VIDMEM: MEASURING MEMORIZATION CAPACITY AND KNOWLEDGE DISPLACEMENT IN VIDEO-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

How many bits can a video-language model store in its parameters? This capacity constrains corpus fit, knowledge displacement during continued training, and exposure of sample-specific information, yet it has not been measured for visual modalities. Existing studies detect replication through similarity, extraction, or attack success but lack a common unit and capacity limit. We propose VidMem, which binds procedurally generated videos to independent random answer strings of known entropy and measures stored information as the reduction in conditional codelength from a uniform reference. Across multiple sizes of a calibration decoder family, the estimate scales linearly at 2.54 bits per trained parameter over the tested range. The protocol applies to a second decoder family and five frozen pretrained encoders, although the coefficient is architecture- and format-specific. Pooled slopes indicate that a video-keyed association costs $1.28\times$ as many parameters as a text-keyed one; tested ablations associate this gap with perception parameters and key length. A capacity ledger then characterizes unavoidable forgetting relative to a specified ceiling. Across three chunk-entropy ratios, onset is consistent with the stream-specific capacity realized by union training. VidMem therefore gives multimodal memorization an explicit unit and budget.

1 INTRODUCTION

Video-language models are becoming a common interface to visual data. They caption archives and answer long-video questions (Liu et al., 2023; Lin et al., 2024), localize and count actions in untrimmed streams (Qiu et al., 2024a;b; Qiu & Yanai, 2026a;b), and increasingly train on web-scale corpora (Bai et al., 2025b). Parametric storage constrains corpus fit, displacement of earlier knowledge, and exposure of sample-specific information. Existing tools use task-specific observables: perceptual similarity for replication (Carlini et al., 2023a; Somepalli et al., 2023; Chen et al., 2024), membership inference for leakage (Shokri et al., 2017; Carlini et al., 2022; Wang et al., 2026), and curated probes for cross-modal recall (Jayaraman et al., 2024; Wen et al., 2025). These observables lack a common unit, do not isolate sample-specific storage from generalization, and do not identify a capacity limit.

The lines of work related to this gap remain largely separate. Visual memorization studies span image diffusion (Somepalli et al., 2023; Kim et al., 2025; Buchanan et al., 2025) and video diffusion (Chen et al., 2024), yet none reports stored information in bits. Continual learning quantifies catastrophic forgetting (Kirkpatrick et al., 2017; Li & Hoiem, 2018; Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017), but relative to a baseline rather than to a measured capacity. A parametric capacity in bits has so far been measured only for text: compression-based measurement estimates roughly 3.6 bits per parameter for language models (Morris et al., 2026), complementing work from canary probes to verbatim extraction (Carlini et al., 2019; 2021; 2023b). Knowledge-capacity scaling laws report a comparable value of about 2 bits per parameter for factual storage (Allen-Zhu & Li, 2024). To our knowledge, no prior work measures the parametric memorization of a visual modality in bits or relates a forgetting bound to a measured content capacity. Figure 1 summarizes this gap.

Two requirements make direct measurement on natural video difficult. The compression framework requires exact log-likelihoods, which are unavailable for pixels in many generative models

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

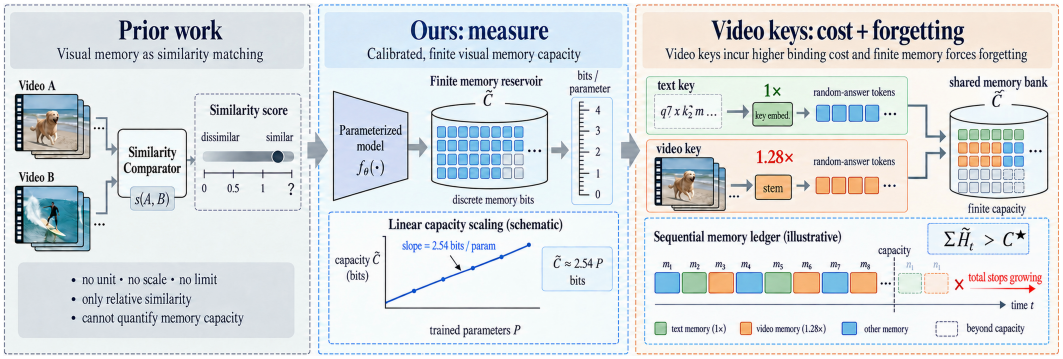

Figure 1: From detecting to measuring. **Left:** most prior visual-memorization studies characterize replication through similarity or attack success, without a common unit. **Middle:** VidMem estimates stored information in bits; the estimate is approximately linear in parameter count over the measured range. **Right:** the pooled slopes imply a $1.28\times$ parameter cost for video-keyed associations, and a capacity ceiling implies unavoidable forgetting once written entropy exceeds it. Schematic illustration.

and are only bounded for diffusion models. It also requires sample-specific storage to be separated from generalization, whereas natural video contains substantial statistical redundancy. We therefore encode the identifying assumptions in the data construction. For an answer assignment drawn independently and uniformly at random, the expected predictive performance on an unseen assignment equals the uniform baseline, whose entropy is known exactly.

