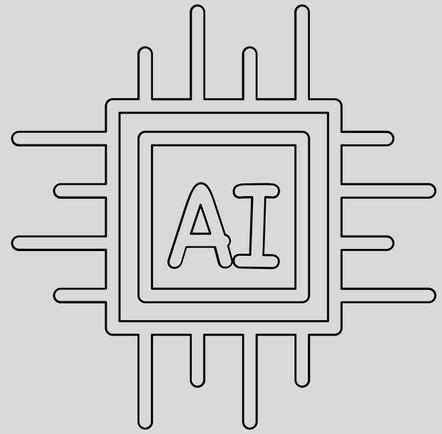
SCPC



2025 Samsung Collegiate Programming Challenge

code7monkey 이승현

Problem Definition

SAMSUNG

train_data	⋯▶	test에 비해 적은 60개의 샘플
파라미터 수 제한	⋯▶	실제 로드 되는 파라미터 수: 3B(30억개) 제한
선다형 질문	⋯▶	closed-ended 형태, 분류에 가까움

개발환경

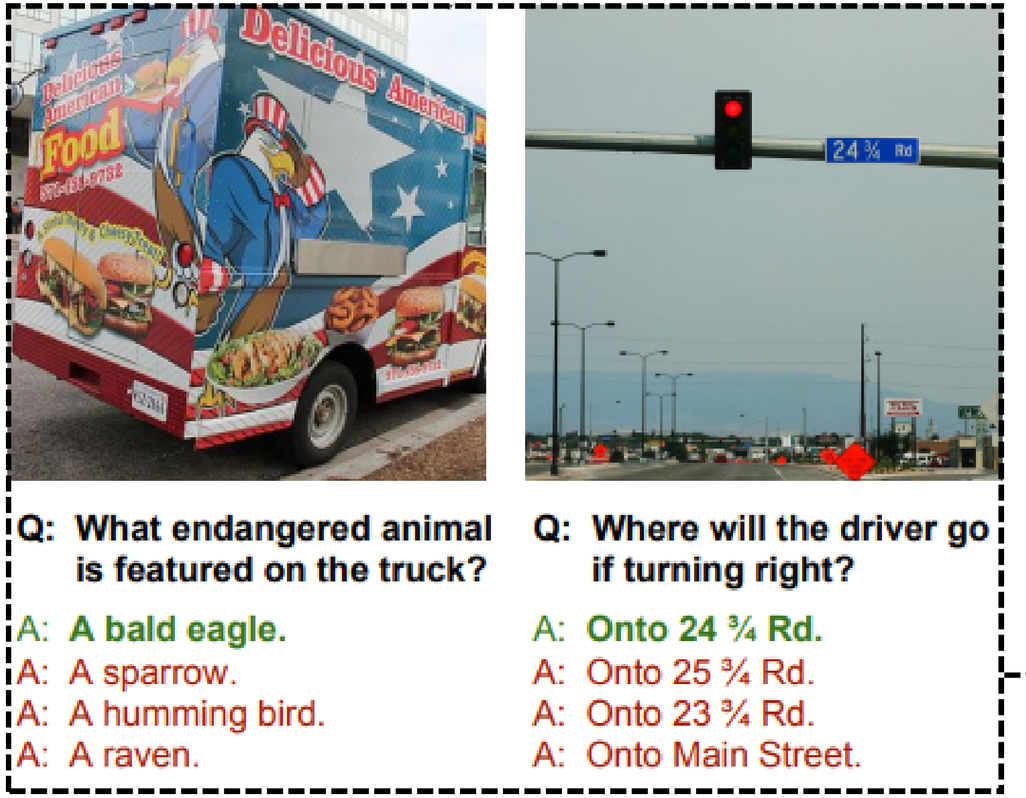
SAMSUNG

플랫폼	Google Colab (A100 40GB)	**NVIDIA-SMI**	550.54.15
OS	Linux-6.1.123 glibc 2.35	**Driver Version**	550.54.15
Python	3.11.13	**CUDA**	12.4
Pandas	2.2.2	**PyTorch**	2.6.0

