조건부 변분 오토인코더를 활용한 대규모 언어 모델의 추론 성능 향상 기법

나준호°, 김기응
한국과학기술원
{jhn9803, kekim}@kaist.ac.kr

Enhancing reasoning performance of Large Language Models via Conditional Variational Autoencoder

요 약

대규모 언어 모델의 추론 능력을 향상시키기 위한 테스트 단계 연산 방법 중 기존의 탐색 기반 방법이나 검증 기반 기법은 계산 비용이 높거나 대규모 모델에 의존해 유연성이 제한되는 문제가 있다. 본 연구에서는 조건부 변분 오토인코더를 통해 문제에 대한 정답으로 이르는 잠재적 추론 패턴을 모델링하여 효율적인 검증 절차를 제공하는 새로운 방법을 제안한다. 제안된 방법은 질문과 풀이과정 데이터를 조건부 변분 오토인코더로 학습하여 추론 패턴을 잠재 공간에 축적한 후, 테스트 단계에서 대규모 언어 모델이 생성한 여러 후보 답안 중 학습된 추론 패턴과 가장 잘 부합하는 답변을 엔트로피 기반으로 선택한다. GSM8K와 MATH 수학 추론 데이터셋에서 대규모 언어 모델을 대상으로 한 실험 결과, 제안된 방법은 기존의 방법들과 대비하여 모든 조합에서 일관되게 상회하는 성능을 보였다. 이러한 결과는 조건부 변분 오토인코더를 활용한 효율적인 훈련 및 계산만으로 대규모 언어 모델의 추론 성능을 향상시킬 수 있음을 실증적으로 보여준다.

1. 서론

최근 대규모 언어 모델의 추론 성능을 높이기 위한 연구에서는 테스트 단계에서의 효율적인 연산 전략이 주목받고 있다. 특히, 모델이 단일 답안을 직접 생성하는 기존 방식 대신 여러 후보 답안이나 중간 추론 단계를 생성하고 이들 중 신뢰도 높은 응답을 선택하는 방식이 성능 향상에 효과적이라는 보고가 잇따르고 있다 [1, 2]. 대표적인 예로 탐색 기반 방식은 다양한 추론 경로를 전개해 최적의 해를 찾는 데 강점을 보이지만, 연산량 증가로 인해 지연 시간 및 비용 문제가 발생할 수 있다 [3]. 검증 기반 방식 역시 성능 개선에는 유리하나, 추가적인 대형 모델 학습이 요구되어 시스템 복잡성과 자원 부담이 크다 [4].

이러한 한계를 극복하기 위해 본 연구는 조건부 변분 오토인코더를 활용한 새로운 추론 전략을 제안한다. 구체적으로, 조건부 변분 오토인코더는 문제와 풀이과정 데이터를 통해 공통 추론 패턴을 잠재 공간에 학습하고 테스트 단계에서는 대규모 언어 모델이 생성한 후보 중 해당 잠재 패턴과 가장 부합하는 답변을 선택한다. 이로써 복잡한 추론 구조를 효과적으로 모델링하고, 기존 전략 대비 더 높은 정확도를 달성할 수 있다.

실험 결과, 제안된 방법은 수학 추론 벤치마크에서 다양한 대규모 언어 모델에 적용해도 일관된 성능 향상을 보였다. 이는 고비용의 추가 모델 없이도 테스트 단계의 연산 전략을 통해 성능을 개선할 수 있음을 시사하며, 대규모 언어 모델 효율성 향상에 기여하는 유의미한 방법론으로 평가된다.[1]

1) 이 논문은 2025년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. RS-2022-II220311, 일상적 물건들의 다접촉 로봇 조작을 위한 목적지향 강화학습 기술 개발)

2. 연구 배경

최근 복잡한 추론 과제에서는 문제 해결 과정이 단순히 입력과 출력의 대응을 넘어, 일정한 패턴이나 구조를 공유하는 잠재적 과정을 거친다는 점이 밝혀지고 있다 [5]. 예컨대, 유사한 문제 해결 전략이나 핵심 단계를 보이는 솔루션들은 t-SNE 기반 시각화에서 자연스럽게 군집을 이루는데, 이는 문제 해결에서 "어떻게 답에 도달하는가"를 특정 패턴의 잠재적 표현으로 해석할 수 있음을 시사한다.

