# Hallucination Mitigation

2023 여름방학세미나

NLP&AI 강명훈

# Theme of the seminar

- How can we mitigate hallucination in terms of factuality?
  - 다양한 Hallucination의 양상 중 non-factual error에 대한 mitigation은 어떻게 할 수 있을까?

- How can we mitigate hallucination efficiently?
  - Hallucination mitigation을 목표로 데이터셋을 만들지 않고도 문제를 해결하는 방법은 없을까?

- How can we get evidence effectively for hallucination mitigation?
  - Factual hallucination을 해결하기 위한 근거 문서 추출을 task마다 어떻게 효과적으로 진행할 수 있을까?

# Papers

## Zero-shot Faithful Factual Error Correction

**Kung-Hsiang Huang**[♠] **Hou Pong Chan**[♡] **Heng Ji**[♠]
[♠]Department of Computer Science, University of Illinois Urbana-Champaign
[♡]Faculty of Science and Technology, University of Macau
[♠]{khhuang3, hengji}@illinois.edu
[♡]hpchan@um.edu.mo

## SELFCHECKGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models

**Potsawee Manakul, Adian Liusie, Mark J. F. Gales**
Department of Engineering, University of Cambridge
pm574@cam.ac.uk, al826@cam.ac.uk, mjfg@eng.cam.ac.uk

## Precise Zero-Shot Dense Retrieval without Relevance Labels

**Luyu Gao**[*†] **Xueguang Ma**[*‡] **Jimmy Lin**[‡] **Jamie Callan**[†]
[†]Language Technologies Institute, Carnegie Mellon University
[‡]David R. Cheriton School of Computer Science, University of Waterloo
{luyug, callan}@cs.cmu.edu, {x93ma, jimmylin}@uwaterloo.ca

# Zero-shot Faithful Factual Error Correction

**Kung-Hsiang Huang**[♠] **Hou Pong Chan**[♡] **Heng Ji**[♠]
[♠]Department of Computer Science, University of Illinois Urbana-Champaign
[♡]Faculty of Science and Technology, University of Macau
[♠]{khhuang3, hengji}@illinois.edu
[♡]hpchan@um.edu.mo

ACL 2023 main accepted

Natural Language Processing & Artificial Intelligence

고려대학교 KOREA UNIVERSITY

# Problem Setting

Inconsistent correction

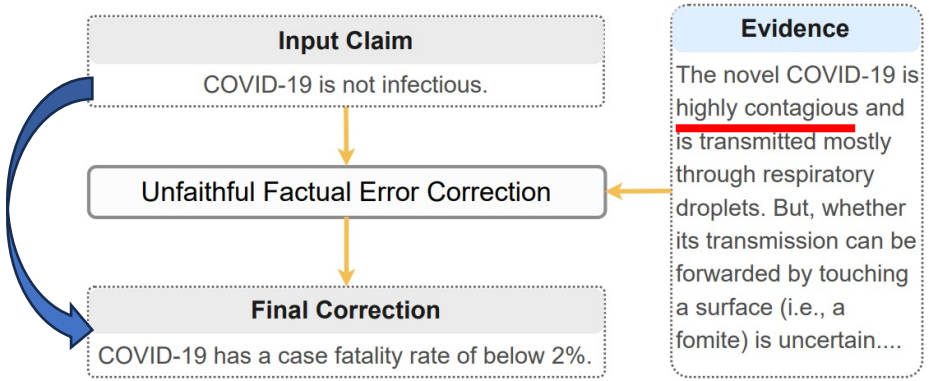



Figure 1: An example of a factual but unfaithful correction leading to misleading information. While it is technically true that the majority of people infected with COVID-19 will recover, there is no information in the evidence that supports the final correction. Additionally, when this statement is taken out of context, it could mislead people to believe that COVID-19 is not dangerous and that there is no need for precautions, which is false. A factual and faithful correction is "COVID-19 is highly contagious.".

Why Faithfulness matters in FEC?

1. Corrected claim과의 일관성 문제 무시

2. 단순 correction은 claim이 가진 context를 제외하게 됨

→Evidence를 무시하는 단편적인 AI 개입의 위험성 강조

# ZeroFEC

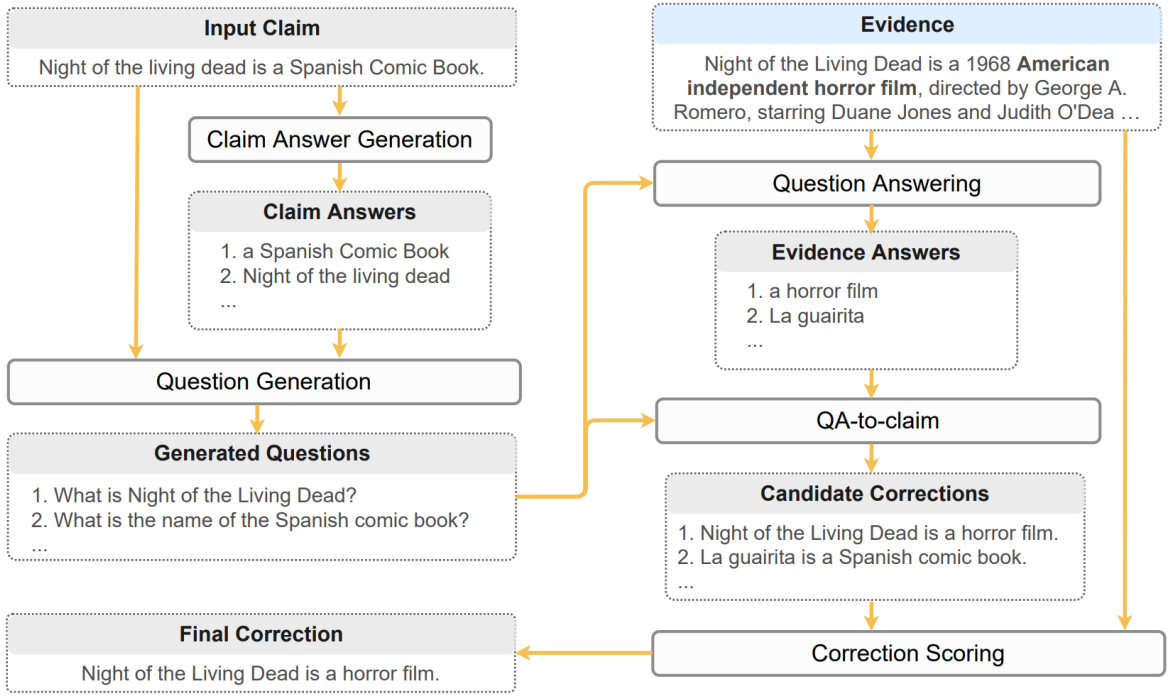



Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- 5 sub-task

1. Claim answer generation: answer가 될 만한 phrase 추출
2. Question generation: answer와 관련 있는 question 생성
3. Question Answering: Evidence안에서 Question에 대한 Answer 생성
4. QA-to-claim: 생성된 QA pair를 선언문으로 교정
5. Correction scoring: evidence와 corrected output간의 faithfulness 측정

# ZeroFEC

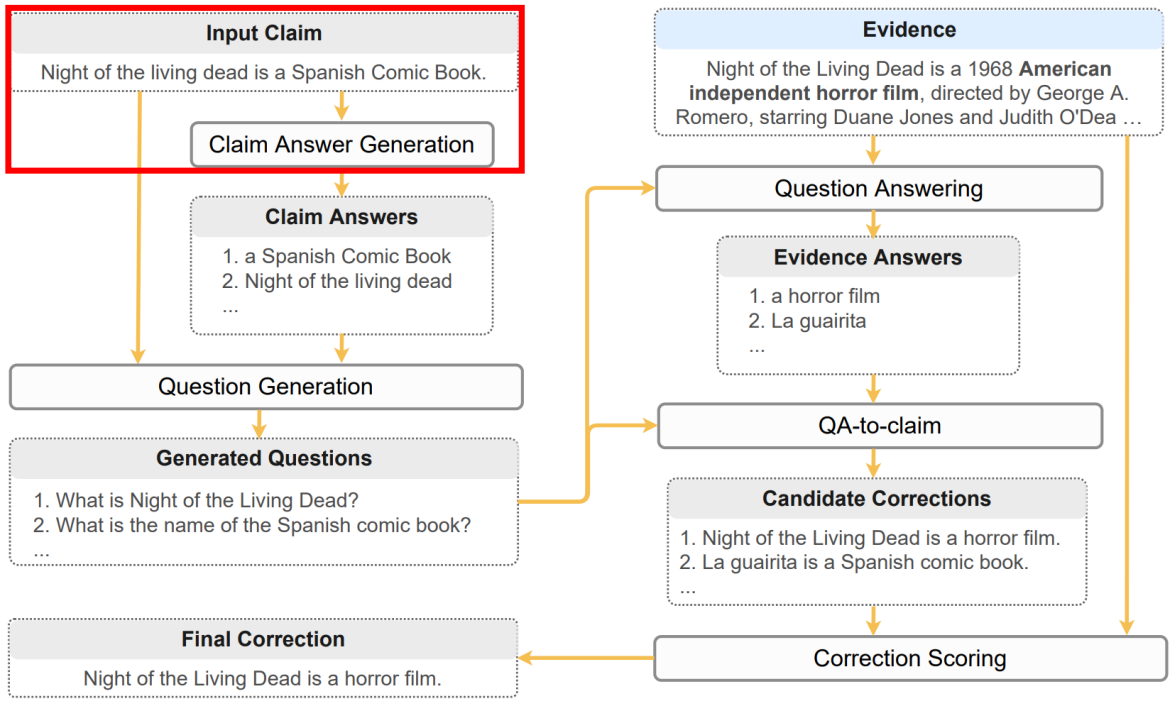



Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- Claim answer generation

Input: $C$

output: $A^C = \{A_1^C, A_2^C, ..., A_n^C\}$

- Spacy, Stanza 이용하여 noun, verb, adjective, adverb, noun phrase, verb phrase 추출
- not, never같은 negation도 추출

