Learning To Retrieve Prompts for In-Context Learning

Ohad Rubin, Jonathan Herzig, Jonathan Berant

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies

1. Overview

[배경]
In-context learning(ICL) - 질문이 몇개의 훈련예제와 더불어 input으로 주어지면, 파라미터의 업데이트 없이 바로 output를 출력하는 NLP의 패러다임. ICL의 매력적인 속성은 semantic parsing과 같은 여러 다운스트림 태스크에 단일 모델을 제공한다는 점에서 매력적.

[연구질문]
ICL의 성능은 훈련예제, 즉 프롬프트에 크게 좌우됨(Liu et al., 2021a). ICL을 위한 프롬프트를 효율적으로 retrieve할 수 있는 방법이 없을까?

기존 연구들은 바로 적용이 가능한 비지도학습 기반의 유사성지표를 사용하거나, surface similarity를 기반으로 예제를 선택하도록 prompt retriever를 지도학습(Das et al., 2021).

[가설]
prompt retriver에 surface similarity heuristics를 적용하는 것보다, 언어모델 그 자체를 이용해 학습을 시키는 것이 더 좋은 성능을 보일 것이다.

1) 주어진 입력-출력 쌍(x,y)에서 출발. 언어모델(scoring-LM)을 통해 프롬프트로 "x + candidate training example"이 주어졌을 때 y가 출력될 확률을 추정. 2) 이 확률에 따라 training example을 긍정/부정으로 레이블링하고, 이를 기반으로 efficient dense retriever를 훈련. 3) 이 retriever를 바탕으로 test 시 training example을 검색하고, 언어모델(inference LM)을 바탕으로 답변을 생성.

scoring-LM와 inference-LM을 분리하여 접근하고 있는데, scoring LM이 작은 모델일 경우에는 retriever를 위한 레이블을 저렴하고 효율적에 생성할 수 있다. socring-LM과 inference-LM이 동일한 경우에도, parameter에 직접 접근할 수 없는 상황에서 유사도 함수 학습만 담당하는 retriever 훈련 함수를 생성. 점점 거대해지고 있는 언어모델들과 상호작용하기 위한 효율적인 프롬프트를 학습하기 위한 접근이라는 점에서 Efficient Prompt Retrieval(EPR)이라 명명.

[실험]
자연어 발화를 구조화된 의미표현으로 매핑하는 세가지 seq-to-seq 태스크(MTop, SM-CalFlow, BREAK)를 대상으로, 두가지 방식으로 실험(scoring과 inference에 모두 동일한 LM(GPT-NEO)을 사용한 경우와, inference에는 더 큰 모델(GPT-J, GPT0-3, Codex)를 사용한 경우). 전반적으로 이전 작업들보다 훨씬 뛰어난 성능을 보였다.

형식화된 정의

training set = input-ouput sequences $\mathscr{D} = \{(x_i, y_i)\}_{i=1}^{n}$ 와 test example x_{test}이 주어져있다고 하자. 우리의 목적은 $m \ll n$ 인 프롬프트 $\mathscr{P} = \{(x_j, y_j)\}_{j=1}^{m} \subset \mathscr{D}$의 부분집합을 찾아내도록 retriever $R(x_{test}, \mathscr{D})$를 학습시키는 것이다.

inference LM, g가 주어진다고 하자., x_{test}와 $\mathscr{P}$가 결합되어 모델 g에 prefix로 입력되면, 경우 목표 출력 시퀀스를 생성해야 한다. 구체적으로, $g([\mathscr{P}; x_{test}])$에서 디코딩하면 y_{test}를 출력해야 한다.

선행 연구

Liu et al.(2021a)은 다양한 프롬프트가 GPT-3의 다운스트림 성능에 미치는 영향을 조사하여 prompt retriever의 필요성을 입증. 비지도 문장 인코더를 사용하여 훈련 예시를 인코딩하고 테스트 인스턴스에 대해 최근접 이웃을 검색.

