2026 하계 세미나

KV Cache Compression

Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

Presented By
양진철

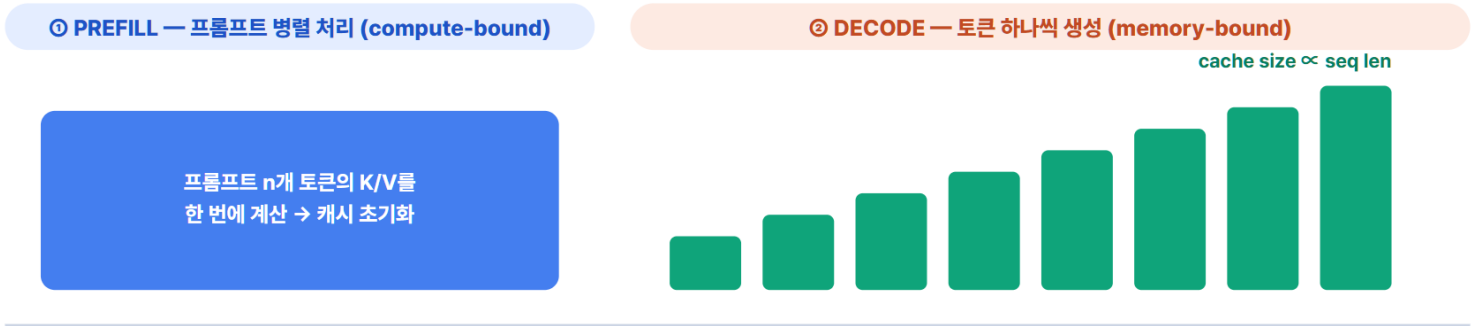
Outline

- Background
  - KV Cache
  - KV Cache Compression

- Papers
  - PatternKV: Flattening KV Representation Expands Quantization Headroom [ICML 2026]
  - KeyDiff: Key Similarity-Based KV Cache Eviction for Long-Context LLM Inference in Resource-Constrained Environments [NeurIPS 2025]

Background

- KV Cache란?
  - Autoregressive decoding에서 매 step마다 과거 token의 Key (K), Value (V)를 재계산하는 대신 한 번 계산한 K, V를 저장해 재사용하는 추론 최적화 기법
    - Decoding step당 연산량을 $O(n)$으로 줄여 추론 최적화
    - 하지만, 메모리 비용 증가
  - Prefill vs. Decoding stage
    - Prefill: 입력 prompt의 n개 토큰의 K, V를 한 번에 계산하여 KV Cache 초기화
    - Decode: 매 step 마다 token의 K, V를 cache에 append
      - Cache가 생성 길이에 선형으로 증가



Background

- KV Cache란?
  - KV Cache Compression의 필요성
    - Cache는 sequence 길이 및 배치에 선형으로 증가
      - KV Cache 크기 계산

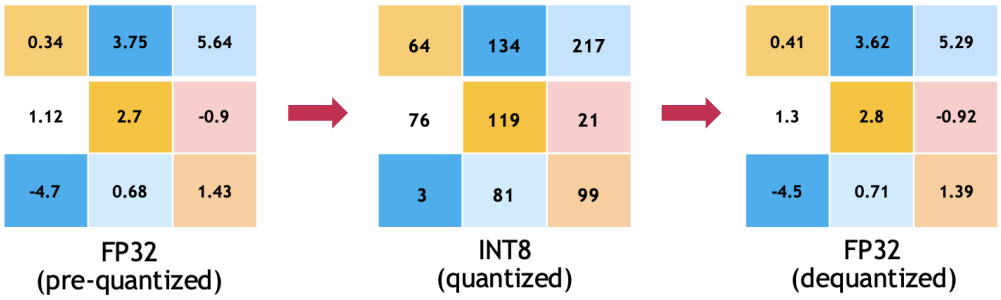
```
KV Cache = 2(K·V) × n_layers × n_kv_heads × d_head × seq_len × batch × 2 bytes
```

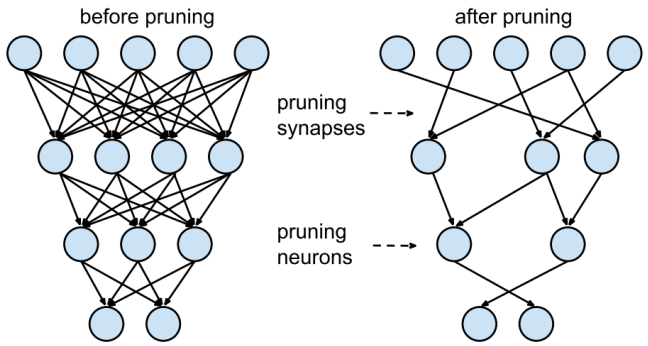
    - Context에 따른 KV Cache 크기
      - 긴 context, reasoning, 영상 입력 등

모델	가중치 크기	KV 구성	토큰당 캐시	역전 지점 (BATCH 1)	BATCH 8일 때
Llama-2-7B	**~14 GB**	MHA · 32L × 32H × 128d	0.50 MB	~28K tokens	~3.5K tokens
Llama-3-8B	**~16 GB**	GQA · 32L × 8H × 128d	0.13 MB	~125K tokens	~16K tokens
Qwen2.5-7B	**~15 GB**	GQA · 28L × 4H × 128d	0.06 MB	~270K tokens	~34K tokens
Llama-3-70B	**~140 GB**	GQA · 80L × 8H × 128d	0.33 MB	~430K tokens	~54K tokens
DeepSeek-V3 (MLA)	**~1.3 TB** (671B)	latent 576d × 61L	~0.07 MB	latent 캐싱으로 역전 자체를 구조적으로 지연	

