추상화 및 추론 문제 해결을 위한 대조학습

구교준[O2] · 심우창[1] · 임재균[2] · 김세진[1] · 김선동[1Ψ]
광주과학기술원 AI대학원[1], 광주과학기술원 전기전자컴퓨터공학부[2]
{gugyojoon, woochang, ijk000829}@gm.gist.ac.kr, {sejinkim, sundong}@gist.ac.kr

Using Contrastive Learning for Abstraction and Reasoning task

Gyojoon Gu[O2] · Woochang Sim[1] · Jaegyun Im[2] · Sejin Kim[1] · Sundong Kim[1Ψ]
GIST AI[1], GIST EECS[2]

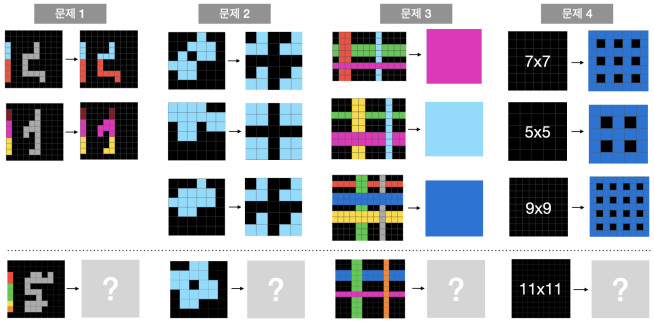
요 약

추상화 및 추론 문제를 풀 때는 사전 지식이 중요한 역할을 한다. 특히 문제에 주어진 정보가 적다면 사전 지식은 더 중요하다. 사람은 인공지능에 비해 경험을 토대로 한 많은 사전 지식이 있다. 따라서 인공지능이 문제를 푸는 성능을 향상시키기 위해서는 적절한 사전 지식을 제공해주는 것이 필요하다고 판단했다. 본 연구는 추상화 및 추론 문제의 벤치마크 데이터셋인 ARC를 문제 유형별로 분류해 사전정보로 사용할 수 있도록 한다. ARC 문제에서 주어지는 하나의 입력-출력 쌍을 하나의 표현 벡터로 나타내고, 대조학습을 통해 같은 문제끼리는 벡터의 거리를 가깝게 하고 다른 문제끼리는 벡터의 거리를 멀리하는 방법을 이용한다. 결과적으로 비슷한 유형의 문제끼리는 인접한 벡터 공간 안에 표현되고, 이 벡터를 이용해 문제 유형별로 분류를 수행했다. 본 연구에서 분류한 문제 유형을 사전 정보로 제공한다면 추상화 및 추론 문제를 해결하는데 큰 도움이 될 것으로 기대된다.

1[1]. 서 론

2019년 François Chollet는 인공지능의 일반화된 지능을 측정하고자 ARC(Abstract and Reasoning Corpus) [1] 데이터셋을 소개했다. ARC는 일반화 능력, 스킬 습득 능력과 같이 지능의 중요한 요소들을 측정해 인공지능이 인간 수준의 추상적 추론 능력을 달성했는지 평가한다. 사람은 ARC 문제를 80%의 정확도로 해결할 수 있지만 [2], 현재 가장 좋은 성과를 보이는 머신러닝 해결 방법은 약 31%의 상대적으로 낮은 정확도를 보인다 [3].

이와 같이, 인간과 인공지능의 차이가 뚜렷한 이유는 사전 지식의 차이 때문이다 [1]. 인간은 ARC 문제의 적은 예제만으로도 해당 문제의 패턴을 쉽게 파악할 수 있다. 이는 인간이 '객체', '색', '대칭', '복사', '행과 열' 등의 여러 사전 개념을 가지고 있기 때문이다. 그러므로 인공지능에게 이와 같은 사전 지식을 준다면 적은 데이터로 충분히 학습할 것으로 예상된다.



[그림 1] 서로 다른 네 가지의 ARC 문제 [4].

이러한 사전 지식 중 문제 유형에 대한 정보가 상당히 중요할 것으로 예상된다. 왜냐하면 인간 또한 문제를 풀기 위해서 해당 문제가 어떤 유형인지 먼저 판단하기 때문이다. 그렇기 때문에 본 논문에서는 ARC 문제 유형을 분류하는 학습 방법에 대한 연구를 진행했으며 추후 ARC 문제 해결 연구 혹은 더 나아가 일반 인공 지능(AGI) 연구에 도움이 되고자 한다.양질의 데이터 개수가 부족한 ARC 문제의 특성상 단순한 형태의 지도 학습을 이용하기에는 학습에 어려움이 클 것이며 양질의 데이터를 확보하는데 많은 비용이 들 것으로 판단했다. 이를 해결하기 위해 본 연구에서는 대조 학습 [5]을 활용해 주어진 데이터에서 중요한 특징들을 뽑아내도록 사전 학습을 시켰다. 이후 사전 학습된 모델의 파라미터를 활용해, 분류 학습을 수행하도록 했다.