We introduce VidMem, a protocol that binds simulated videos to random answer strings of known entropy and estimates stored information as that entropy minus the model's conditional codelength. The protocol compares the parameter cost of associations keyed by text, visual appearance, or motion; the ratio of these costs is the *cross-modal binding cost*. We also formulate a capacity ledger: relative to a specified capacity ceiling, information written beyond that ceiling must displace previously stored information. In experiments, we distinguish the finite-grid capacity estimate from the stream-specific retention ceiling realized by union training. Our contributions are as follows.

- We propose VidMem, to our knowledge the first protocol that estimates the parametric memorization capacity of video-language models in bits without a reference model or a tractable video likelihood. Over the tested range, the finite-grid estimate scales linearly at 2.54 bits per trained parameter for the calibration decoder family. The unit applies to a second decoder family, frozen pretrained encoders, and sparse-expert layers, although the numerical constant does not transfer unchanged.
- We introduce the cross-modal binding cost, which compares the parameter cost of the same random association under different keys. The pooled estimates give a $1.28\times$ cost for video-keyed storage. In the tested ablations, this gap is associated with perception parameters and key length, and motion and appearance keys reach the same cost at the largest measured sizes.
- We derive a capacity ledger that lower-bounds forgetting relative to a specified capacity ceiling. Empirically, the first lossy chunk is consistent with the stream-specific ceiling realized by union training at all three tested chunk-entropy ratios.
- We evaluate continual-learning methods against this empirical retention ceiling. Replay saturates below the union-training result, while naive fine-tuning both forgets earlier chunks and progressively loses plasticity.

2 Related Work

Language-model memorization capacity. Networks can fit random labels (Zhang et al., 2017), and some memorization aids long-tail generalization (Feldman, 2020). Compression-based measurement gives GPT-family models 3.6 bits/param (Morris et al., 2026); factual-knowledge scaling gives 2 bits/param (Allen-Zhu & Li, 2024); the framework has since been adapted to explain grokking (Song & Ye, 2026). Capacity also shapes compute-optimal scaling of model and data size (Kaplan et al., 2020; Hoffmann et al., 2022). All operate on text alone.

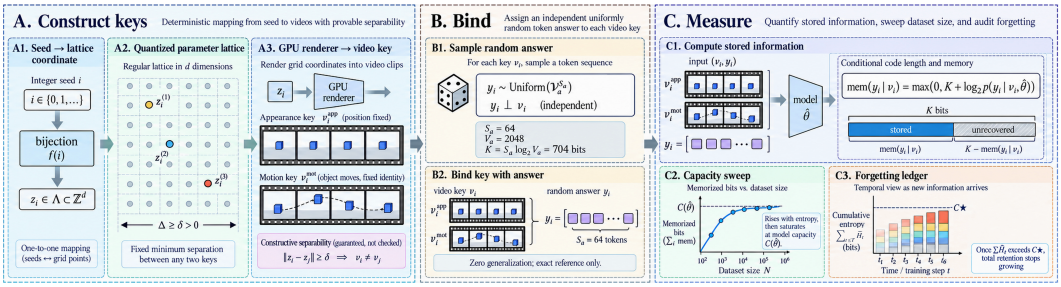

Figure 2: Overall framework of VidMem. **(A) Key generation:** keys are deterministic functions of an index and occupy distinct points of a parameter lattice, so separation holds by construction. **(B) Binding:** each key is paired with an independently generated random answer, making the uniform distribution the reference in expectation. **(C) Measurement:** stored bits are estimated as answer entropy minus conditional codelength (Eq. 1); sweeping dataset size yields an operational capacity estimate (Eq. 2), and sequential streams instantiate the forgetting ledger (Eq. 5).

Memorization in visual generative models. Replication and extraction are established for image diffusion (Carlini et al., 2023a; Somepalli et al., 2023; Kim et al., 2025; Buchanan et al., 2025) and extend to video diffusion and text-to-video attacks (Chen et al., 2024; Wang et al., 2026). None of these reports bits; Buchanan et al. (2025) predicts a model-size threshold for memorization, not a capacity in bits.

Information analyses of visual tokens. Recognition limits per vision token (Zhuang et al., 2026), optical context compression (Wei et al., 2025), and cross-modality recall probes (Jayaraman et al., 2024; Wen et al., 2025) all use downstream accuracy as the currency; none measure what the parameters themselves store.

Continual learning and forgetting bounds. Regularization and rehearsal methods (Kirkpatrick et al., 2017; Rebuffi et al., 2017) are evaluated by relative forgetting; an information-theoretic account bounds forgetting via task-identity entropy (Cheng, 2026); none derive limits from a measured content capacity.

