AI Day 12 정리 (AI 모델 활용)

생성일 @2025년 10월 26일 오후 6:54

📒 요약

1. AI모델 활용 “가속기 구동을 위한 100% 정수 연산 양자화”

1. 100% 정수 연산 양자화의 필요성

1-1. AI 가속기의 구동 특성

AI가속기는 정수 연산만 지원하거나 , 정수 연산에서 훨씬 효율적으로 동작함

- Google Coral Edge TPU
- ARM Ethos-U NPU 시리즈

1-2. 일반 양자화와 100% 정수 연산 양자화의 차이

- 일반적 양자화 : r이라는 실수 값을 q라는 정수로 표현하고 싶다.
- 무한한 실수를 모두 나타내는 것 불가능 - 유한한 관심 범위를 지정
- 8비트 정수 [0,255] 로 [r min , rmax] 범위의 실수를 표현하려면 2가지 파라미터 필요

- 스케일링 파라미터 S (실수)
- 오프셋 파라미터 Z (정수)

$$r = S(q - Z)$$

$$S = \frac{r_{max} - r_{min}}{2^8 - 1},\ Z = \left\lfloor -\frac{r_{min}}{S} \right\rceil_8$$

💡 문제는 스케일링 파라미터가 실수라는 것

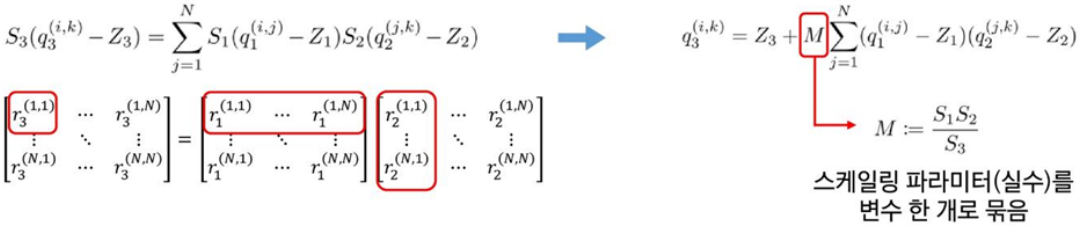
- 양자화 후, 정수 q만 사용해서 연산할 것 같지만,

실제로는 실수 값 r에 매핑하기 위해 숨어있는 스케일링 파라미터 S가 개입하여 실수 연산이 필요함 (AI 가속기 구동 불가)

2. 정수 연산 양자화

2-1. 행렬 곱 연산 양자화

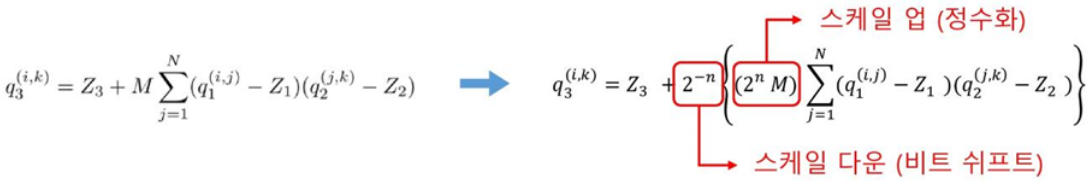
실수 행렬의 연산과정을 양자화 관점에서 유도



비트 쉬프팅

- 스케일링 파라미터를 뭉쳐서 M으로 나타냈으므로, M을 별도로 미리 양자화시키면 실수 연산 전멸
- 문제점 : M 이 실제론 (0,1)사이의 값 → 정수로 나타내기 어려움
- 해결책 : 비트 쉬프팅

임시로 스케일 업 → 양자화 및 연산 → 스케일 다운



2-2. 배치 정규화 계층 폴딩

배치 정규화

: 학습 중 정규화를 통해 입력 분포의 변화를 줄여 안정적인 학습을 도움.

$$\mathrm{BN}(x) = \gamma \cdot \frac{x - \mu_{\text{batch}}}{\sqrt{\sigma^2_{\text{batch}} + \epsilon}} + \beta$$