Background

- KV Cache Compression
  - Quantization
    - 모델 파라미터의 표현 정밀도를 낮추는 과정
      - Floating point (FP) value → INT value
  - Pruning
    - 모델 파라미터에서 필요 없는 파라미터를 덜어내는 방법

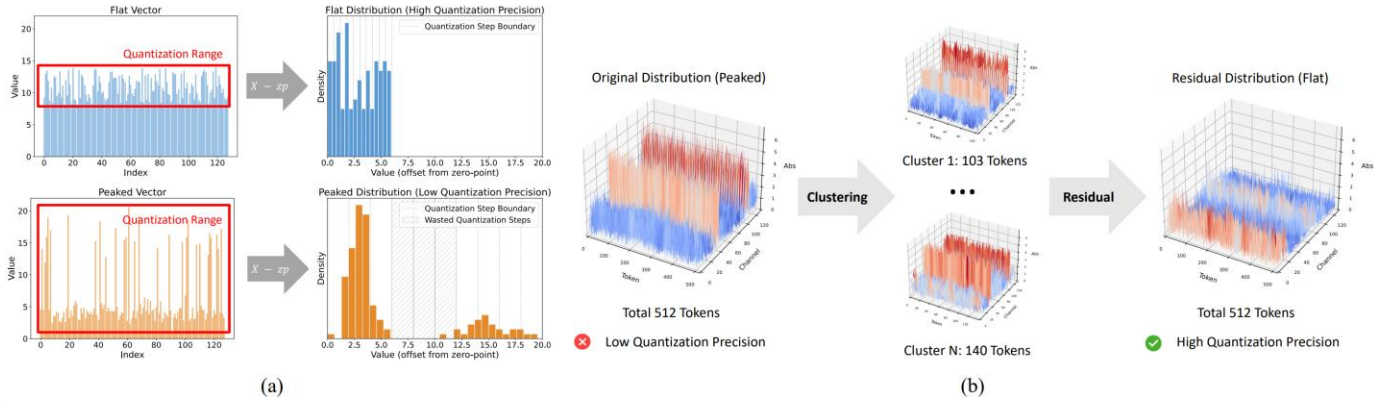




PatternKV: Flattening KV Representation Expands Quantization Headroom [ICML 2026]

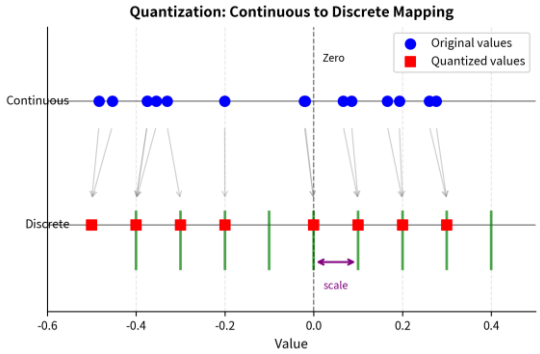
PatternKV[1)]

- Problem statements
  - KV quantization의 정확도는 KV 분포의 flatness에 크게 의존
    - 분포가 peaked하고 outlier가 존재할수록 quantization range가 넓어져 오차 증가
  - 기존 연구는 outlier를 분리 및 보호하는 데 집중
    - 전체 분포를 flatten 하지 못하고 low-bit 환경에서 성능이 취약해지는 근본적 한계 존재
- Key contributions
  - KV quantization을 variance decomposition 관점으로 재해석
    - Outlier 보호 → 분포 flattening
  - KV cache의 pattern을 분석하여 pattern 기반 residual quantization인 PatternKV 제안



PatternKV[1)]

- A Variance Decomposition View
  - Quantization scaling factor
    - 분포의 range에 비례 → variance를 flatness의 proxy로 사용



$$Q(X) = \left\lfloor \frac{X - z}{s} \right\rceil, \qquad X_{\text{deq}} = s \cdot Q(X) + z, \qquad s = \frac{\max(X) - \min(X)}{2^n - 1}$$

  - KV의 variance를 줄이기 위한 방법
    - Law of total variance

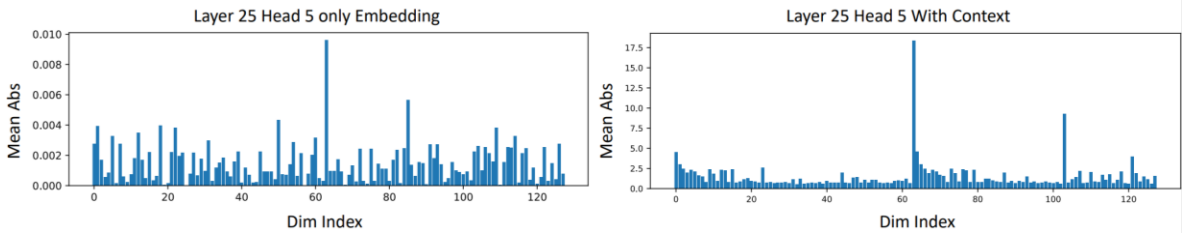
고정 시

$$\text{Var}(Z) = \underbrace{\mathbb{E}[\text{Var}(Z \mid M)]}_{\text{intra-pattern variance}} + \underbrace{\text{Var}(\mathbb{E}[Z \mid M])}_{\text{inter-pattern variance}}$$

