Deep Q-Networks를 활용한 Speculative Decoding의 적응형 임계값 최적화*

안태현[0], 신동군
성균관대학교 전자전기컴퓨터공학과
thahn1230@gmail.com, dongkun@skku.edu

Adaptive Threshold Optimization for Speculative Decoding via Deep Q-Networks

Taehyun An[0], Dongkun Shin
Department of Electrical and Computer Engineering, Sungkyunkwan University
thahn1230@gmail.com, dongkun@skku.edu

요 약

대규모 언어 모델(Large Language Model, LLM)의 효율적 추론은 최근 연구의 중요한 관심사이다. 특히 BiLD에서 제안한 Speculative Decoding 기법은 모델 추론의 속도와 정확도를 향상시키기 위해 fallback과 rollback 정책을 활용한다. 그러나 이러한 정책의 성능은 두 정책이 의존하는 임계값(threshold) 설정에 크게 좌우되며, 이를 수동으로 최적화하는 것은 어렵고 비효율적이다. 본 논문에서는 최근 강화학습 분야에서 뛰어난 성능을 보이는 Deep Q-Network(DQN) 알고리즘을 활용하여 fallback 및 rollback 정책의 임계값을 자동으로 최적화하는 시스템을 제안한다. 실험 결과, DQN을 활용하여 최적화된 시스템은 기존의 고정 임계값을 사용한 BiLD보다 더욱 향상된 성능을 보였으며, 추론 시간이 4.32% 감소하였다

1. 서 론

최근 Transformer 기반의 LLM은 자연어 처리 분야에서 획기적인 발전을 이루며, 다양한 분야에서 탁월한 성능을 보이고 있다. 그러나 LLM의 뛰어난 성능은 높은 계산 비용과 긴 추론 시간을 요구하여 실제 응용 환경에서의 활용을 제한하는 중요한 문제로 작용하고 있다. 이러한 배경에서, 효율적인 추론을 가능하게 하는 기술 개발은 중요한 연구 주제가 되고 있다

BiLD는 이러한 효율성 문제를 해결하기 위해 Speculative Decoding 기법을 제안하였다.[1] 이 기법은 fallback과 rollback이라는 두 가지 정책을 활용하여 추론 과정에서 모델의 속도와 정확도를 균형 있게 조정하는 것을 목표로 한다. 그러나 Speculative Decoding의 성능은 fallback 및 rollback 정책에서 사용되는 임계값(threshold)의 설정에 크게 좌우된다. 기존 BiLD 접근법은 Grid Search를 통해 고정된 임계값을 설정하며, 이는 입력 데이터의 특성에 따라 동적으로 적응하지 못하고 다양한 상황에서 일관된 성능을 보장하지 못하는 단점이 있다. 또한 최적의 임계값 설정을 위한 Grid Search 방식은 많은 시간과 계산 자원을 요구한다는 문제점이 존재한다.

본 논문에서는 이러한 한계를 극복하기 위해 최근 강화학습 분야에서 주목받는 Deep Q-Network(DQN) 알고리즘을 이용하여 fallback 및 rollback 정책의 임계값을 런타임에 자동으로 최적화하는 시스템을 제안한다. 이를 통해 인간의 개입을 최소화하면서 입력 데이터의 특성에 따라 적응적으로 최적의 임계값을 선택하여 성능을 극대화하는 것을 목표로 한다. 본 연구는 제안된 DQN 기반 자동 임계값 최적화 시스템의 설계, 구현, 성능 평가를 통해 기존의 수동 설정 대비 성능 향상을 입증하고, 보다 효율적이고 범용적인 LLM 추론 시스템 개발 가능성을 제시한다.

2. 배 경

* 이 논문은 삼성전자 미래기술육성센터의 지원을 받아 수행된 연구(과제번호 SRFC-IT2402-15)

2.1 Speculative Decoding과 BiLD

LLM은 일반적으로 auto-regressive 방식으로 텍스트를 생성하며, 이는 이전 단계에서 생성된 토큰을 바탕으로 다음 토큰을 순차적으로 예측하는 방식을 따른다. 그러나 auto-regressive 방식의 특성상 모델이 클수록 추론 시간이 길어지는 문제가 발생한다. 이를 해결하기 위해 제안된 Speculative Decoding은 작은 드래프트 모델(Draft Model)을 사용하여 빠르게 여러 토큰을 미리 예측하고, 더 큰 타겟 모델(Target Model)이 이를 검증하는 방식을 통해 추론 속도를 향상시킨다. SpecInfer 논문은 이러한 접근 방식을 소개하였으며, BiLD(Big Little Decoder)는 여기에 fallback과 rollback이라는 두 가지 정책을 추가하여 모델의 성능을 최적화하였다.[1][2][3] fallback 정책은 드래프트 모델의 예측이 부정확할 경우 타겟 모델의 예측을 직접 사용하는 정책이며, rollback 정책은 예측 오류가 발생할 경우 일정 단계 이전으로 되돌아가는 정책이다. 그러나 기존 BiLD 논문은 Grid Search 방식을 사용하여 고정된 임계값을 결정하기 때문에 다양한 입력 데이터 상황에서 동적으로 최적화된 성능을 보장하기 어렵다.