3 Method

3.1 Overall Framework of VidMem

Figure 2 illustrates the overall framework of VidMem. A dataset $\mathcal{D} = \{(\boldsymbol{v}_i, y_i)\}_{i=1}^N$ pairs a key video $\boldsymbol{v}_i$ with an answer string y_i of S_a tokens sampled independently and uniformly from a vocabulary of size V_a. Each answer therefore has entropy $K = S_a \log_2 V_a$, and the random assignment has total answer entropy $H(\mathcal{D}) = NK$. Under this assignment distribution, y_i is independent of $\boldsymbol{v}_i$ and of the other pairs; expected prediction on a newly sampled assignment is therefore at the uniform baseline. In the finite implementation, the answers are generated by an independently seeded counter-mode generator, and held-out-index tests provide an empirical check that the model has not learned its structure. The protocol consequently measures sample-specific key–answer storage through the answer channel rather than pixel storage. The uniform distribution has codelength K per answer and supplies the reference without training an additional model. We define the per-sample memorization score as the number of bits saved by the model's conditional codelength relative to this reference:

$$\text{mem}_U(y_i \mid \boldsymbol{v}_i) = \max\big(0,\ K + \log_2 p(y_i \mid \boldsymbol{v}_i, \hat{\theta})\big), \tag{1}$$

where $p(y_i \mid \boldsymbol{v}_i, \hat{\theta})$ is the model's likelihood of the answer given the key video, the logarithm converts negative log-likelihood to an arithmetic-coding codelength in bits, and truncation at zero assigns no negative storage to samples compressed worse than the uniform reference. We use the sum of Eq. 1 as the protocol's dataset-level score. The operational capacity estimate is the largest score observed as dataset size increases:

$$\widehat{\mathrm{C}} \ = \ \max_{N \in \mathcal{N}} \sum_{i=1}^{N} \text{mem}_U(y_i \mid \boldsymbol{v}_i), \tag{2}$$

where $\mathcal{N}$ is the finite grid of dataset sizes at a fixed data distribution and only converged runs enter the maximum. The parameter count in every capacity-per-parameter result includes exactly the

trained parameters; a frozen encoder does not enter the denominator. Within a run, memorization-versus-step curves are not monotone: after their peak they can decline by a few percent even without weight decay. We therefore save the running-best checkpoint and re-evaluate it on the full dataset. Because the search spans a finite set of dataset sizes, optimization settings, and training budgets, $\widehat{\mathrm{C}}$ is a lower bound on the architecture's attainable capacity under this answer format, not a universal upper bound. We use "capacity" below when the operational meaning is clear and retain this qualification for all reported constants.

3.2 Keys, Entropy Accounting, and Binding Costs

Under the random-assignment distribution, the key carries no answer information, $I(\boldsymbol{v}_i; y_i) = 0$, yet its representation affects the cost of storing the association. VidMem measures this cost with key tracks $m \in \{\text{text}, \text{appearance}, \text{motion}\}$ that share one answer distribution and differ only in the map from sample index to key:

$$\boldsymbol{v}_i = g_m\big(\pi_m(i)\big), \qquad \pi_m : \{1, \dots, N\} \hookrightarrow \Lambda_m, \qquad \min_{i \neq j} \big\|\pi_m(i) - \pi_m(j)\big\| \;\geq\; \delta_m, \tag{3}$$

where Λ_m is a lattice over the generation parameters of track m with spacing δ_m, π_m is the restriction of a keyed bijection on Λ_m, and g_m is a deterministic renderer. Every track thus supplies the same $\log_2 N$ bits of sample identity and varies only the surface form g_m: random token prefixes for text, which anchors the reference slope α_{text}; static visual attributes for appearance; and trajectory parameters under a fixed appearance for motion, so that separating two motion keys requires temporal integration. Two properties follow from Eq. 3: since δ_m is independent of N, key crowding cannot produce a spurious plateau in the capacity fit, and since $g_m \circ \pi_m$ depends on the index alone, the dataset is defined implicitly rather than stored and is exactly reproducible. A keyless track, $\boldsymbol{v}_i = \varnothing$, replicates the original text-only protocol as a pipeline calibration; without a key the first answer token is unpredictable, so its per-sample ceiling is $K - \log_2 V_a$ and it does not enter Eq. 4. Fitting $\widehat{\mathrm{C}}_m = \alpha_m P$ across model sizes within each track, with P the trained parameter count, gives the capacity slope α_m, and the relative cost of a modality is the ratio:

$$\gamma_m \;=\; \frac{\alpha_{\text{text}}}{\alpha_m}, \qquad m \in \{\text{appearance}, \text{motion}\}, \tag{4}$$

where $\gamma_m > 1$ means a fact keyed by modality m consumes γ_m times the parameter budget of the same fact keyed by text, and the gap between γ_{motion} and $\gamma_{\text{appearance}}$ compares binding across several frames with binding within a single frame.