2. 방 법

2.1. 표현 벡터 추출

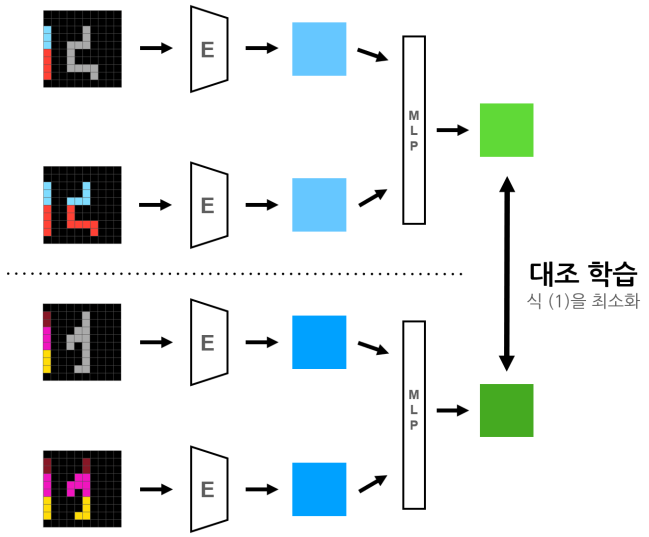
ARC에 대조학습을 적용시키기 위해, [그림 2]와 같이 ARC 문제의 입력과 출력 정보를 포함하는 표현 벡터를 추출했다. 입력 이미지와 출력 이미지는 섹션 2.3에서 설명할 VAE 인코더를 통과해 해당 이미지를 표현하는 벡터(🟦)로 변환된다. 이 벡터들을 하나의 벡터로 합치고, 이를 레이어에 통과시켜 ARC 문제의 입력과 출력을 정보를 담은 하나의 표현 벡터(🟩)를 추출했다.

2.2. 대조 학습

본 논문에서는 섹션 2.1의 과정을 통해 추출한 표현 벡터를 활용해 대조 학습을 진행했다. 이와 같은 학습을 통해 유사한 문제 유형의 벡터끼리는 잠재 공간 내에서 가깝게, 상이한 벡터끼리는 잠재 공간 내에서 멀어지도록 모델을 학습시켰다. 이를 위해서 코사인 유사도를 활용했다. 유사한 유형의 문제의 벡터들은 코사인 유사도를 최대화하고 반대의 경우는 코사인 유사도를 최소화하는 방식을 이용했다. [그림 3]의 임베딩 공간에서 노란색으로 색칠한 부분이 같은 문제끼리 비교한 것이고 유사도가 최대가 되는 부분이다. 그리고 나머지 하얀색 부분은 유사도가 최소가 되는 부분이다. 손실함수는 대칭 교차 엔트로피를 사용해 계산했다.

[1] 이 논문은 과학기술정보통신부의 재원으로 한국연구재단과 정보통신기획평가원의 지원을 받아 수행된 연구임 (RS-2023-00240062, RS-2023-00216011, 2019-0-01842)

본 논문에서 [그림 3]과 같이 자기 주도 학습 (Self Supervised Learning. SSL)과 지도 대조 학습 (Supervised Contrastive Learning. SCL) [6]등 2종류의 대조 학습을 이용했다.

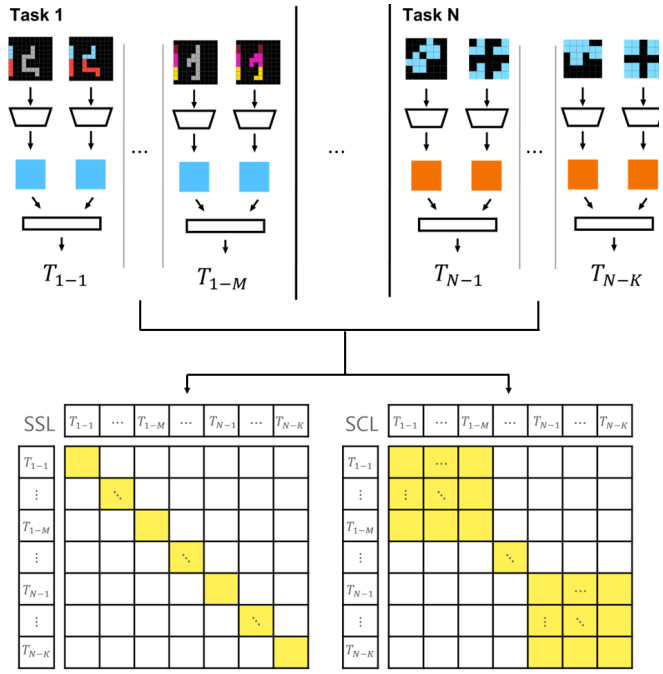


[그림 2] ARC 문제의 입력-출력 쌍에 대해 표현 벡터를 추출하고 대조 학습을 하는 과정을 나타낸 간단한 모식도.