- 그러나 BN 계층에서 별도로 양자화를 진행하면 비효율적임
- 그래서 폴딩사용

배치 정규화 계층 폴딩

- BN 폴딩 : BN연산을 바로 앞의 Conv 계층 연산에 통합해서 함께 양자화

2-3 . 비전 트랜스 포머 연산 양자화

- Softmax
  : exponential연산 양자화, 정수 파트와 소수파트 분리하고 소수파트만 정수로 근사

- GELU
  : 1.702를 비트 쉬프팅을 통해 근사, sigmoid함수는 softmax 양자화 방식 그대로 차용
- LayerNorm
  : 루트 연산을 정수화하기 위해 비트 쉬프팅과 반복적 탐색을 통한 근사값 도출

2. AI 모델 활용 "테스트 타임 도메인 적용"

1. TTA

분포이동

- AI가 실제 쓰인 순간, 학습 데이터에서 경험하지 못했던 낯선 데이터에 맞닥뜨리는 현상
- AI 성능하락 원인

테스트 타임 도메인 적응 -TTA

- AI가 실제 쓰이는ㄴ 순간, 테스트 데이터의 분포를 파악하여 모델을 유연하게 적응시킴

2. TTA기법

2-1. CNN기반

TENT : 배치 정규화 계층을 테스트 데이터에 최적화

- 평균 분산 업데이트 : 테스트 데이터 기준으로 재계산, 즉각적응
- 어파인 파라미터 업데이트 : 엔트로피를 손실함수로 설정, 예측의 확신도를 높이는 방향으로 적응

SAR

: TENT보다 학습이 안정적이고 테스트 환경에서도 성능 유지

2-2. 비전언어모델 기반

TPT: 일관성있는 예측

- 6개의 평균 엔트로피를 로스로 세팅, 이를 최소화하도록 텍스트 프롬프트 업데이트

Prompt Align : 학습 데이터와 테스트 데이터를 직접 정렬

- 학습시 얻은 feature분포와 테스트 feature분포를 정렬시키는 방식\

PromptAlign : 학습 데이터와 테스트 데이터를 직접 정렬시키자.

- 추가적으로 엔트로피 로스를 함께 사용하여 안정적인 예측 유지
- 최종적으로 텍스트 프롬프트 + 이미지 프롬프트 모두 업데이트

3. AI 모델 활용 - “적응적 센싱”

- Overview
  - 초거대 AI의 근본 원리와 한계점은 무엇일까?
  - 현재 AI 학습방식과 인간인지체계의 차이점은 무엇일까?
  - 적응적 센싱과 분포 이동 억제의 개념은 무엇일까?

1. 초거대 AI의 근본 원리와 한계

1-1. 스케일링 법칙과 초거대 AI

- 초거대 AI : 파라미터 규모가 매우 큰 인공지능 모델
  - 파라미터: AI 뇌세포 개수 (파라미터가 많을수록 똑똑함)
- 대량의 데이터를 학습하여 여러 작업을 수행
- 데이터가 많고, 모델이 클수록 성능이 좋아짐(스케일링 법칙)
- 분포 이동의 문제 존재
  - 분포 이동: AI가 학습할 때 경험한 세상과, 실제로 쓰이는 순간의 세상이 달라지는 현상
  - Ex) 공부 열심히 했는데 시험 범위가 다름

- 일러스트 이미지만 학습한 AI 모델은 실제 고양이 이미지를 잘 구분하지 못함

- 분포 이동의 리스크를 줄이기 위해 규모로 보험을 들었음
  - 웹에서 방대한 데이터를 모으고,
  - 이 데이터를 추가 변형까지 시켜서 드문 경우로 흉내낼 수 있게 증강하고,
  - 그 지식을 담을 초거대 모델을 설계해서,
  - 고성능 대규모 컴퓨팅으로 오래 학습