# ZeroFEC

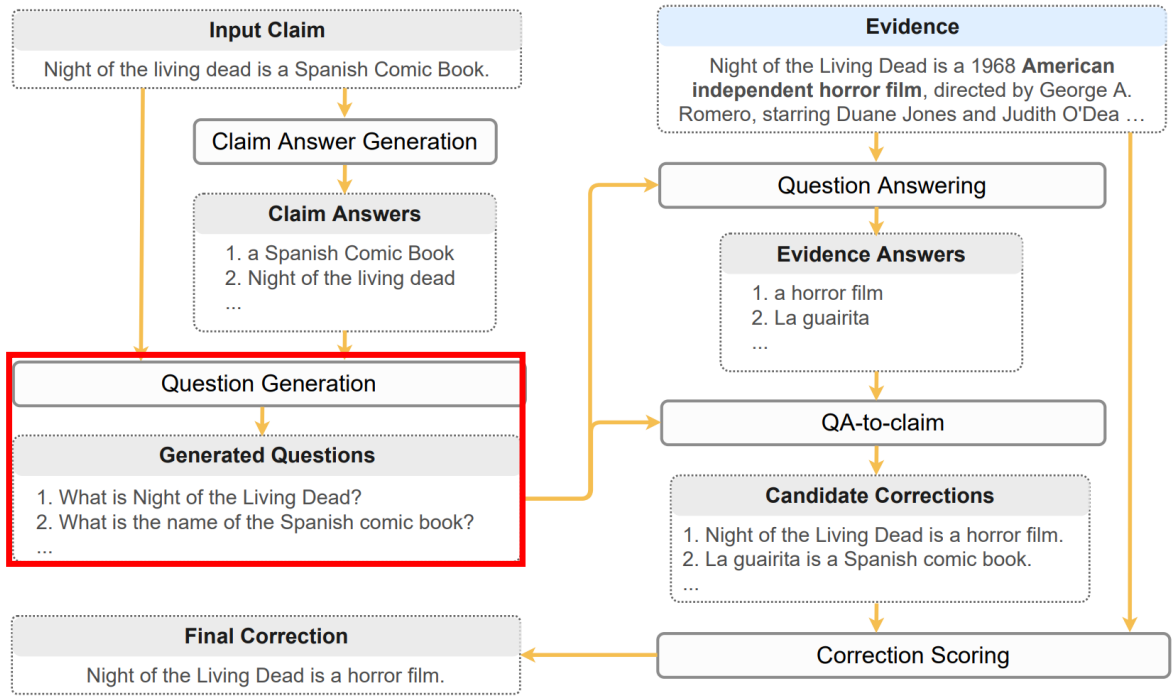



Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- Question Generation

Input: $C$,

output: $Q_i$

- MixQG 모델(g)을 활용하여 claim과 claim answers를 입력으로 하여 question 생성
- $Q_i = g(A_i^C, C)$

# ZeroFEC

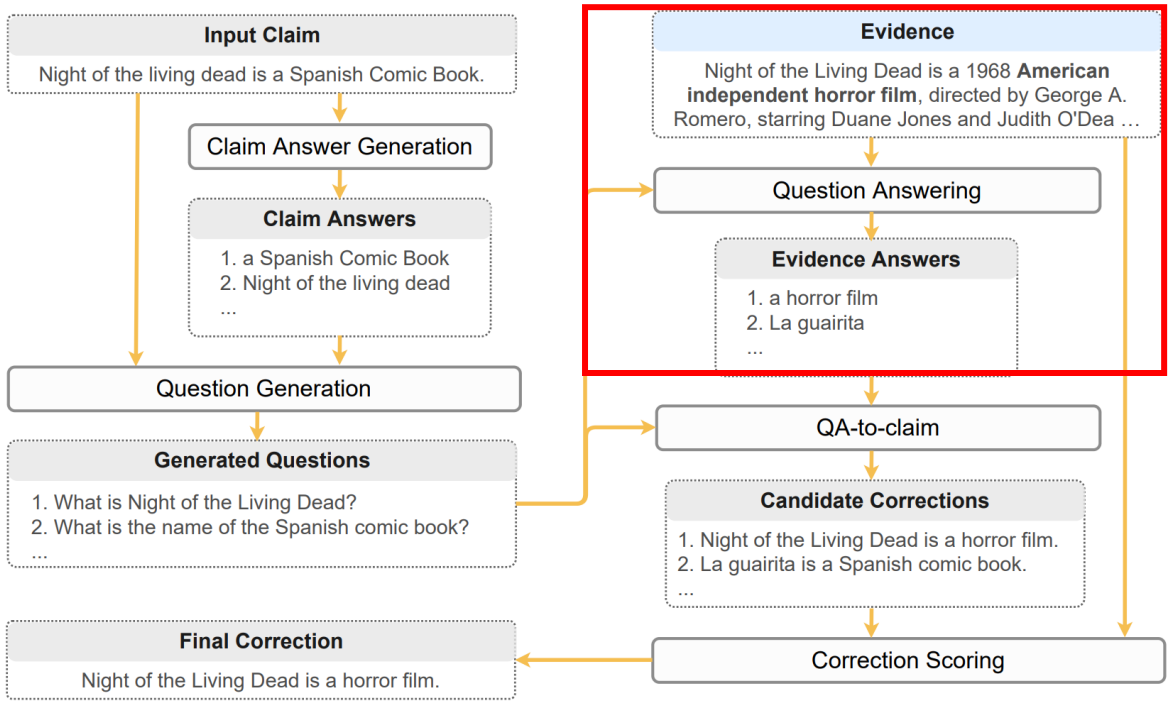



Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- Question Answering

Input: $Q_i$, $\varepsilon$

output: $A_i^{\varepsilon} = F(Q_i, \varepsilon)$

- UnifiedQA-v2 (T5기반)를 이용하여 QA 진행
- 20개의
- Abstractive QA 방식으로 수행

# ZeroFEC

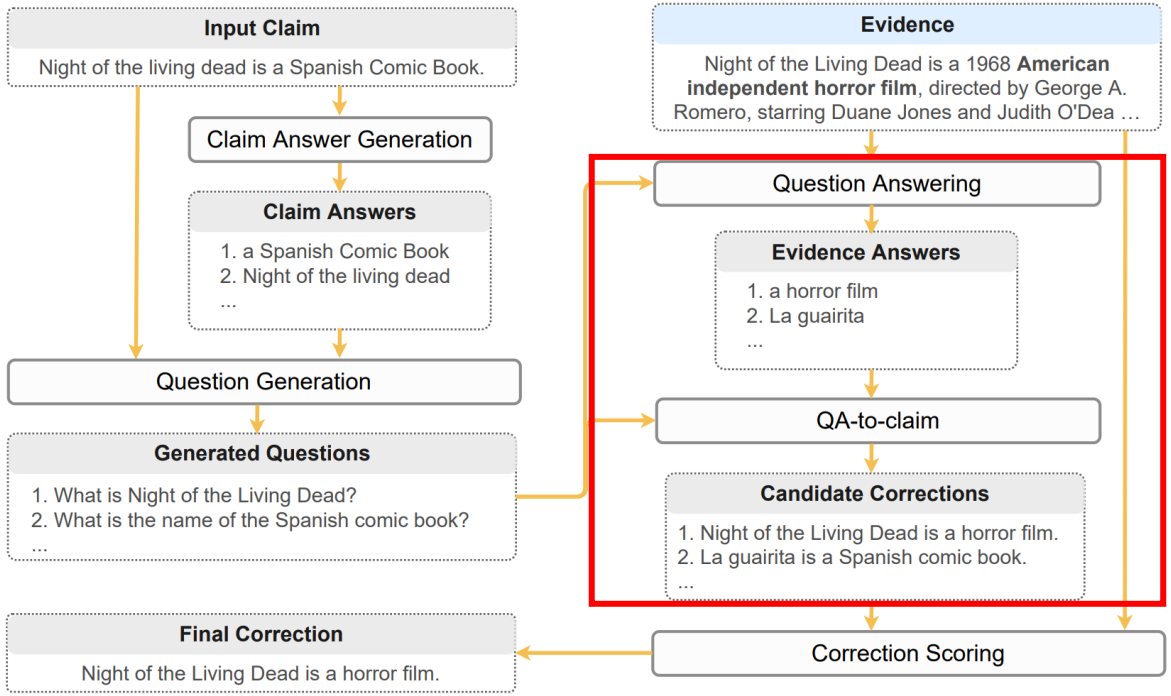


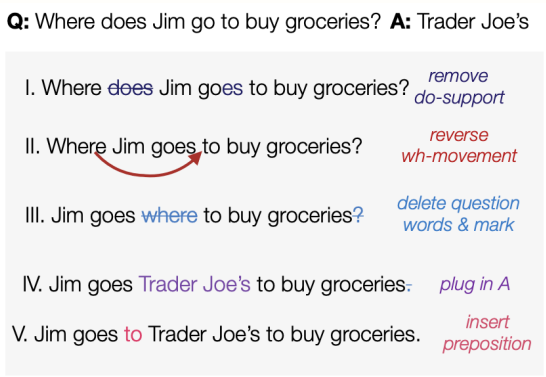

Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- QA-to-claim

Input: $Q_i, A_i^{\varepsilon}$

output: $S_i = M(Q_i, A_i^{\varepsilon})$

- UnifiedQA-v2 (T5기반)모델 $M$을 이용하여 QA 진행
- QA를 선언문으로 변경하는 task 데이터셋으로 학습한 모델 사용
- QA2D, BoolQ, SciTail