변분 오토인코더 잠재 구조를 효과적으로 포착하기 위한 대표적인 접근법 중 하나가 변분 오토인코더 (Variational Autoencoder)이다 [6]. 변분 오토인코더는 관측 데이터 x와 잠재 변수 z 간의 확률적 관계 $p_\theta(z)$ 및 $p_\theta(x \mid z)$를 학습하고, 이를 통해 새로운 데이터를 생성하거나 x의 잠재 표현 z를 추론한다. 학습 목표는 $p_\theta(x)$를 최대화하는 것과 동등하며, 통상 다음의 ELBO (Evidence Lower BOund) 형태로 정리된다. $q_\phi(z \mid x)$는 실제 근사 후방분포(posterior distribution)을 대체하는 변분 분포를 의미한다.

$$\mathcal{L}(\theta, \phi; x) \;=\; \mathbb{E}_{z \sim q_\phi(z|x)}\big[\log p_\theta(x \mid z)\big] \;-\; \mathbb{D}_{\mathrm{KL}}\big[q_\phi(z \mid x) \,\|\, p(z)\big]$$

조건부 변분 오토인코더 문제 자체의 맥락이나 추가 정보를 통합해야 하는 경우에는 단순 변분 오토 인코더로는 한계가 존재한다. 이를 보완하기 위해 제안된 조건부 변분 오토인코더(Conditional Variaitional Autoencoder)는 입력 x 외에도 추가 조건 y를 사용해 학습함으로써, 보다 세밀하고 목표 지향적인 잠재 표현을 획득한다. 조건부 변분 오토 인코더에서는 다음과 같이 조건화

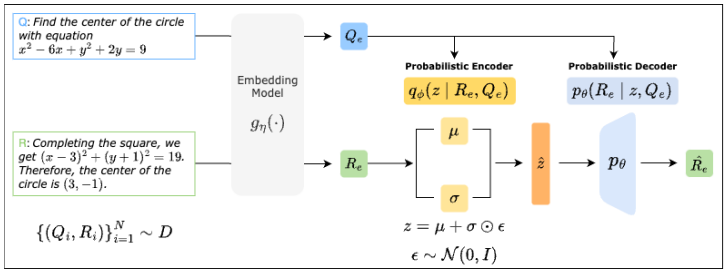

그림 1: 조건부 변분 오토인코더 학습.

된 ELBO를 최대화하도록 학습한다.

$$\mathcal{L}(\theta, \phi; x, y) = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{KL}}$$

$$\mathcal{L}_{\text{recon}} = \mathbb{E}_{q_\phi(z|x,y)} \left[\log p_\theta(x \mid z, y)\right]$$

$$\mathcal{L}_{\text{KL}} = -\mathbb{D}_{\text{KL}} \left(q_\phi(z \mid x, y) \,\|\, p(z \mid y)\right)$$

여기서 $p_\theta(z \mid y)$는 조건 y에 대한 잠재 변수 z의 사전(prior) 분포이다.

본 연구에서는 이러한 조건부 변분 오토인코더 접근을 복잡한 추론 문제에 적용함으로써, 문제 자체가 지니는 맥락 정보와 정답 도출에 필요한 핵심 단계를 잠재 공간에서 효율적으로 학습하고자 한다. 이를 통해 다양한 문제 상황에서 공통된 추론 패턴을 포착해, 적은 비용으로도 정확하고 견고한 해답을 제시할 수 있는 기반을 마련하는 것이 본 연구의 주요 배경이 된다.

표기 본 논문에서는 대규모 언어 모델의 추론 과제에서 다음과 같은 표기법을 사용한다. Q와 R은 각각 데이터셋 D에서 문제와 그에 대응되는 풀이과정을 나타내며 z는 질문에서 정답으로 이어지는 본질적인 추론 과정을 담고 있으며, 모델은 이를 잠재 변수로 활용해 Q 및 R 사이의 관계를 학습하고 재구성할 수 있다.