Das et al.(2021)은 지식 기반 질문에 대한 답변을 위해 지도형 프롬프트 리트리버를 훈련시켰습니다. 이 리트리버는 지식 기반 쿼리에 맞게 조정된 감독을 통해 훈련되었으며, 공식 쿼리 간의 표면 유사성에 의존했습니다. 반면, 우리의 접근 방식은 생성형 LM 자체를 활용하므로 훨씬 일반화가능성이 높음.

Shin et al.(2021)은 GPT-3을 사용하여 semantic parsing을 위한 few-shot 예제를 선택했음. 다만 별도의 retriever를 훈련시키는 대신, 훈련세트에서 무작위 쌍을 추출한 뒤, GPT-3에게 바로 질문과 유사한 것을 선택하도록 함. 각 테스트 인스턴스에 대해 GPT-3를 수백 번 실행하는 고비용 추론 절차가 발생.

3. EPR

이제 EPR을 훈련시키고, 이를 바탕으로 추론하는 방법에 대해 논의.

3.1 How to gererate the labeled data

candidate set

기본적인 발상은 훈련예제들이 다른 훈련예제들을 위한 좋은 프롬프트로 기능할 수 있다는 것이다. 그러나 모든 예제들에 대해 서로에 대한 유사성을 계산할 경우, 시간복잡도가 $|\mathscr{D}|$의 제곱이 되므로, 먼저 소수의 후보집합 $\bar{\mathscr{E}} \subset \mathscr{D}$을 구해 유사도 계산을 실시한다.

이때 우리는 테스트 상황이 아니기 때문에 타겟 시퀀스 y를 활용할 수 있다. 주어진 훈련예제 (x,y)에 대해 유사도가 높은 후보집합 $\bar{\mathscr{E}} = R_u\left((x, y), \mathscr{D}\right)$를 구하기 위해, (x,y) 모두를 활용하는 방법과 y만을 활용하는 방법, BM25와 SBERT를 테스트. 최종적으로 y에 대해 BM25를 바탕으로 L개의 예제로 구성된 후보집합 $\bar{\mathscr{E}} = \{\bar{e}_1, \cdots, \bar{e}_L\}$를 구했다.

scoring the candidate set

이제, 후보집합의 각 원소 $\bar{e}_l \in \bar{\mathscr{E}}$에 대해 LM을 바탕으로 점수를 매긴다. 입력으로 $\bar{e}_l, x$가 들어왔을 때 y가 출력될 조건부 확률에 대한 $\hat{g}$를 기반으로 한 추정값이다. inferenceLM g에 대한 프록시로서 scoring LM $\hat{g}$를 사용했다.

$$s(\overline{e}_l) = \text{Prob}_{\hat{g}}(y \mid \overline{e}_l, x)$$

이제 이를 바탕으로 top-k를 $\mathscr{E}_{pos}$에, bottom-k를 $\mathscr{E}_{neg}$에 할당한다. 전자가 좋은 프롬프트의 예시가 된다면, 후자의 경우 이미 BM25를 바탕으로 유사도가 높은 후보예제 중에서 선별된 만큼, 'hard negative'가 된다.

3.2 Training and Inference

1) training

이제 각 training example들에 대해 $\mathscr{E}_{pos}$와 $\mathscr{E}_{neg}$가 주어져있고, 훈련과정에서는 이를 바탕으로 대조적 학습(DPR)을 수행한다. input tokens x는 $E_X(\cdot)$을 통해, 후보 prompt(input-output token)는 $E_P(\cdot)$을 통해 인코딩된다.

training instance는 $\langle x_i, e_i^+, e_{i,1}^-, \ldots, e_{i,2B-1}^- \rangle$와 같은 형태로 주어진다. 먼저 B개의 training examples가 미니배치의 형태로 주어진다. 유사성을 판단하게 될 i번째 input x_i가 있고, 긍정적 사례로서 $\mathscr{E}_{pos}^{(i)}$로부터 샘플링된 example e_i^+가 있다. 그리고 $\mathscr{E}_{neg}^{(i)}$에서 샘플링된 e_i^-와 더불어, 같은 배치 내 다른 B-1개의 instance로부터 하나의 긍정사례와 하나의 부정사례를 샘플링한다. 결과적으로 하나의 positive example과 $1 + (B-1) + (B-1) = 2B - 1$개의 hard negatives가 선택된다.