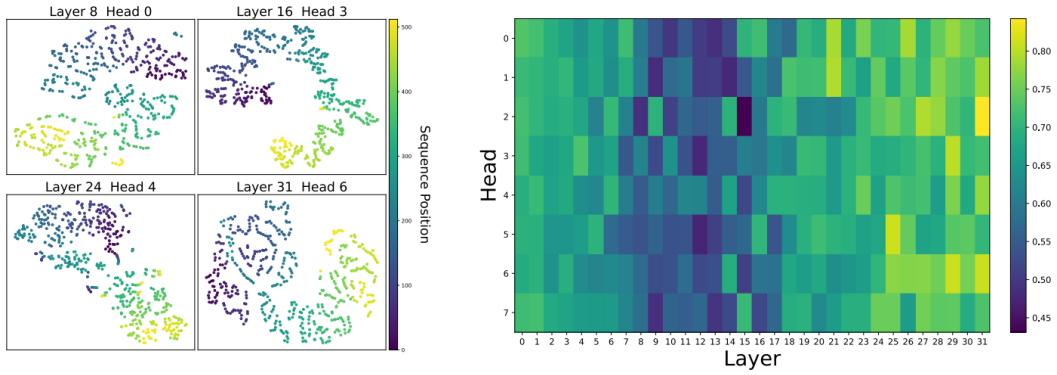
    - Pattern set M을 고정하면 inter-pattern 항이 소거 → 목표: "Intra-pattern variance로 축소"
      - Pattern set M: 데이터를 그룹화
  - 실제 분포의 오차를 줄이는 것 → Flat한 quantization 대상을 만드는 pattern M을 선택

PatternKV[1)]

- KV Pattern Analysis
  - K cache
    - 모델 내부 구조에서 기인하는 stable pattern, context는 rescale

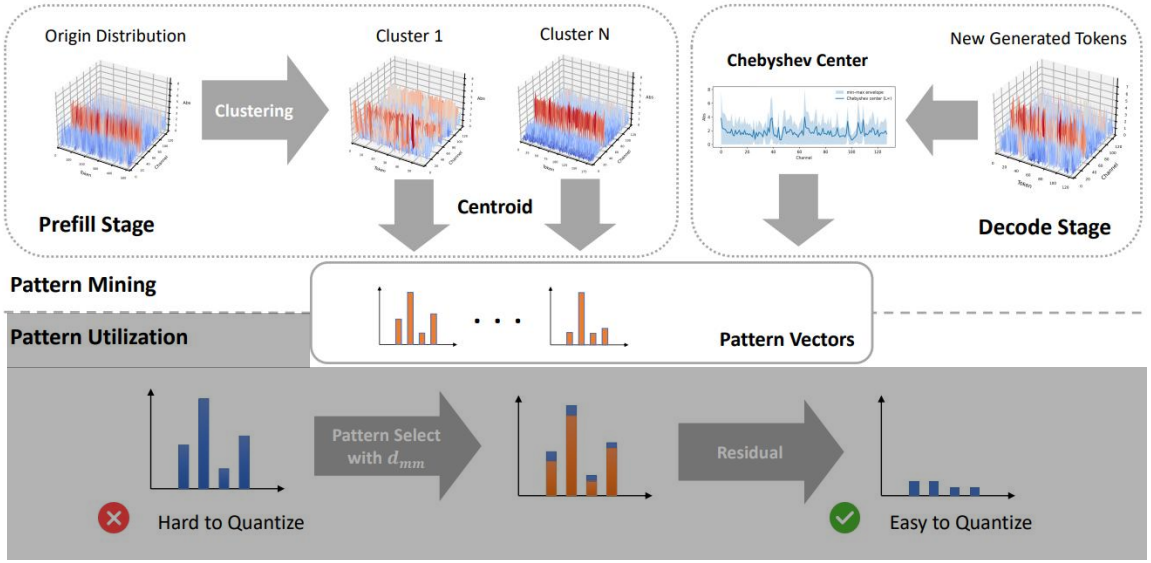


    - Context, RoPE에 의해 decoding stage에서 attention head별로 완만하게 변화
  - V cache
    - 토큰 의미와 연관된 latent pattern 존재, 대부분 layer에서 token–cluster 연관성 강함



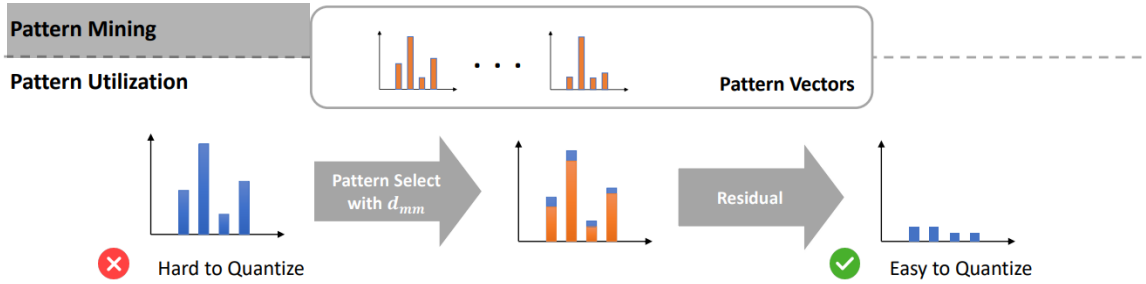
PatternKV[1)]