이때, 조건부 변분 오토인코더에서의 조건부 분포는 아래와 같이 표기한다.

$$p_\theta(R \mid z, Q), \quad q_\phi(z \mid R, Q).$$

3. 본론

조건부 변분 오토인코더 학습 조건부 변분 오토 인코더에 입력되기 전에, 질문 Q와 추론 과정 R은 각각 사전에 준비된 임베딩 모델 $g_\eta(\cdot)$을 거쳐 벡터로 포현된다. 이렇게 얻어진 임베딩 (Q_e, R_e)는 조건부 변분 오토인코더의 인코더와 디코더 네트워크의 입력으로 사용되어, 질문과 그 내재적 추론이 잠재 변수 모델링에 적합한 형태로 표현되도록 돕는다.

$$Q_e = g_\eta(Q)$$

$$R_e = g_\eta(R)$$

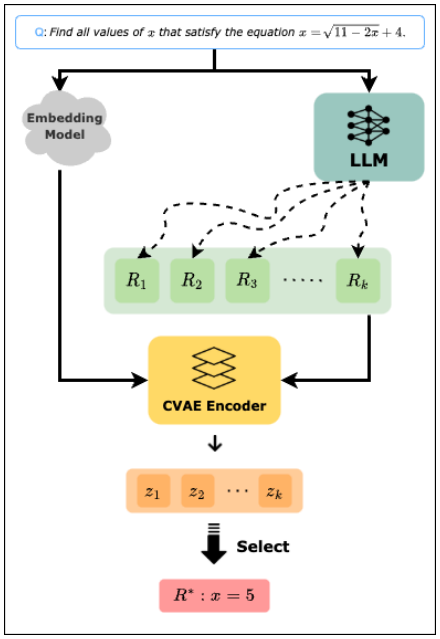

그림 2: 테스트 단계 연산.

인코더와 디코더 모델은 각각 q_ϕ와 p_θ라 하고, 잠재 변수 z에 대한 사전 분포는 표준 정규분포 $\mathcal{N}(0, I)$로 설정한다.

$$\mathcal{L}(\theta, \phi; R_e, Q_e) = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{KL}}$$

$$\mathcal{L}_{\text{recon}} = \mathbb{E}_{q_\phi(z|R_e,Q_e)} \left[\log p_\theta(R_e \mid z, Q_e)\right]$$

$$\mathcal{L}_{\text{KL}} = -\mathbb{D}_{\text{KL}} \left(q_\phi(z \mid R_e, Q_e) \,\|\, p(z)\right)$$

테스트 단계 연산 테스트 단계에서 본 연구는 학습된 조건부 변분 오토 인코더의 q_ϕ를 활용하여 대규모 언어 모델의 추론 과정을 향상시킨다. 구체적으로, 대규모 언어 모델 π_ψ는 주어진 문제 Q를 입력으로 받아 샘플링을 통해 k개의 후보 답안 $\{R_i\}_{i=1}^k$을 생성한다. 생성된 각 후보 답안과 문제 Q는 사전 학습된 임베딩 모델 $g_\eta(\cdot)$를 통해 임베딩된 후, 학습된 인코더 q_ϕ에 입력되어 각 후보 답안에 대응되는 추론 패턴 z_i를 획득한다.

$$\{R_i\}_{i=1}^k = \pi_\psi(\cdot \mid Q)$$

$$z_i = q_\phi(z \mid Q_e, R_{i,e})$$

이후, 학습된 인코더가 표준 정규분포를 형성하도록 하기 때문에, 각 후보의 추론 패턴으로부터 가우시안 분포의 엔트로피를 계산하고, 이때 엔트로피가 가장 작은 후보 답안을 최종 정답으로 선택한다. 즉, 낮은 엔트로피를 가질 수록 인코더가 더 확신하는 추론 경로임을 의미한다.

제안된 방법은 추론 단계에서 대규모 언어 모델을 추가로 사용할 필요가 없다는 점에서 기존의 테스트 단계 연산 기법과 차별화되며 계산 비용과 시간 측면에서 효율적이라는 이점을 가진다.

4. 실험

본 연구에서 제안한 방법을 활용하였을 때 대규모 언어 모델의 수학 추론 문제에서의 성능 향상을 관찰하기 위하여 다양한 수학 추론 데이터셋들을 이용하여 실험을 수행하였다. 또한 다양한 크기의 모델과 테스트 단계 연산 방법들과의 비교를 진행하였다.

실험 세팅 LLM의 추론 능력을 평가하기 위해 수학 추론 데이터셋인 GSM8K [7]와 MATH [8]를 이용하였다. 임베딩 모델로는 사전 학습된 오픈 소스 모델[2]을 사용하였으며 임베딩 차원은 1024로 지정하였고 잠재 변수 z의 차원으로는 256을 선택하였다. 대규모 언어 모델은 오픈 소스 모델인 SmolLM[3]과 Llama[4]를 사용하였다. 본 연구에서 제안한 조건부 변분 오토인코더 방법과 직접적인 비교를 위한 다른 베이스라인 방법들로는 Greedy Decoding, Top-k, Top-p 방식들을 선택하였다. temperature는 0.7로 고정하였고 Top-k의 k는 32, Top-p의 p는 0.9로 고정하였다. 평가 점수는 대규모 언어 모델이 생성한 답변과 실제 정답이 완전히 일치하였는지(Exact-Match)를 기준으로 하였다.