이제 다음과 같은 손실함수를 최소화하도록 $E_X(\cdot)$와 $E_P(\cdot)$를 최적화한다.

$$L(x_i, e_i^+, e_{i,1}^-, \ldots, e_{i,2B-1}^-) = -\log \frac{e^{\text{sim}(x_i, e_i^+)}}{e^{\text{sim}(x_i, e_i^+)} + \sum_{j=1}^{2B-1} e^{\text{sim}(x_i, e_{i,j}^-)}}$$

2) inference

추론 상황에서는 먼저 미리 모든 training examples에 대해 $E_P(\cdot)$를 적용한 벡터를 저장해두고, x_{test}에 대해 MIPS(Maximum Inner-Product Search) 연산을 통해 L개의 프롬프트를 검색한다. 그리고 inference-LM의 maximal context size를 고려하여 $\sum_{i=1}^{L'} |e_i| + |x_{\text{test}}| + |\overline{y}'| \leq C$를 만족하는 $L' \leq L$개의 예제를 선택한다. 그리고 최종적으로 이를 모두 input에 넣어 $g\left([e_{L'}; e_{L'-1}; \ldots; e_1; x_{\text{test}}]\right)$를 얻는다.

4. Experimental Results

4.1 Datasets(table1)

BREAK(Wolfson et al., 2020) : 44K/7K/8K. 자연어를 순차적인 원소들로 분해하는 매핑.

MTop(Li et al, 2021) : 16K/2K/4K. 자연어를 11개 도메인의 작업으로 분할하는 매핑.

SMCalFlow(Andreas et al.) : 44K(/7K/8K). 자연어를 달력, 날씨, 장소검색 등을 위한 API 호출, 함수작성 등으로 변환하는 매핑.

4.2 Baselines and Orcacles

Dataset	Size	Utterance	Meaning Representation

BREAK	60K	*There are more birds in the image on the right than in the image on the left.*	`1) return right image;`
			`2) return birds in #1;`
			`3) return number of #2;`
			`4) return left image;`
			`5) return birds in #4`
			`6) return number of #5;`
			`7) return if #3 is higher than #6;`

MTOP	22K	*call Zoey's wife.*	`[IN:CREATE_CALL =`
			`[SL:CONTACT = [IN:GET_CONTACT =`
			`[SL:CONTACT_RELATED = Zoey]`
			`[SL:TYPE_RELATION = wife]]]]`

SMCALFLOW	148K	*Can you create me a new meeting on thursday morning?*	`(Yield (CreateCommitEventWrapper`
			`(CreatePreflightEventWrapper`
			`(Event.start_?`
			`(DateTimeConstraint (Morning)`
			`(NextDOW (Thursday)))))))`

Table 1: Examples from each of the datasets we evaluate on.

1) unsupervised baselines

RANDOM : training set에서 랜덤으로 선택된 예제.

SBERT : 사전학습된 sentence transformer(paraphrase-mpnet-base-v2, 110M)를 바탕으로 x_{test}를 인코딩한 뒤 가장 거리가 가까운 예제를 선택.

BM25 : BM25를 바탕으로 x_{test}와 가장 유사한 예제를 선택.

BruteForce : training set에서 임의로 200개를 샘플링한뒤, x_{test}와의 유사도를 inference-LM에게 직접 평가하도록 함. 매회차 과도한 비용발생.