- 스케일링 법칙
  - 모델크기↑ + 학습 데이터양↑ + 컴퓨팅 자원↑ → Test Loss(오류)↓

1-2. 스케일링 전략의 한계

- 이 세상에서 일어날 수 있는 모든 경우의 수를 학습 데이터로 준비하는 것은 불가능함
- AI는 LLM에서 피지컬 AI로 진화하고 있는데, 단순 학습데이터가 아닌 적응적 센싱이 필요

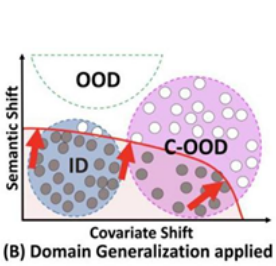
2. 적응적 센싱을 통한 분포 이동 억제

2-1. 발상의 전환: 인간의 인지체계

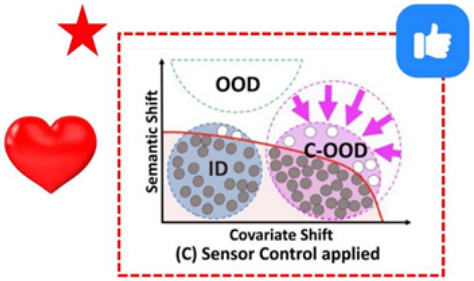
- 초거대 AI는 모든 문제를 '맨 땅에 헤딩'으로 해결하려함(ex. 눈이 나쁘면 흐린글씨를 5000개 학습)
- 인간은 눈이 나쁘면 안경을 맞춤

2-2. 적응적 센싱: AI를 위한 안경

- 뇌 = 모델, 센서 = 감각기관 ⇒ 뇌만 거대화하지 말고, 감각기관을 고도화하자!
- `적응적 센싱`: 환경에 맞춰서 **센서(카메라)**를 조절해서 좋은 데이터만 받음
- 분포 이동 억제: 모델 일반화 VS 적응적 센싱
  - 모델 일반화 → 모델 지식 영역을 확장하여 분포 이동 억제
  - 적응적 센싱 → 센서 제어를 통해 다양한 데이터를 모델이 아는 영역 안으로 넣어줌으로써 분포 이동 억제



[그림2-3] 모델일반화 방식



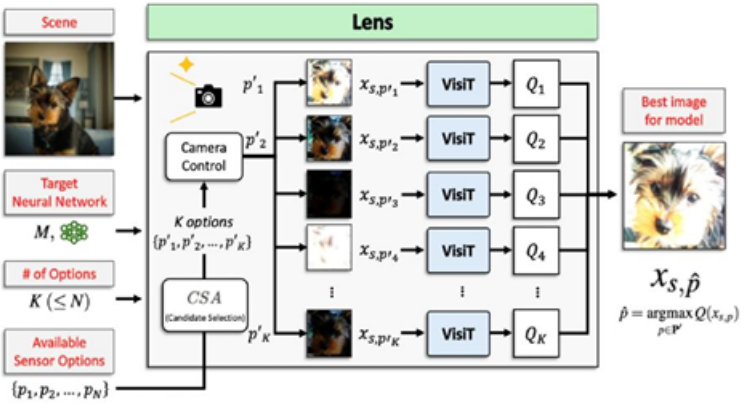
[그림2-4] 적응적 센서 방식

2-3. 적응적 센싱을 위한 데이터셋

- 같은 장면을 카메라 센서 파라미터와 조도를 변화시키며 캡쳐
  - ImageNet-ES (Luminus): TV스크린에 띄운 이미지를 카메라로 촬영(발광체)
  - ImageNet-ED (Diverse): 종이에 프린트된 이미지를 카메라로 촬영

