REVEAL: Retrieval-Augmented Visual-Language Pre-Training with Multi-Source Multimodal Knowledge Memory

CVPR 2023

Ziniu Hu[1], Ahmet Iscen[2], Chen Sun[2], Zirui Wang[2], Kai-Wei Chang[1], Yizhou Sun[1],
Cordelia Schmid[2], David A. Ross[2], Alireza Fathi[2]
[1]University of California, Los Angeles, [2]Google Research

2024.04.22

발제자:
윤예준

연구 배경

- 최신 large-scale model들은 상당한 양의 world knowledge을 저장 가능
  → 매우 큰 규모의 파라미터, 데이터, 연산 능력이 필요
  → 새로운 world knowledge를 업데이트하기 위해선 모델 재학습 필요

- NLP, Vision 분야에서는 위 문제의 대안으로 적은 리소스로 world knowledge를 활용할 수 있는 Retrieval-augmented model들이 주목 받고 있음
  → 주로 Single-modality backbone을 사용하여 Query, knowledge corpora의 모든 정보를 활용하지 않음

위 한계를 해결하는 end-to-end retrieval-augmented visual-language pre-training 방법 제안

연구 목표

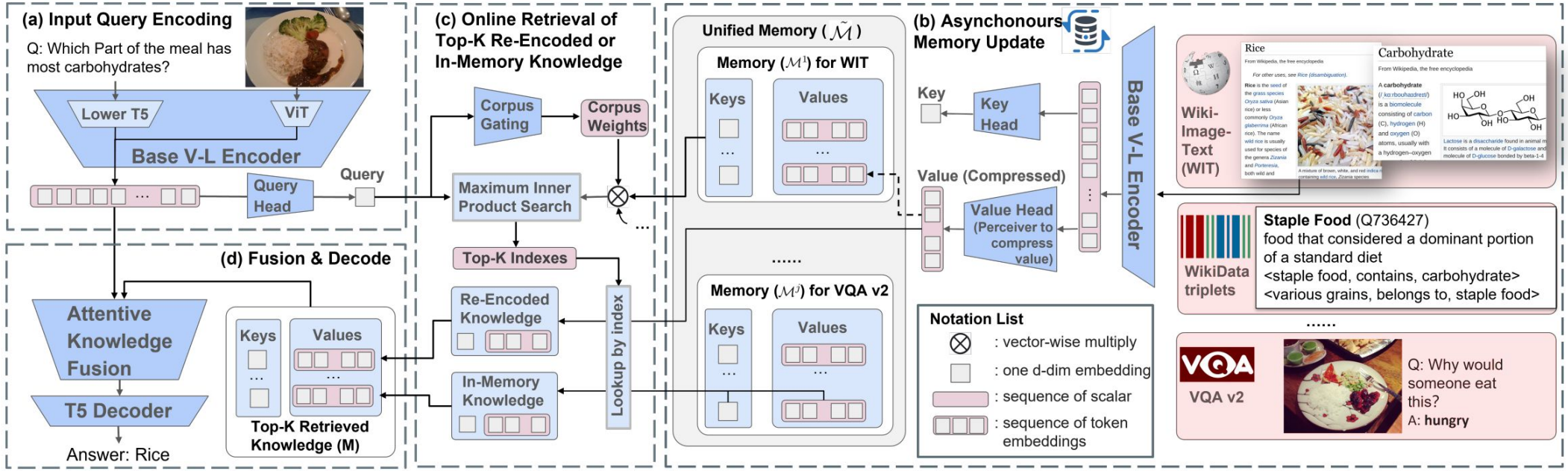
- End-to-end retrieval augmented vision-language pre-training을 통해 knowledge-intensive task를 푸는 방법 제안
  - Multi-source multimodal knowledge memory: encoding various sources of multimodal world knowledge

knowledge-intensive task: 외부 지식이 있어야 해결 가능한 작업

방법

- REVEAL(**Re**trieval-Augmented **Vi**sual **L**anguage Model)
  - 구성 요소: knowledge encoding, memory, retrieval and generation
  - Given an input query x, retrieve K possibly helpful entries $M = \{m_1, \cdots, m_k\}$ from the memory corpora $\mathcal{M}$
    - m is a memory entry containing the encoded single key embedding and a sequence of value embeddings
  - It uses the retrieved set M and the original input query x to generate output y