2) training set으로 훈련된 prompt retriever

BM25를 바탕으로 50개의 훈련예제로 구성된 후보집합 $\tilde{\mathcal{E}}$을 선정한 뒤, 각 원소에 대해 유사성 점수를 매기도록 하여 이를 기반으로 pos, neg 예제를 선정하고 대조학습 훈련을 진행.

DR-BM25 : BM25 기반으로 점수를 매김

CBR(Case-Based Reasoning; Das et al. 2021) : 두 관계 집합들의 논리적 표현을 바탕으로 F1-score 계산.

EPR(Efficient Prompt Retrieval) : 우리의 모델

3) Orcacle

BM25-Oracle : y_{test}를 바탕으로 DR-BM25를 계산하고 학습진행. BM25기반으로 이론적으로 가능한 최대한의 성과.

LM-Oracle : y_{test}를 바탕으로 LM-score 계산하고 학습진행. LM 기반으로 이론적으로 가능한 최대한의 성과.

(AnyCorrect-Oracle) : $\tilde{\mathcal{E}}$의 모든 원소에 대해 평가 진행.

4.3 Experimental Details

1) Language models

scoring-LM :

GPT-NEO(2.7B model trained on 825GB corpus 'The Pile')

inference-LM :

GPT-J(7B model trained on 825GB corpus 'The Pile')

GPT-3(175B model from filtered subset of common crawl)

CODEX(175B model finetuned on code from GitHub)

2) Evaluation

BREAK -> dev set에 대해서는 LF-EM, test set에 대해서는 NEM(Normalized Exact Match)

MTop & SM-CalFlow -> EM(Exact Match)

4.4 Results

1) LM-as-a-service

	Model	BREAK	MTOP	SMCALFLOW
Unsuper.	RANDOM	1.7	7.3	8.9
	SBERT	21.6	48.7	43.6
	BM25	26.0	52.9	46.1
	BRUTEFORCE	7.7	18.1	11.1
Super.	DR-BM25	23.6	50.2	43.1
	CBR	25.7	57.0	51.4
	EPR (ours)	**31.9**	**64.2**	**54.3**
Oracle	BM25-ORACLE	32.3	58.9	47.3
	LM-ORACLE	43.1	71.6	73.7

Table 2: Development results when GPT-NEO is the scoring and inference LM. Numbers for BREAK are LF-EM, and for MTOP and SMCALFLOW are EM.

	Model	BREAK	MTOP
Unsuper.	BM25	17.6	49.0
Super.	CBR	18.4	57.5
	EPR (ours)	**23.9**	**64.4**

Table 3: Test results where GPT-NEO is the scoring and inference LM. Numbers for BREAK are NEM, the official metric, and for MTOP are EM.

(dev) (test)

	Model	One-shot	Full-context
Unsuper.	RANDOM	1.1	1.7
	BM25	15.2	26.0
Super.	DR-BM25	14.1	23.6
	CBR	14.5	25.7
	EPR	**23.0**	31.9
Oracle	BM25-ORACLE	18.0	32.3
	LM-ORACLE	33.3	43.1
	ANYCORRECT-ORACLE	53.6	-

Table 4: Development results on BREAK with GPT-NEO in the one-shot setting. Numbers are LF-EM. Full-context is the corresponding numbers from Table 2.

2) LM-as-a-proxy

	BREAK				MTOP				SMCALFLOW			
Method	RANDOM	BM25	CBR	EPR	RANDOM	BM25	CBR	EPR	RANDOM	BM25	CBR	EPR
GPT-3	4.2	20.1	21.3	25.3	7.6	52.5	54.8	62.6	5.8	35.3	41.6	46.5
CODEX	8.9	24.5	24.2	29.5	10.8	60.6	59.4	66.1	7.2	45.1	48.7	50.3
GPT-J	3.3	26.7	26.7	31.5	8.8	56.6	58.0	65.4	10.6	50.4	50.9	57.4
GPT-NEO	1.0	22.8	25.8	29.9	7.6	52.8	55.4	63.6	8.0	46.1	50.1	53.5

Table 5: Results on a random sample of 1,000 examples from the development set when using GPT-Neo as a scoring LM across different inference LMs and datasets.