2.2 Deep Q-Network

Deep Q-Network(DQN)은 강화학습에서 활용되는 대표적인 알고리즘 중 하나로, Q-learning에 신경망을 결합하여 상태-행동 쌍에 대한 가치(Q-value)를 학습하는 방식이다.[4][5] DQN은 경험 리플레이(Experience Replay) 기법과 타겟 네트워크(Target Network)를 도입하여 학습의 안정성과 효율성을 높인다. DQN은 상태 공간에서 최적의 행동을 결정하기 위해 Q-value를 이용하며, 이를 통해 장기적인 기대 보상을 최대화하는 정책을 학습한다. DQN은 특히 이산화된 행동 공간 문제에 효과적으로 적용 가능하며, 다양한 응용 분야에서 뛰어난 성과를 보이고 있다.

2.3 강화학습 기반 임계값 최적화

최근 연구에서는 강화학습을 활용하여 모델의 성능을 동적으로 최적화하는 접근이 이루어지고 있다. 특히 ATD[6]에서는

Monte Carlo Tree Search(MCTS)를 사용하여 임계값 탐색 전략을 수립하고 있다.[6] MCTS는 탐색과 활용의 균형을 통해 임계값을 효과적으로 결정하지만, 런타임에 실시간으로 적응적인 결정을 수행하기에는 연산 부담이 큰 단점이 있다. 본 논문에서는 이러한 한계를 극복하기 위해 보다 가벼우면서도 실시간 적응이 가능한 DQN을 사용하여 fallback과 rollback 정책의 임계값을 런타임에서 동적으로 최적화하고자 한다.

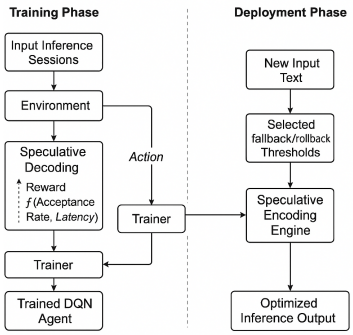

그림 1. DQN 강화학습 알고리즘 Overview

3. 본 론

본 논문에서는 LLM의 Speculative Decoding 과정에서 fallback과 rollback 정책의 임계값을 동적으로 최적화하기 위해 Deep Q-Network(DQN) 기반 강화학습 시스템을 제안한다. 시스템 설계의 주요 구성 요소는 다음과 같다.

3.1 상태 공간 (State Space)

상태 벡터 $s_t \in R^7$는 Speculative Decoding의 성능과 관련된 다음과 같은 요소를 포함한다:

$$s_t = [a_{avg}, v_{acc}, l_{avg}, f_{last}, r_{last}, l_{text}, p_{prog}]$$

각 요소들은 다음과 같다. 지금까지의 평균 수락 길이(a_{avg}), 누적된 값의 점수(v_{acc}), 평균 추론 지연 시간(l_{avg}), 이전 단계에서 사용한 fallback 임계값(f_{last}), 이전 단계에서 사용한 rollback 임계값(r_{last}), 입력된 텍스트의 길이(l_{text}), 현재 진행된 추론의 비율(p_{prog}). 이러한 상태 정보는 현재 추론 상황을 정확히 반영하며, 시스템이 더 나은 결정을 내리도록 돕는다

3.2 이산 행동 공간 (Discrete Action Space)

Fallback과 rollback 임계값을 효과적으로 관리하기 위해 연속적인 행동 공간을 다음과 같이 이산화하였다:

$$F = \{f_1, f_2, \cdots, f_5\} \quad where \quad f_i \in [0.1, 0.9]$$

$$R = \{r_1, r_2, \cdots, r_5\} \quad where \quad r_i \in [0.2, 1.0]$$

이로 인해 행동 공간은 총 25가지 조합으로 구성되며, 탐색 및 활용의 균형을 효과적으로 조절할 수 있다.

3.3 DQN 알고리즘

DQN은 강화학습의 한 종류로, 각 상태에서 어떤 행동이 가장 좋은지 학습하는 방식이다. DQN은 Q-네트워크라는 신경망을 사용해 각 상태와 행동 쌍의 가치를 계산한다. 이 가치(Q-value)는 특정 행동을 했을 때 앞으로 얻을 수 있는 보상을 나타낸다. 본 논문에서 사용하는 Q-네트워크는 다음과 같은 구조를 가진다:

◎ 입력 상태를 받아 첫 번째 hidden layer에서 처리(128개의 뉴런 사용)
◎ 두 번째 hidden layer에서 추가로 처리(128개의 뉴런 사용)
◎ 마지막으로 각 행동에 대한 Q-value를 출력

학습 과정에서는 실제 경험한 데이터(상태, 행동, 보상, 다음 상태)를 모아서 저장하고, 이를 다시 이용해 신경망을 훈련한다. 이렇게 하면 안정적이고 효율적인 학습이 가능해진다.

