DSBA Open QA Study

Open-Domain Question Answering Paper Review

KAIST AI대학원

LK Lab

발표자 : 오한석

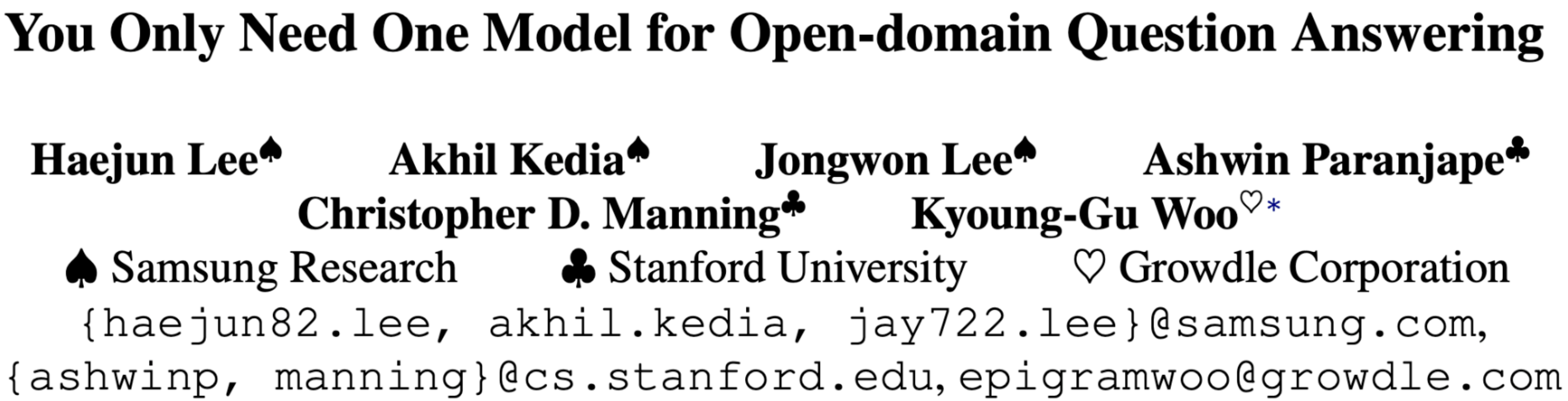
00 발표 목차

Contents

01 You Only Need One Model for Open-domain Question Answering

02 GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

03 Research Ideas



You Only Need One Model for Open-domain Question Answering

Haejun Lee[♠] Akhil Kedia[♠] Jongwon Lee[♠] Ashwin Paranjape[♣]
Christopher D. Manning[♣] Kyoung-Gu Woo[♡*]
♠ Samsung Research ♣ Stanford University ♡ Growdle Corporation
{haejun82.lee, akhil.kedia, jay722.lee}@samsung.com,
{ashwinp, manning}@cs.stanford.edu, epigramwoo@growdle.com

https://arxiv.org/abs/2112.07381

01

You Only Need One Model for Open-domain Question Answering

Motivation

Single language model YONO (You Only Need One Model)

- Target task : Efficient Model, Strongly coupled embedding
- Dataset : Natural Questions & TriviaQA open setting (Retrieval + QA)

- 최근 Neural language modeling 기반의 ODQA는 크게 두 가지 흐름의 접근법을 가능하게 함
  - 1) Model에 학습된 Internal knowledge를 접근 (Parametric Retrieval)
  - 2) Dense representation space상에서 매칭한 query, knowledge의 유사도를 기반으로 External knowledge를 접근 (Dense Retrieval)

- Parametric Retrieval의 단점
  - 1) large number of model parameters
  - 2) non-expandable knowledge without re-training
  - 3) hallucinations

- Dense Retrieval의 단점
  - 1) 대부분의 모델들이 reader, reranker, retriever에 대해서 분리된 parameter들을 갖고, weak coupling을 가짐

YONO 모델은 single model로서 retrieval, reranking, reading을 internal attention function으로 일반화시킴. 이를 통해서 representation을 공유하고 Model parameter들을 더 효율적으로 사용할 수 있음

제시한 모델을 end-to-end 방식으로 학습 가능하게함

01

You Only Need One Model for Open-domain Question Answering

Approach

First 12 encoder layers

Next 4 encoder layers

Remaining 8 encoder layers + 6 decoder layers

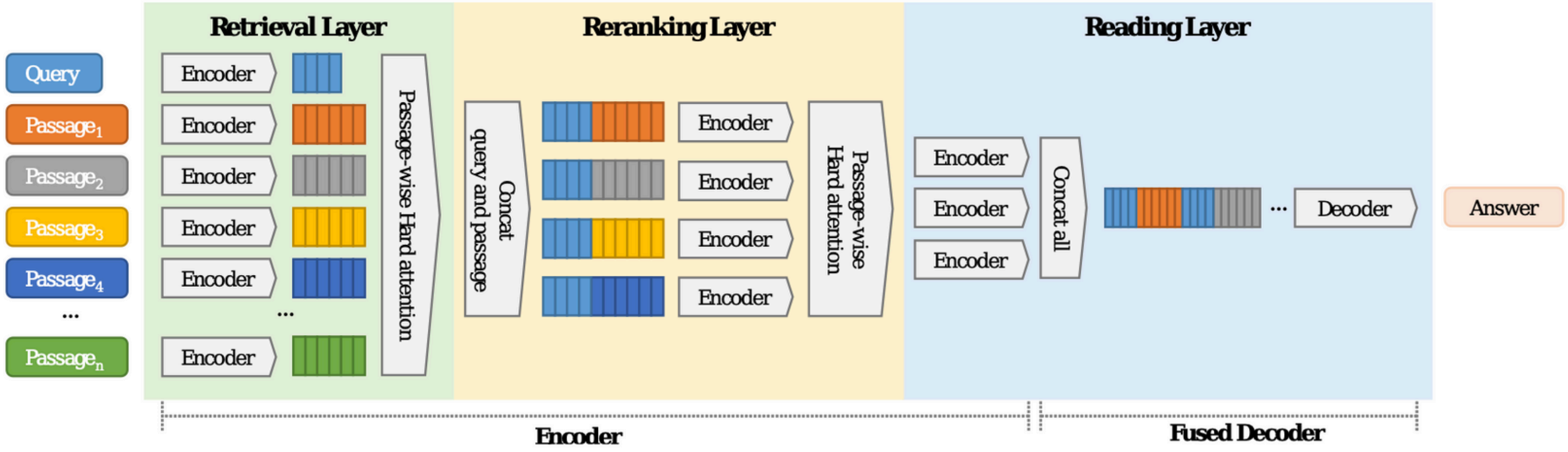


Figure 1: The overall architecture of our proposed model YONO.