2-4. 적응적 센싱 기법: Lens

- 센서 파라미터 후보 선택 → 이미지 캡쳐 → 비전 테스트 → 최적의 파라미터 선택



2-5. 성능 효과

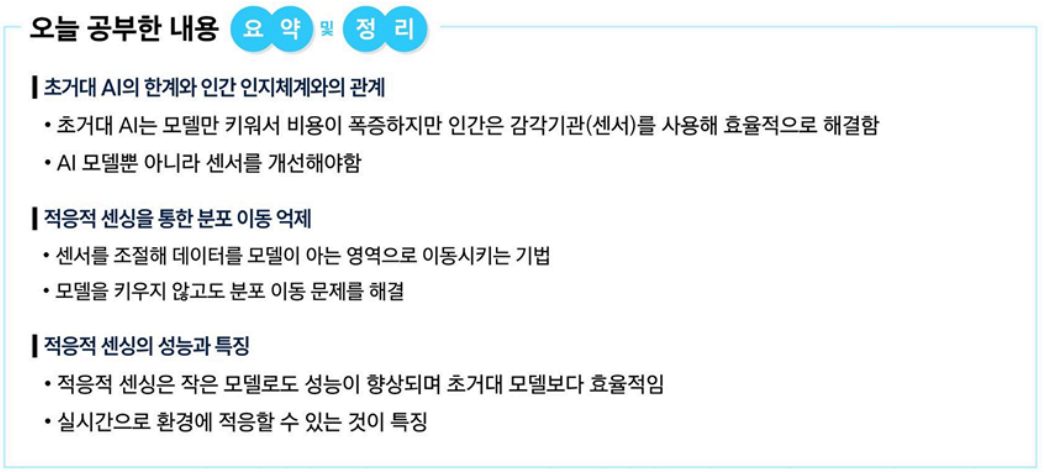
- Lens: 센서 제어로 모델이 보기 좋은 이미지를 만듦 (효과가 좋음)
- AE: 사람이 보기 좋은 이미지를 만들기 위한 제어 기법

→ AE 모델이 Lens보다 모델 파라미터 50배/학습 데이터 2,000,000배 많은데 **Lens 정확도가 4~10%** 높음

⇒ 모델뿐만이 아니라 센서도 중요하다

2-6. 특징과 함의

- 사람에게 보기 좋은 이미지와 AI 모델이 좋아하는 이미지는 다르다
- 최적의 센서 파라미터는 모델/환경/피사체에 따라 달라짐(적응적 센싱의 필요성)



4. AI 모델 활용 - "응용 분야 전문 지식을 활용한 모델 설계"

- Overview
  - 우리가 배운 AI 모델을 특정 도메인에 특화시키는 방법
  - AI 전문가 홀 설계 vs 도메인 전문가의 협업의 질적 차이
  - 경량화 기법 없이 경량화 가능한 지의 여부

1. 도메인 전문지식의 필요성

- 도메인 특화 AI 모델을 제작하고자 한다면 그 도메인의 `전문지식`을 이해해야 함

2. 의료 응용: 수면 의학을 위한 AI

2-1. 수면의학의 기본적 이해

- 수면다원검사: 10여종 이상의 센서를 부착하고 하룻밤 수면 → 전문 수면기사가 시계열 파형을 분석

2-2. 수면다원검사의 한계

- 인력 부족, 비용 부담, 정확도 문제
- 첫날 밤 효과(샘플의 부정확성): 낯선 장소에서 여러 개의 센서를 부착하고 잠
- 하룻 밤 문제(샘플 부족): 수면 상태는 매일 변함

3. 개발 사례 1: 수면다원검사 자동 채점 AI

3-1. 수면단계 진단 AI의 필요성

- 수면 단계
  - 기본 시간 단위: 30초
  - 5단계 분류: Wake, N1, N2, N3, REM - 단계별 생리학적 신호특징 가짐
- 자동진단 AI 개발의 도전성
  - 블랙박스 문제
    - 의료진단은 결과에 책임을 져야함
    - 의료진이 AI를 신뢰하려면 AI는 설명할 수 있어야 함
  - 멀티모달 신호의 통합 분석
    - 여러 종류의 생체신호에 대한 종합적 설명력을 탑재하기 위해서는 **복잡한 멀티모달 시계열 신호 분석 AI** 개발이 필요