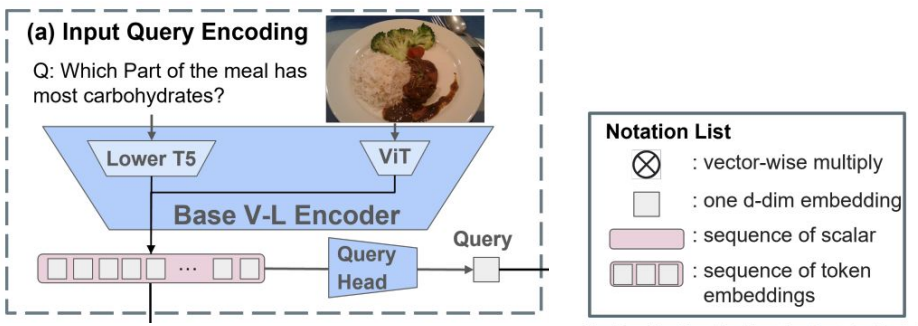
$$p(y \mid x) = \sum_{M \subset \tilde{\mathcal{M}}} \underbrace{p(M \mid x)}_{retrieval} \cdot \underbrace{p(y \mid x, M)}_{generation}. \qquad (1)$$



방법

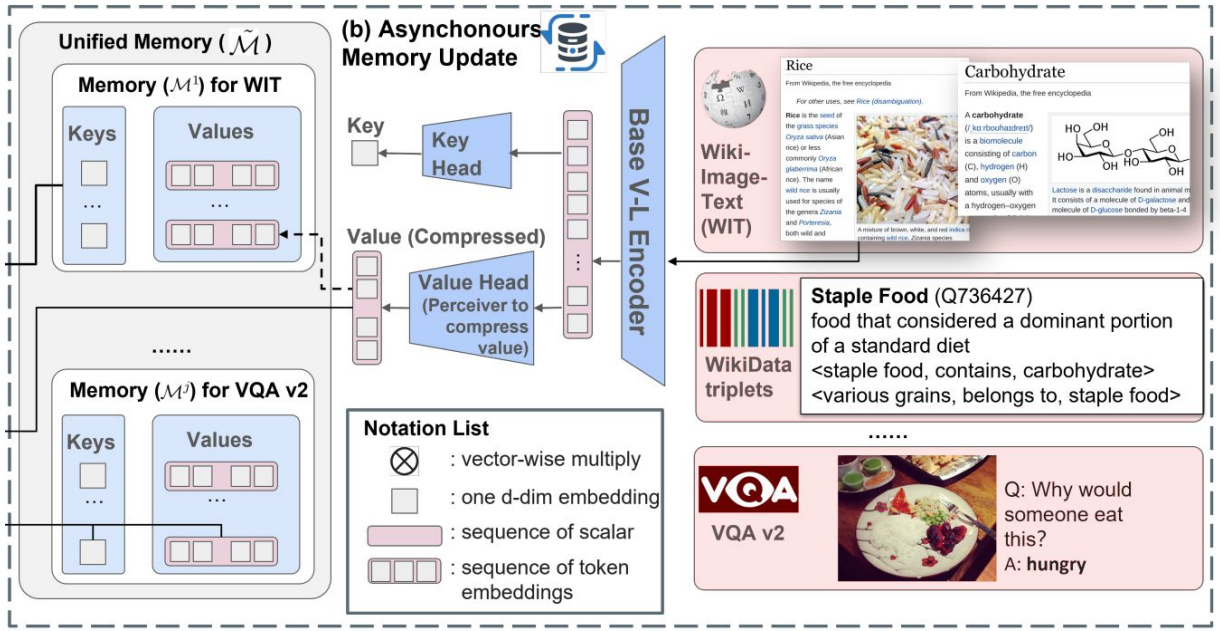
- Query Encoding
  - Base V-L Encoder
    - Vision encoder: Vision Transformer
    - Text encoder: T5 encoder의 token embedding layer를 포함한 l개의 하위 layers(Lower T5)
    - input: text-only, image-only or text-image pairs
    - output: 두 개의 모달리티를 concatenate

  - Query Head: T5 encoder의 마지막 l개의 layer(upper-layer)
    - input: Base V-L Encoder의 output
    - output: [CLS]토큰의 출력에 projection layer와 L2-normalization을 거친 d차원의 벡터