- Method: PatternKV
  - Pattern Mining
    - Prefill : K-means로 pattern vector를 고정하여 intra-pattern variance 최소화
      - 유사한 데이터들을 cluster N (32) $\min_{\mathcal{P}_h=P_1,\ldots,P_k} \sum_{j=1}^{k} \sum_{\boldsymbol{x_i}\in P_j} \|\boldsymbol{x_i} - \overline{\boldsymbol{x}}_{\boldsymbol{P_j}}\|_2^2$
      - 각 cluster의 평균값 → Pattern vectors
    - Decoding : Chebyshev center로 pattern 갱신 (128 tokens) → 분포의 점진적 진화 추적
      - 각 cluster의 중간 값 → Pattern vectors $M_{h,d}^{new} = \frac{1}{2}\left(\min_i X_{h,i,d} + \max_i X_{h,i,d}\right)$



PatternKV[1)]

- Method: PatternKV
  - Pattern Utilization
    - Pattern vector 선택
      - Min-max distance $M^\star = \underset{M\in\mathcal{M}_h}{argmin}\, d_{\mathrm{mm}}(X, M)$ $d_{mm}(\boldsymbol{x}, \boldsymbol{m}) = \max_i(x_i - m_i)$
    - Residual quantization
      - 분포 flatten $R = X - M^\star$
  - Only V cache
    - Token–cluster 연관성이 낮은 중간 layer
      - 중간 layer는 semantic 정렬이 약해 pattern 정렬이 오히려 오차를 증가
      - One-sided z-test 기반 adaptive threshold로 선택적 적용 $\rho = R_{\mathrm{flat}}/R_{\mathrm{raw}} \le \rho_*(d, \alpha)$



PatternKV[1)]

- System Optimization
  - Prefill stage pattern extraction
    - Head마다 별도 pattern set이 필요 → GPU-parallel K-means로 clustering latency 최소화
  - Decoding stage KV reconstruction
    - Decode마다 KV 재구성 overhead가 큼 → 두 개의 custom CUDA kernel로 fusion
      - Kernel 1: K cache의 pattern index 복원 + dequantization + QK^T matmul을 하나로 융합
      - Kernel 2: Dequantization + pattern index 복원 + V에 대한 attention weight 적용을 하나로 융합
- 효과
  - PyTorch 구현 대비 decode latency 대폭 감소
    - Peak throughput 약 15%↑, per-sample latency 약 10%↓
  - Overhead
    - Pattern vector 저장은 prefill 0.39%, decode 0.78% 수준으로, 전체 KV 메모리의 극히 일부

PatternKV[1)]

- Experiments
  - Quantitative Results
    - Benchmarks: LongBench(long-context)
    - Models: Llama-3.1-8B/70B, Qwen2.5-7B
    - 2-bit LongBench: baseline 대비 일관되게 향상

Model	Method	MQA	SQA	Summ.	Few-shot	Synth.	Code	Avg
Llama3.1-8B-Instruct	FP16	36.63	46.56	25.54	61.16	59.99	59.42	46.59
	KIVI	34.86	43.96	24.98	60.35	54.43	55.53	44.33
	ZipCache	32.65	40.52	24.02	59.86	47.44	60.91	42.49
	SKVQ	34.81	42.59	24.83	59.74	52.81	61.45	44.25
	OTT	34.34	43.41	**25.19**	59.64	55.45	**62.48**	44.84
	PatternKV	**35.49**	**45.08**	25.12	**60.58**	**57.89**	56.55	**45.33**
Llama3.1-70B-Instruct	FP16	52.68	49.56	25.67	66.18	72.67	46.80	51.81
	KIVI	52.41	48.92	**25.45**	65.73	72.58	46.62	51.48
	ZipCache	36.98	45.44	23.28	58.57	67.92	58.37	46.55
	SKVQ	-	-	-	-	-	-	-
	OTT	40.72	47.36	24.74	60.05	68.50	**59.97**	48.43
	PatternKV	**52.45**	**49.19**	25.21	**65.76**	**72.67**	47.65	**51.61**
Qwen2.5-7B-Instruct	FP16	38.03	45.40	23.37	59.85	58.83	62.84	46.13
	KIVI	35.77	42.73	**22.80**	58.13	51.50	56.25	43.08
	PatternKV	**36.36**	**43.93**	22.77	**59.21**	**55.17**	**56.67**	**44.18**

PatternKV[1)]

- Experiments
  - Ablation studies
    - Components : K/V pattern, new pattern 각각 기여, 특히 V-threshold 제거 시 급락
    - Cluster groups: M이 클수록 단조 향상, M=4로도 이득의 절반 확보
  - Efficiency & Overhead
    - Prefill의 pattern mining이 주 overhead지만 latency +6%에 불과
    - FP16 대비 throughput 1.5×, batch 1.25× / KIVI 대비 peak memory +0.42%에 불과

Component	LongBench Avg	GSM8K
KIVI	44.33	72.96
PatternKV	45.33	75.58
w/o K Pattern	44.53	73.91
w/o V Pattern	44.96	74.60
w/o New Pattern	45.37	75.49
w/o V Threshold	24.67	0.30

$\|\mathcal{M}\|$	LongBench Avg	GSM8K
KIVI	44.33	72.96
2	44.57	73.72
4	44.92	75.26
8	44.92	75.94
16	45.28	75.20
32	45.33	75.58

Component	Latency	Throughput	GFLOPs
Pattern mining	+6.60%	−6.10%	6.91
Pattern selection	+2.59%	−2.47%	2.07
Chebyshev-center updates	+3.11%	−2.94%	2.78

KeyDiff: Key Similarity-Based KV Cache Eviction for Long-Context LLM Inference in Resource-Constrained Environments

[NeurIPS 2025]