결과 표 1는 베이스라인 방법들과 본 연구에서 제안한 조건부 오토인코더(CVAE)를 사용하였을 때의 각 벤치마크에서의 성능 결과를 보여주고 있다. 표 1에서 확인할 수 있듯이 본 연구에서 제안한 방법을 사용하였을 때 모든 모델에서 다른 방법들보다 높은 추론 성능을 보임을 확인할 수 있다. 이러한 결과는 조건부 변분 오토인코더를 통해 문제를 풀기 위한 추론 패턴을 학습시킬 수 있으며 학습된 인코더를 이용하여 대규모 언어 모델의 답변 중 가장 신뢰도가 높은 추론 과정을 가진 답변을 선택함으로써 모델의 추론 성능을 향상시킬 수 있음을 보여준다. 더불어 본 연구에서 제안한 방식은 높은 계산 비용 또는 대규모 언어 모델에 대한 의존 없이도 대규모 언어 모델의 추론 능력을 향상시킬 수 있음을 확인할 수 있다.

5. 결론

본 연구에서는 대규모 언어 모델의 추론 능력을 향상하기 위한 테스트 단계 활용 방안과 관련된 방법을 탐구하였다. 조건부 변분 오토인코더는 문제와 풀이과정이 있는 데이터로 학습하여 문제를 풀기 위한 공통된 추론 패턴을 포착하고, 테스트 단계에서 이를 활용해 대규모 언어 모델이 생성한 답변들 중 잠재적 추론 구조와 가장 잘 부합하는 답변을 선택한다. 이 방법을 통해 조건부 변분 오토인코더를 활용한 효율적인 훈련 및 계산만으로도 테스트 단계 연산에서의 대규모 언어 모델의 추론 성능 향상을 이끌어낼 수 있음을 실험적으로 확인하였다.

2) huggingface.co/jinaai/jina-embeddings-v3
3) huggingface.co/HuggingFaceTB/SmolLM2-1.7B-Instruct
4) huggingface.co/meta-llama/Llama-3.2-1B-Instruct

Model	Method	GSM8K	MATH
SmolLM	Greedy Decoding	23.4	9.9
	Top-k	37.2	13.6
	Top-p	40.7	15.2
	R-CVAE	**43.2**	**18.4**
Llama	Greedy Decoding	23.4	10.8
	Top-k	37.2	18.2
	Top-p	39.0	19.8
	R-CVAE	**41.9**	**21.7**

표 1: 실험 결과. CVAE는 본 연구에서 제시한 테스트 단계 시 조건부 변분 오토 인코더를 활용한 방법을 의미함.

참고 문헌

[1] C. Snell, J. Lee, K. Xu, and A. Kumar, "Scaling llm test-time compute optimally can be more effective than scaling model parameters," *arXiv preprint arXiv:2408.03314*, 2024.

[2] X. Li, "A survey on llm test-time compute via search: Tasks, llm profiling, search algorithms, and relevant frameworks," *arXiv preprint arXiv:2501.10069*, 2025.

[3] D. Zhang, S. Zhoubian, Z. Hu, Y. Yue, Y. Dong, and J. Tang, "Rest-mcts*: Llm self-training via process reward guided tree search," *Advances in Neural Information Processing Systems*, vol. 37, pp. 64735–64772, 2024.

[4] A. Hosseini, X. Yuan, N. Malkin, A. Courville, A. Sordoni, and R. Agarwal, "V-star: Training verifiers for self-taught reasoners," *arXiv preprint arXiv:2402.06457*, 2024.

[5] Z. Xu, H. Wang, D. Bespalov, X. Wu, P. Stone, and Y. Qi, "Lars: Latent reasoning skills for chain-of-thought reasoning," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3624–3643, 2024.

[6] D. P. Kingma, M. Welling, *et al.*, "Auto-encoding variational bayes," 2013.

[7] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, *et al.*, "Training verifiers to solve math word problems," *arXiv preprint arXiv:2110.14168*, 2021.

[8] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt, "Measuring mathematical problem solving with the math dataset," *arXiv preprint arXiv:2103.03874*, 2021.