# ZeroFEC

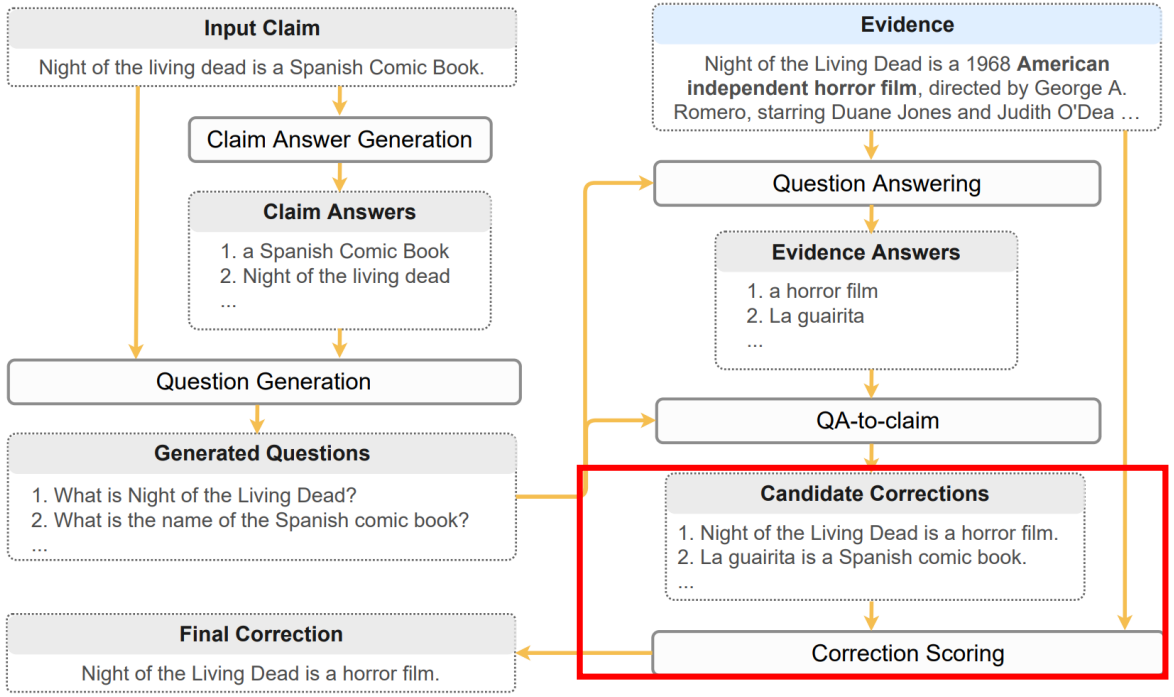



Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- Correction Scoring

Input: $S_i$, $\varepsilon$, $C$

output: $V(S_i), \acute{C}$

- 생성된 선언문 중 best corrected claim을 선정하기 위한 과정
- NLI score와 ROUGE 점수를 합산한 점수로 faithfulness 측정
- Faithfulness score: $V(S_i) = \mathrm{DocNLI}(S_i, \varepsilon) + \mathrm{ROUGE{-}1}(S_i, C)$
- Best correction: $\acute{C} = \mathrm{argmax}\, V(S_i)$

# ZeroFEC + Domain Adaptation

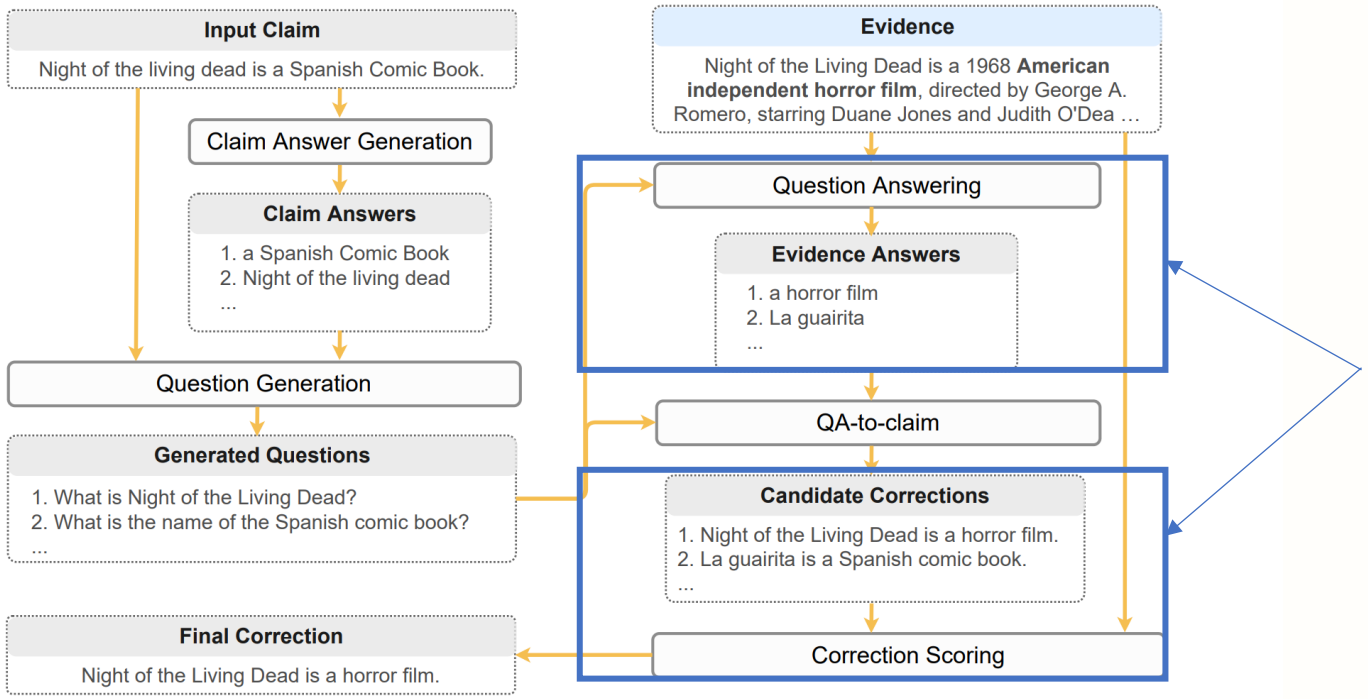



Figure 2: An overview of our framework. First, given an input claim, we generate the *claim answers* by enumerating all information units in the input claim. Second, conditioned on each extracted answer and the input claim, a question is generated. Third, each question is then fed to a question answering model to produce an *evidence answer* using the given evidence as context. Fourth, using a sequence-to-sequence approach, each *evidence answer* and the corresponding question are transformed into a statement, which serves as a *candidate correction*. Finally, the *final correction* is produced by scoring candidate corrections based on faithfulness.

- Biomedical domain adaptation

- 사전실험에서 Biomedical domain에서의 낮은 성능 발견
- QA와 Correction scoring 부분에서 biomedical dataset으로 추가 Fine-tuning 실시
- PubMedQA, BioASQ 데이터셋 사용
- 차후 실험에서 -DA로 Domain adaptation 효과 증명

# Experiment Setup

- Dataset Construction

## Raw dataset

- Fact Verification을 위한 데이터셋을 변형하여 사용
- FEVER: 정치 사회 도메인 fact verification
- SCiFact: 과학 도메인 fact verification

**Supports Example**

```
{
    "id": 62037,
    "label": "SUPPORTS",
    "claim": "Oliver Reed was a film actor.",
    "evidence": [
        [
            [<annotation_id>, <evidence_id>, "Oliver_Reed", 0]
        ],
        [
            [<annotation_id>, <evidence_id>, "Oliver_Reed", 3],
            [<annotation_id>, <evidence_id>, "Gladiator_-LRB-2000_film-RRB-", 0]
        ],
        [
            [<annotation_id>, <evidence_id>, "Oliver_Reed", 2],
            [<annotation_id>, <evidence_id>, "Castaway_-LRB-film-RRB-", 0]
        ],
        [
            [<annotation_id>, <evidence_id>, "Oliver_Reed", 1]
        ],
        [
            [<annotation_id>, <evidence_id>, "Oliver_Reed", 6]
        ]
    ]
}
```

FEVER 데이터셋 예시

## Dataset Construction Process

1. Raw dataset에서 진실로 판명된 claim만을 추출
2. 이 claim을 unfaithful claim 변경하는 작업 실시
   - Knowledge Base Informed Negations(KBIN) 방법으로 생성

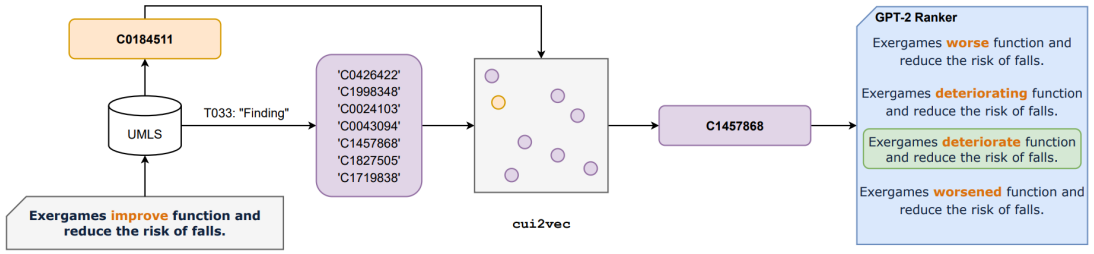



Figure 2: KBIN method. We start with NER and linking to UMLS using scispaCy. We then find the most similar concepts with the same type using `cui2vec`, replace the entity in the source sentence using the canonical name and aliases of similar entities, and rank them using GPT-2. Finally, from the highest ranked replacements, we select the claim which maximizes contradiction with the original claim using an external NLI model.