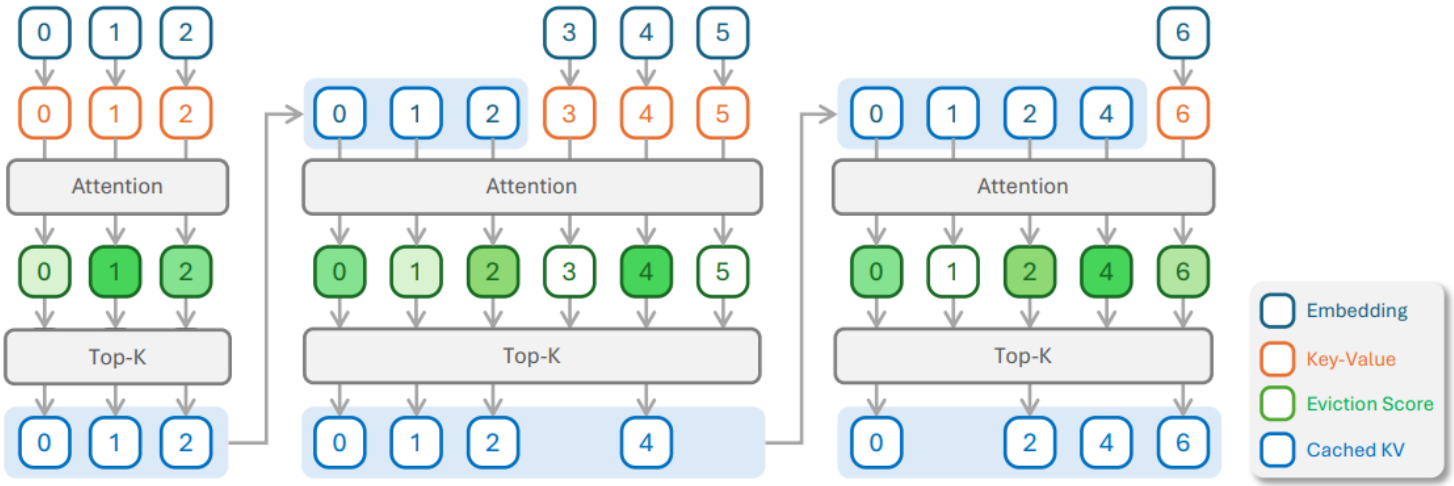
KeyDiff[1)]

- Problem statements
  - Edge-device 환경에서 제약 존재
    - Prompt 전체를 한 번에 처리하면 계산 과정에서 메모리 제약
  - 기존 연구들은 block-wise inference로 우회
    - 하지만 attention 기반 eviction은 정확도가 하락

- Key contributions
  - Cosine similarity 분석
    - 낮은 key일수록 높은 attention score를 받음
  - Attention score 없이 key 유사도만으로 eviction하는 KeyDiff 제안
  - Key diversity가 token 중요도의 proxy임을 이론적으로 정당화

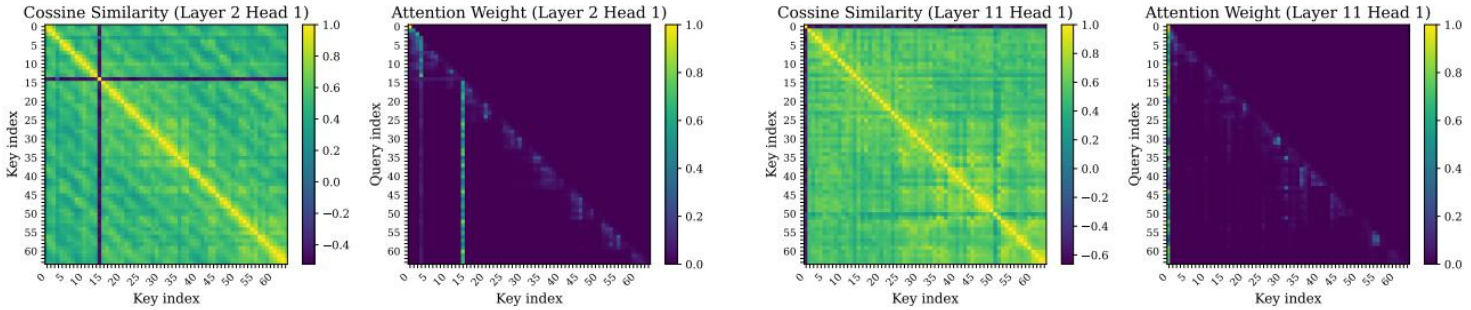
KeyDiff[1)]

- Block Prompt Processing
  - 입력 prompt를 작은 block으로 나누어 순차적으로 처리 (128 words)
    - 추론 전 과정에서 메모리 제약 준수
  - 각 block 처리 후 eviction score로 budget N을 초과하는 KV를 제거
- Attention 기반 eviction의 한계
  - Block i의 eviction이 이후 block i+1의 cache에 영향
    - 부분만 보고 중요도를 판단하여 미래에 중요해질 token이 미리 제거



KeyDiff[1)]

- Cosine similarity 분석
  - Insight : Key similarity와 attention score의 음의 상관관계
    - Attention sink 현상
      - LLM은 Query와 무관하게 특정 소수 토큰에 높은 attention을 할당
    - Hypothesis
      - 높은 attention score는 QK 조합이 아니라 Key 자체의 고유 특성으로 결정 ?
    - Results
      - Key들과 평균 cosine similarity가 낮은 K가 Q와 무관하게 더 높은 attention score
        - ✓Key diversity가 미래 token 접근 없이도 global token importance의 강한 지표