3.4 행동 변환 (Action Transformation)

Q-네트워크는 최적의 행동을 숫자(인덱스)로 반환하는데, 이를 실제 fallback/rollback 임계값을 바꾸는 규칙은 다음과 같다:

◎ fallback 값 : F [인덱스 / 5의 몫]
◎ rollback 값 : R [인덱스를 5로 나눈 나머지]

3.5 성능 평가 기준

본 시스템의 성능은 두 가지 주요 지표를 사용하여 평가한다:

◎ 평균 보상 (average reward): 시스템이 얼마나 좋은 결정을 내렸는지 평가하는 지표
◎ 수락률 (acceptance rate): 예측된 토큰이 타켓 모델에서 수락되는 비율로, 높을수록 좋다

본 논문의 시스템은 기존의 수작업이나 Grid Search 방식보다 더 빠르고 적응적으로 임계값을 결정하여, 전체적인 추론 시간을 줄이고 성능을 높이는 것을 목표로 한다.

4. 실 험

4.1 실험 환경

표 1. 평가 환경

Names	Configuration
OS	Ubuntu 22.04.6 LTS 64-bit
CPU	Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz
Memory	32GB
GPU	Nvidia Geforce RTX 4090

본 연구에서는 LLM 추론 최적화 성능을 검증하기 위해 다음과 같은 실험을 설계하였다:

◎ 모델 : Speculative Decoding을 지원하는 LLM(OPT family)
◎ 비교 방법:
  ◎ Single Model (Speculative Decoding 미적용, OPT-6.7B)
  ◎ Naive Speculative Decoding (Draft(OPT-125M)가 예측하고 모든 토큰을 Target(OPT-6.7B)이 검증)
  ◎ BiLD (고정된 fallback=0.103, rollback=4.697 사용)
  ◎ BiLD with RL (DQN 기반 adaptive threshold 적용)
◎ 측정 지표:
  ◎ Inference Latency(ms): 문장 생성 1회 소요 평균 시간

4.2 결과 (Results)

표 2는 각 방법에 대한 평균 Inference Latency를 나타낸다.

표 2. Inference Latency 측정

방법	Inference Latency (ms)
Single Model	208.13
Naive Speculative Decoding	124.30
BiLD (FB: 0.103, RB: 4.697)	101.24
BiLD w/ RL (본 연구)	**96.86**

◎ 데이터셋은 Wikitext2를 사용하였고, 실험은 Speculative Sampling 논문에서 제안한 아키텍쳐를 가지고 진행하였다.[7]
◎ Speculative Decoding 기법을 사용하면 Single Model 대비 약 40% 이상의 속도 향상을 달성할 수 있었다.
◎ BiLD 기법을 적용하면 추가적인 임계값 조정으로 더 빠른 추론이 가능해졌다.
◎ 특히, 본 연구에서 제안한 **BiLD w/ RL** 방법은 기존 BiLD 대비 약 **4.3% 추가 지연 시간 감소**를 달성하였다. 이는 런타임에 입력에 적용하는 임계값 설정이 고정값 대비 성능 이점을 가진다는 것을 의미한다.

4.3 Threshold 동적 변화 (Dynamic Threshold Behaviour)

그림 2는 BiLD w/ RL 실험에서 100회 테스트 동안 fallback과 rollback 임계값이 어떻게 동적으로 변화하는지를 보여준다.

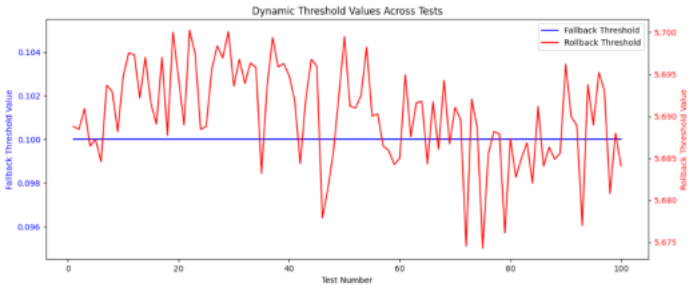

그림 2. fallback, rollback 임계값 변화 양상

◎ Fallback Threshold: 거의 0.1 부근의 고정되어 있음
◎ Rollback Threshold: 입력 데이터 특성에 따라 변동
이 결과는 학습된 DQN 에이전트가 fallback은 고정하고, rollback만 상황에 따라 세밀하게 조정한다는 것을 보여준다.