KBIN framework

# Experiment Setup

- Evaluation Metric

논문의 Problem setting을 따름: Correction이 Evidence로부터 비롯되었는가?

- BS (BARTScore): Correction과 Evidence의 semantic overlap 측정
- FC (FactCC): Correction이 Evidence를 entail하는지 측정
- QFE (QAFactEval): QAG를 이용하여 Correction이 Evidence로부터 비롯되었는지 측정
- SARI: 기존 Factual Error Correction task에서 사용된 lexical 기반 방법

| **Document** |
| --- |
| The Knicks beat the Rockets. The fans were excited. |
| **Summary** |
| The Knicks beat the Bucks. |

| **Entailment Matrix** [Contra, Neutral, Support] | **Selected Answer** |
| --- | --- |
| $\begin{bmatrix} 0.90 & 0.07 & 0.03 \\ 0.02 & 0.90 & 0.08 \end{bmatrix}$ | the Bucks |
| | **Generated Question** Who did the Knicks beat? |
| | **QA Output** the Rockets |
| **Max Support Score** 0.08 | **Answer Overlap Score** 0.20 |

QAFactEVal 방법 예시

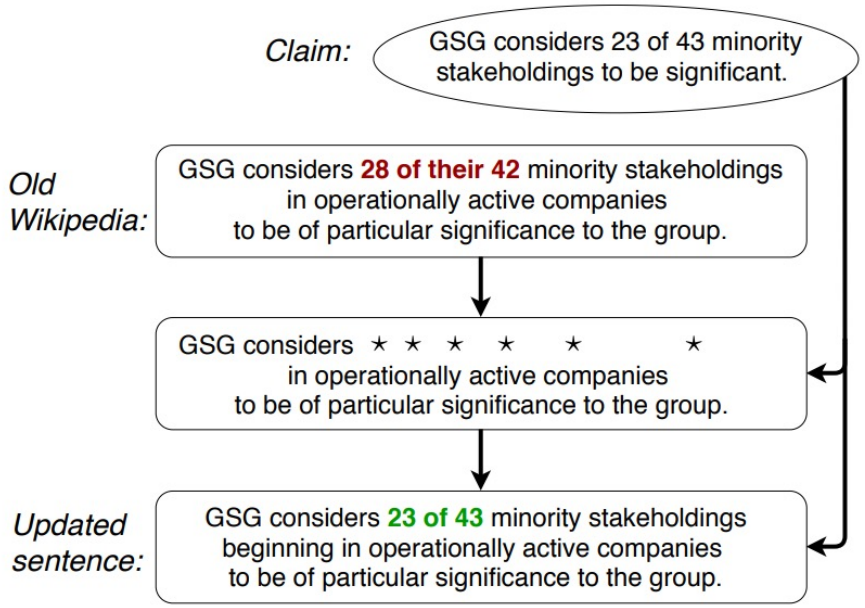
# Experiment Setup

- Model

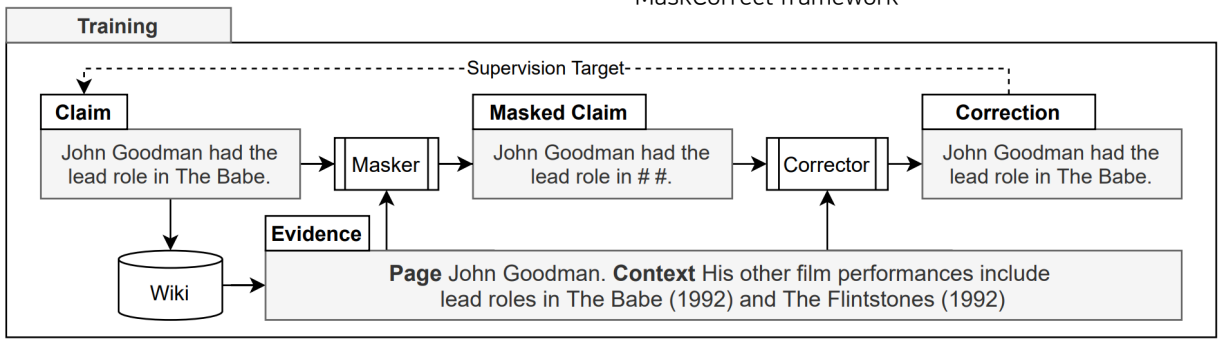
비교대상: Fully supervised Factual Error Correction

Model

- T5-Full: T5-base 기반 Claim + Evidence 투입해서 T5로 Factual Error Correction 생성
- MaskCorrect: Bi-LSTM 이용 Masking 방식으로 Factual Error Correction
- T5-Distant: T5-base 기반 Masking 방식으로 Factual Error Correction
- ReviseRef: entity swap 방식 Data augmentation이용 하여 학습
- CompEdit: entity insertion 방식 Data augmentation이용 하여 학습




MaskCorrect framework




T5-Distant framework

# Main result

- Quantitative Result

| Method | FEVER | | | | SciFact | | | |
|---|---|---|---|---|---|---|---|---|
| | SARI (%) | BS | QFE | FC (%) | SARI (%) | BS | QFE | FC (%) |
| *Fully-supervised* | | | | | | | | |
| T5-Full | 35.50 | -2.74 | 1.40 | 41.91 | 35.07 | -3.12 | 1.23 | 50.17 |
| *Distantly-supervised* | | | | | | | | |
| MaskCorrect | 25.66 | -4.48 | 0.67 | 20.12 | 15.21 | -4.31 | 0.54 | 34.92 |
| T5-Distant | 36.01 | -2.90 | 1.12 | 32.28 | 20.08 | -3.51 | 0.99 | 44.77 |
| *Zero-shot* | | | | | | | | |
| ReviseRef | 20.52 | -5.27 | 0.30 | 26.00 | 17.53 | -4.58 | 0.97 | **52.44** |
| CompEdit | 25.51 | -2.83 | 1.23 | 39.46 | 25.41 | -3.31 | 1.12 | 50.62 |
| ZeroFEC (Ours) | 39.16* | **-2.58*** | **2.06*** | **47.08*** | 29.67 | -3.22 | 1.12 | 47.84 |
| ZeroFEC-DA (Ours) | **40.65*** | -2.67* | 2.03* | 45.75* | **31.93** | **-3.21** | **1.30*** | 50.10 |

Table 1: Main results on the FEVER and SciFact datasets. BS denotes BARTScore, QFE denotes QAFactEval, and FC denotes FactCC. ZeroFEC-DA is our framework with the QA and entailment components further fine-tuned on biomedical QA datasets. Among distantly-supervised and zero-shot results, the best scores per metric are marked in **boldface**. Models achieving performance better than the fully-supervised model are marked in gray. Statistical significance over previous best methods computed with the paired bootstrap procedure (Berg-Kirkpatrick et al., 2012) are indicated with * ($p < .01$).

Zero-shot 제안 방법이 Full-shot을 모두 이김

- ZeroFEC가 FEVER의 경우 모둔 부분에서 Full-shot을 앞지르는 성능 + 그 차이가 모두 통계적으로 유의미함
→ 제안 방법의 우수성을 확연하게 보여줌

- 비록 SciFact 데이터셋에서 모든 부분의 성능을 앞지르지는 못했으나 비슷한 성능을 도출함

- 두 데이터셋 모두에서 Domain Adaptation(DA)의 효과를 볼 수 있었음

# Main result

- Qualitative Result

| | |
|---|---|
| **Example 1** | |
| **Input claim**: Clathrin stabilizes the spindle fiber apparatus during anaphase.<br>**Evidence**: ...but is shut down during mitosis, when clathrin concentrates at the spindle apparatus...<br>**Gold correction**: Clathrin stabilizes the spindle fiber apparatus during mitosis. | |
| **Claim answer**: anaphase<br>**Evidence answer**: mitosis<br>**DocNLI + ROUGE-1**: 0.0165 + 0.8235 | **Generated question**: Clathrin stabilizes the spindle fiber apparatus during what phase?<br>**Candidate correction**: Clathrin stabilizes the spindle fiber apparatus during mitosis phase.<br>**ZEROFEC's output**: Clathrin stabilizes the spindle apparatus during anaphase? |
| **Claim answer**: anaphase<br>**Evidence answer**: mitosis<br>**DocNLI + ROUGE-1**: 0.9999 + 0.8235 | **Generated question**: Clathrin stabilizes the spindle fiber apparatus during what phase?<br>**Candidate correction**: Clathrin stabilizes the spindle fiber apparatus during mitosis phase.<br>**ZEROFEC-DA's output**: Clathrin stabilizes the spindle fiber apparatus during mitosis phase. |
| **Example 2** | |
| **Input claim**: Fuller House (TV series) won't air on Netflix.<br>**Evidence**: Fuller House is an American family sitcom and sequel to the 1987-95 television series Fuller House, airing as a Netflix original series...<br>**Gold correction**: Fuller House (TV series) airs on Netflix. | |
| **Claim answer**: won't air on Netflix<br>**Evidence answer**: Yes<br>**DocNLI + ROUGE-1**: 0.7222 + 0.7143 | **Generated question**: Does Fuller House air on Netflix?<br>**Candidate correction**: Fuller House airs on Netflix.<br>**ZEROFEC's output**: Fuller House airs on Netflix. |
| **T5-DISTANT's output**: Fuller House ( TV series ) isn't airing on HBO. | |

Table 2: Example outputs from different approaches. The outputs from our framework are directly interpretable, as the generated questions and answers reflect which information units in the input claim are erroneous and which information in the evidence supports the final correction. We only show the generated answers and questions directly related to the gold correction. In the first example, ZEROFEC-DA corrects a mistake made by ZEROFEC thanks to domain adaptation. In the second example, ZEROFEC successfully produces a faithful factual error correction, whereas the output of T5-DISTANT, the distantly-supervised baseline, is factual yet unfaithful to the evidence.