2.2.1. 자기 주도 학습

자기 주도 학습방식의 대조학습은 [그림 3]의 SSL과 같이 배치 단위 학습시, 표현 벡터들이 서로 상이하게 끔 학습하는 방식이다. 해당 방식은 분류 클래스에 대한 라벨링이 없더라도 사용할 수 있는 방식이다.

2.2.2. 지도 대조 학습

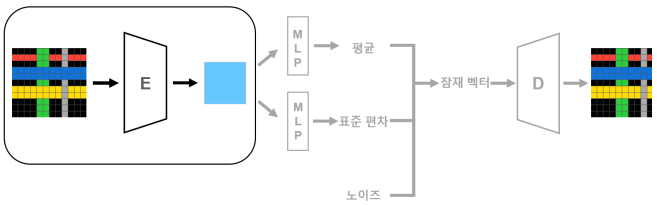


[그림 3] 각각의 문제를 나타내는 표현 벡터를 활용해 대조 학습을 진행하는 그림. 그림에서 T_{N-K}는 N번째 문제의 K번째 입력-출력 쌍을 나타내는 표현 벡터이다.

지도 대조 학습은 [그림 3]의 SCL과 같이 동일한 문제 유형의 예제들끼리는 서로 유사한 표현 벡터를 생성하도록 하고 다른 유형의 표현 벡터와는 다르게 생성하도록 학습시키는 방식이다. 해당 방식은 자기 주도 학습과 달리 사전에 분류 클래스에 대한 라벨링이 되어 있어야 한다.

2.3. 인코더

인코더는 Variational Auto Encoder(VAE)를 사용했다. VAE는 [그림 4]의 구조와 같이 주어진 데이터의 분포를 잘 표현할 수 있도록 표현 벡터를 학습한다. 이러한 특징은 대조 학습에 도움이 될 것으로 판단해 본 연구에서 VAE의 인코더를 사용하게 되었다.



[그림 4] VAE 구조. 인코더에서 입력 데이터를 압축하는 과정을 거치고 압축된 정보에 평균과 표준편차를 표현하는 값을 파라미터를 통해 출력한다. 그리고 예측한 표준편차에 가우시안 노이즈를 곱한 후, 평균을 더해 표현 벡터를 생성한다.

3. 실 험

본 연구에서는 ARC 데이터셋과 ConceptARC 데이터셋 [7]을 이용해 분류 실험을 진행했다. 이때, 모델 구조로 VAE 인코더를 사용했다. 학습을 위한 손실 함수는 SimCLR [5]와 이전 많은 연구에서 이용된 NT-Xent (정규화된 온도스케일 교차 엔트로피 손실)를 사용했다. 또한, 대조 학습을 적용하기 전과 후의 분류 성능을 평가하기 위해 KNN과 선형 탐색 방법을 사용했다.

3.1. 데이터 셋

ARC 데이터셋과 ConceptARC 데이터셋 [7]을 이용해 분류 실험을 진행했다. ARC 데이터 셋의 경우, 400개의 문제들이 서로 다른 문제 유형이라고 가정한 채 실험을 진행했으며 ConceptARC의 경우, 사전에 정의된 16가지의 분류 클래스를 기준으로 실험을 진행했다.

ConceptARC는 ARC 문제 유형 중, Above and Below', 'Center', 'Clean Up' 등, 16가지의 분류 클래스를 정의했고 이를 토대로 데이터셋을 만들었다.

3.2. 손실 함수

T_{N-K}가 N번째 문제의 K번째 입력-출력 쌍을 나타내는 표현 벡터이고, $sim(u,v) = u^T v / ||u|| \cdot ||v||$(코사인 유사도)일 때, 손실 함수는 다음과 같다.

$$l_{i,j} = -log \frac{exp(sim(T_{N-K}, T_{N-K}) / \tau)}{\sum_{k=1}^{2N} 1_{[k \neq i]} exp(sim(T_{N-K}, T_{N-K}) / \tau)} \quad (1)$$

여기서 $1_{[k \neq i]} \in \{0, 1\}$ 은 지시 함수로 $k \neq i$ 일 때 1의 값을 가진다. τ는 온도 파라미터이다. 최종 손실 값은 미니 배치에서 모든 양의 값을 가지는 쌍 (i, j), (j, i) 에 대해 계산된다.