Base : pre-trained T5-large
Total : 24 encoder, 6 decoder layers
(18 decoder layers discarded)
=> 220M (for retriever + reranker)
=> 220M (for reader)

retrieval layers

- query와 passage는 분리해서 encoding하여 passage representation을 미리 계산가능
- passage-wise Hard-attention via Dot product similarity (PHD)을 통해서 전체 Knowledge base에서 initial relevant passage를 retrieve

reranking layers

- retrieval layer에서 얻은 일부의 passage와 query representation을 함께 encoding해서 더 expressive coupled representation을 얻음
- passage-wise Hard-attention via Cross-attention (PHC)을 통해서 더욱 관련된 passage를 선택함

reading layers

- 최종적으로 얻어진 query-passage embedding pair는 모두 concat해서 Transformer encoder를 통해서 Deeper representations으로 변환된 뒤 generative decoder를 활용해서 answer를 생성

01 You Only Need One Model for Open-domain Question Answering

Approach

Final Loss:

$$\boxed{\mathcal{L} = \mathcal{L}_{gen.} + \mathcal{L}^{phd}_{approx.} + \mathcal{L}^{phc}_{approx.} \qquad (3)}$$

- Passage-wise Hard-attention via Dot-product similarity (PHD)
    - Passage와 가장 유사도(Dot product score)가 높은 top-N Passage를 retrieve
    - 모든 passage의 score를 훈련 과정에 구하기 어려움
    - random negative passage (P^N), retrieved passage (P^R)을 사용

$$Q_q = LayerNorm(q_0 W_q)$$
$$K_P = LayerNorm(P_0 W_p)$$
$$score_{phd}(q, P) = \frac{Q_q K_P^T}{\sqrt{d_k}} \qquad (1)$$

- Passage-wise Hard-attention via Cross-attention (PHC)
    - 앞서 Retriever level에서 추출된 P^R과 query representation을 concat해서 encode
    - cross-attention을 통해서 더 표현력있는 representation H를 배우게 됨

$$H = Transformer(q \oplus P^R)$$
$$score_{phc}(q, P^R) = LayerNorm(H_0) W_{qp} \qquad (2)$$

- Hard attention을 사용해서 non-differentiable 해졌기에 attention approximation loss를 사용
    - approximate the soft-attention using KL-divergence
    - soft attention score G_soft는 decoder의 attention으로부터 구함

$$\mathcal{L}_{approx.} = D_{KL}(G_{soft} \| S_{hard}) \qquad (5)$$

$$score_{soft}(P) = \left(\sum_{l=0}^{N_l} \sum_{h=0}^{N_h} \sum_{t_p=0}^{N_{p,t}} \frac{SG(att_{dec}(0, l, h, t_p))}{N_l N_h N_t} \mid p \in P\right) \qquad (6)$$

$$\mathcal{L}_{gen.} = -\log \prod_{t=1}^{T_A} p(a_t \mid a_{<t}, q, P^R) \qquad (4)$$

Generation loss
(autoregressive language modeling loss)

You Only Need One Model for Open-domain Question Answering

Approach

- Pre-training
  - pre-trained된 encoder-decoder 구조를 YONO에 맞게 학습시키기 위해서 pre-train을 실시함
  - end-to-end로 한 모델에서 학습되기에 pre-training을 위해서 ‘input-passage-output’ 구조의 데이터가 필요함

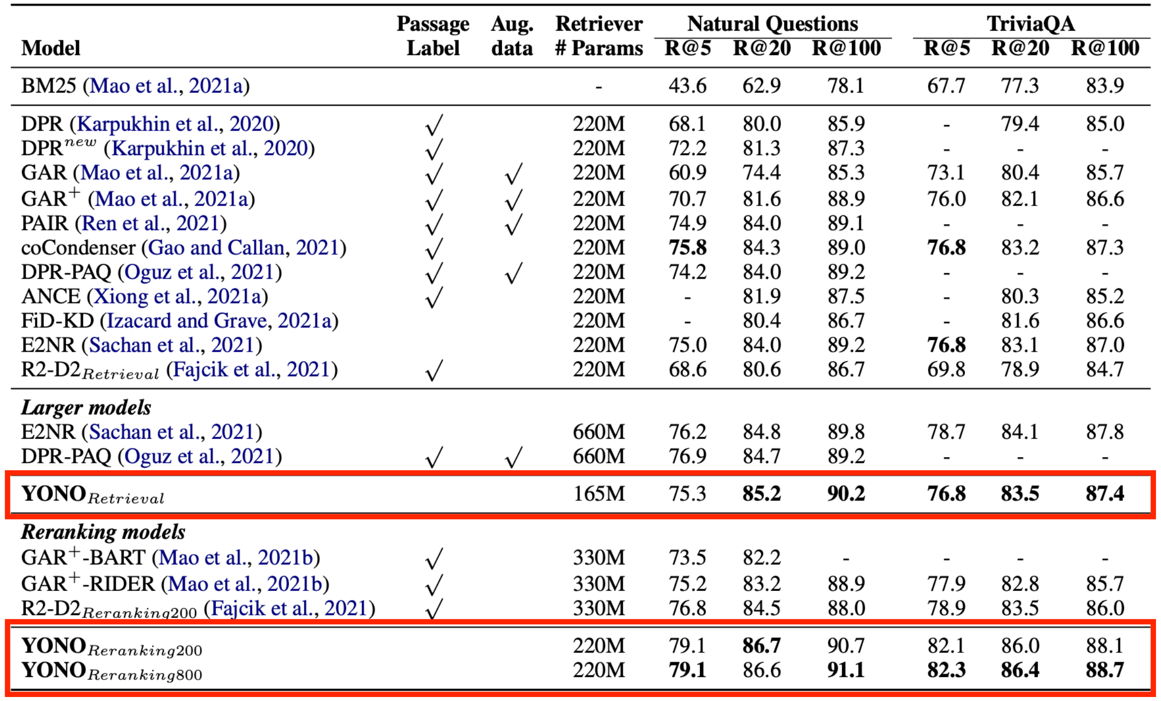
- Explicit Masked Salient Span (eMSS)
  - 문장에서 하나의 Named Entity를 뽑아서 해당 entity를 모두 masking처리함 [input & output]
  - Masking 처리된 entity를 포함하는 이웃한 passage를 explicit하게 ground truth passage로 사용함 [passage]
  - 해당 방식을 통해서 53M triples을 Wikipedia내에서 구축함

=> 기존 REALM에서 사용된 Salient Span Masking은 [question, answer]만 존재했다면, passage를 추가해서
제시된 single 모델의 포맷에 맞게 ‘query-passage-answer’ 형태를 구축함