3.3 The Capacity Ledger and Continual-Learning Audit

Let $\mathrm{C}^\star$ denote a valid upper bound on the amount of answer information that the fixed model, precision, and answer format can retain. Consider a task stream that presents chunks $\mathcal{D}_1, \dots, \mathcal{D}_T$ sequentially, where chunk t carries H_t bits of answer entropy and $R_t(T)$ is the score from Eq. 1 retained for that chunk after training on all T chunks. If $\sum_t R_t(T) \leq \mathrm{C}^\star$, the total forgetting satisfies

$$F(T) \;=\; \sum_{t=1}^{T} \big(\tilde{H}_t - R_t(T)\big) \;\geq\; \max\Big(0,\; \sum_{t=1}^{T} \tilde{H}_t \;-\; \mathrm{C}^\star\Big), \tag{5}$$

where $\tilde{H}_t = \min(H_t, \mathrm{C}^\star)$. Equation 5 is a lower bound on forgetting conditional on a valid capacity ceiling; it does not specify which samples are forgotten or how closely a training algorithm approaches the ceiling. Importantly, the finite-grid estimate $\widehat{\mathrm{C}}$ in Eq. 2 is a lower bound and therefore cannot by itself substitute for $\mathrm{C}^\star$ in a universal claim. In the experiments, we instead use the maximum retention achieved by union training on the same stream as a stream-specific empirical ceiling, denoted $\mathrm{C}_{\mathrm{stream}}$. This ceiling is an observed reference rather than a theorem. We compare naive fine-tuning and replay (Rolnick et al., 2019) with it and report their retention gaps in bits.

4 Experiments

4.1 Protocol and Model Grid

Models. We train decoder-only video-language models from scratch. A lightweight 3D-patch stem embeds 16-frame 64×64 clips and feeds a GPT-2-style decoder. The grid crosses depth $\{1, 2, 4, 8\}$

Table 1: Capacity per trained parameter (bits/param; bf16, lower bounds, converged runs only) at every grid point, with the pooled origin-through fit and the equal-weight per-model mean. $\gamma = \alpha_{\text{text}}/\bar{\alpha}_{\text{video}}$ averages the binding costs of Eq. 4 over the two video tracks; Params counts the video-keyed models. †: single capacity-limited dataset size (lower bound).

Depth×Width	Params	α_{keyless}	α_{text}	$\alpha_{\text{appear.}}$	α_{motion}	γ
1×64	0.39M	3.28	3.18	1.67	1.96	1.76
2×64	0.44M	3.11	3.07	1.77	2.02	1.62
1×128	0.87M	3.23	3.11	1.88	2.14	1.55
2×128	1.07M	3.45	3.34	2.09	2.29	1.53
4×128	1.47M	3.48	3.46	2.42	2.45	1.42
2×256	2.92M	3.43	3.41	2.33	2.45	1.43
4×256	4.50M	3.44	3.44	2.59	2.61	1.32
8×256	7.66M	3.20	2.90	2.44	2.42	1.19
4×512	15.30M	3.38†	3.42	2.51†	2.50†	1.36
8×512	27.91M	3.15	3.21	2.56	2.56	1.26
pooled fit	—	3.20	3.25	2.54	2.54	1.28
per-model mean	—	3.32	3.25	2.23	2.34	1.44
s.d.	—	0.14	0.19	0.35	0.23	0.17

with width $\{64, 128, 256, 512\}$, giving ten sizes whose trained-parameter counts run from 0.19M to 27.9M across the four tracks. A separate real-video anchor replaces the synthetic key pathway with a frozen pretrained encoder and tests whether the constants transfer to natural visual statistics.

Data. Unless otherwise specified, answers use S_a=64 tokens over a vocabulary of V_a=2048, so each sample carries K=704 bits. The absolute constants are specific to this answer format, while every ratio we report holds it fixed. Dataset sizes sweep eight points per model on a log grid spanning one-eighth to eight times the model's predicted capacity crossing, so every sweep straddles the diagonal, the knee, and the plateau.

Metrics. We report per-sample memorization by Eq. 1 and model capacity by Eq. 2, both in bits, the capacity slope α in bits/param as a pooled origin-through regression and as a per-model mean with its standard deviation, the binding cost γ_m of Eq. 4, and the retention $R_t(T)$ in bits.

4.2 Implementation Details

All models are trained with Adam at batch size 2048 under bf16 autocast for at most 4×10^5 steps, with one seed per grid configuration and seed spread quantified at replicated points. The learning rate is 2×10^{-3}, reduced to 5×10^{-4} at width 512 where the higher rate suppresses capacity, and weight decay is zero because decoupled decay erodes stored bits once the loss plateaus. Training stops under a patience rule on the memorization record, and a run that reaches the step cap counts only if its record growth has fallen below 0.1% per 10^4 steps. Every likelihood that enters a reported bit count is evaluated on a single fixed GPU pool with a pinned software stack: training may be heterogeneous, measurement is not.