방법

- Memory
  - Key Head: T5 encoder의 마지막 l개의 layer(upper-layer)
  - 각 corpus는 $C^j = \{z_1^j, \ldots, z_N^j\}$로 정의, $z_i^j \in C^j$는 knowledge item(image-text pair, text only 등)
  - unified knowledge corpus: $\tilde{C} = C^1 \cup C^2 \cdots \cup C^S$, $|\tilde{C}| = S$
  - $\tilde{\mathcal{M}} = [\mathcal{M}^1, \ldots, \mathcal{M}^{|\tilde{C}|}]$: external knowledge corpora를 unified memory로 인코딩
  - knowledge item z_i는 key/value pair $m_i = (Emb_{key}(z_i), Emb_{value}(z_i))$로 인코딩
  - Value Head(ψ): perceiver architecture를 통해 knowledge item 압축
    - $Emb_{Value}(z) = \psi(b(z)) \in R^{c \times d}, c = 32$, b: Base V-L Encoder
  - $\tilde{\mathcal{M}}$ 은 1000 training steps마다 비동기적으로 업데이트됨



$$\mathcal{L}_{\text{decor}} = \sum_{i,j=1}^{K} \left\| \text{Covariance}\left(\psi\left(b(z_i)\right), \psi\left(b(z_j)\right)\right) \right\|_F^2$$

$$\mathcal{L}_{\text{align}} = \left| 1 - \frac{\sum_z \|\psi\left(b(z)\right)\|_2}{\sum_x \|b(x)\|_2} \right|$$

방법

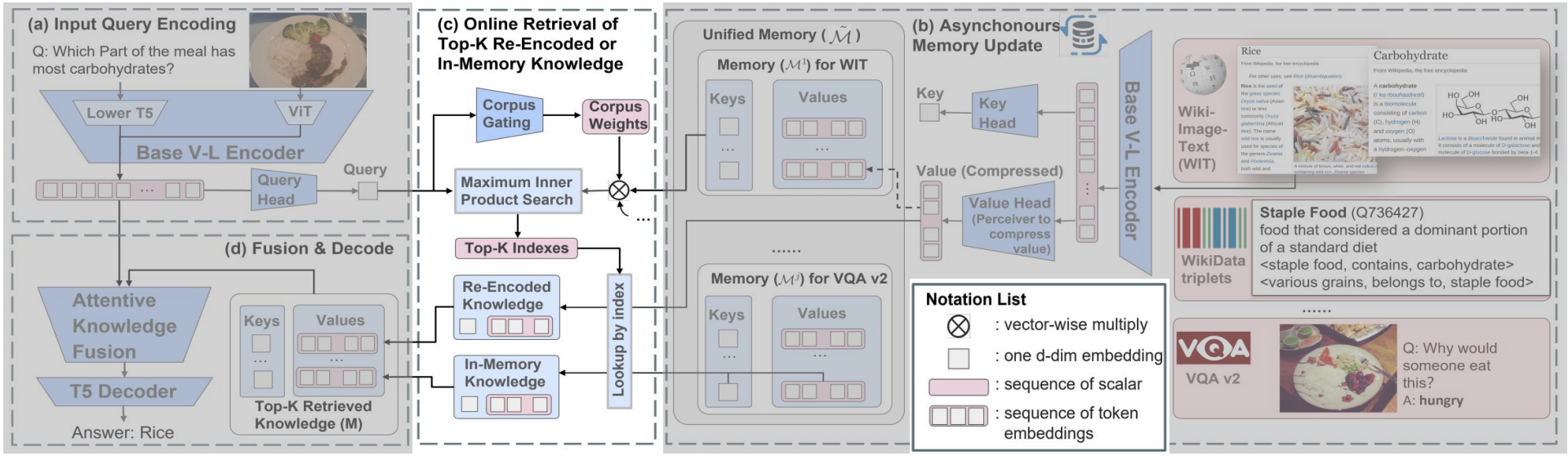
- Retrieval
  - query x가 주어지면 $\tilde{\mathcal{M}}$에서 top-K memory entries M을 찾는 것
  - 가장 적절한 knowledge sources를 선택하기 위해 각 memory corpus에서 검색할 확률을 모델링 하는 gating function을 학습
  - Encoder의 end-to-end training을 지원하기 위해 Top-K의 일부(10%) 다시 인코딩

$$p(m_i^j \mid x) = p(\mathcal{M}^j \mid x) \cdot p(m_i^j \mid x; \mathcal{M}^j) \quad (2)$$

$$= Gate_{\mathcal{M}^j}(x) \cdot \frac{\exp\left(Rel(x, m_i^j)/\tau\right)}{\sum_{m_k^j \in \mathcal{M}^j} \exp\left(Rel(x, m_k^j)/\tau\right)} \quad (3)$$