01

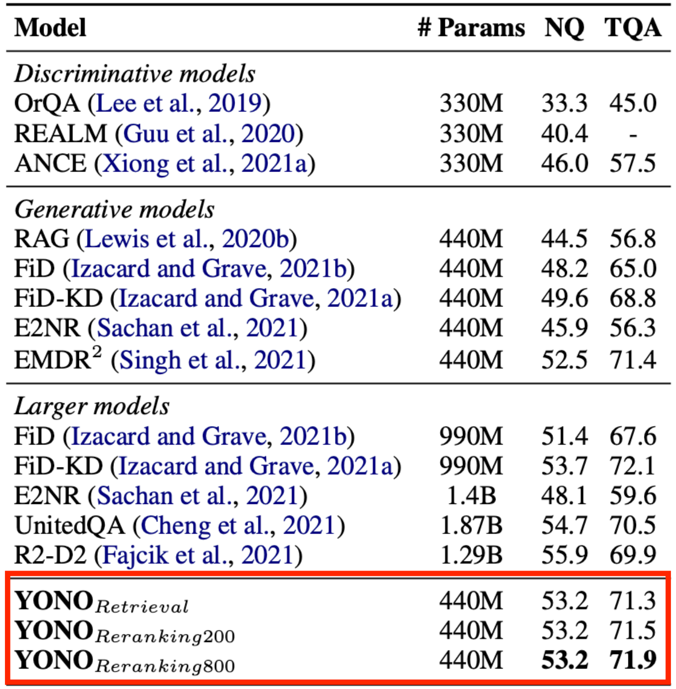
You Only Need One Model for Open-domain Question Answering

Experiments



Model	Passage Label	Aug. data	Retriever # Params	Natural Questions R@5	Natural Questions R@20	Natural Questions R@100	TriviaQA R@5	TriviaQA R@20	TriviaQA R@100
BM25 (Mao et al., 2021a)			-	43.6	62.9	78.1	67.7	77.3	83.9
DPR (Karpukhin et al., 2020)	√		220M	68.1	80.0	85.9	-	79.4	85.0
DPR^{new} (Karpukhin et al., 2020)	√		220M	72.2	81.3	87.3	-	-	-
GAR (Mao et al., 2021a)	√	√	220M	60.9	74.4	85.3	73.1	80.4	85.7
GAR^{+} (Mao et al., 2021a)	√	√	220M	70.7	81.6	88.9	76.0	82.1	86.6
PAIR (Ren et al., 2021)	√	√	220M	74.9	84.0	89.1	-	-	-
coCondenser (Gao and Callan, 2021)	√		220M	**75.8**	84.3	89.0	**76.8**	83.2	87.3
DPR-PAQ (Oguz et al., 2021)	√	√	220M	74.2	84.0	89.2	-	-	-
ANCE (Xiong et al., 2021a)	√		220M	-	81.9	87.5	-	80.3	85.2
FiD-KD (Izacard and Grave, 2021a)			220M	-	80.4	86.7	-	81.6	86.6
E2NR (Sachan et al., 2021)			220M	75.0	84.0	89.2	**76.8**	83.1	87.0
R2-D2$_{Retrieval}$ (Fajcik et al., 2021)	√		220M	68.6	80.6	86.7	69.8	78.9	84.7
Larger models									
E2NR (Sachan et al., 2021)			660M	76.2	84.8	89.8	78.7	84.1	87.8
DPR-PAQ (Oguz et al., 2021)	√	√	660M	76.9	84.7	89.2	-	-	-
YONO$_{Retrieval}$			165M	75.3	**85.2**	**90.2**	**76.8**	**83.5**	**87.4**
Reranking models									
GAR^{+}-BART (Mao et al., 2021b)	√		330M	73.5	82.2	-	-	-	-
GAR^{+}-RIDER (Mao et al., 2021b)	√		330M	75.2	83.2	88.9	77.9	82.8	85.7
R2-D2$_{Reranking200}$ (Fajcik et al., 2021)	√		330M	76.8	84.5	88.0	78.9	83.5	86.0
YONO$_{Reranking200}$			220M	79.1	**86.7**	90.7	82.1	86.0	88.1
YONO$_{Reranking800}$			220M	**79.1**	86.6	**91.1**	**82.3**	**86.4**	**88.7**

Table 1: Recall@N results on Natural Questions and TriviaQA test sets. The best retrieval and reranking scores except larger models are indicated in bold. Reranking200/800 refer to reranking the 200/800 retrieved passages.



Model	# Params	NQ	TQA
Discriminative models			
OrQA (Lee et al., 2019)	330M	33.3	45.0
REALM (Guu et al., 2020)	330M	40.4	-
ANCE (Xiong et al., 2021a)	330M	46.0	57.5
Generative models			
RAG (Lewis et al., 2020b)	440M	44.5	56.8
FiD (Izacard and Grave, 2021b)	440M	48.2	65.0
FiD-KD (Izacard and Grave, 2021a)	440M	49.6	68.8
E2NR (Sachan et al., 2021)	440M	45.9	56.3
EMDR^{2} (Singh et al., 2021)	440M	52.5	71.4
Larger models			
FiD (Izacard and Grave, 2021b)	990M	51.4	67.6
FiD-KD (Izacard and Grave, 2021a)	990M	53.7	72.1
E2NR (Sachan et al., 2021)	1.4B	48.1	59.6
UnitedQA (Cheng et al., 2021)	1.87B	54.7	70.5
R2-D2 (Fajcik et al., 2021)	1.29B	55.9	69.9
YONO$_{Retrieval}$	440M	53.2	71.3
YONO$_{Reranking200}$	440M	53.2	71.5
YONO$_{Reranking800}$	440M	**53.2**	**71.9**

Table 2: End-to-end Open QA Exact-Match results on Natural Questions and TriviaQA test sets. Our model uses top 100 retrieved or reranked passages to generate answers. The best EM scores except larger models are indicated in bold.

Retriever results

- Retrieval only로만 비슷한 크기의 모델들보다 뛰어난 성능을 보이고, 훨씬 적은 크기로 Larger models 와 comparable한 결과를 보임
- 더 많은 passage를 reranking에 사용할 경우 성능이 좋은 경향

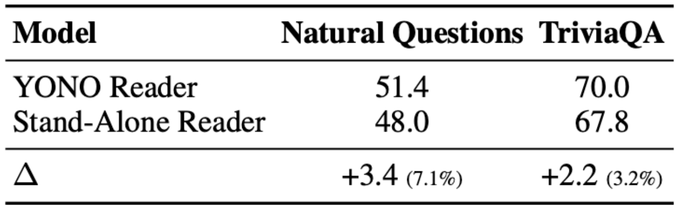
End-to-end QA results

- 같은 크기의 모델 사이즈에서 가장 뛰어난 end-to-end 성능을 보임

01

You Only Need One Model for Open-domain Question Answering

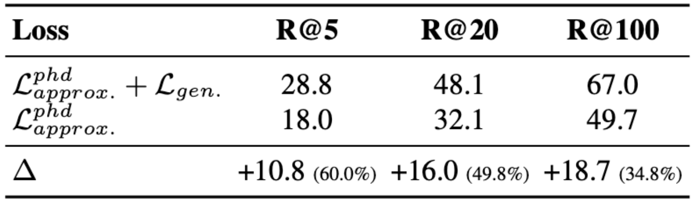
Experiments



Model	Natural Questions	TriviaQA
YONO Reader	51.4	70.0
Stand-Alone Reader	48.0	67.8
Δ	+3.4 (7.1%)	+2.2 (3.2%)

Table 3: Effect of sharing of retrieval and reranking representations on exact match scores of reader models that use 220M parameters on NQ and TQA development sets.