Data Processing

SAMSUNG

- 외부데이터



7W 유형: What, Where, When, Who, Why, How, Which

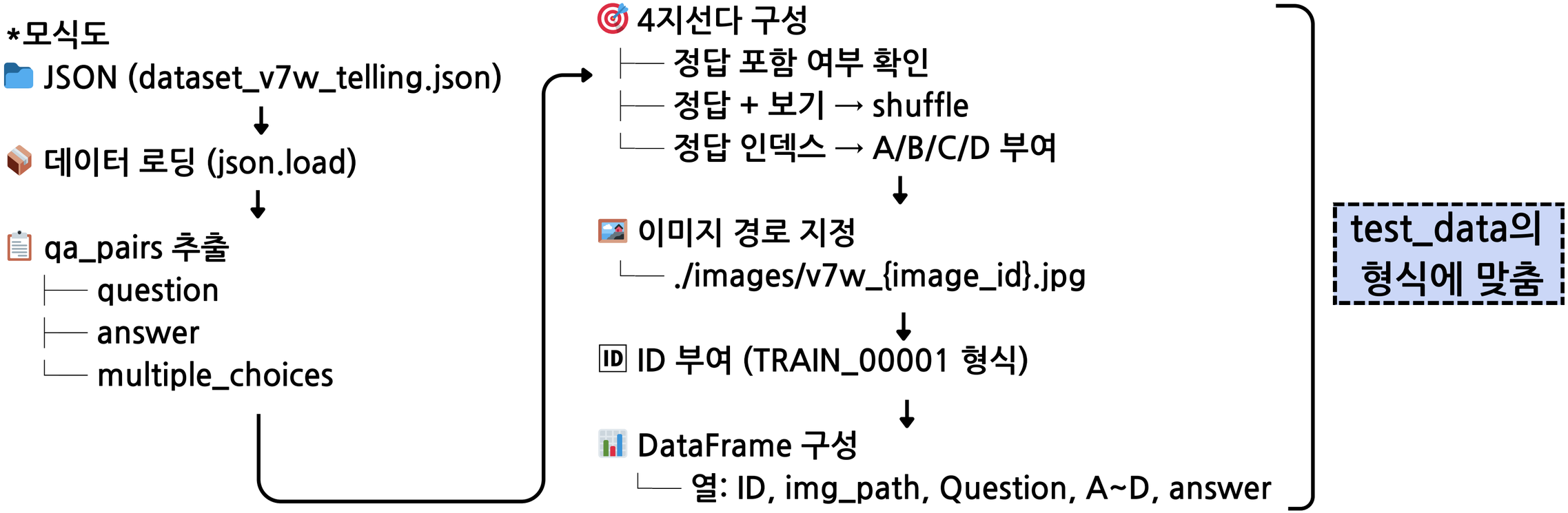
데이터 명	Visual7W: Grounded Question Answering in Images
라이선스	• MIT (Visual7W annotations) • CC BY 4.0 (coco이미지)
데이터 규모	• 이미지: 약 47,300장 (MS-COCO 기반) • 질문: 총 327,939개 • 선택지(보기): 각 질문당 4개 보기 포함
정답 유형	• 텍스트 정답 (선다형) • 위치 정답 (bounding box, 일부 질문에 한함)

Data Processing

SAMSUNG

📊제공된 train데이터는 미사용
→ 수집한 외부데이터와 형식이 맞지 않고, 수가 너무 적음

*모식도



Modeling

SAMSUNG

- 모델 선택

Models	#Trainable Params	VQAv2 test-dev	VQAv2 test-std
Open-ended generation models			
ALBEF (Li et al., 2021)	314M	75.84	76.04
BLIP (Li et al., 2022)	385M	78.25	78.32
OFA (Wang et al., 2022a)	930M	82.00	82.00
Flamingo80B (Alayrac et al., 2022)	10.6B	82.00	82.10
BLIP-2 ViT-g FlanT5$_{XL}$	1.2B	81.55	81.66
BLIP-2 ViT-g OPT$_{2.7B}$	1.2B	81.59	81.74
BLIP-2 ViT-g OPT$_{6.7B}$	1.2B	**82.19**	**82.30**
Closed-ended classification models			
VinVL	345M	76.52	76.60
SimVLM (Wang et al., 2021b)	~1.4B	80.03	80.34
CoCa (Yu et al., 2022)	2.1B	82.30	82.30
BEIT-3 (Wang et al., 2022b)	1.9B	**84.19**	**84.03**

Table 4. Comparison with state-of-the-art models fine-tuned for visual question answering.