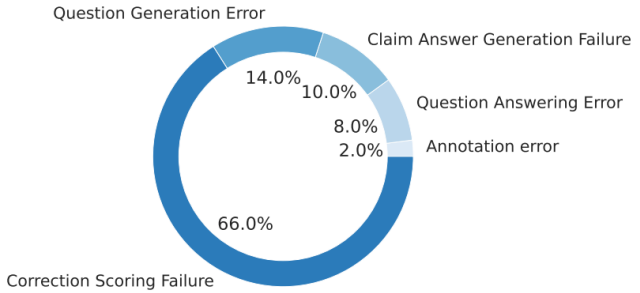



Figure 3: Distributions of errors.

## 제안 방법의 Faithfulness 증명

- Domain Adaptation을 통한 DocNLI 성능 향상 및 output 변경 과정 제시
- 특히 비교대상 T5-distant의 결과를 비교해볼 때 Hallucination 문제를 mitigation할 수 있다는 단서로도 볼 수 있을듯
- 다만 저자가 위에서 밝히듯이 scoring 부분에서 나타나는 error가 가장 많이 발견된 것으로 보아 best correction ranking의 어려움을 알 수 있음

# Main result

- Human Evaluation

| Method | FEVER | | | SciFact | | |
|---|---|---|---|---|---|---|
| | *Intel.* | *Fact.* | *Faith.* | *Intel.* | *Fact.* | *Faith.* |
| T5-Full | 0.983 | 0.516 | 0.509 | 0.972 | 0.683 | 0.610 |
| T5-Distant | 0.891 | 0.471 | 0.412 | 0.628 | 0.186 | 0.116 |
| ZeroFEC | **0.951** | 0.797 | 0.797 | 0.826 | 0.413 | 0.413 |
| ZeroFEC-DA | 0.893 | **0.835** | **0.835** | **0.953** | **0.628** | **0.628** |

Table 3: Human evaluation on the FEVER and SciFact datasets. *Intel.* denotes *Intelligibility*, *Fact.* denotes *Factuality*, and *Faith.* denotes *Faithfulness*.

Human eval을 통한 Faithfulness 우수성 증명

- 3명의 graduate school student에게 총 140개 case에 대한 평가 진행
- 평가자에게 gold correction, gold evidence, model output을 보여주고 평가를 진행 (Krippendorff alpha 68.85%)

Intel (Intelligibility): correction의 fluency 측정
Fact (Factuality): gold와 correction간의 fact 일치도
Faith (Faithfulness): correction과 evidence간의 consistency

- 제안 방법이 Faithfulness 부분에서 큰 성능격차를 보이며 제시해온 faithfulness 부분에서의 강점을 다시 한번 증명

# Main result

- Correlation btw Human & Automatic metrics

| Metric | FEVER | | | SCIFACT | | |
|---|---|---|---|---|---|---|
| | *Intel.* | *Fact.* | *Faith.* | *Intel.* | *Fact.* | *Faith.* |
| SARI | 0.017 | **0.370** | **0.383** | -0.026 | **0.379** | **0.412** |
| BARTSCORE | **0.137** | 0.071 | 0.104 | **0.104** | 0.118 | 0.119 |
| QAFACTEVAL | -0.045 | 0.360 | 0.379 | 0.084 | 0.234 | 0.272 |
| FACTCC | 0.053 | 0.203 | 0.225 | -0.119 | -0.073 | -0.076 |

Table 4: Correlation between automatic metrics and human judgments on the FEVER and SCIFACT datasets computed using Kendall's Tau.

추후 연구를 위한 insight 제시

- 본 연구에서 평가에 사용한 metric과 human과의 correlation 측정
  - SARI, BARTScore를 제외한 metric들이 Summarization task에서 비롯된 factual consistency score이므로
  - 이 metric이 factual error correction에서 적용되는 transferability 확인

- Lexical 기반의 SARI가 Fact, Faith 부분에서는 가장 높은 상관관계.
  즉 저자가 강조하는 faithfulness 관련 부분에서는 효용성이 있다고 말할 수 있음
- BART는 semantic similarity를 측정하는 metric인만큼 fluency를 일부 담보할 수 있는 metric임을 Intel correlation에서 확인할 수 있음
- 다만 QAFactEval, FactCC는 모든 부분에서 낮은 correlation을 보임
  - 즉 Factual error correction task에서 faithfulness를 고려한 고도화된 metric 개발의 필요성도 볼 수 있음

# SELFCHECKGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models

**Potsawee Manakul, Adian Liusie, Mark J. F. Gales**
Department of Engineering, University of Cambridge
pm574@cam.ac.uk, al826@cam.ac.uk, mjfg@eng.cam.ac.uk

Arxiv (EMNLP 2023 제출 예상)

# Problem Setting

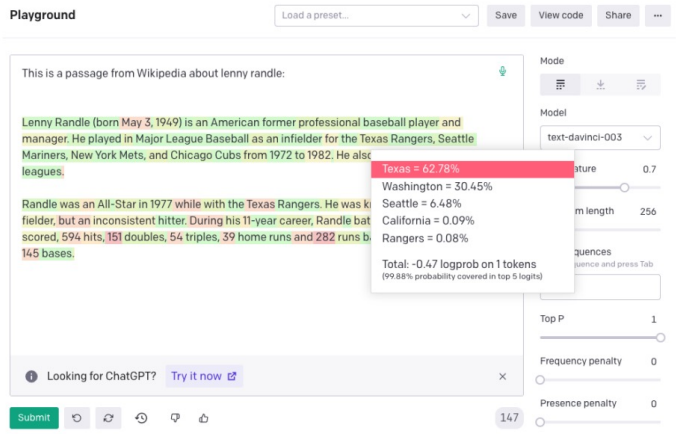



**Figure 2:** Example of OpenAI's GPT-3 interface with token probabilities displayed.


**Figure 3** Examples of P(IK) scores from a 52B model. Token sequences that ask harder questions have lower P(IK) scores on the last token. To evaluate P(IK) on a specific full sequence, we simply take the P(IK) score at the last token. Note that we only train P(IK) on final tokens (and not on partial questions).

LM이 생성한 sequence의 confidence를 확인하는 방법으로
*token probability*를 확인하는 방법이 존재
←적용하기 쉬운 방법이면서 이해하기 쉬운 점에서 강점

그러나 다음의 문제가 존재

1. 최근의 LLM(특히 ChatGPT, GPT4)는 token probability를 알 수 없음
2. 이런 경우 전용 모듈을 추가하거나 마지막 단계에 linear layer를 추가학습해야 하는 문제
3. 특정 task에 대한 평가를 위해선 데이터셋이 추가로 필요

→ 추가 데이터셋 및 모듈을 사용하지 않고(zero-resource)
모델의 출력 결과를 그대로 사용하는(black-box) Hallucination 정도 평가 방법 제안

# SelfCheckGPT

- Intuition

LLM이 생성한 N개의 output sampling 결과를 바탕으로 Hallucination 정도 측정
- Sample간의 일치도가 높다는 것은 모델은 사실에 대해서 이해를 하고 있다는 것
- Sample간의 일치도가 낮다면 모델은 그 사실에 대해서 이해를 하지 못한 상태로 generation한다는 가정

Sample간의 일치도를 파악하는 3가지 전략으로 LM의 생성 output에 대한 Hallucnation을 측정하는 SelfCheckGPT 제안
- zero-resource: 추가 학습데이터를 사용하지 않음
- Black-Box method: GPT-3 모델을 활용하여 token probability를 활용하지 않고 모델의 생성 output만을 활용하여 평가함

1. SelfCheckGPT with BERTscore
2. SelfCheckGPT with Question Answering
3. SelfCheckGPT with n-gram

# SelfCheckGPT with BERTscore

$$\mathcal{S}_{\text{BERT}}(i) = 1 - \frac{1}{N}\sum_{n=1}^{N} \max_{k} \left(\mathcal{B}(r_i, s_k^n)\right) \qquad (6)$$

reference 문서의 문장 r과 sample 문서의 문장 s와의 BERTscore를 기반으로 hallucination정도 측정

문장 단위로 점수 측정 (0점: valid, 1점: Hallucinated)

Notation

- $r_i$: Reference 문서 $R$의 i번째 문장
- $S_k^n$: n번째 sample $S^N$의 k번째 문장
- $B$: BERTScore 함수