- retriever, reranker layer와 representations을 공유한 경우 Reader 성능이 더 좋음



Loss	R@5	R@20	R@100
$\mathcal{L}^{phd}_{approx.} + \mathcal{L}_{gen.}$	28.8	48.1	67.0
$\mathcal{L}^{phd}_{approx.}$	18.0	32.1	49.7
Δ	+10.8 (60.0%)	+16.0 (49.8%)	+18.7 (34.8%)

Table 4: Effect of generation loss on zero shot retrieval performance after the first iteration of pre-training on Natural Questions development set.

- generation loss를 반영했을 때 retrieval 성능이 큰 폭으로 향상되고, retrieved된 passage 수가 적을 때 상대적인 향상 폭은 훨씬 큼

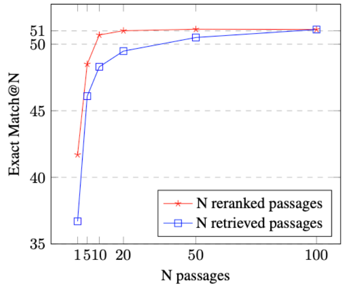


Figure 2: Exact Match scores for given N retrieved or reranked passages on NQ development set. Rerank EM scores are from reranking only 100 retrieved passages.

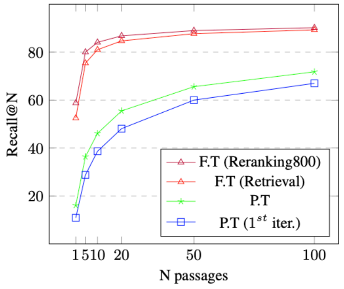


Figure 3: Recall@N at each training stage on NQ development set. P.T denotes pre-training, F.T denotes fine-tuning.

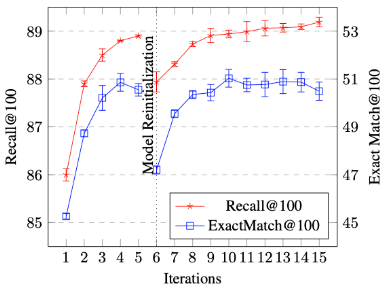


Figure 4: Average Recall and Exact Match scores at each fine-tuning iteration with error bars from 3 runs on NQ development set. The model is once reinitialized at the 6^{th} iteration.

- retrieve N의 수가 적을 때 reranking & retriever only 차이가 가장 크고 N이 늘어나며 차이가 줄음
- 먼저 Pre-training 후 Fine-tuning을 실시할 때 성능이 큰 폭으로 개선됨
- Fine-tuning시 over fitting을 방지하고자 reinitialization을 한 것의 효과

01 You Only Need One Model for Open-domain Question Answering

Discussion Points

- **End-to-End 방식으로 학습하는 방식이 더 효과적인가?**
  - 본 연구에서는 reader의 generation loss를 반영했을 때 retriever의 성능도 더 향상되는 경향성을 보임
  - DPR에서는 Joint training과 pipeline training을 비교했을 때 joint training 방식이 성능 개선으로 이뤄지지 않았음
  - Pipeline training은 Dense representation에 대한 index를 한 번만 형성하면 되기에 더 효율적이라는 관점도 존재함

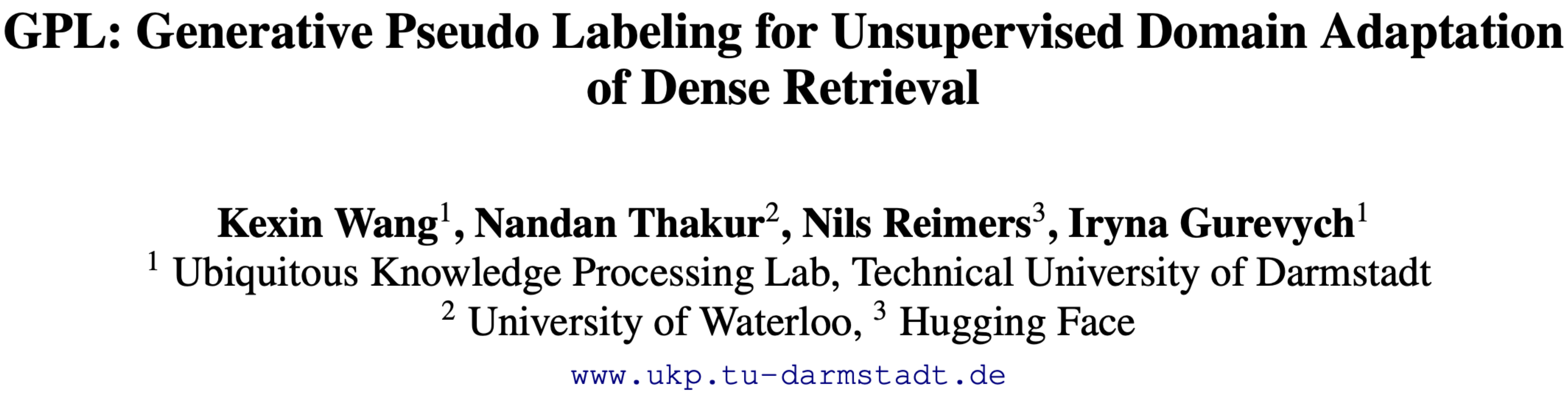
- **해당 논문에서는 제한된 모델 크기에서 효율성을 강조함. 만약 같은 구조의 모델을 더 큰 크기로 키웠을 때는 성능이 향상이 될까? 그렇지 않다면 왜 그럴까?**

01

You Only Need One Model for Open-domain Question Answering

Related Papers to Read

- Izacard, Gautier, and Édouard Grave. "Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering." Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. 2021. [link]



GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Kexin Wang[1], **Nandan Thakur**[2], **Nils Reimers**[3], **Iryna Gurevych**[1]
[1] Ubiquitous Knowledge Processing Lab, Technical University of Darmstadt
[2] University of Waterloo, [3] Hugging Face
www.ukp.tu-darmstadt.de

https://arxiv.org/pdf/2112.07577.pdf

02

GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Motivation

Generative Pseudo Labeling (GPL)

- Target task: Model generalization across diverse set of domains and task types
- Dataset : domain-specific retrieval dataset from BeIR benchmark (Retriever)

기존 Dense retrieval 방법론의 단점

- 좋은 성능을 얻기 위해서는 큰 규모의 훈련 데이터에 의존적임
- 대부분의 Specific Retriever Domain에서는 많은 양의 데이터를 얻기 힘듦
- Domain shift가 일어날 때 성능이 현저히 떨어지는 문제가 존재함