4.3 Main Results: Capacity of Video-Language Models

Table 1 reports the operational capacity estimate per trained parameter at every grid point together with the fitted constants, and Figure 4 shows the memorization curve behind every cell. Figure 3 plots memorized bits against dataset size: the curves follow the diagonal in the data-limited regime and approach a plateau in the capacity-limited regime. Across the ten measured sizes, the pooled origin-through fit gives $\alpha_{\text{video}} = 2.54$ bits/param in bf16 on both video tracks, compared with $\alpha_{\text{text}} = 3.25$ bits/param on the same stack. The per-model video mean is lower because the ratio varies with size and the pooled fit is dominated by the largest models. We therefore report both statistics and interpret the pooled value only over the measured range. The keyless replication calibrates the full pipeline: mid-range sizes fall within the published 3.51 ± 0.1 bits/param (Morris et al., 2026), and the largest variation occurs in the deepest, narrowest shapes, as in the original grid. Independence in the random-assignment distribution, together with the held-out-index check, supports interpreting the measured reduction in codelength as sample-specific storage. All constants remain lower bounds under the stated optimization and finite search.

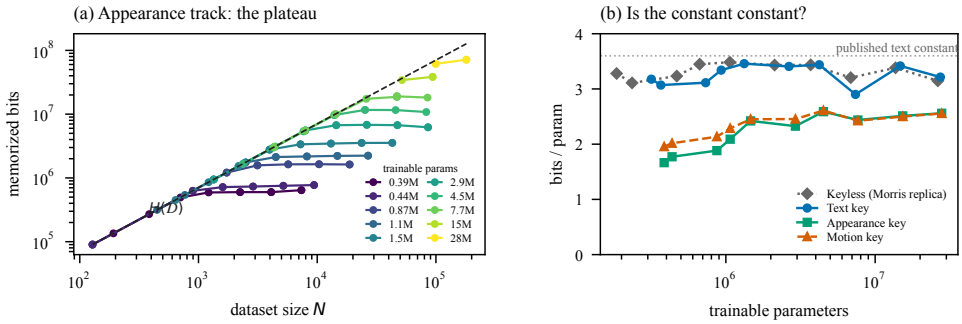

Figure 3: **(a)** Memorized bits against dataset size on the appearance track, one curve per model size, with the dataset entropy $H(\mathcal{D})$ dashed. **(b)** Capacity per trained parameter against model size on all four tracks, with the published text constant dotted.

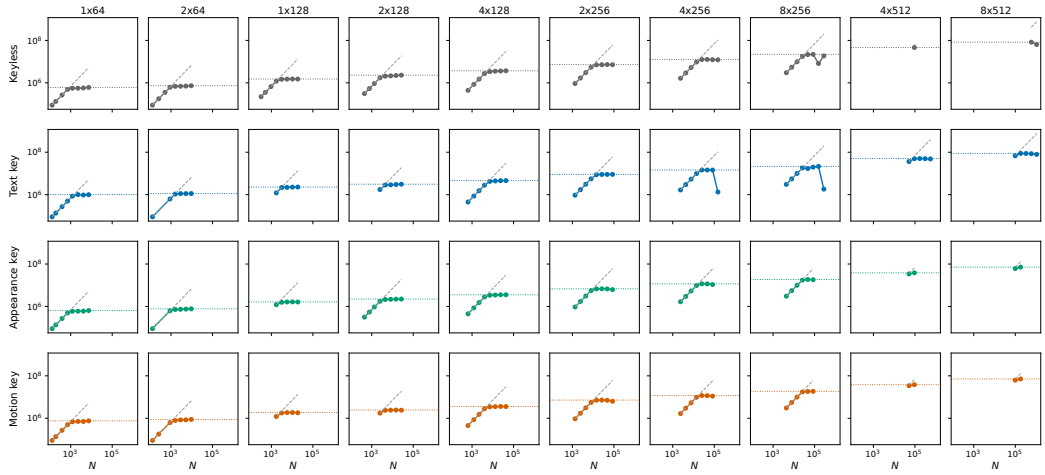

Figure 4: The raw curve behind every cell of Table 1: memorized bits against dataset size N (log–log), one panel per model size and key track. The grey dashed line is the dataset entropy $H(\mathcal{D})$; the dotted line marks the reading the table prints, the maximum over N.

4.4 Cross-Modal Binding Costs

The γ column of Table 1 compares the parameter cost of the same 704-bit random association across key modalities. The pooled slopes give $\gamma = 1.28\times$ for a video key relative to a text key. The two video tracks share one pooled slope and agree closely at all but one of the larger sizes. The per-size γ generally declines across the grid, although not monotonically, consistent with a fixed perception overhead being amortized as the trainable store grows.

Two measured factors are associated with the binding-cost gap. Excluding the perceptual stem from the denominator moves the video-to-text capacity ratio from above one at the smallest size to below one at the largest. Only the stem is excluded because each stem parameter adds less than one stored bit, whereas each text-key embedding parameter adds at least 2.9 at nine of ten sizes. In a matched 2×256 ablation, enlarging the patch until the video and text keys contain the same number of tokens removes the remaining difference under this normalization. Within the tested configurations, the observed gap is therefore consistent with two factors: the parameter cost of the perceptual stem and the effect of key length on per-token storage. We do not claim that visual content is irrelevant beyond these configurations.