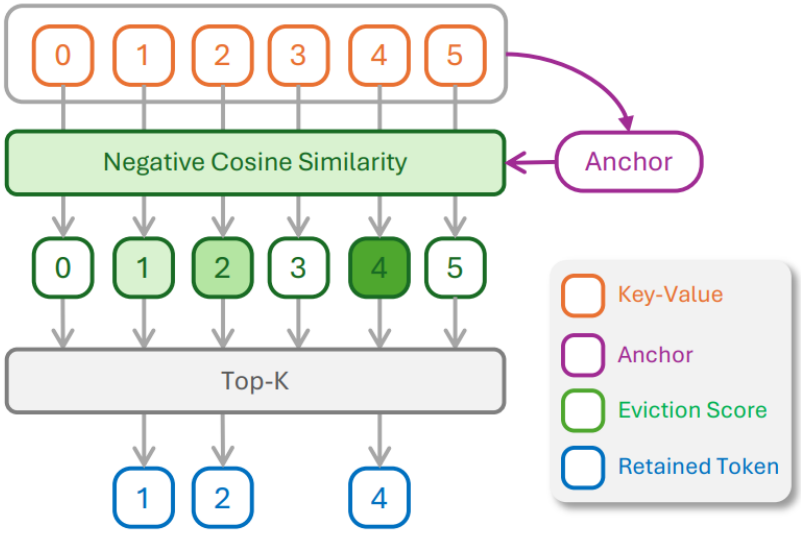
(a) Layer 2 and Head 1 (b) Layer 11 and Head 1

KeyDiff[1)]

- Method : KeyDiff
  - 각 key에 대해 다른 key와의 pairwise cosine similarity 합이 큰 Key부터 제거
    - 유사도가 낮은 N 개만 생존

$$S = \texttt{topk}(-\mathrm{CosSim}(K)\mathbf{1}, N) \longrightarrow S = \texttt{topk}(-\mathrm{CosSim}(\mu(\hat{K}), \hat{k}_i), N)$$

    - Efficient variant : $O(n^2) \rightarrow O(n)$
      - Anchor vector μ(K)(전체 Key 평균)와의 유사도만 계산
  - 유사도가 낮은 token이 중요한 정보!



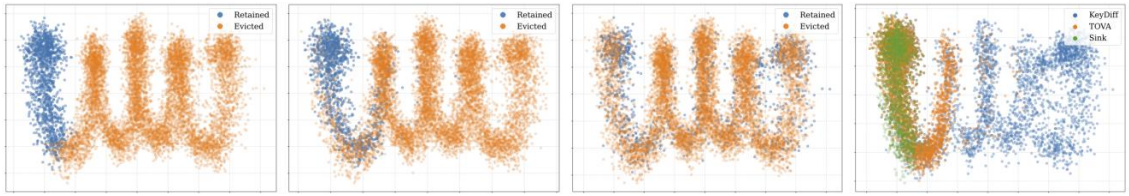
KeyDiff[1)]

- Why KeyDiff Works : A Theoretical Perspective
  - Lemma
    - Key가 높은 attention score를 받으려면 Query와 높은 cosine similarity 필요
  - Theorem

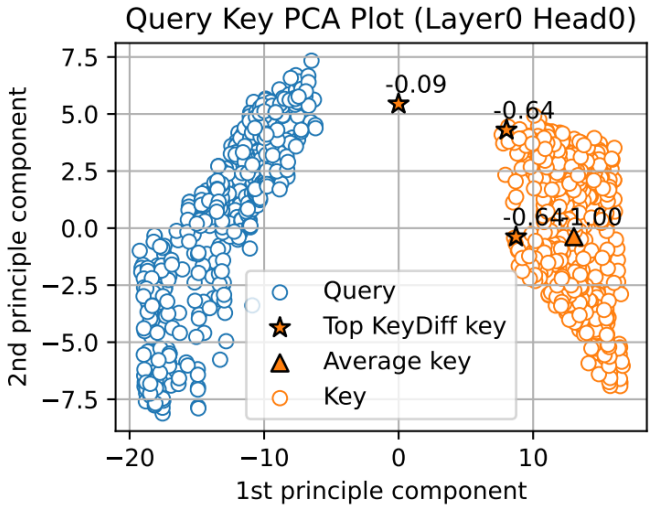
$$CosSim(k^*, q) = \beta_q > 0 \; and \; CosSim(\bar{k}, q) = \alpha_q < 0.$$

$$CosSim(\bar{k}, k^*) \leq 1 + \alpha_q\beta_q - 0.5\alpha_q^2 - 0.5\beta_q^2.$$

    - 1. 중요 Key와 Query가 정렬되고 2. 평균 Key가 Query와 유사도가 낮으면
      - 중요 Key와 평균 Key의 낮은 유사도
  - 따라서, 평균 Key와 유사도가 낮은 Key 가 중요
    - "Key diversity가 미래 token 접근 없이도 global token importance의 강한 지표"



(a) Sink Attention (b) TOVA (c) KeyDiff (d) Retained keys only



KeyDiff[1)]

- Experiments
  - Quantitative Results
    - Benchmarks : LongBench (long-context)
    - Models : Llama-3.1-8B, Llama-3.2-3B
    - 6K budget ≤1.5%, 8K budget ≤0.04% 손실 (33%·23% 압축), 대부분 task에서 우수