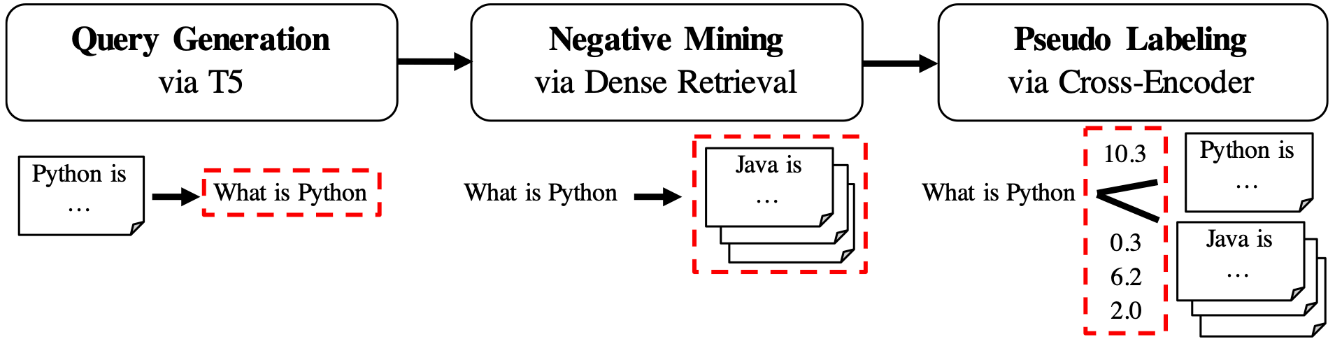
기존의 Domain Adaptation & Pseudo Labeling 방법의 한계 (e.g. QGen)

- Retriever task에서는 pre-training을 이어나갈 in-domain의 labeled data를 얻기 어려움
- Pseudo Labeling을 통해서 synthetic하게 형성된 query가 항상 좋은 품질이 아님 & pseudo-labels이 이를 탐지하기 어려움
- Pseudo Labeling을 활용해서 얻은 hard negative case를 훈련에 사용하기 어려움 (실제 정답 passage와 그럴 듯한 passage를 잘 구분하지 못함)
- Domain adaptive 세팅에서 다양한 pre-training 방법들에 다양한 충분한 탐색이 이뤄지지 않음

Dense Retrieval 모델을 위한 비지도 학습 기반의 Domain Adaptation 기술을 제안
cross-encoder를 통한 pseudo labeling을 실시해서 query generator로 사용

02 GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Approach



Query Generation

- T5 encoder-decoder를 활용해서 Target corpus에 대해서 각 passage별 3개의 query를 형성함

Negative Mining

- Generated query들에 대해서 기존의 retrieval system을 활용해서 50개의 negative passage들을 retrieve함

Pseudo Labeling

- Cross-Encoder를 활용해서 각 (Query,Passage) pair의 점수를 산출
- Soft-label (query,passage) pairs

앞선 과정을 통해서 얻은 Synthetic data를 MarginMSE loss를 이용해서 query와 passage를 같은 벡터 공간에 매핑하는 domain adapted dense retriever를 학습시킴

=> Dense retriever가 positive와 negative query-passage쌍에 대해서 score margin을 모방하도록 학습함

02 GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Approach

QGen에서 사용한 MultipleNegativeRanking (MNRL) loss의 문제점?

- MNRL은 query와 matching되는 passage만을 relevant하다고 학습하고 나머지 passage는 irrelevant하게 학습됨
- 하지만 query encoder를 통해서 query는 결함이 있을 수 있고, passage들로부터 답을 얻지 못할 수 있음
- 또한 주어진 query에 다른 passage 역시 매칭 될 수 있음 (False Negative)

$$L_{\mathrm{MNRL}}(\theta) = -\frac{1}{M}\sum_{i=0}^{M-1}\log\frac{\exp\left(\tau\cdot\sigma(f_\theta(Q_i), f_\theta(P_i))\right)}{\sum_{j=0}^{M-1}\exp\left(\tau\cdot\sigma(f_\theta(Q_i), f_\theta(P_j))\right)}$$

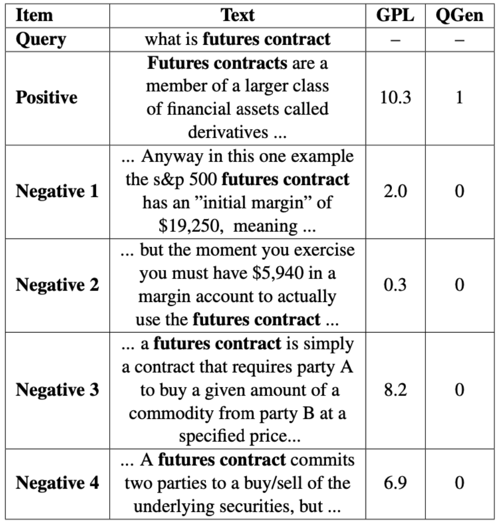
MarginMSE loss의 장점

- passage가 주어졌을 때 badly generated query는 낮은 점수를 받음
- False negative는 cross-encoder에서 높은 점수를 받기에 dense retriever는 해당 embedding에 large distance를 주지 않아도 됨

MarginMSE loss

- cross-encoder를 이용해서 (query,passage)쌍에 대해서 0,1이 아닌 연속적인 점수로 soft-label을 부여
- Dense retriever를 앞서 형성한 postivie, negative (query,passage) 간의 Margin을 모방하도록 학습시킴

$$L_{\mathrm{MarginMSE}}(\theta) = -\frac{1}{M}\sum_{i=0}^{M-1}|\hat{\delta}_i - \delta_i|^2 \tag{1}$$

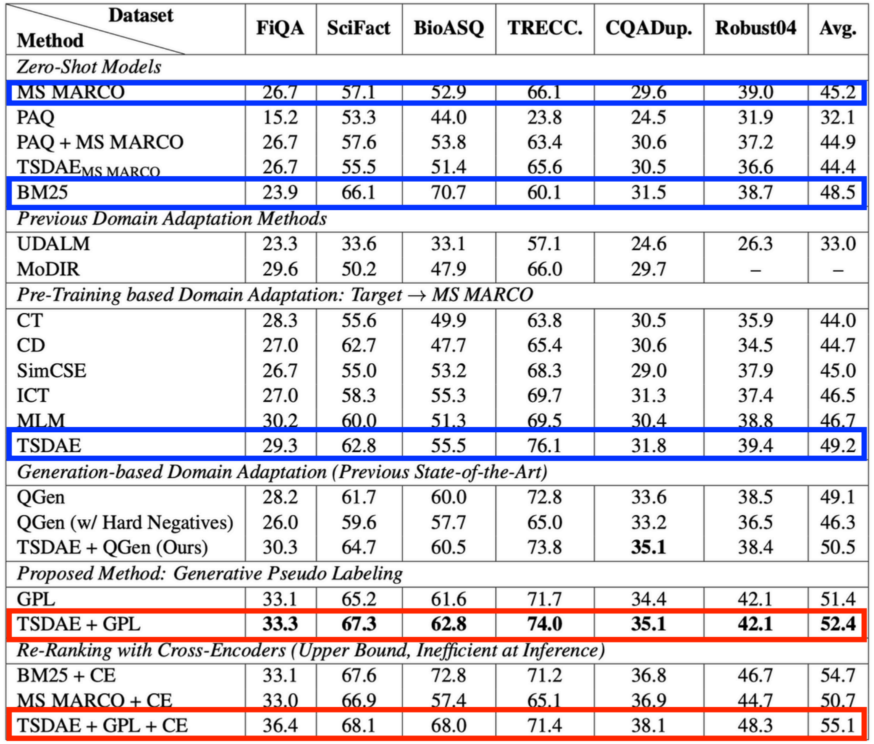


Item	Text	GPL	QGen
Query	what is **futures contract**	–	–
Positive	**Futures contracts** are a member of a larger class of financial assets called derivatives ...	10.3	1
Negative 1	... Anyway in this one example the s&p 500 **futures contract** has an ”initial margin” of $19,250, meaning ...	2.0	0
Negative 2	... but the moment you exercise you must have $5,940 in a margin account to actually use the **futures contract** ...	0.3	0
Negative 3	... a **futures contract** is simply a contract that requires party A to buy a given amount of a commodity from party B at a specified price...	8.2	0
Negative 4	... A **futures contract** commits two parties to a buy/sell of the underlying securities, but ...	6.9	0