4.5 The Forgetting Onset

Figure 5 tracks retention through a ten-chunk stream whose cumulative entropy crosses the stream-specific union-training ceiling at the third chunk. Under naive fine-tuning, retention of a chunk

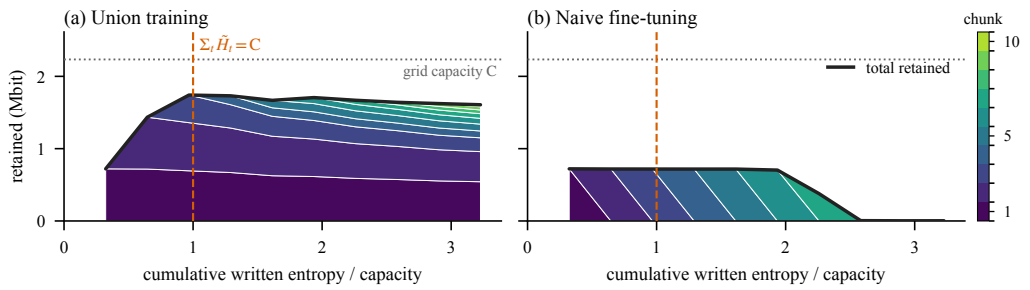

Figure 5: Forgetting onset under **(a)** union training and **(b)** naive fine-tuning. Retained bits per chunk are stacked by chunk index, so band thickness is one chunk's retained information and the top of the stack (black) is the total. The dotted line, labelled C in the plot, is the finite-grid estimate $\widehat{\mathrm{C}}$, which also normalizes the horizontal axis. The dashed line marks cumulative written entropy equal to $\widehat{\mathrm{C}}$, and the flat top of the union-training stack is the stream-specific ceiling $\mathrm{C}_{\mathrm{stream}}$.

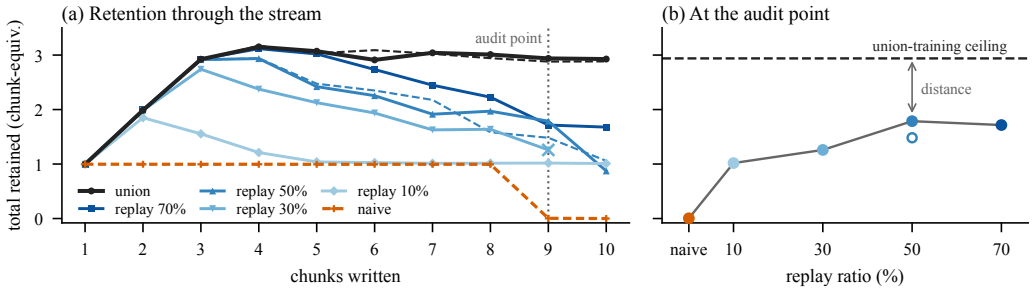

Figure 6: Continual-learning audit on the ten-chunk stream, in chunk-equivalents of 2.97×10^6 bits. **(a)** Total retention after each chunk. Second seeds are drawn thin and without markers, and the cross marks the replay 30% run, which terminated in chunk 10. **(b)** Retention at the nine-chunk audit point against replay ratio, with the union-training ceiling dashed and the second seed hollow.

drops to zero after training on the next chunk even below this empirical ceiling. At all three tested sizes, the steps required to acquire each new chunk increase until acquisition fails. This behavior is consistent with the loss of plasticity described by Dohare et al. (2024). Under union training, retention is nearly lossless below its plateau, and total retention saturates about one fifth below the finite-grid estimate at all three sizes and in one replicated seed. At all three sizes most phases stop before a joint run peaks, so limited per-phase optimization may explain the gap. Across the three tested chunk-entropy ratios, the first lossy chunk is consistent with the union-training ceiling, which is estimated from the same runs, while the independent grid estimate places it up to two chunks later. Below that ceiling, forgetting is attributable to optimization; above it, every tested replay configuration exhibits forgetting. These experiments support the ledger interpretation for the tested protocols but do not establish the union curve as a universal capacity upper bound.

4.6 Auditing Continual Learning Against the Ledger

Figure 6 compares each method with the highest retention observed on the same stream, achieved by union training. Panel (a) follows total retention through the stream, and panel (b) evaluates all methods after nine chunks, when cumulative entropy exceeds twice the finite-grid estimate. The union result is a stream-specific empirical ceiling, not the information-theoretic lower bound on forgetting in Eq. 5. Retention increases with the replay ratio and then saturates below the union result. Increasing the ratio further does not improve retention in these runs: both seeds at 50% replay peak midway through the stream and then decline. At the same evaluation point, naive fine-tuning retains neither earlier chunks nor the most recently trained chunk because it fails to acquire the latter. Figure 7 shows the per-chunk retention behind these totals.