3.3. 성능 평가 방법

3.3.1. 성능 지표

문제에 대한 정확도는 전체 예측 개수 중 올바르게 예측된 개수의 백분율을 의미하며, 0과 100 사이의 실수로 계산된다.

3.3.2. 분류 성능 평가 방법

KNN 분류기: KNN (K-Nearest Neighbors) 분류기는 KNN 알고리즘을 기반으로 한 분류 알고리즘이다. 주어진 데이터 포인트의 분류를 결정하기 위해 학습 데이터셋의 데이터 포인트들과 거리를 기반으로 가장 가까운 'K'개의 이웃을 찾는다. 그리고 이 K개의 이웃 중 가장 많이 나타나는 클래스로 주어진 데이터 포인트를 분류한다.

선형 탐색: 선형 탐색 방법은 주로 사전 학습된 모델의 성능을 평가하는 방식으로 사용된다. 본 논문에서는 해당 방식을 이용해 대조 학습 적용 유무에 따른 분류 성능을 측정했다. 대조 학습을 적용한 경우, 먼저 대조 학습 모델 구조 및 학습된 파라미터는 고정한다. 그 후, 마지막 레이어에 분류를 위한 선형 레이어를 추가해 해당 레이어만을 학습 시켜 성능을 평가한다. 이와 같은 방식으로 성능을 평가할 경우, 분류 작업을 위한 대조 학습 모델의 질을 파악할 수 있다. 대조 학습을 적용하지 않는 경우, 입력과 출력을 각각 VAE에 통과시켜 벡터를 얻고 결합시킨다. 그 후, 결합시킨 벡터를 분류 작업을 위한 선형 레이어를 통과시켜 결과를 얻는다. 해당 방식 또한, 성능을 평가하기 전, 학습과정을 통해 분류 작업을 위한 선형 레이어를 적절하게 학습시켜줘야 한다.

3.4. 실험 결과 및 분석

입력-출력 쌍의 표현벡터에 대조 학습을 사용했을 때 (🟩)와 사용하지 않았을 때 (🟦) 의 분류 결과를 비교했다. 선형과 KNN의 두 가지 분류 방법을 사용했다.

[표 1] ARC 데이터셋과 ConceptARC 데이터셋에 대해서 대조학습 유무에 따른 정확도를 KNN과 선형 탐색 방식을 이용해 평가한 결과표.

		CL 적용 X (🟦)	CL 적용 O (🟩)	
데이터 셋	분류기		SSL	SCL
ARC	KNN	17.31%	28.85%	32.93%
	선형	23.56%	38.94%	40.14%
ConceptARC	KNN	11.36%	19.89%	16.48%
	선형	20.11%	22.90%	22.90%

해당 실험을 통해, 데이터셋과는 별개로 대조 학습을 적용했을 때, KNN에서는 5%에서 많게는 15% 정도의 성능향상을 보였으며 선형 탐색에서는 최대 약 17%의 성능 향상을 보여주었다.

4. 결 론

본 연구에서는 ARC 문제 해결에 앞서, 인공지능의 사전 지식 활용과 사전 지식의 중요성에 대해서 언급했고 이 중에서 문제 유형 정보가 중요한 사전 지식이 될 수 있다고 판단했다.

현재, ConceptARC 이외에 ARC 문제를 분류한 데이터 셋이 없으며 이러한 데이터 셋 부족 문제와 문제 유형에 대한 적절한 표현 벡터 추출을 위해 사전에 대조 학습으로 학습하는 방식을 제시했다. 실험 결과를 통해, 대조 학습을 적용한 경우, 적용하지 않은 경우보다 ARC 문제를 유형별로 잘 분류할 수 있다는 사실을 확인했다. 추후 연구에서 해당 방식을 활용해 추출해낸 문제 유형을 사전 지식으로 사용한다면, 문제 해결에 큰 도움이 될 수 있을 것으로 기대된다.

참고문헌

[1] François Chollet, On the measure of intelligence. arXiv, 2019.
[2] Aysja Johnson et al., Fast and flexible: Human program induction in abstract reasoning tasks. arXiv, 2021.
[3] 현재 진행되고 있는 ARC 대회에 대한 정보, https://lab42.global/arcathon/.
[4] 심우창, et al., ARC 문제해결을 위한 프롬프트 엔지니어링의 가능성. KCC, 2023.
[5] Chen, Kornblith, et al., A simple framework for contrastive learning of visual representations. PMLR, 2020.
[6] Khosla, Prannay, et al., Supervised contrastive learning. NeurIPS, 2020.
[7] Arseny Moskvichev, TheConceptARCBenchmark: Evaluating Understanding and Generalization in the ARC Domain. arXiv, 2023.