02 GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

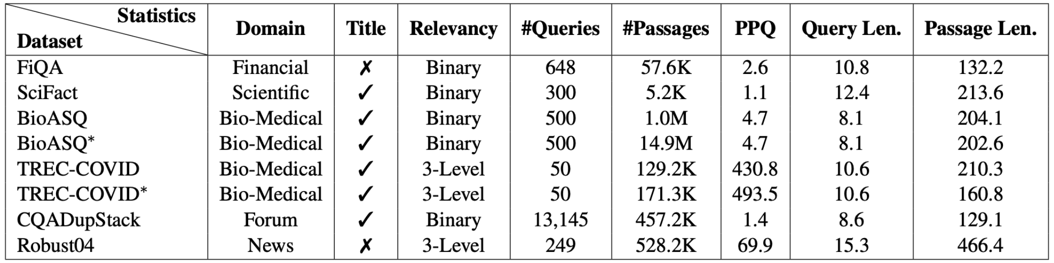
Experiments

- evaluate on domain-specific datasets from the BeIR benchmark



Dataset / Method	FiQA	SciFact	BioASQ	TRECC.	CQADup.	Robust04	Avg.
Zero-Shot Models							
MS MARCO	26.7	57.1	52.9	66.1	29.6	39.0	45.2
PAQ	15.2	53.3	44.0	23.8	24.5	31.9	32.1
PAQ + MS MARCO	26.7	57.6	53.8	63.4	30.6	37.2	44.9
$\text{TSDAE}_{\text{MS MARCO}}$	26.7	55.5	51.4	65.6	30.5	36.6	44.4
BM25	23.9	66.1	70.7	60.1	31.5	38.7	48.5
Previous Domain Adaptation Methods							
UDALM	23.3	33.6	33.1	57.1	24.6	26.3	33.0
MoDIR	29.6	50.2	47.9	66.0	29.7	–	–
Pre-Training based Domain Adaptation: Target → MS MARCO							
CT	28.3	55.6	49.9	63.8	30.5	35.9	44.0
CD	27.0	62.7	47.7	65.4	30.6	34.5	44.7
SimCSE	26.7	55.0	53.2	68.3	29.0	37.9	45.0
ICT	27.0	58.3	55.3	69.7	31.3	37.4	46.5
MLM	30.2	60.0	51.3	69.5	30.4	38.8	46.7
TSDAE	29.3	62.8	55.5	76.1	31.8	39.4	49.2
Generation-based Domain Adaptation (Previous State-of-the-Art)							
QGen	28.2	61.7	60.0	72.8	33.6	38.5	49.1
QGen (w/ Hard Negatives)	26.0	59.6	57.7	65.0	33.2	36.5	46.3
TSDAE + QGen (Ours)	30.3	64.7	60.5	73.8	**35.1**	38.4	50.5
Proposed Method: Generative Pseudo Labeling							
GPL	33.1	65.2	61.6	71.7	34.4	42.1	51.4
TSDAE + GPL	**33.3**	**67.3**	**62.8**	**74.0**	**35.1**	**42.1**	**52.4**
Re-Ranking with Cross-Encoders (Upper Bound, Inefficient at Inference)							
BM25 + CE	33.1	67.6	72.8	71.2	36.8	46.7	54.7
MS MARCO + CE	33.0	66.9	57.4	65.1	36.9	44.7	50.7
TSDAE + GPL + CE	36.4	68.1	68.0	71.4	38.1	48.3	55.1

Table 1: Evaluation using nDCG@10. The best results of the single-stage dense retrievers are bold. TRECC. and CQADup. are short for TREC-COVID and CQADupStack. Our proposed GPL significantly outperforms other domain adaptation methods. For the first time, we investigate the TSDAE pre-training in domain adaptation for dense retrieval and find it can significantly improve both QGen and GPL.



Statistics / Dataset	Domain	Title	Relevancy	#Queries	#Passages	PPQ	Query Len.	Passage Len.
FiQA	Financial	✗	Binary	648	57.6K	2.6	10.8	132.2
SciFact	Scientific	✓	Binary	300	5.2K	1.1	12.4	213.6
BioASQ	Bio-Medical	✓	Binary	500	1.0M	4.7	8.1	204.1
BioASQ*	Bio-Medical	✓	Binary	500	14.9M	4.7	8.1	202.6
TREC-COVID	Bio-Medical	✓	3-Level	50	129.2K	430.8	10.6	210.3
TREC-COVID*	Bio-Medical	✓	3-Level	50	171.3K	493.5	10.6	160.8
CQADupStack	Forum	✓	Binary	13,145	457.2K	1.4	8.6	129.1
Robust04	News	✗	3-Level	249	528.2K	69.9	15.3	466.4

Table 7: Statistics of the target datasets used in the experiments. Column **Title** indicates whether there is (✓) a title for each passage or not (✗). Column **PPQ** represents number of Passages Per Query. Query/passage lengths are counted in words. Symbol $*$ marks the original version from the BeIR benchmark (Thakur et al., 2021b)

Wikipedia (DPR)
: 21 M passage

MS MARCO
: 8.8M passage

Baselines

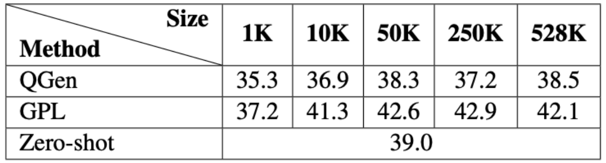
- Zero-Shot Models
  - MS MARCO
  - PAQ
  - PAQ + MS MARCO
  - TSDAE_MS MARCO
  - BM25
- Previous Domain Adaptation Methods
  - UDALM
  - MoDIR
- Pre-Training based Domain Adaptation
- Generation-based Domain Adaptation
- Re-Ranking with Cross-Encoders

- BM25는 zero-shot 모델 중에서 여러 Dataset에서 안정적인 결과를 보임 (BeIR paper)
- Pre-Training과 Generation 기반의 다른 Domain Adaptation 방법들과 비교했을 때 GPL은 뛰어난 성능을 보임
- TSDAE : denoising auto-encoder architecture
- Re-Ranking을 실시할 경우 Avg 성능이 오르지만 (+2.7), inference 속도가 느려지는 단점이 존재

02

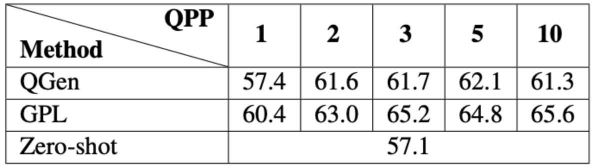
GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Experiments



Method \ Size	1K	10K	50K	250K	528K
QGen	35.3	36.9	38.3	37.2	38.5
GPL	37.2	41.3	42.6	42.9	42.1
Zero-shot	39.0				

Table 2: Influence of corpus size on performance on Robust04. The full size is 528K. GPL can achieve the best performance with as little as 50K passages.