3-2. 도메인 전문지식 주입 과정

1. 이미지 데이터 포멧 변환
   - 수면기사들은 생체신호 파형을 **'눈'**으로 보고 채점함(파형의 **시각적 모양**이 중요)

     ⇒ 데이터를 시계열에서 **이미지**로 변환
2. 비전 트랜스포머 기반 신경망 학습
   - 본 작업에서는 CNN보다 비전 트랜스포머가 적합함
3. 에폭 간 맥락 파악 및 결과 보장
   - 실제 수면은 30초 단위로 끊기기 보단 연속적 특징을 가짐

     ⇒ 에폭간 맥락을 파악하는 트랜스포머를 통해 최종 수면단계 진단

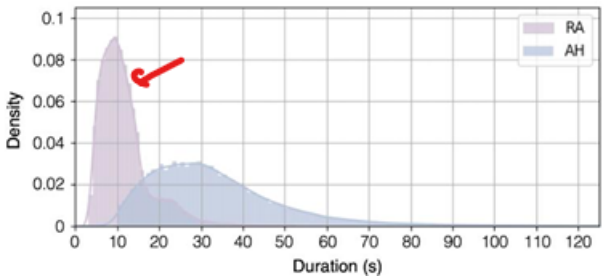
4. 개발 사례2: 비접촉식 수면 무호흡증 진단 AI

4-1. 비접촉식 수면 무호흡증 진단 AI의 필요성

- 수면 중 **미세 움직임 분석**으로 수면 신호 추출
- 카메라에서 직접 온디바이스 AI 구동 후 **원본 영상 삭제**

4-2. 도메인 전문지식 주입 과정

- 무호흡증(AH) 감지 → `호흡 각성(RA)` (숨 쉬려고 몸 뒤척이는거) 감지



⇒ 호흡 각성 감지 후 무호흡/저호흡 이벤트 역추적

- 호흡 각성 vs 자발적 각성(그냥 뒤척이는거)

  ⇒ 호흡 각성이 움직임 범위가 더 크고, 가슴 위쪽 움직임 강도가 더 큼

- 온디바이스 AI 파이프라인 설계
  - 30초마다 쪼개서 클립 구성 → 호흡 각성 감지 → 진단 결과만 보존(원본 영상 삭제)
  - 전체 수면의 호흡 각성 이벤트 빈도 계산 → 무호흡/저호흡 이벤트 빈도로 변환

4-3. 성능 평가

- 경량 모델, 엣지 디바이스, 빠른 실행
  - CPU만 써도 30초 이내 추론 가능
  - 감지 타깃 전환 전략을 통해 복잡한 영상 AI 과제를 온디바이스 AI로 해결

오늘 공부한 내용 요약 및 정리

도메인 전문지식 주입의 필요성

- 도메인 특화 AI 모델을 제작할 때는 해당 분야의 전문지식을 이해하고 데이터/모델 설계에 적극 반영해야 함
- 전문지식을 모델 구조에 주입하면 AI가 불필요한 시행착오 없이 효율적으로 학습할 수 있음.

수면다원검사 자동 채점 AI

- 수면 전문가가 30초 단위로 수동 채점하던 수면다원검사를 비전 트랜스포머 기반 AI로 자동화
- 시계열 데이터를 이미지로 변환하고, 에폭간 맥락을 파악하는 트랜스포머를 적용하여 전문가 수준의 정확도로 수면단계를 진단

오늘 공부한 내용 요약 및 정리

비접촉식 수면 무호흡증 진단 AI

- 적외선 영상만으로 수면 무호흡증을 진단하기 위해 무호흡 이벤트 대신 호흡 각성(RA) 감지로 문제를 전환
- 경량 비디오 모델(MoViNet)을 활용한 온디바이스 AI 파이프라인을 구축하여 엣지 디바이스에서도 30초 이내에 실시간 진단이 가능하도록 설계