2023년 1월,BLIP-2 논문에 소개된 VQAv2 데이터셋에 대한 SOTA 모델들.

Fine-tuning on VQAv2 (Visual Question Answering)

The detailed instructions can be found at `get_started_for_vqav2.md`.

initialized checkpoint	resolution	augmented data	test-dev	test-std	#params
beit3_base_patch16_224	480x480	-	77.65	-	228M
beit3_base_indomain_patch16_224	480x480	-	78.46	-	228M
beit3_large_patch16_224	480x480	-	81.85	-	683M
beit3_large_indomain_patch16_224	480x480	-	82.53	-	683M

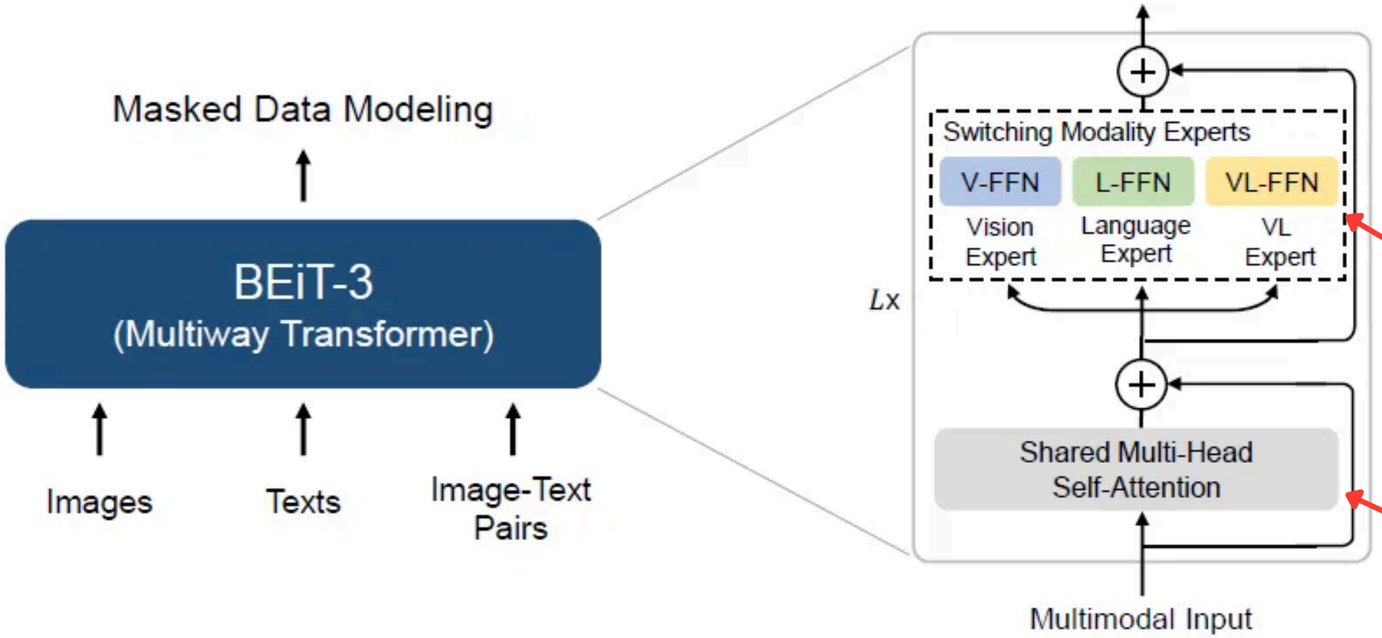
Beit-3

1. closed-ended models에서 최고성능
2. encoder-only구조
3. parms: 683M(0.68B)

Modeling

SAMSUNG

- 모델 선택



Pretraining Task
- Masked Data Modeling 수행

각 modality 전용 FFN으로
정보 정제 (Expert 방식)