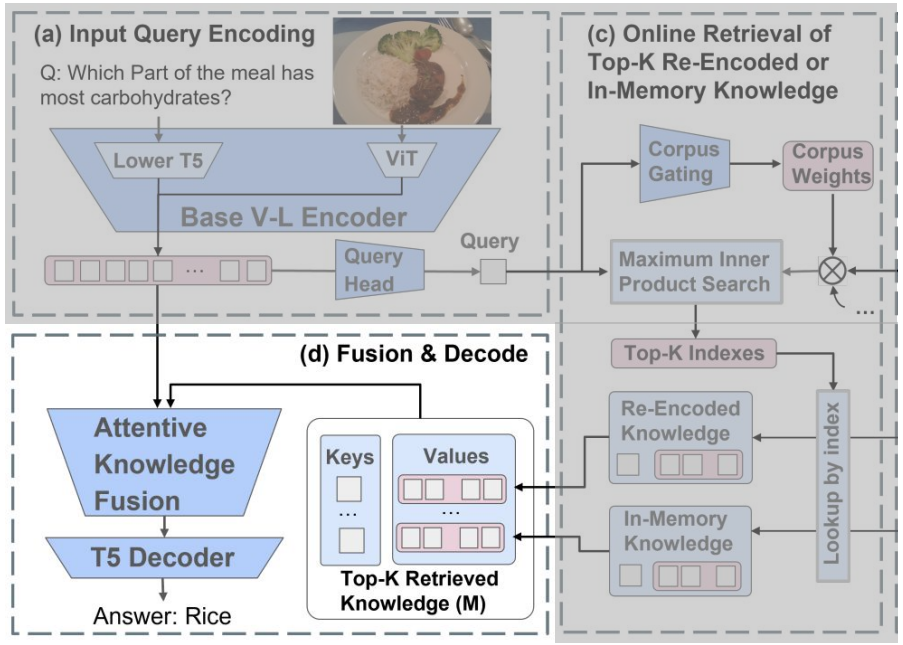
$$Gate_{\mathcal{M}^j}(x) = Softmax\left(W \cdot Emb_{Query}(x) + b\right)[j]$$

$$Rel\left(x, m_i^j\right) = Emb_{query}(x)^T \cdot Emb_{key}(z_i^j)$$



방법

- Generator
  - 검색된 Top-K개의 memory values는 query embedding과 concatenated
    $X = [b(x), \psi(b(z_1)), \ldots, \psi(b(z_K))] \in R^{(I+c\cdot K)\times d}$,($I$: query token 수, c : 압축된 token 수)
  - latent soft attention mask: query와 key간 latent representation 학습
  - Attentive Fusion Layer



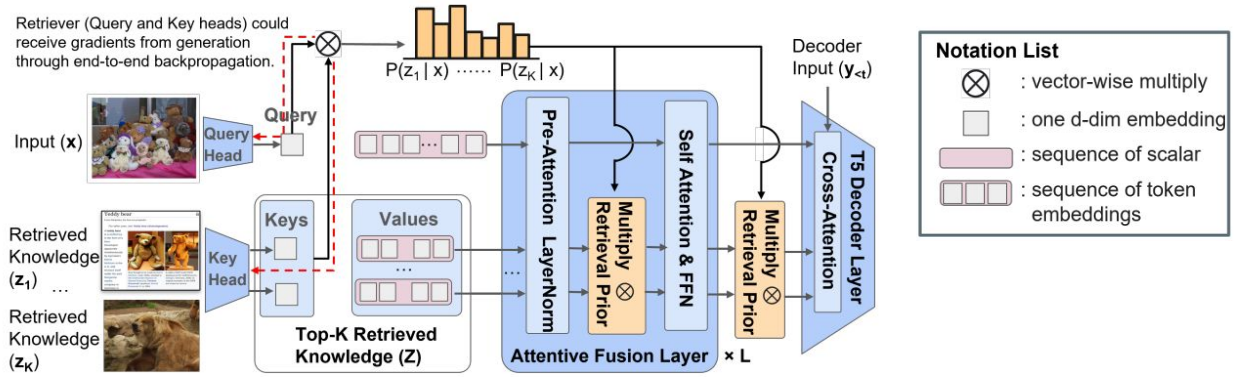


Figure 3. Detailed procedure of attentive knowledge fusion module. We inject retrieval probability as a prior to knowledge token embeddings, so the retriever can receive gradients via back-propagating over {self/cross}-attention part.