3) Analysis

		EPR	CBR
Test Example	Utterance	*Give the code of the airport with the least flights.*	

	Meaning Representation	`1) airports`
		`2) flights of #1`
		`3) number of #2 for each #1`
		`4) #1 where #3 is lowest`
		`5) code of #4`	

Top-1	Utterance	*What is the code of the city with the most students?*	*What destination has the fewest number of flights?*

	Meaning Representation	`1) cities`
		`2) students in #1`
		`3) number of #2 for each #1`
		`4) #1 where #3 is highest`
		`5) code of #4`
			`1) destinations`
			`2) flights of #1`
			`3) number of #2 for each #1`
			`4) #1 where #3 is lowest`

Top-2	Utterance	*Return the code of the city that has the most students.*	*Which destination has least number of flights?*

	Meaning Representation	`1) cities`
		`2) students in #1`
		`3) number of #2 for each #1`
		`4) #1 where #3 is highest`
		`5) code of #4`
			`1) destinations`
			`2) flights to #1`
			`3) number of #2 for each #1`
			`4) #1 where #3 is lowest`

Top-3	Utterance	*Find the count and code of the job has most employees.*	*What is the number of airports per country, ordered from most to least?*

	Meaning Representation	`1) jobs`
		`2) employees of #1`
		`3) number of #2 for each #1`
		`4) #1 where #3 is highest`
		`5) employees of #4`
		`6) number of #5`
		`7) code of #4`
		`8) #6 , #7`
			`1) countries`
			`2) airports in #1`
			`3) number of #2 for each #1`
			`4) #3 sorted by most to least`

Table 6: An example from BREAK development set where EPR is correct and CBR is incorrect along with the top-3 training examples retrieved from each retriever.

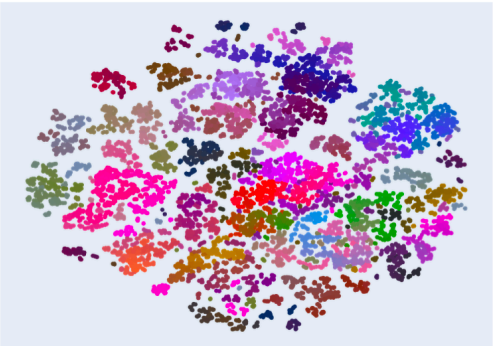

Figure 3: A t-SNE projection and clustering of the representations learned by EPR for the training examples in BREAK. An interactive version displaying individual examples is available here.

4) Prompt Copying

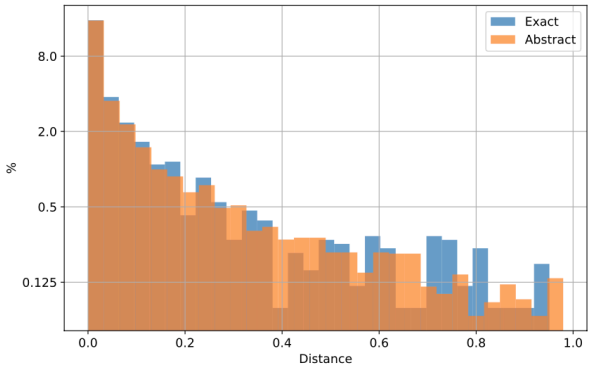

Figure 4: On the subset of copied patterns we plot the distribution of the distance from the test instance to the example containing the pattern. Shown on the BREAK validation set using EPR in the LM-as-a-service setup using GPT-NEO. Note that the y-axis is in log-scale.

5. Q&A

(스스로의 질문)

- ODQA와 ICL의 차이점은?
- ODQA 모델 중에서는 어떤 것과 가장 가까운가