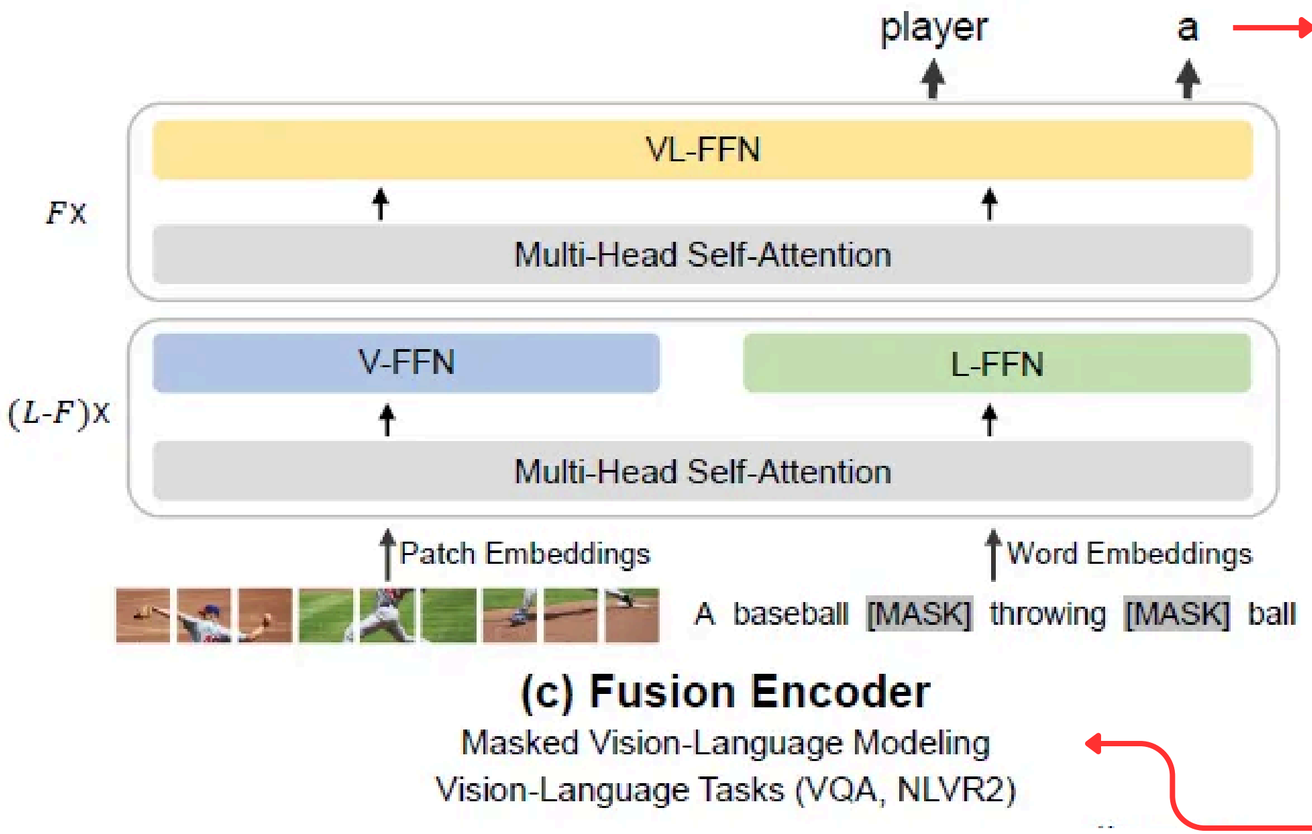
Shared attention으로 modality
간 alignment 학습

Figure 2: Overview of BEIT-3 pretraining. We perform masked data modeling on monomodal (i.e., images, and texts) and multimodal (i.e., image-text pairs) data with a shared Multiway Transformer as the backbone network.