Pre-training

- Data
  - Web-Image-Text dataset[1]에서 노이즈 제거한 1.3 billion image caption pair 사용

- Main Objective
  - $\mathcal{D}$: pre-training Web-Image-Text dataset
  - $x = (img, txt)\ from\ \mathcal{D}$
  - 랜덤으로 sample의 prefix length T_p선택
  - input: $x < T_p$
  - output: $x \geq T_p$, autoregressively generate

$$\begin{aligned}\mathcal{L}_{\text{PrefixLM}} &= -\,\mathbb{E}_{x\sim\mathcal{D}}\left[\log p(x_{\geq T_p} \mid x_{<T_p})\right] \\ &= -\,\mathbb{E}_{x\sim\mathcal{D}}\Big[\sum_{i\geq T_p}\log p(x_i \mid x_{<i})\Big].\end{aligned} \tag{4}$$

  - Warm Starting the Model
    - end-to-end 사전학습을 하려면 retrieval의 능력이 필수적
    - 랜덤 가중치로 시작시 관련 없는 메모리를 반환하는 경우인 cold-start problem을 막기 위해 pseudo ground-truth knowledge를 갖춘 initial retrieval dataset 구성 제안
      - Wikipedia-Image-Text의 passage를 query 이미지와 조합하여 입력 query에 해당하는 pseudo ground-truth knowledge($\hat{z}$)로 활용

$$\mathcal{L}_{contra} = -\text{logSoftmax}(\text{Emb}_{\text{Query}}(x)^T \text{Emb}_{\text{Key}}(\hat{z}))$$

[1] Lit: Zero-shot transfer with locked-image text tuning.

구성 요소

- Knowledge Sources

Knowledge Source	Corpus Size	Type of Text	Avg. Text Length
WIT [37]	5,233,186	Wikipedia Passage	258
CC12M [5]	10,009,901	Alt-Text Caption	37
VQA-V2 [12]	123,287	Question Answer	111
WikiData [40]	4,947,397	Linearlized Triplets	326

- Architecture

Model Name	T5 Variant	Image Encoder	# params.	GFLOPs
REVEAL-Base	T5-Base	ViT-B/16	0.4B	120
REVEAL-Large	T5-Large	ViT-L/16	1.4B	528
REVEAL	T5-Large	ViT-g/14	2.1B	795

결과

- Knowledge-Based VQA: OK-VQA, A-OKVQA
  - Fine-tuning: Top-K 50, base V-L encoder 파라미터 고정
  - LLM에 의존하지 않고도 좋은 결과를 보임

Q: What days might I most commonly go to this building?
A: Sunday

VQA Model Name	Knowledge Sources	Accuracy (%)	Memory (GB)
MUTAN+AN [29]	Wikipedia + ConceptNet	27.8	-
ConceptBERT [11]	Wikipedia	33.7	-
KRISP [28]	Wikipedia + ConceptNet	38.4	-
Visual Retriever-Reader [27]	Google Search	39.2	-
MAVEx [46]	Wikipedia+ConceptNet+Google Images	39.4	-
KAT-Explicit [13]	Wikidata	44.3	1.5
PICa-Base [48]	Frozen GPT-3	43.3	350
PICa-Full [48]	Frozen GPT-3	48.0	350
KAT [13] (Single)	Wikidata + Frozen GPT-3	53.1	1.5 + 352+ 500
KAT [13] (Ensemble)	Wikidata + Frozen GPT-3	54.4	4.6 + 352 + 500
ReVIVE [24] (Single)	Wikidata + Frozen GPT-3	56.6	1.5 + 354 + 500
ReVIVE [24] (Ensemble)	Wikidata+Frozen GPT-3	58.0	4.6 + 354 + 500
REVEAL-Base	WIT + CC12M + Wikidata + VQA-2	55.2	0.8 + 7.5 + 744
REVEAL-Large	WIT + CC12M + Wikidata + VQA-2	58.0	2.8 + 10 + 993
REVEAL	WIT + CC12M + Wikidata + VQA-2	**59.1**	4.2 + 10 + 993

Table 3. **Visual Question Answering** results on OK-VQA, compared with existing methods that use different knowledge sources. For the memory cost, we assume all models use bfloat16. Green means on-device model parameters that are learnable, Blue means on-device memory of frozen model parameters, and Red means CPU/disk storage cost that are not involved in computation.