# SelfCheckGPT with Question Answering

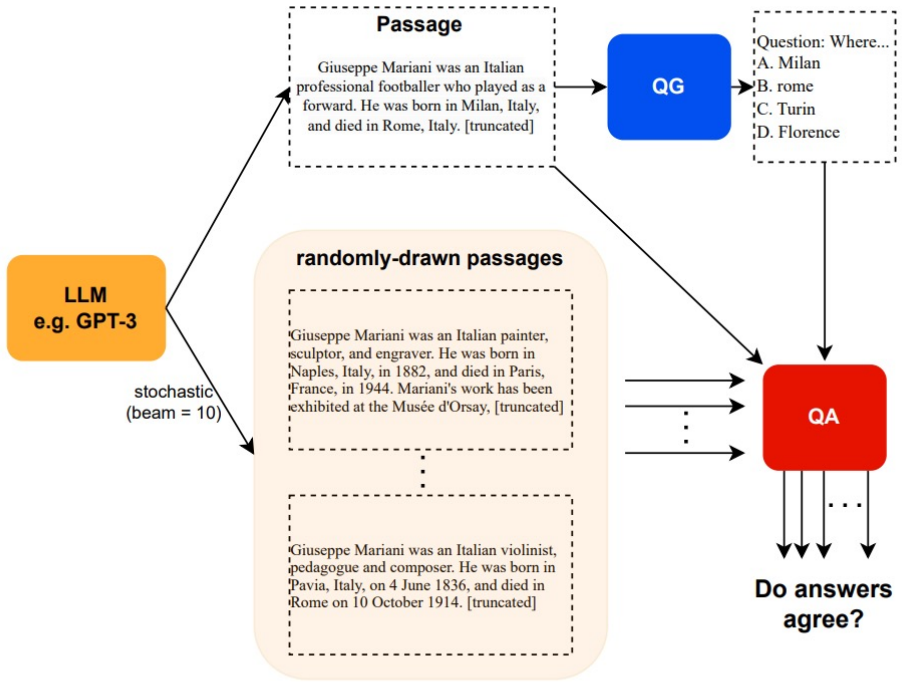



**Figure 1:** SelfCheckGPT with Question Answering.

$R$과 sampled output $S^N$의 MCQA 결과를 바탕으로 Hallucination 정도 평가

Multiple choice Question Answer Generation(MQAG) 활용

1. T5기반 QAG 시스템 활용하여 $R$에 관한 QA pair 생성
2. 나머지 distraction answerer를 생성하기 위한 T5기반 distract generation 수행
3. T5기반 QA시스템 활용하여 Answer 추출

$$a_R = \underset{k}{\operatorname{argmax}} \left[ P_{\mathrm{A}}(o_k | q, R, \mathbf{o}) \right] \qquad (9)$$

$$a_{S^n} = \underset{k}{\operatorname{argmax}} \left[ P_{\mathrm{A}}(o_k | q, S^n, \mathbf{o}) \right] \qquad (10)$$

# SelfCheckGPT with Question Answering

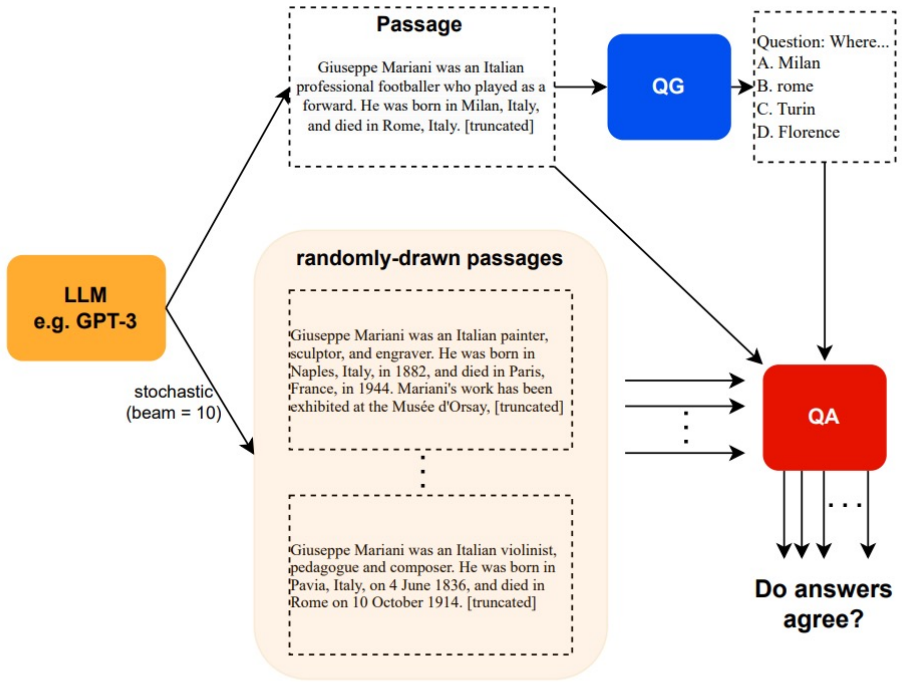



**Figure 1:** SelfCheckGPT with Question Answering.

$R$로부터 도출된 answer와 $S^N$ 추출 Answer의 일치, 불일치 count를 바탕으로 Hallucination 정도 측정

$N_M$ : Answer 일치 count

$N_n$ : Answer 불일치 count

$$\mathcal{S}_{\mathrm{QA}}(i,q) = \frac{\gamma_2^{N'_{\mathrm{n}}}}{\gamma_1^{N'_{\mathrm{m}}} + \gamma_2^{N'_{\mathrm{n}}}} \qquad (11)$$

# SelfCheckGPT with n-gram

$$\mathcal{S}^{\text{Avg}}_{\text{n-gram}}(i) = -\frac{1}{J}\sum_{j}\log \tilde{p}_{ij} \qquad (13)$$

$$\mathcal{S}^{\text{Max}}_{\text{n-gram}}(i) = \max_{j}\left(-\log \tilde{p}_{ij}\right) \qquad (14)$$

측정 대상 모델과 비슷한 token generation probability를 따르도록 학습된 n-gram model의 log-probability를 활용

문장 단위로 점수 측정 (0점: valid, 1점: Hallucinated)

Notation

- $\widetilde{p_{ij}}$: i번째 문장의 j번째 token의 생성확률

# Experimental Setup

- Dataset & Annotation

WikiBio 데이터셋에 대하여 GPT-3에게 다음의 prompt를 주고 synthetic article을 생성하도록 지시
→ "This is a Wikipedia passage about {concept}".

이렇게 GPT-3가 생성된 문단을 문장 단위 labeling을 진행
- 생성된 데이터셋에서 201개를 2명의 annotator에서 annotation진행
- 만약 두 annotator가 서로 다른 label을 제안할 경우 두 label중 더 낮은 라벨로 annotation 진행
  Cohen's K 3-label: 0.595, 2-label: 0.748

- **Major Inaccurate** (Non-Factual, **1**): The sentence is entirely hallucinated, i.e. the sentence is unrelated to the topic.
- **Minor Inaccurate** (Non-Factual, **0.5**): The sentence consists of some non-factual information, but the sentence is related to the topic.
- **Accurate** (Factual, **0**): The information presented in the sentence is accurate.

| #Passages | #Sentences | #Tokens/passage |
|---|---|---|
| 238 | 1908 | 184.7±36.9 |

**Table 1:** The statistics of **WikiBio GPT-3 dataset** where the number of tokens is based on the OpenAI GPT-2 tokenizer.

# Experimental Setup

- Baseline

Main baseline: GPT-3의 token probability, entropy를 활용한 Hallucination 정도 측정

Proxy LLM: 다른 LLM의 token probability를 근사할 수 있는 LLaMA 사용하여 Hallucination 정도 측정

SelfCheckGPT: 제안한 방법을 적용 GPT-3활용 Hallucination 정도 측정

- temp=1.0에 20개의 sample을 생성해서 평가 진행

# Main result

- Sentence-level detection task (Black-box based approach capacity)

| Method | Sentence-level (AUC-PR) | | | Passage-level (Corr.) | |
|---|---|---|---|---|---|
| | NonFact | NonFact* | Factual | Pearson | Spearman |
| Random | 72.96 | 29.72 | 27.04 | - | - |
| GPT-3's probabilities (*LLM, grey-box*) | | | | | |
| Avg($-\log p$) | 83.21 | 38.89 | 53.97 | 57.04 | 53.93 |
| Avg($\mathcal{H}$)† | 80.73 | 37.09 | 52.07 | 55.52 | 50.87 |
| Max($-\log p$) | 87.51 | 35.88 | 50.46 | 57.83 | 55.69 |
| Max($\mathcal{H}$)† | 85.75 | 32.43 | 50.27 | 52.48 | 49.55 |
| LLaMA-30B's probabilities (*Proxy LLM, black-box*) | | | | | |
| Avg($-\log p$) | 75.43 | 30.32 | 41.29 | 21.72 | 20.20 |
| Avg($\mathcal{H}$) | 80.80 | 39.01 | 42.97 | 33.80 | 39.49 |
| Max($-\log p$) | 74.01 | 27.14 | 31.08 | -22.83 | -22.71 |
| Max($\mathcal{H}$) | 80.92 | 37.32 | 37.90 | 35.57 | 38.94 |
| **SelfCheckGPT** (*black-box*) | | | | | |
| w/ BERTScore | 81.96 | 45.96 | 44.23 | 58.18 | 55.90 |
| w/ QA | 84.26 | 40.06 | 48.14 | 61.07 | 59.29 |
| w/ Unigram (max) | 85.63 | 41.04 | 58.47 | 64.71 | 64.91 |
| Combination | 87.33 | 44.37 | 61.83 | 69.05 | 67.77 |

**Table 3:** AUC-PR for sentence-level detection tasks. Passage-level ranking performances are measured by Pearson correlation coefficient and Spearman's rank correlation coefficient w.r.t. human judgements. The results of other proxy LLMs, in addition to LLaMA, can be found in the appendix. †GPT-3 API returns the top-5 tokens' probabilities, which are used to compute entropy.