4.3.1 Fallback Threshold가 0.1로 고정된 이유

Speculative Decoding 특성 상, Draft Model이 생성한 다수의 토큰 후보를 Target Model이 검증하는 과정에서는 "빠르게 많이 Draft"하는 것이 이득이지만, 동시에 "Draft된 토큰이 Target에서 많이 Rejected 되면" 오히려 rollback/추가 검증 오버헤드가 커져서 전체 속도가 늦어진다.
◎ **Fallback Threshold**는 Draft가 실패했을 때 얼마나 쉽게 target 모델로 넘어갈지를 결정한다.
◎ 임계값이 높으면 Draft 결과를 신뢰하는 경향이 강해지고, 낮으면 바로 Target에 의존하려는 경향이 커진다.
◎ 0.1 수준으로 fallback threshold를 고정하면 **Draft 결과를 매우 엄격하게 검증**하고, 실패하면 빠르게 Target으로 fallback 하는 전략을 채택하는 것이다.
대부분의 Draft Token은 완전히 정확하지 않기 때문에, 낮은 fallback threshold가 불필요한 rollback/추가 검증 overhead를 줄이고, 결국 latency를 최소화하는 데 유리하기 때문이다. 특히 큰 모델일수록 이 경향은 더욱 두드러진다.
즉, 빠르게 fallback하여 Time Waste를 최소화하는 전략이 latency를 효과적으로 줄이는 것을 실험을 통해 확인할 수 있었고, 이로 인해 학습된 DQN은 fallback threshold를 낮은 값(0.1)으로 유지하는 방향으로 최적화 되었다.

5. 결 론

본 논문은 LLM의 효율적인 추론을 위해 Speculative Decoding 기법의 핵심 구성 요소인 fallback 및 rollback 임계값을 런타임에서 동적으로 최적화하는 DQN 기반 시스템을 제안한다. 기존 BiLD는 고정된 임계값을 사용하여 다양한 입력 상황에 적응하지 못하는 한계가 있었다. 이를 해결하기 위해 본 연구는 강화학습을 통해 입력 특성에 따라 최적의 임계값을 실시간으로 조정하는 방식을 제안하고, 이를 통해 평균 추론 지연 시간을 더욱 줄일 수 있음을 실험적으로 입증하였다.
실험 결과, 제안된 BiLD w/ RL 방법은 기존 BiLD 대비 약 4.3%의 추가적인 추론 시간 단축을 달성하였으며, fallback 임계값을 낮게 설정하고 rollback 임계값을 유연하게 조정하는 정책을 스스로 학습하는 경향을 보였다. 특히 fallback threshold가 낮게 유지된 것은 Speculative Decoding 특성상 빠른 실패 감지와 효율적인 추론이 전체 성능 향상에 기여한다는 점을 시사한다.
본 연구는 Speculative Decoding 최적화에 강화학습을 접목할 수 있는 가능성을 제시하였으며, 향후 다양한 모델 아키텍처나 추가적인 탐색 기법과 결합하여 더욱 범용적이고 강건한 LLM 추론 최적화 시스템으로 확장될 수 있을 것이다.

6. 향후 연구

향후 연구에서는 다양한 LLM 아키텍처에 대해 본 방법론의 일반화 가능성을 검증하고, Soft Actor-Critic(SAC)이나 Proximal Policy Optimization(PPO)과 같은 연속 행동 공간 강화학습 기법을 적용하여 더욱 정밀한 임계값 조정이 가능하도록 확장할 예정이다. 또한, 수락률과 지연 간의 다목적 최적화(multi-objective optimization)를 고려하여 보다 균형 잡힌 추론 성능을 달성하는 방향으로 연구를 이어갈 계획이다.

참 고 문 헌

[1] Kim, Sehoon, et al. "Speculative decoding with big little decoder."Advances in Neural Information Processing Systems36 (2023): 39236-39256.
[2] Leviathan, Yaniv, Matan Kalman, and Yossi Matias. "Fast inference from transformers via speculative decoding."International Conference on Machine Learning. PMLR, 2023.
[3] Miao, Xupeng, et al. "Specinfer: Accelerating large language model serving with tree-based speculative inference and verification."Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3. 2024.
[4] Mnih, Volodymyr, et al."Playing atari with deep reinforcement learning."arXiv preprint arXiv:1312.5602 (2013).
[5] Fan, Jianqing, et al. "A theoretical analysis of deep Q-learning."Learning for dynamics and control. PMLR, 2020.
[6] Liu, Xukun, et al. "Adaptive draft-verification for efficient large language model decoding."Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 39. No. 23. 2025.
[7] Chen, Charlie, et al. "Accelerating large language model decoding with speculative sampling."arXiv preprint arXiv:2302.01318(2023).