4.7 Ablation Studies

Entropy accounting sanity check. Sweeping answer length at one mid-sized model crosses the data-limited/capacity-limited boundary; measured memorization tracks the parameter-free predic-

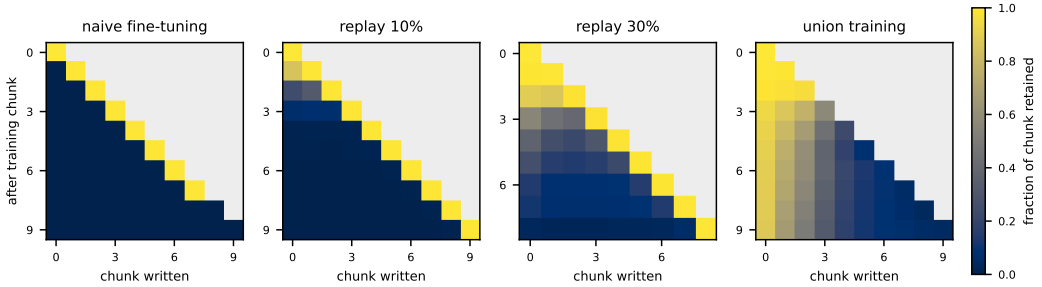

Figure 7: Retention matrices over the ten-chunk stream. Entry (t, j) is the fraction of chunk j retained after training through chunk t. Naive fine-tuning keeps at most the diagonal, replay widens the retained region with its budget, and union training fills the triangle up to its plateau. Chunks are indexed from 0, and the replay 30% run stops after nine chunks.

Table 2: Encoder sweep on the ActivityNet anchor at three decoder sizes; only the frozen tower changes. α (all) divides capacity by every trained parameter; α (excl. proj.) removes the single $D \times w$ projection. Two seeds per cell.

Frozen tower	Decoder	D	Proj. params	α (all)	α (excl. proj.)
DINOv2-L	2×128	1024	131K	3.20 / 3.19	**3.83 / 3.82**
SigLIP2-so400m	2×128	1152	148K	3.14 / 3.15	**3.84 / 3.84**
Qwen2.5-VL-3B	2×128	2048	262K	2.78 / 2.76	**3.87 / 3.85**
Qwen3-VL-4B	2×128	2560	328K	2.49 / 2.50	**3.72 / 3.72**
Qwen3-VL-8B	2×128	4096	524K	2.08 / 2.07	**3.72 / 3.69**
DINOv2-L	2×64	1024	66K	3.04 / 3.04	**3.88 / 3.88**
SigLIP2-so400m	2×64	1152	74K	2.97 / 2.97	**3.90 / 3.89**
Qwen2.5-VL-3B	2×64	2048	131K	2.55 / 2.57	**3.97 / 4.00**
Qwen3-VL-4B	2×64	2560	164K	2.26 / 2.25	**3.82 / 3.81**
Qwen3-VL-8B	2×64	4096	262K	1.80 / 1.80	**3.79 / 3.80**
DINOv2-L	1×64	1024	66K	3.03 / 3.04	**4.10 / 4.11**
SigLIP2-so400m	1×64	1152	74K	2.89 / 2.94	**4.04 / 4.10**
Qwen2.5-VL-3B	1×64	2048	131K	2.40 / 2.43	**4.10 / 4.14**
Qwen3-VL-4B	1×64	2560	164K	2.10 / 2.11	**3.95 / 3.97**
Qwen3-VL-8B	1×64	4096	262K	1.69 / 1.67	**4.06 / 4.02**
spread (max/min), 2×128 / 2×64 / 1×64				1.54× / 1.69× / 1.80×	1.04× / 1.05× / 1.04×

tion $\min(\widehat{\mathrm{C}}, H(\mathcal{D}))$ closely on both sides of it, replicating the original protocol's consistency check in the conditional setting.

Real-video anchor. We repeat the protocol on natural video, where ActivityNet (Caba Heilbron et al., 2015) clips pass through a *frozen* pretrained vision tower (Bai et al., 2025b), with only a projection and the decoder trainable. Small datasets are memorized almost completely, and the capacity-limited points approach a common plateau across a several-fold range of N. At a comparable decoder size, the estimate is higher than on the synthetic video tracks, consistent with lower perception overhead when a pretrained encoder supplies the features. Additional decoder sizes and one replicated seed show similar values; at the largest decoder, the available corpus remains data-limited and therefore provides only a lower bound. These results support, but do not by themselves establish, transfer of the synthetic-key scaling pattern to natural-video features.

The anchor estimate varies modestly across the tested domains. Repeating the anchor at a fixed configuration on four corpora with different visual statistics—ActivityNet, Charades (Sigurdsson et al., 2016), QVHighlights (Lei et al., 2021) (deduplicated to one clip per source video), and the Kinetics-derived Countix (Kay et al., 2017; Dwibedi et al., 2020)—yields estimates that agree closely across domains and seeds. The limited variation suggests that the estimate is not driven by a single tested dataset, although broader domain coverage is needed before treating it as dataset-independent.