Modeling

SAMSUNG

- 모델 선택



(c) Fusion Encoder

Masked Vision-Language Modeling

Vision-Language Tasks (VQA, NLVR2)

출력: 예측된 단어들 (예: player, a)

인코더 구성:

- 하위 계층 (L-F)(L-F)(L-F):
  - ├─V-FFN: 이미지 패치 전용
  - └─L-FFN: 텍스트 전용

>>동일한 Self-Attention 층을 공유하여 상호작용

- 상위 계층 FFF:
  - └─VL-FFN: 이미지와 텍스트 정보를 통합

>>Cross-modal self-attention을 통해 융합

입력 구성:

왼쪽: 이미지 → Patch Embeddings

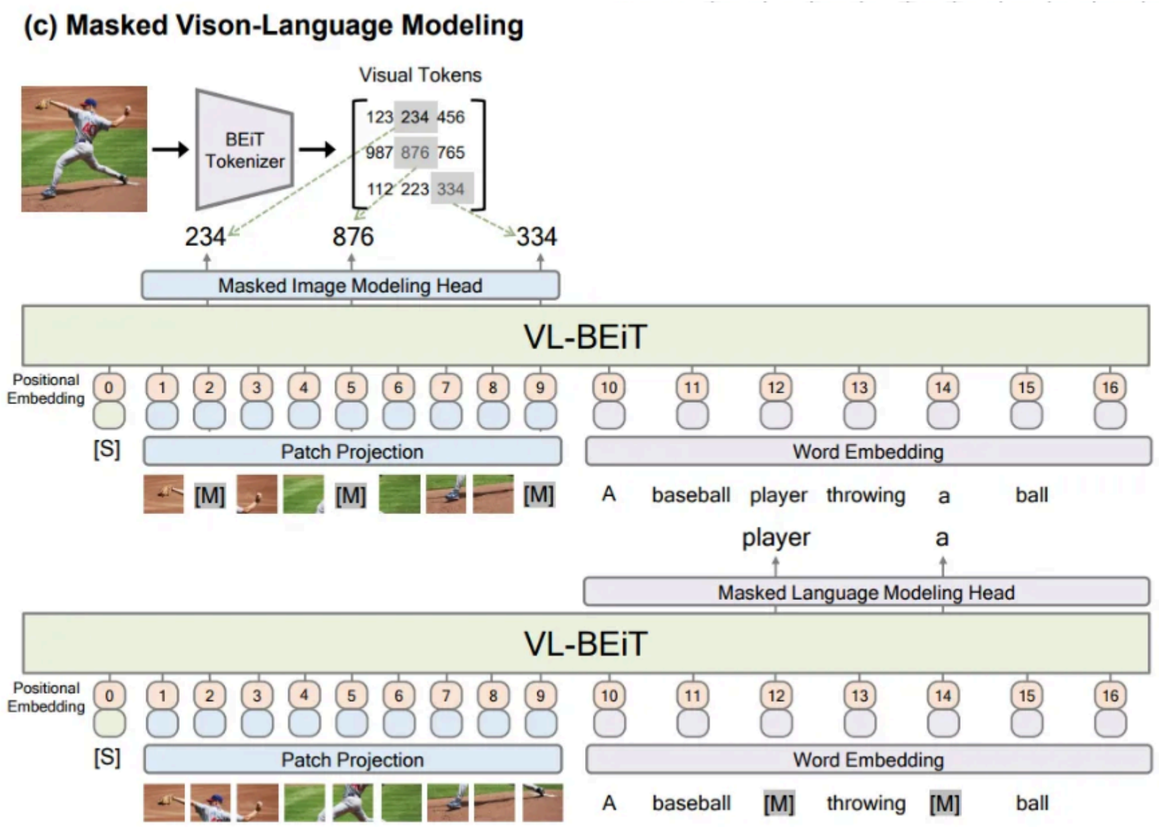
오른쪽: 텍스트 → Word Embeddings

(예: "A baseball [MASK] throwing [MASK] ball")

Modeling

SAMSUNG

- Masked vison-language modeling



🧩**상단 구조: Image 중심**

- BEiT Tokenizer를 통해 이미지가 시각 토큰으로 변환
- 일부 이미지 패치는 [M]으로 마스킹
- 텍스트도 포함되지만, 이 구조에서 핵심은 이미지 복원

마지막에 연결된 헤드:

- Masked Image Modeling Head → 이미지 패치 복원

📝**하단 구조: Language중심**

- 동일한 이미지와 텍스트 입력 사용
- 텍스트의 일부 단어가 [MASK] 처리됨
- 이 구조의 목적은 텍스트 복원

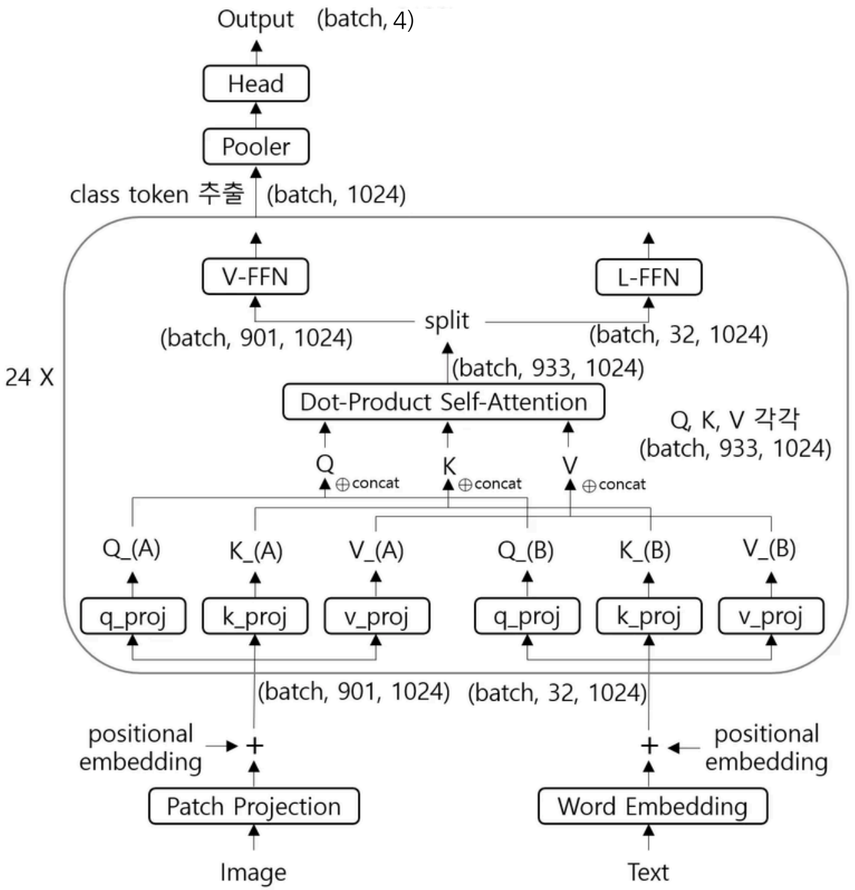
마지막에 연결된 헤드:

- Masked Language Modeling Head → 단어 복원

Modeling

SAMSUNG

- **모델 구조도**



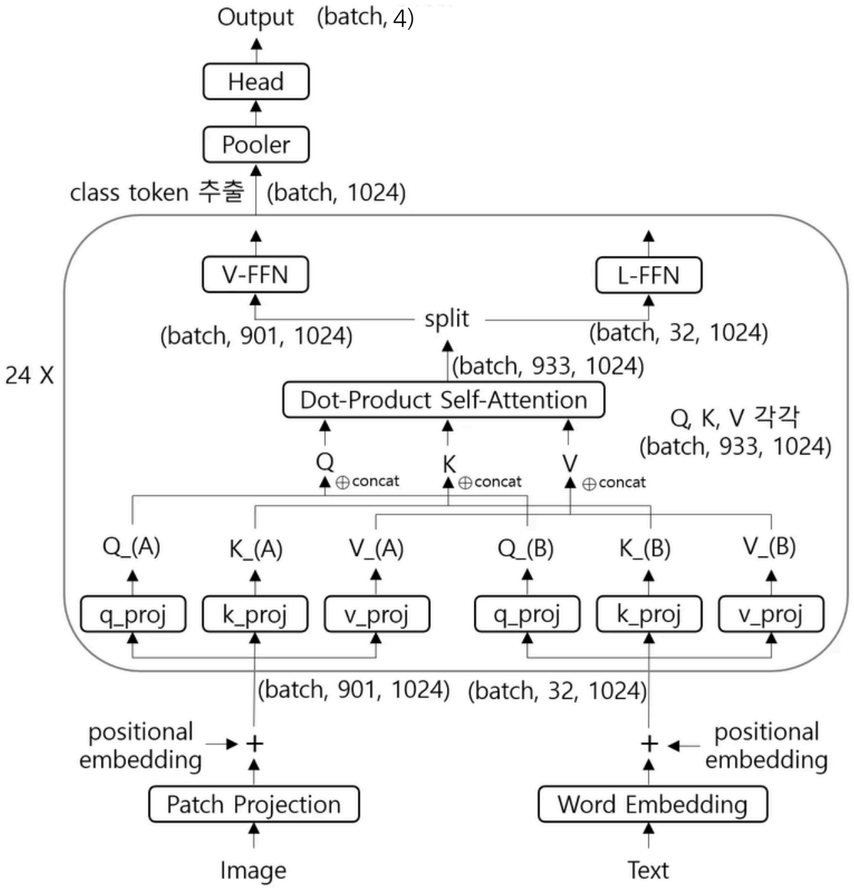
1. <u>임베딩 레이어</u>
   - Word Embedding
     - SentencePiece로 토큰화된 텍스트 → 1024-dim 벡터
   - Patch Projection
     - 입력 이미지 → 16×16 패치 → Conv2d → 1024-dim 벡터