VQA Model Name	Accuracy (%)
ViLBERT [26]	30.6
LXMERT [38]	30.7
ClipCap [30]	30.9
KRISP [28]	33.7
GPV-2 [21]	48.6
REVEAL-Base	50.4
REVEAL-Large	51.5
REVEAL	**52.2**

Table 4. **Visual Question Answering** results on A-OKVQA.

결과

- Image Captioning: MSCOCO, NoCaps
  - CIDEr metric 기준 최신 baselines 보다 좋은 성능을 보임

Model Name	MSCOCO	NoCaps	# params.
Flamingo [2]	138.1	-	80B
VinVL [52]	140.9	105.1	0.4B
SimVLM [45]	143.3	112.2	1.5B
CoCa [49]	143.6	122.4	2.1B
REVEAL-Base	141.1	115.8	0.4B
REVEAL-Large	144.5	121.3	1.4B
REVEAL	**145.4**	**123.0**	2.1B

Table 5. **Image Captioning** results on MSCOCO (Karpathy-test split) and NoCaps (val set). Evaluated using the CIDEr metric.

결과

- Analyzing Effects of Key Model Components
  - Does utilizing multiple knowledge sources enhance performance?
    - 단일 source를 사용하는 것보다 여러 source를 사용하는 것이 더 효과적임

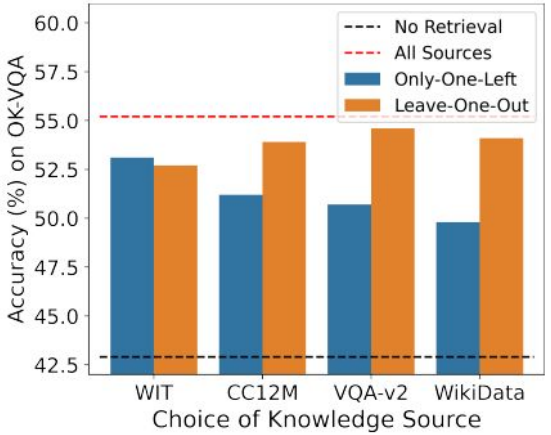


Figure 5. OKVQA Accuracy of REVEAL using 1) **Only-One-Left**: only use a single knowledge source; 2) **Leave-One-Out**: use all without this knowledge source.

Figure 6. OKVQA Accuracy of REVEAL using all **Pair of Knowledge Sources**. Results show that combining multiple sources could consistently improve performance.

결과

- Analyzing Effects of Key Model Components
  - Does the proposed attentive fusion surpass existing end-to-end retrieval training methods?
    - 기존 방법보다 attentive fusion이 효율적이고 효과적임을 보임
  - Can we add knowledge by only updating the memory without modifying model parameters?
    - 단순히 메모리를 업데이트함으로써 새로운 지식에 적응할 수 있음을 보임

Retrieval Method	Acc@10	Acc@100	OKVQA Acc.	GFLOPs
ALIGN [20] (fixed)	0.638	0.793	44.7	-
Attention Distill [17]	0.674	0.835	45.9	119
EMDR^2 [47]	0.691	0.869	46.5	561
Perplexity Distill [18]	0.704	0.886	46.7	561
Ours (Attentive Fusion)	0.726	0.894	47.3	120

Table 6. **Analysis of Retrieval Training Method**: We train REVEAL-Base (frozen generator, only train randomly initialized retriever) to retrieve from the WIT dataset (only text passage without image), and show the retrieval accuracy at the first 10 or 100 results, as well as fine-tuned OKVQA accuracy.