		Single Doc. QA			Multi Doc. QA			Summarization			Fewshot Learning			Synthetic		Code		
		Narrative QA	Qasper	MF-en	HotpotQA	2WikiMQA	Musique	GovReport	QMSum	MultiNews	TREC	TriviaQA	SAMSum	PCount	PR-en	Lcc	RB-P	Avg.
Llama3.1-8B		30.05†	47.00	56.12	57.33	47.81	32.25	34.86	25.32	27.02	73.00	91.61	43.37	8.33	99.50	61.66	51.94	49.20
H2O	2K	1.74	21.15	25.33	26.11	24.15	8.78	2.17	2.70	16.78	44.00	29.36	7.62	2.25	5.88	40.15	12.14	16.89
	4K	4.07	36.16	36.00	33.52	32.87	17.78	6.66	5.95	24.09	55.00	47.65	17.41	4.00	24.50	54.85	21.43	26.37
	6K	8.52	43.31	44.80	40.03	42.46	21.68	11.85	8.78	26.03	62.00	56.39	25.72	5.75	45.50	58.62	29.53	33.19
	8K	13.85	44.94	47.81	43.64	44.90	23.65	18.78	11.35	26.49	69.50	69.05	33.41	5.25	62.50	59.74	36.26	38.20
TOVA	2K	22.57	37.26	39.43	45.74	34.48	14.77	28.87	21.17	26.95	62.50	90.73	42.74	0.00	18.00	62.68	52.48	37.52
	4K	22.68	44.55	47.87	46.76	44.54	20.56	30.95	22.13	26.96	61.50	90.56	43.27	3.00	43.50	61.62	53.40	41.49
	6K	24.59	45.93	53.92	55.09	47.43	25.07	32.33	24.10	27.00	68.50	90.81	43.89	4.25	67.00	61.50	52.39	45.24
	8K	24.86	46.78	54.83	54.52	49.00	26.40	33.44	24.76	27.00	71.00	91.11	43.29	6.25	87.00	61.49	51.79	47.09
Sink	2K	21.83	34.27	29.24	38.64	29.50	12.59	28.51	20.21	26.62	65.00	89.46	42.20	2.00	25.50	64.95	59.54	36.88
	4K	22.94	43.01	39.08	44.04	41.39	19.09	31.08	21.57	26.78	70.00	91.53	42.29	3.00	38.50	62.12	58.84	40.95
	6K	25.41	47.40	44.13	47.39	45.73	21.90	32.53	22.19	26.87	72.00	91.25	43.41	3.08	52.50	62.22	56.24	43.39
	8K	23.53	46.63	48.68	49.61	47.16	21.14	33.10	23.20	26.92	72.00	91.29	43.79	3.25	66.00	62.18	56.43	44.68
SnapKV	2K	21.81	37.22	37.19	46.10	35.42	16.53	29.83	21.05	26.77	61.00	88.84	42.56	4.03	51.50	62.37	51.45	39.60
	4K	24.79	44.22	47.30	48.49	46.73	20.55	32.19	22.68	26.95	67.50	90.98	43.14	5.17	89.50	61.44	51.20	45.18
	6K	24.10	45.57	50.44	53.12	48.41	24.27	33.43	23.53	27.03	71.50	92.28	43.58	5.25	98.00	61.32	52.16	47.12
	8K	25.15	46.55	53.39	56.00	48.75	27.82	33.67	24.85	27.01	72.50	91.78	43.54	5.08	100.00	61.48	51.41	48.06
KeyDiff	2K	26.64	41.73	50.99	51.59	46.47	22.84	29.02	23.86	26.76	66.50	85.92	39.26	3.17	96.00	59.17	39.42	**44.33**
	4K	28.70	45.62	56.06	54.58	49.31	28.25	32.30	25.03	27.07	70.00	90.85	42.84	4.21	99.00	60.80	48.00	**47.66**
	6K	29.90	46.33	55.11	56.80	49.50	31.52	33.44	24.58	26.98	72.00	90.99	43.10	5.27	99.50	61.40	49.70	**48.51**
	8K	33.57	46.77	55.48	56.87	49.37	30.88	34.17	25.12	27.01	72.50	92.28	42.81	5.83	99.50	61.48	50.90	**49.03**
Llama3.2-3B		23.76	40.23	50.09	50.69	42.29	26.84	33.09	24.30	25.21	72.50	90.11	42.58	3.00	96.50	56.22	56.52	45.87
H2O	2K	1.63	19.96	20.20	18.02	19.56	2.88	0.78	1.55	15.97	41.00	21.97	9.83	0.50	0.50	39.71	13.91	14.25
	4K	2.92	31.94	33.23	24.49	28.08	7.55	5.44	6.30	22.77	53.00	38.85	20.33	1.50	7.50	51.23	22.94	22.38
	6K	4.62	38.81	39.06	34.66	35.52	15.21	10.51	10.01	24.25	61.50	53.23	27.37	0.50	13.00	54.55	32.29	28.44
	8K	9.65	39.66	43.20	38.09	40.41	21.46	17.80	13.28	24.67	70.00	64.30	32.19	2.00	24.50	55.00	39.09	33.46
TOVA	2K	17.14	30.14	32.44	35.96	30.05	13.08	26.15	19.70	25.04	56.50	87.81	40.48	2.50	11.50	55.51	52.36	33.52
	4K	20.52	39.53	42.47	44.12	38.42	18.22	29.36	21.36	24.96	63.50	88.98	41.50	3.00	23.50	55.72	56.66	38.24
	6K	20.22	39.78	45.86	49.08	41.54	20.43	30.50	22.17	25.11	66.50	89.00	42.50	4.00	46.50	55.57	57.53	41.02
	8K	21.08	40.67	49.07	48.69	41.93	23.05	31.64	22.85	25.21	69.00	89.25	42.19	2.50	71.00	55.77	57.47	43.21
Sink	2K	16.85	30.69	26.58	33.26	25.27	13.82	26.74	19.15	25.15	65.00	86.17	40.79	1.50	19.50	56.65	52.73	33.74
	4K	19.46	38.61	36.22	41.97	35.84	13.37	29.34	20.19	25.06	71.00	88.06	41.31	2.50	35.50	56.48	52.43	37.96
	6K	19.33	40.29	37.95	46.48	40.29	15.31	30.43	21.35	25.14	71.50	88.93	42.04	3.50	47.00	56.55	54.11	40.01
	8K	20.15	40.02	41.94	48.15	42.24	16.01	31.64	22.10	25.20	73.00	89.26	42.37	3.50	62.50	56.86	56.63	41.97
SnapKV	2K	17.38	31.37	31.48	37.77	30.05	11.54	27.03	19.93	24.97	59.00	88.13	40.48	3.50	32.50	56.32	55.91	35.46
	4K	19.85	39.22	39.86	46.70	37.98	16.64	29.79	21.21	25.01	65.50	89.35	40.95	2.50	62.50	55.74	56.88	40.60
	6K	20.83	39.65	44.48	49.30	40.18	20.28	31.27	22.73	25.09	69.00	89.95	41.47	4.00	85.00	55.69	57.82	43.55
	8K	20.49	40.80	48.16	48.78	41.65	24.79	31.81	23.46	25.17	70.00	90.17	41.99	5.00	94.00	55.77	57.29	44.96
KeyDiff	2K	18.29	36.65	45.44	46.09	35.41	13.79	28.16	21.45	25.01	60.00	85.24	37.00	1.00	60.50	54.13	42.01	**38.14**
	4K	22.34	40.60	49.15	50.14	40.30	21.65	31.38	23.44	25.06	66.50	87.92	41.41	2.50	88.50	55.55	52.24	**43.67**
	6K	22.29	40.68	50.14	51.74	42.19	24.83	32.39	23.53	25.19	71.00	90.02	42.00	3.00	95.00	55.86	54.39	**45.27**
	8K	22.41	40.77	50.10	49.83	43.58	28.09	32.78	23.60	25.17	72.00	90.17	42.46	3.50	96.50	55.85	55.65	**45.78**