2. <u>위치 정보 추가 (Positional Embedding)</u>
   - 텍스트 위치 인코딩 (Embedding A)
     - 입력 토큰 수: 최대 32개 (질문 + [CLS]/[SEP] 포함)
     - 테이블 크기: 33 × 1024 (인덱스 0-32)
   - 이미지 위치 인코딩 (Embedding B)
     - 패치 개수: 30×30 = 900 패치 + [CLS] 토큰 = 901
     - 테이블 크기: 902 × 1024 (인덱스 0-901)
   - 멀티웨이(PositionalEmbedding A/B) 구조
     - 서로 독립된 두 개의 임베딩 테이블
     - 텍스트·비전 모달리티별 위치 특징을 각기 학습

Modeling

SAMSUNG

- **모델 구조도**



3. <u>Multiway Transformer ×24 layers</u>
   - 공유 Self-Attention
     - 텍스트(A) · 비전(B) 모달리티 모두에 공통으로 적용 → 멀티모달 정렬
   - 모달리티별 Feed-Forward Network (FFN)
     - A(텍스트) / B(비전) 각각 독립 FFN
     - 구조: 1024 → 4096 → 1024, DropPath & Dropout 적용

4. <u>Pooler & 분류 헤드</u>
   - Pooler
     - CLS 토큰(1024-dim) → LayerNorm → Linear → Tanh → 요약 벡터 생성
   - Head (4-class)
     - Linear(1024→2048)
     - LayerNorm
     - GELU
     - Linear(2048→4) → Softmax

Modeling

SAMSUNG

- **모델 선택: hoon-bari/DACON_VQA**
  - MIT license
  - colab환경에 맞게 수정된 버전

📂 **dataset.py**
- CustomDataset 클래스 추가
- task2dataset에 "vqacustom" 항목 추가

🛠️ **utils.py**
- import torch._six → import torch
- pos_tokens = pos_tokens.float() 추가
- torch.distributed.barrier() 삭제

→ Single GPU 평가를 위한 수정

⚙️ **engine_for_finetuning.py**
- get_handler 함수 내

→ args.task == "vqacustom" 조건 분기 추가

🧠 **modeling_finetune.py**
- "vqacustom" 모델 구조 정의 추가
- num_classes → 라벨 수에 맞춰 수정

🚀 **run_beit3_finetuning.py**
- parser.add_argument에 --task 옵션에 "vqacustom" 추가
- args.eval 시 "vqacustom" 처리 로직 (367~370줄) 추가

Modeling

SAMSUNG

- **모델 구조 수정**
  - closed-ended에 맞게 추가 수정
  - 기타 추가 오류 수정

📂 **dataset.py**

- answer = i['answer'] → answer = normalize_word(i['answer'])

→ 정답 전처리 일관성 확보 (단어 정규화 적용)