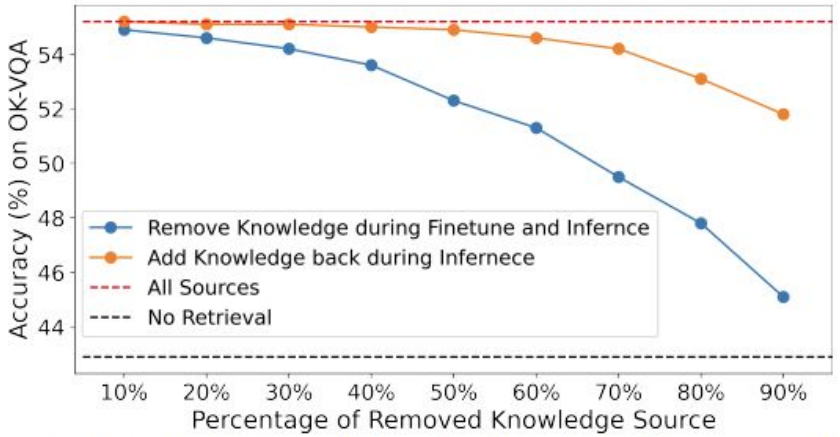


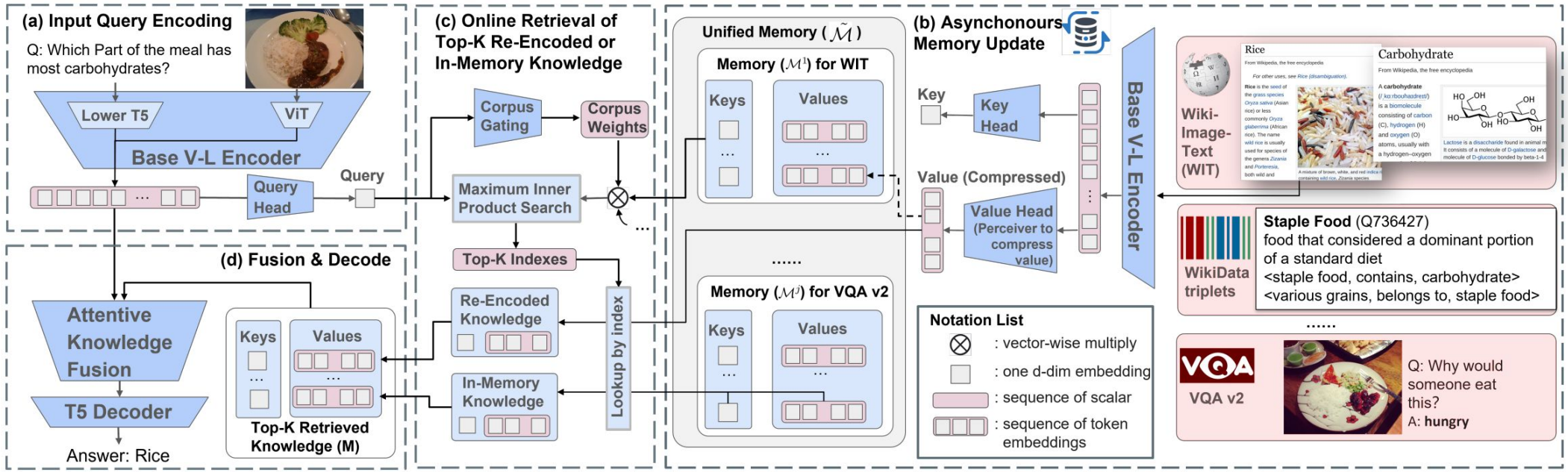
Figure 7. **Study of Knowledge Update**. The blue curve shows result by removing certain percentage of knowledge during both fine-tuning and inference stage. The orange curve shows results by still first removing the knowledge, and then adding the knowledge back during inference, which simulates the knowledge update.

결론

- Knowledge-intensive visual-language tasks를 풀기 위해 대규모 지식 메모리를 활용하여 학습하는 end-to-end V-L pre-training paradigm 최초 제안
- 다양한 multimodal world knowledge를 통합하여 대규모 메모리 구축하는 방법 제시
- 이전 연구보다 적은 파라미터로 여러 knowledge-intensive VQA와 Image Captioning에서 SOTA 달성

Open Questions

- 리소스가 부족한 상황에서 효과적이고 효율적인 retrieval augmented Vision-Language Pre-training을 하기 위해선 어떻게 해야하는가?



감사합니다.