KeyDiff[1)]

- Experiments
  - Ablation studies
    - 실험 세팅: Average of Full Longbench results for Llama 3.2-3BInstruct
    - Anchor vector
      - Pairwise, Median은 더 정확한 방식이지만 연산 복잡도가 큼
    - Similarity metric
      - Cosine similarity가 모든 budget에서 우위

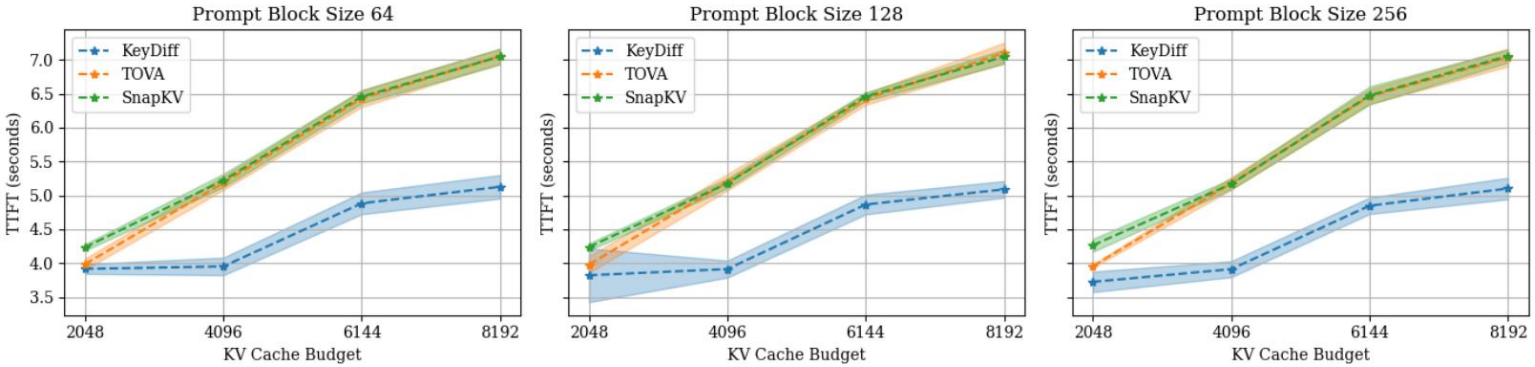
	2K	4K	6K	8K
KEYDIFF	35.09	40.28	41.88	42.40
Pairwise	35.24	40.61	41.87	42.45
Median	35.43	40.67	41.89	42.26

	2K	4K	6K	8K
KEYDIFF	35.09	40.28	41.88	42.40
DotProd	30.14	38.01	41.09	42.23
Euclidean	13.68	21.06	25.91	29.53

KeyDiff[1)]

- Experiments
  - Efficiency
    - 복잡도 O (N+B)로 TOVA와 동일, H2O, SnapKV보다 낮음
    - FlashAttention 호환으로 TTFT 최대 30% 감소, block size와 무관한 낮은 latency
    - Cache가 커질수록 TOVA, SnapKV 대비 scoring latency 크게 우위

	Runtime Complexity	Memory Complexity
KeyDiff	$\mathcal{O}(N+B)$	$\mathcal{O}(N+B)$
TOVA	$\mathcal{O}(N+B)$	$\mathcal{O}(N+B)$
H2O	$\mathcal{O}(NB+B^2)$	$\mathcal{O}(NB+B^2)$
SnapKV	$\mathcal{O}((N+B)L)$	$\mathcal{O}((N+B)L)$
Sink	$\mathcal{O}(N)$	$\mathcal{O}(1)$



감사합니다.