards imperceptible document manipulations against neural ranking models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6648–6664, 2023d.
- C. L. Clarke, N. Craswell, and I. Soboroff. Overview of the trec 2009 web track. Technical report, Waterloo University, 2009.
- D. Cohen, B. Mitra, K. Hofmann, and W. B. Croft. Cross domain regularization for neural ranking models using adversarial learning. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1025–1028, 2018.
- N. Craswell, B. Mitra, E. Yilmaz, D. Campos, and E. M. Voorhees. Overview of the trec 2019 deep learning track. In *Text REtrieval Conference*, Mar 2020.
- N. Craswell, B. Mitra, E. Yilmaz, and D. Campos. Overview of the trec 2020 deep learning track. *Text REtrieval Conference, Text REtrieval Conference*, 2021a.

- N. Craswell, B. Mitra, E. Yilmaz, D. Campos, and J. Lin. Ms marco: Benchmarking ranking models in the large-data regime. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 1566–1576, 2021b.
- Z. Dai and J. Callan. Context-aware sentence/passage term importance estimation for first stage retrieval. *arXiv preprint arXiv:1910.10687*, 2019.
- J. Guo, Y. Fan, L. Pang, L. Yang, Q. Ai, H. Zamani, C. Wu, W. B. Croft, and X. Cheng. A deep look into neural ranking models for information retrieval. *Information Processing & Management*, 57(6):102067, 2020.
- Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *AIRWeb*, volume 5, pages 39–47. Citeseer, 2005.
- G. Izacard, M. Caron, L. Hosseini, S. Riedel, P. Bojanowski, A. Joulin, and E. Grave. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*, 2021.
- M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, 2017.

- P. Kasela, G. Pasi, R. Perego, and N. Tonellotto. Desire-me: Domain-enhanced supervised information retrieval using mixture-of-experts. In *European Conference on Information Retrieval*, pages 111–125. Springer, 2024.
- O. Kurland and M. Tennenholz. Competitive search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2838–2849, 2022.
- T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, et al. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.
- D. Lee, S.-w. Hwang, K. Lee, S. Choi, and S. Park. On complementarity objectives for hybrid retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13357–13368, 2023.
- Y. Li, N. Yang, L. Wang, F. Wei, and W. Li. Multiview identifiers enhanced generative retrieval. In *61st Annual Meeting of the Association for Computational Linguistics*, pages 6636–6648, 2023.
- J. Lin, M. Efron, G. Sherman, Y. Wang, and E. M. Voorhees. Overview of the trec-2013 microblog track. In *TREC*, volume 2013, page 21, 2013.

- J. Liu, Y. Kang, D. Tang, K. Song, C. Sun, X. Wang, W. Lu, and X. Liu. Order-disorder: Imitation adversarial attacks for black-box neural ranking models. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2025–2039, 2022.
- Y.-A. Liu, R. Zhang, J. Guo, M. de Rijke, W. Chen, Y. Fan, and X. Cheng. Black-box adversarial attacks against dense retrieval models: A multi-view contrastive learning method. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 1647–1656, 2023a.
- Y.-A. Liu, R. Zhang, J. Guo, M. de Rijke, W. Chen, Y. Fan, and X. Cheng. Topic-oriented adversarial attacks against black-box neural ranking models. In *SIGIR*, page 1700–1709, 2023b.
- Y.-A. Liu, R. Zhang, J. Guo, M. de Rijke, Y. Fan, and X. Cheng. Multi-granular adversarial attacks against black-box neural ranking models. In *SIGIR*, 2024a.
- Y.-A. Liu, R. Zhang, J. Guo, M. de Rijke, Y. Fan, and X. Cheng. Robust neural information retrieval: An adversarial and out-of-distribution perspective. *arXiv preprint arXiv:2407.06992*, 2024b.