제안한 SelfCheckGPT가 Grey-box, 다른 Black Box 방법 대비 우수한 detection 성능을 보임

NonFact*는 전체 passage의 hallucination정도가 높지 않고 일부 문장이 major inaccurate한 경우를 말함.
저자는 이것이 challenging한 task라고 설명

제안한 모든 방법을 조합한 Combination이 가장 성능이 좋음

# Main result

- Ablation study(zero-resource approach capacity)

| Method | Sent-lvl AUC-PR | | | Passage-lvl | |
|---|---|---|---|---|---|
| | NoFac | NoFac* | Fact | Pear. | Spear. |
| SelfCk-BERT | 81.96 | 45.96 | 44.23 | 58.18 | 55.90 |
| WikiBio+BERT | 81.32 | 40.62 | 49.15 | 58.71 | 55.80 |
| SelfCk-QA | 84.26 | 40.06 | 48.14 | 61.07 | 59.29 |
| WikiBio+QA | 84.18 | 45.40 | 52.03 | 57.26 | 53.62 |
| SelfCk-1gm | 85.63 | 41.04 | 58.47 | 64.71 | 64.91 |
| WikiBio+1gm | 80.43 | 31.47 | 40.53 | 28.67 | 26.70 |

**Table 4:** The performance when using SelfCheckGPT samples versus external stored knowledge.

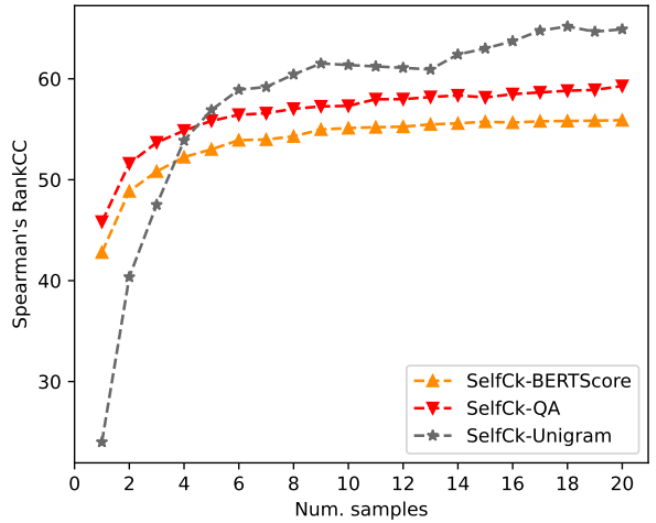



**Figure 7:** The performance of SelfCheckGPT methods on ranking passages (Spearman's) versus the number of samples.

# Precise Zero-Shot Dense Retrieval without Relevance Labels

**Luyu Gao**[*†] **Xueguang Ma**[*‡] **Jimmy Lin**[‡] **Jamie Callan**[†]

[†]Language Technologies Institute, Carnegie Mellon University
[‡]David R. Cheriton School of Computer Science, University of Waterloo

{luyug, callan}@cs.cmu.edu, {x93ma, jimmylin}@uwaterloo.ca

ACL 2023 main accepted

# Problem Setting

없으면 학습 불가

$$L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-) \qquad (2)$$
$$= -\log \frac{e^{\mathrm{sim}(q_i, p_i^+)}}{e^{\mathrm{sim}(q_i, p_i^+)} + \sum_{j=1}^{n} e^{\mathrm{sim}(q_i, p_{i,j}^-)}}.$$

새로운 Task, Domain에 대해서 DPR류의 retriever 학습의 어려운 점은 *Relevance label(positive sample)*이 반드시 필요하다는 것

이는 DPR모델이 새로운 task, domain에서 zero-shot setting으로 진행할 때의 성능 보장이 어려움을 의미하기도 함

RQ: 이 positive sample없이도 DPR류의 모델을 zero-shot setting에서 유의미한 성능을 도출할 수 있는 방법이 없을까?
→ Positive Sample을 Generation하면 되지!

# HyDE

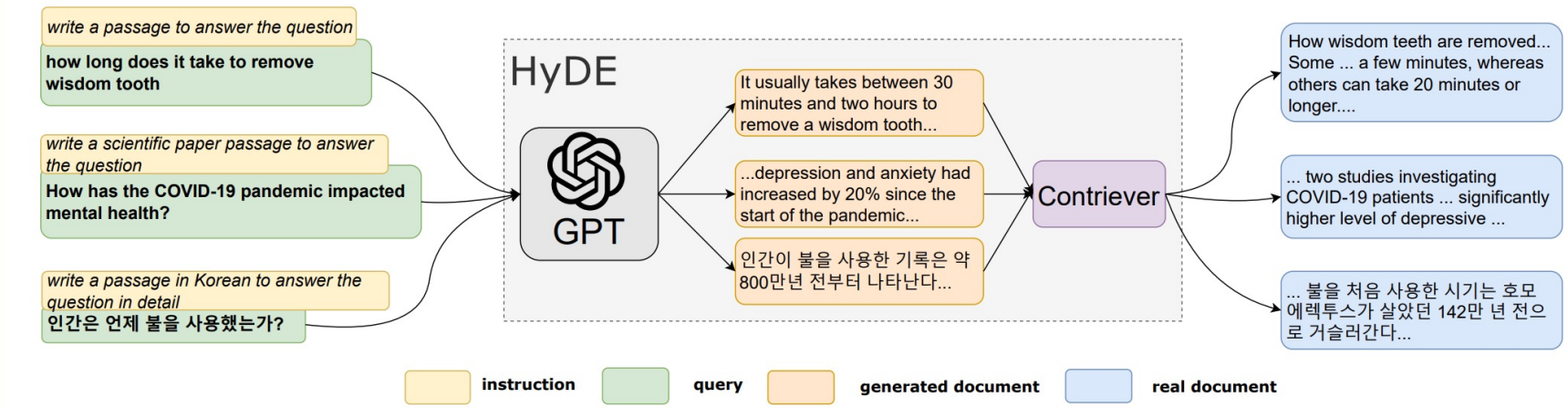



Figure 1: An illustration of the HyDE model. Document snippets are shown. HyDE serves all types of queries without changing the underlying InstructGPT and Contriever/mContriever models.

Instruction-tuned LLM으로 query relevant한 document generation을 통한 zero-shot dense retriever HyDE 제안

Instruction-tuned LLM에게 task별 적절한 instruction을 부여하여 query-relevant한 hypothetical document를 생성
해당 hypothetical document와 가장 relevance가 높은 passage를 retriever pool에서 추출
→ zero-shot setting 적용 가능

# HyDE

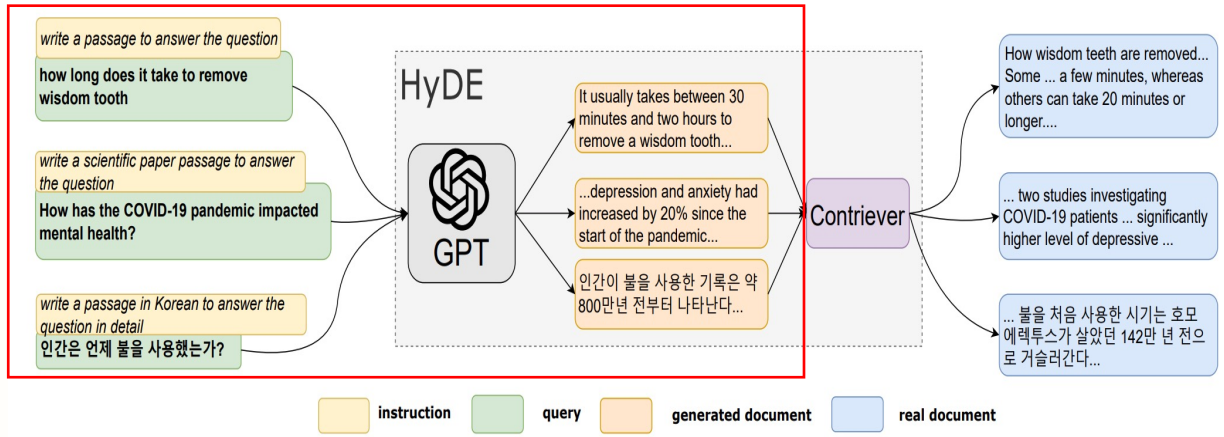



Figure 1: An illustration of the HyDE model. Document snippets are shown. HyDE serves all types of queries without changing the underlying InstructGPT and Contriever/mContriever models.

1. Hypothetical document generation
L개의 task와 관련된 prompt를 바탕으로 prompt + query 형식으로 query 관련 N개의 'hypothetical document' 생성

이러한 query relevant한 document generation 자체가 기존 dual-encoder 방식의 query-document similarity를 대체

document를 생성하므로 기존 dual-encoder의 document embedding space에 mapping하기 용이

다만 생성하는 Document의 factuality는 보장할 수 없음

# HyDE

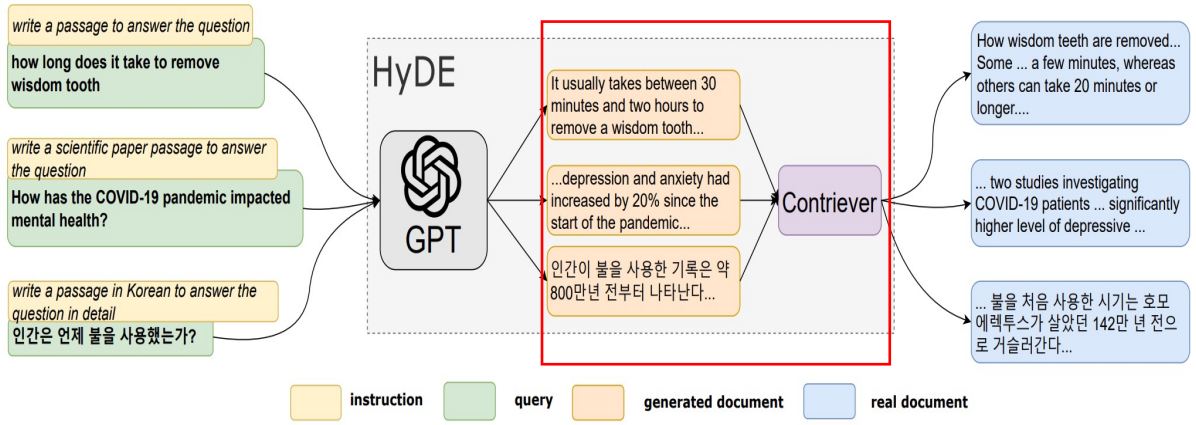



Figure 1: An illustration of the HyDE model. Document snippets are shown. HyDE serves all types of queries without changing the underlying InstructGPT and Contriever/mContriever models.