🛠️ **utils.py**

- Checkpoint 불러올 때 weights_only=False 명시
- args.resume 경로에서 로드 시에도 weights_only=False 추가
- optimizer.load_state_dict → 주석 처리
- torch.load(...) 관련 들여쓰기 오류 수정

🧠 **modeling_finetune.py**

- 출력 클래스 수(num_classes)를 4개 고정
  - nn.Linear(embed_dim, num_classes) → nn.Linear(embed_dim, 4)
  - nn.Linear(embed_dim*2, num_classes) → nn.Linear(embed_dim*2, 4)
- self.head = nn.Linear(...) → 117번 줄 명시적으로 수정

Modeling

SAMSUNG

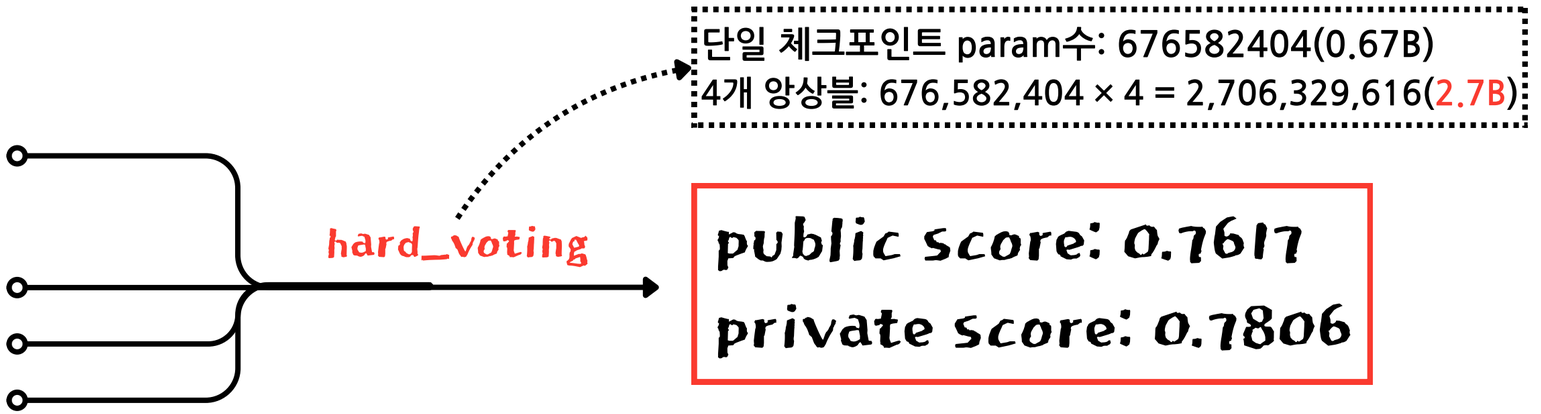
- **Params & Ensemble**

주요 파라미터:

- model: beit3_large_patch16_480_vqacustom
- epochs: 5, lr: 2e-5
- finetune: pretrained checkpoint (.pth)
- resume: 이전 epoch 체크포인트

체크포인트별 평가(public score):

- checkpoint-0.pth: 0.7403
- checkpoint-1.pth: 0.7583
- checkpoint-2.pth: 0.7506
- checkpoint-3.pth: 0.7574
- checkpoint-4.pth: 0.7600
- checkpoint-best.pth: 0.7591



References

SAMSUNG

Dataset:
We use the Visual7W dataset introduced by Zhu et al. (CVPR 2016).
The QA annotations are licensed under the MIT License, and the images are from the COCO dataset, licensed under CC BY 4.0.
https://ai.stanford.edu/~yukez/visual7w/

Pretrained model:
This project uses code from the [hoon-bari / DACON_VQA] repository, which is licensed under the MIT License.
MIT License
Copyright (c) 2023 거의있다
https://github.com/hoon-bari/DACON_VQA

Papers:
[1] Zhu, Y., Groth, O., Bernstein, M., & Fei-Fei, L. (2016). Visual7W: Grounded question answering in images. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4995-5004.

[2] Bao, H., Dong, L., & Wei, F. (2022). Image as a foreign language: BEiT pretraining for all vision and vision-language tasks. arXiv preprint arXiv:2208.10442.

Thank You!