- Y.-A. Liu, R. Zhang, M. Zhang, W. Chen, M. de Rijke, J. Guo, and X. Cheng. Perturbation-invariant adversarial training for neural ranking models: Improving the effectiveness-robustness trade-off. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024c.
- Q. Long, Y. Deng, L. Gan, W. Wang, and S. J. Pan. Backdoor attacks on dense passage retrievers for disseminating misinformation. *arXiv preprint arXiv:2402.13532*, 2024.
- S. Lupart and S. Clinchant. A study on fgsm adversarial training for neural retrieval. In *European Conference on Information Retrieval*, pages 484–492. Springer, 2023.
- X. Ma, J. Guo, R. Zhang, Y. Fan, Y. Li, and X. Cheng. B-prop: Bootstrapped pre-training with representative words prediction for ad-hoc retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1513–1522, 2021.
- S. MacAvaney, A. Yates, S. Feldman, D. Downey, A. Cohan, and N. Goharian. Simplified data wrangling with ir-datasets. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2429–2436, 2021.
- T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. Ms marco: A human generated machine reading comprehension dataset. In *CoCo@NIPS*, 2016.

- J. Ni, C. Qu, J. Lu, Z. Dai, G. H. Abrego, J. Ma, V. Zhao, Y. Luan, K. Hall, M.-W. Chang, et al. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, 2022.
- G. Penha, A. Câmara, and C. Hauff. Evaluating the robustness of retrieval pipelines with query variation generators. In *European Conference on Information Retrieval*, pages 397–412. Springer, 2022.
- N. Raifer, F. Raiber, M. Tennenholz, and O. Kurland. Information retrieval meets game theory: The ranking competition between documents' authors. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 465–474, 2017.
- G. Sidiropoulos and E. Kanoulas. Analysing the robustness of dual encoders for dense retrieval against misspellings. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2132–2136, 2022.
- C. Song, A. M. Rush, and V. Shmatikov. Adversarial semantic collisions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4198–4210, 2020.

- L. Su, J. Guo, Y. Fan, Y. Lan, and X. Cheng. Controlling risk of web question answering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 115–124, 2019.
- M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to rank answers on large online qa collections. In *proceedings of ACL-08: HLT*, pages 719–727, 2008.
- P. Tasawong, W. Ponwitayarat, P. Limkonchotiwat, C. Udomcharoenchaikit, E. Chuangsawanich, and S. Nutanong. Typo-robust representation learning for dense retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1106–1115, 2023.
- N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych. Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.
- D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, number 2019, 2019.

- K. Wang, N. Thakur, N. Reimers, and I. Gurevych. Gpl: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2345–2360, 2022.
- C. Wu, R. Zhang, J. Guo, W. Chen, Y. Fan, M. de Rijke, and X. Cheng. Certified robustness to word substitution ranking attack for neural ranking models. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2128–2137, 2022a.
- C. Wu, R. Zhang, J. Guo, Y. Fan, and X. Cheng. Are neural ranking models robust? *ACM Transactions on Information Systems*, 41(2):1–36, 2022b.
- C. Wu, R. Zhang, J. Guo, M. de Rijke, Y. Fan, and X. Cheng. Prada: Practical black-box adversarial attacks against neural ranking models. *ACM Transactions on Information Systems*, 41(4):1–27, 2023.
- Y. Yu, C. Xiong, S. Sun, C. Zhang, and A. Overwijk. Coco-dr: Combating distribution shifts in zero-shot dense retrieval with contrastive and distributionally robust learning. *arXiv preprint arXiv:2210.15212*, 2022.

- H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.
- Z. Zhong, Z. Huang, A. Wettig, and D. Chen. Poisoning retrieval corpora by injecting adversarial passages. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2023.
- S. Zhuang and G. Zuccon. Dealing with typos for bert-based passage retrieval and ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2836–2842, 2021.
- S. Zhuang and G. Zuccon. Characterbert and self-teaching for improving the robustness of dense retrievers on queries with typos. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1444–1454, 2022.