Method \ QPP	1	2	3	5	10
QGen	57.4	61.6	61.7	62.1	61.3
GPL	60.4	63.0	65.2	64.8	65.6
Zero-shot	57.1				

Table 3: Influence of number of generated Queries Per Passage (QPP) on performance on SciFact. Using a large QPP (e.g. 5 or 10) cannot further improve the performance.

Robustness against Query generation

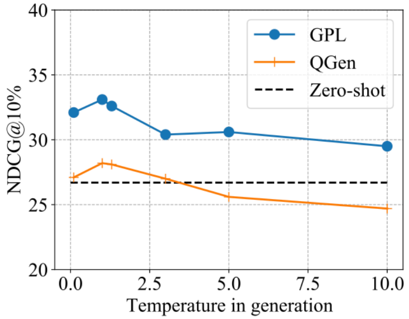
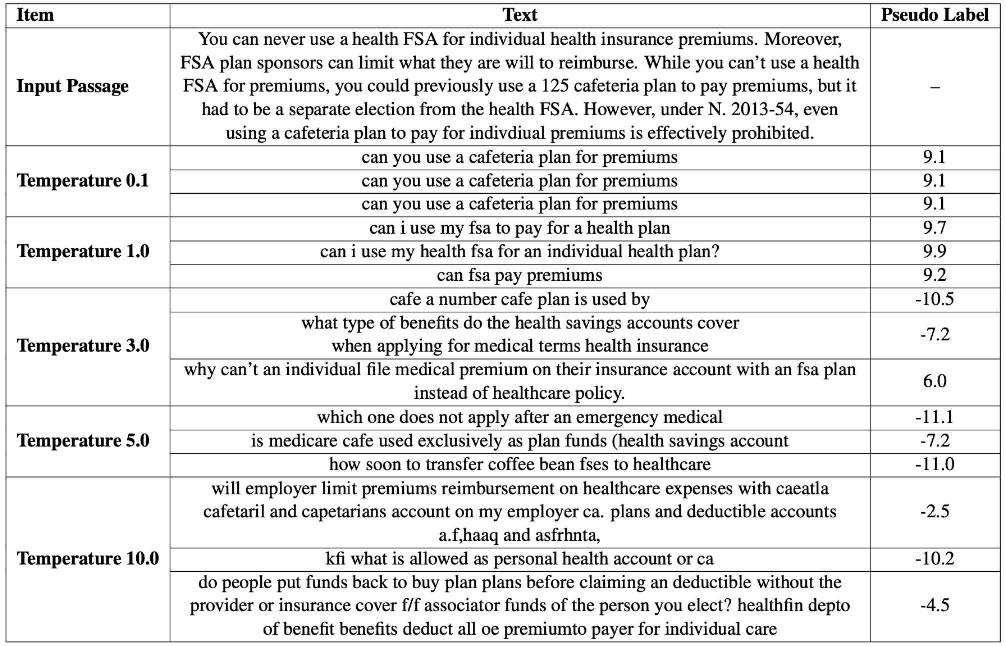


Figure 3: Influence of the temperature in generation on the performance on FiQA. A higher temperature means more diverse queries but of lower quality. GPL can still yield around 3.0-point improvement over the zero-shot baseline with high temperature value of 10.0, where the generated queries have nearly no connection to the passages.



Item	Text	Pseudo Label
Input Passage	You can never use a health FSA for individual health insurance premiums. Moreover, FSA plan sponsors can limit what they are will to reimburse. While you can't use a health FSA for premiums, you could previously use a 125 cafeteria plan to pay premiums, but it had to be a separate election from the health FSA. However, under N. 2013-54, even using a cafeteria plan to pay for indivdiual premiums is effectively prohibited.	–
Temperature 0.1	can you use a cafeteria plan for premiums	9.1
	can you use a cafeteria plan for premiums	9.1
	can you use a cafeteria plan for premiums	9.1
Temperature 1.0	can i use my fsa to pay for a health plan	9.7
	can i use my health fsa for an individual health plan?	9.9
	can fsa pay premiums	9.2
Temperature 3.0	cafe a number cafe plan is used by	-10.5
	what type of benefits do the health savings accounts cover when applying for medical terms health insurance	-7.2
	why can't an individual file medical premium on their insurance account with an fsa plan instead of healthcare policy.	6.0
Temperature 5.0	which one does not apply after an emergency medical	-11.1
	is medicare cafe used exclusively as plan funds (health savings account	-7.2
	how soon to transfer coffee bean fses to healthcare	-11.0
Temperature 10.0	will employer limit premiums reimbursement on healthcare expenses with caeatla cafetaril and capetarians account on my employer ca. plans and deductible accounts a.f,haaq and asfrhnta,	-2.5
	kfi what is allowed as personal health account or ca	-10.2
	do people put funds back to buy plan plans before claiming an deductible without the provider or insurance cover f/f associator funds of the person you elect? healthfin depto of benefit benefits deduct all oe premiumto payer for individual care	-4.5

Table 10: Examples of generated queries under different temperature value for a passage from FiQA.

temperature effect in nucleus sampling GPL은 high temperature에서 다양한 query를 만들어내지만 zero-shot setting보다 우수한 성능을 유지함

02

GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Experiments

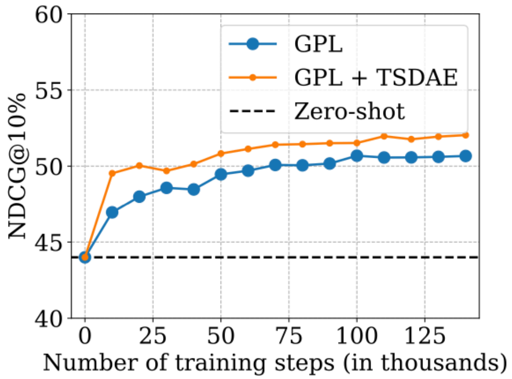
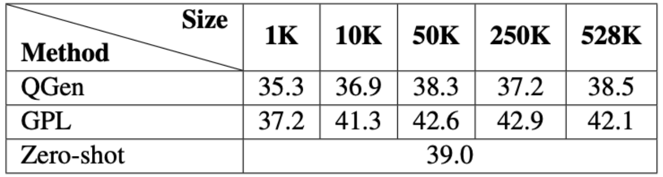


Figure 2: Influence of the number training steps on the averaged performance. The performance of GPL begins to be saturated after 100K steps. TSDAE helps improve the performance during the whole training stage.