2. Query representation generation
Query + N document encoding + average 를 통해 query representation 생성
(query-doc similarity 대체를 위한 중간과정)

Contriever의 document encoder를 활용하여 아래와 같이 query representation 생성

$$\hat{\mathbf{v}}_{q_{ij}} = \frac{1}{N+1}[\sum_{k=1}^{N} f(\hat{d}_k) + f(q_{ij})] \qquad (8)$$

# HyDE

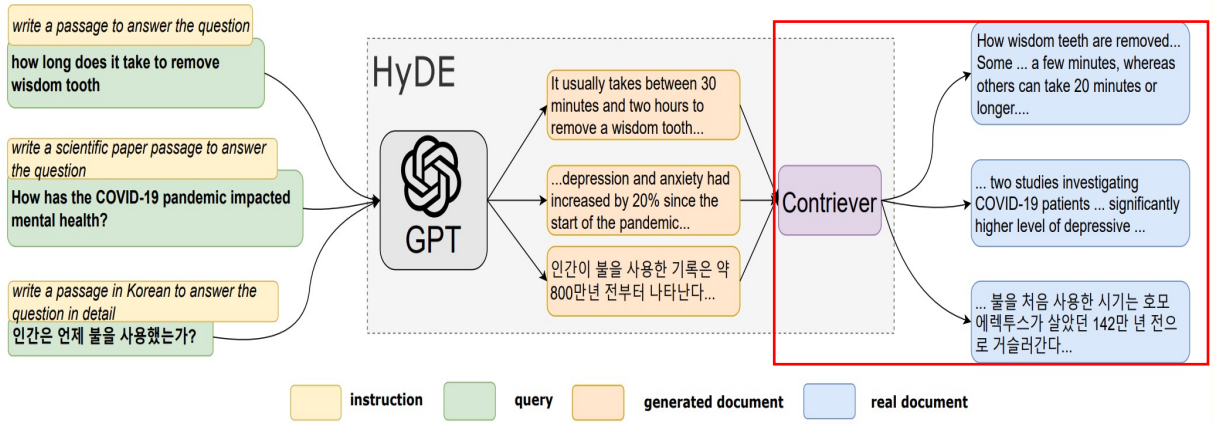



Figure 1: An illustration of the HyDE model. Document snippets are shown. HyDE serves all types of queries without changing the underlying InstructGPT and Contriever/mContriever models.

3. Retrieval

retreival pool에 있는 document representation 과 similarity scoring을 통해 top-1 retrieving

$$\mathbf{sim}(\mathbf{q}_{ij}, \mathbf{d}) = \langle \hat{\mathbf{v}}_{q_{ij}}, \mathbf{v}_d \rangle \quad \forall d \in D_i \qquad (9)$$

저자는 이 scoring과정에서 LLM이 생성한 문서의 factuality filtering이 진행되므로 hallucinated hypothetical document 문제를 일부 완화할 수 있다고 주장

# Experimental Setup

- Dataset & Baseline

Dataset

Zero-shot setting에서 좋은 성능을 보이는 HyDE의 능력을 평가하기 위하여 다양한 task에서 실험을 수행

1. Web search: MS MARCO dataset에서 추출된 TREC DL19, TREC DL20 데이터셋 사용
2. low-resource tasks: low-resource retrieval dataset들에 대한 실험 진행
3. non-English retrieval: 언어에 따른 zero-shot retrieval 실험도 진행

Baseline

주요 비교대상은 SOTA 모델인 Contriever

Unsupervised: Hyde 포함 zero-shot setting으로 진행된 retriever

Supervised: Relevance label로 fine-tuning이 진행된 retriever

# Main result

- Web search experiment

| | DL19 | | | DL20 | | |
|---|---|---|---|---|---|---|
| | mAP | nDCG@10 | Recall@1k | mAP | nDCG@10 | Recall@1k |
| *Unsupervised* | | | | | | |
| BM25 | 30.1 | 50.6 | 75.0 | 28.6 | 48.0 | 78.6 |
| Contriever | 24.0 | 44.5 | 74.6 | 24.0 | 42.1 | 75.4 |
| HyDE | **41.8** | **61.3** | **88.0** | **38.2** | **57.9** | **84.4** |
| *Supervised* | | | | | | |
| DPR | 36.5 | 62.2 | 76.9 | 41.8 | **65.3** | 81.4 |
| ANCE | 37.1 | **64.5** | 75.5 | 40.8 | 64.6 | 77.6 |
| Contriever-ft | 41.7 | 62.1 | 83.6 | **43.6** | 63.2 | **85.8** |

Table 1: Results for web search on DL19/20. Best performing w/o relevance and overall system(s) are marked **bold**. DPR, ANCE and Contriever-ft are in-domain *supervised* models that are fine-tuned on MS MARCO training data.

# Main result

- low-resource retrieval task experiment

| | Scifact | Arguana | Trec-Covid | FiQA | DBPedia | TREC-NEWS | Climate-Fever |
|---|---|---|---|---|---|---|---|
| | | | | nDCG@10 | | | |
| *Unsupervised* | | | | | | | |
| BM25 | 67.9 | 39.7 | **59.5** | 23.6 | 31.8 | 39.5 | 16.5 |
| Contriever | 64.9 | 37.9 | 27.3 | 24.5 | 29.2 | 34.8 | 15.5 |
| HyDE | **69.1** | **46.6** | 59.3 | **27.3** | **36.8** | **44.0** | **22.3** |
| *Supervised* | | | | | | | |
| DPR | 31.8 | 17.5 | 33.2 | 29.5 | 26.3 | 16.1 | 14.8 |
| ANCE | 50.7 | 41.5 | **65.4** | 30.0 | 28.1 | 38.2 | 19.8 |
| Contriever-ft | 67.7 | 44.6 | 59.6 | **32.9** | **41.3** | 42.8 | **23.7** |
| | | | | Recall@100 | | | |
| *Unsupervised* | | | | | | | |
| BM25 | 92.5 | 93.2 | **49.8** | 54.0 | 46.8 | 44.7 | 42.5 |
| Contriever | 92.6 | 90.1 | 17.2 | 56.2 | 45.3 | 42.3 | 44.1 |
| HyDE | **96.4** | **97.9** | 41.4 | **62.1** | **47.2** | **50.9** | **53.0** |
| *Supervised* | | | | | | | |
| DPR | 72.7 | 75.1 | 21.2 | 34.2 | 34.9 | 21.5 | 39.0 |
| ANCE | 81.6 | 93.7 | 45.7 | 58.1 | 31.9 | 39.8 | 44.5 |
| Contriever-ft | 94.7 | 97.7 | 40.7 | **65.6** | **54.1** | 49.2 | **57.4** |

Table 2: Results for a selection of low-resource tasks from BEIR. Best performing w/o relevance and overall system(s) are marked **bold**.

# Main result

- Multi-lingual retrieval task experiment

| | sw | ko | ja | bn |
|---|---|---|---|---|
| *Unsupervised* | | | | |
| BM25 | 38.9 | 28.5 | 21.2 | **41.8** |
| mContriever | 38.3 | 22.3 | 19.5 | 35.3 |
| HyDE | **41.7** | **30.6** | **30.7** | 41.3 |
| *Supervised* | | | | |
| mDPR | 7.3 | 21.9 | 18.1 | 25.8 |
| mBERT | 37.4 | 28.1 | 27.1 | 35.1 |
| XLM-R | 35.1 | 32.2 | 24.8 | 41.7 |
| mContriever-ft | **51.2** | **34.2** | **32.4** | **42.3** |

Table 3: Results on Mr.TyDi in terms of MRR@100. Best performing unsupervised and overall system(s) are marked **bold**.

# Main result

- Ablation study

| Model | DL19 | | DL20 | |
|---|---|---|---|---|
| | mAP | nDCG@10 | mAP | nDCG@10 |
| Contriever | 24.0 | 44.5 | 24.0 | 42.1 |
| HyDE | | | | |
| w/ Flan-T5 | 32.1 | 48.9 | 34.7 | 52.9 |
| w/ Cohere | 34.1 | 53.8 | 36.3 | 53.8 |
| w/ InstructGPT | **41.8** | **61.3** | **38.2** | **57.9** |

Table 4: nDCG@10 on TREC DL19/20 comparing the effects of changing different instruction LMs on *unsupervised* Contriever. Best performing results are marked **bold**.

| | Scifact | FiQA | DBPedia |
|---|---|---|---|
| Contriever | 64.9 | 24.5 | 29.2 |
| HyDE | | | |
| w/ InstructGPT | **69.1** | 27.3 | 36.8 |
| w/ GPT-3 | 65.9 | **27.9** | **40.5** |

Table 5: nDCG@10 comparing InstructGPT vs. 3-shot GPT-3 on BEIR. Best results are marked **bold**.

| Model | DL19 | | DL20 | |
|---|---|---|---|---|
| | mAP | nDCG@10 | mAP | nDCG@10 |
| Contriever-ft | 41.7 | 62.1 | 43.6 | 63.2 |
| + HyDE | 48.6 | 67.4 | 46.9 | 63.5 |
| GTR-XL | 46.7 | 69.6 | 46.9 | 70.7 |
| + HyDE | **50.6** | **71.9** | **51.5** | **70.8** |

Table 6: nDCG@10 on TREC DL19/20 comparing the effects of HyDE on *supervised* models. Best results are marked **bold**.

# Discussion

- 개선 방안
  - SelfCheckGPT같은 방법으로 먼저 output의 hallucination 정도를 파악한 뒤
  - Hallucination이 짙은 output인 경우 그 output의 factual error correction을 진행하는 방향으로 hallucination mitigation
  - Factual error correction과정에서 적절한 근거 문서를 추출할 수 있는 Zero-shot retriever 이용 및 개발

# Thank you