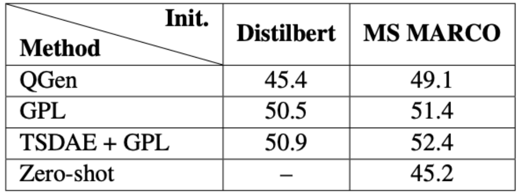
TSDAE pre-training method는 꾸준하게
성능 향상에 도움이 됨



Method \ Size	1K	10K	50K	250K	528K
QGen	35.3	36.9	38.3	37.2	38.5
GPL	37.2	41.3	42.6	42.9	42.1
Zero-shot	39.0				

Table 2: Influence of corpus size on performance on Robust04. The full size is 528K. GPL can achieve the best performance with as little as 50K passages.

Zero-shot 성능을 10k corpus 만으로 뛰어넘을 수 있음
Best performance는 50k corpus



Method \ Init.	Distilbert	MS MARCO
QGen	45.4	49.1
GPL	50.5	51.4
TSDAE + GPL	50.9	52.4
Zero-shot	–	45.2

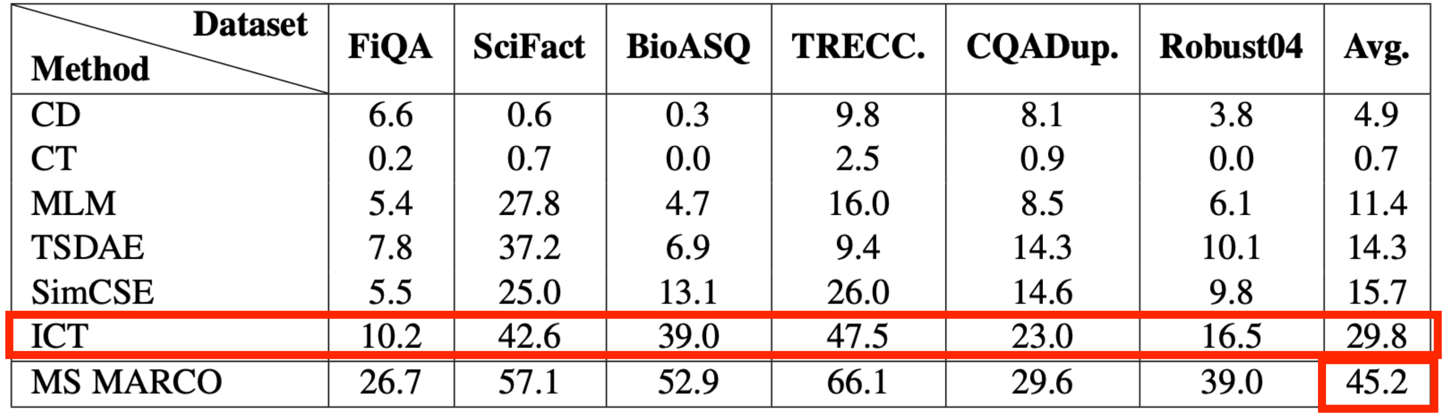
Table 4: Influence of initialization checkpoint on performance in average. GPL yields similar performance when starting from different checkpoints.

Initial checkpoint에 다른 방법들에 비해서
민감하지 않음

02 GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Experiments

Performance of Unsupervised Pre-Training



Dataset Method	FiQA	SciFact	BioASQ	TRECC.	CQADup.	Robust04	Avg.
CD	6.6	0.6	0.3	9.8	8.1	3.8	4.9
CT	0.2	0.7	0.0	2.5	0.9	0.0	0.7
MLM	5.4	27.8	4.7	16.0	8.5	6.1	11.4
TSDAE	7.8	37.2	6.9	9.4	14.3	10.1	14.3
SimCSE	5.5	25.0	13.1	26.0	14.6	9.8	15.7
ICT	10.2	42.6	39.0	47.5	23.0	16.5	29.8
MS MARCO	26.7	57.1	52.9	66.1	29.6	39.0	45.2

CD(ConDensor)
: MLM on top of dense representation

CT (Contrastive Tension)

TSDAE
: denoising auto-encoder architecture

ICT (Inverse Clozer Task)

- retriever task를 위한 Domain adaptation 시나리오에서 최근에 사용되는 6개의 pre-training 방법론들의 성능을 비교함
- ICT pre-training 방법이 가장 높은 성능을 보임
- Target domain pre-training + MS MARCO supervised training에서는 TSDAE가 가장 높은 성능을 보였음
- 하지만 모든 unsupervised pre-training 방법은 MS MARCO zero-shot baseline보다는 좋은 성능을 얻지 못함

02

GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Discussion Points

- False Negative Case에 의한 모델 성능 저하를 막기 위한 또 다른 방법은 무엇이 있을까?
  - 실제로 정답에 해당하지만 Annotation 결과에 의해서 Negative Passage로 구분돼서 query-passage matching이 안 되도록 훈련이 된다면 모델 성능 저하로 이어질 것
  - 본 연구에서는 Cross encoder의 soft label을 통한 margin을 극대화하는 방식으로 훈련하며 False Negative의 경우는 score가 True Negative와 다르게 크게 낮지 않게 학습하여 오류를 방지
- 다른 NLP task처럼 Large corpus에 pre-trained된 Language Model을 적용하는 것이 Retriever task에서 잘 안되는 경우는 무엇일까?
  - Intrinsic Knowledge를 활용할 수 있는 다른 task와 다르게 Retriever는 query-passage간의 관련성을 학습해야함. corpus내에서 다른 passage간의 차이를 학습하는 것이 중요함?

02 GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Related Papers to Read

- Ram, Ori, et al. "Learning to Retrieve Passages without Supervision." arXiv preprint arXiv:2112.07708 (2021). [link]
- Thakur, Nandan, et al. "BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models." arXiv preprint arXiv:2104.08663 (2021). [link]

Research Ideas

03 Recent Trends in ODQA

Summary

- Explicit context retrieval vs. knowledge encoded in models
- Rationale and evidence to support answers
- Answer triggering : knowing when it doesn't know
- Narrative questions and long-form answers
- Multi-turn, conversational QA
- Multi-modal interactions (e.g. VQA, virtual tour guide)
- Full wiki setting
- Interpretability
- multilinguality
- Domain shift + unsupervised method

03 Research Ideas

- Interpretability
- Domain shift + unsupervised method
- Generalization
- Answer-query (1:N match) => multi vector embedding
- What characteristics QA system should have?
- Pre-training task is quite expensive, so using existing datasets efficiently is further researched. (e.g. in-batch negatives)
- Pseudo query, pseudo-evidence … pseudo + @
- Unsupervised (self-supervised) pre-training tasks for ODQA/ Retriever
- User interaction and grounding (Multi-modal interactions)
- Using pre-trained large language model as intrinsic knowledge base