The anchor estimate is similar across the tested encoders after normalization. Swapping the frozen tower across five encoders that differ in pretraining objective and generation—a self-supervised tower that has never seen language (DINOv2; Oquab et al. 2024), a contrastive tower (SigLIP2; Tschannen et al. 2025), and three generative VLM towers from two generations and three sizes (Qwen2.5-VL-3B, Qwen3-VL-4B, Qwen3-VL-8B; Bai et al. 2025b;a)—while holding the

Table 3: Capacity per trained parameter (bits/param) under a second decoder family (RMSNorm, rotary positions, SwiGLU) beside the GPT-2 grid; peak reading, converged runs only. Cells with two values are seeds 0 and 1, and the γ row divides the text slope by the appearance slope; means are over the sizes present in both columns. ‡: seeds matched at N=84,992.

	keyless		text		appearance	
Size	GPT-2	modern	GPT-2	modern	GPT-2	modern
1×64	3.28	3.44 / 3.45	3.18	3.45 / 3.43	1.67	1.75 / 1.74
2×64	3.11	3.45 / 3.49	3.07	3.49 / 3.49	1.77	2.01
1×128	3.23	3.44 / 3.42	3.11	3.42 / 3.39	1.88	2.04
2×128	3.45	3.78 / 3.81	3.34	3.99 / 3.98	2.09	2.53 / 2.52
4×128	3.48	4.19 / 4.18	3.46	4.28 / 4.20	2.42	3.19 / 3.19
2×256	3.43	3.88 / 3.89	3.41	3.95 / 3.94	2.33	2.45 / 2.50
4×256	3.44	4.12 / 4.03	3.44	4.20 / 4.14	2.59	2.85 / 2.84
8×256	3.20	3.92 / 3.87	2.90	4.17 / 4.18‡	2.44	3.05
mean ± sd	3.33 ± 0.14	3.79 ± 0.30	3.24 ± 0.20	3.87 ± 0.36	2.15 ± 0.34	2.49 ± 0.52
γ at 2×128				1.60 (GPT-2) vs. 1.58 (modern)		

clips, decoder, and N fixed produces substantial variation when all trained parameters are counted. Much of this variation is associated with the projection layer, whose width follows the encoder feature dimension. After excluding the projection from the denominator, the five estimates agree closely at each of three decoder sizes (Table 2). This pattern suggests that the measured constant is primarily associated with the trainable decoder in these configurations. Because the projection can itself store information, especially for the smallest decoders, the projection-excluded values should be interpreted as a diagnostic normalization rather than a literal decomposition.

Beyond one architecture. Table 3 repeats three tracks under a modern decoder family with everything else held fixed. This family stores more per parameter on every measured track and size and varies more across sizes, confirming that the numerical constant is architecture-specific. At the one matched size, its binding cost is close to the GPT-2 value; additional matched sizes are needed to determine whether that ratio is architecture-stable. Sparse-expert results indicate that capacity is more closely associated with the total parameter count than with the parameters activated per token.

Seeds and stability. Sixteen capacity-defining points carry multiple seeds, with spreads of about one percent or less; no seed reverses any ordering reported here.

Per-sample analysis. Membership inference from the raw memorization score remains saturated across the anchor sweep, so partial memorization confers no membership privacy here, and memorability tracks visual distinctiveness. Storage thins as the dataset grows, yet no trained clip's trace reaches zero while never-trained clips score below uniform, so a model held far past its capacity still marks every sample it has seen.

Frame shuffling (negative result). We expected shuffling frame order to destroy motion keys while sparing appearance keys, and it does neither, leaving both video tracks' capacity unchanged to within the seed spread. Even unordered, the set of object positions encodes speed and acceleration, so trajectory identity is invariant to permutation and the motion track measures motion-derived visual statistics rather than temporal order itself.

Precision. Replacing bf16 autocast with fp32 arithmetic, with fp32 weights in both cases, brings the keyless calibration point to within a fraction of a percent of the value reported for fp32 language models (Morris et al., 2026), raises every track by 4–15% at matched dataset sizes, and leaves γ unchanged, so the bf16 constants are conservative.

5 Conclusion

We proposed VidMem, which measures the memorization capacity of video-language models in bits. Over the tested range, the estimate scales linearly at 2.54 bits per trained parameter for the calibration decoder family, although the coefficient is architecture- and format-specific. Video-keyed storage costs 1.28× the parameters of text-keyed storage, a gap largely explained by perception parameters and key length. Forgetting onset is consistent with the stream-specific union-training ceiling, so forgetting can be measured in the same unit as storage. VidMem thus gives multimodal memorization a common unit, which supports capacity-aware model design and continual learning.

AI USE STATEMENT

We used generative AI tools mainly to implement parts of the training and measurement code and to polish the writing. AI tools also gave feedback on the research hypothesis and experimental design and helped interpret results, while the research questions, design decisions and conclusions were made by the authors and every reported claim was checked against the underlying run outputs. The schematic illustrations in Figures 1 and 2 were rendered with AI assistance. We take responsibility for all content of this work.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. It involves no human subjects or animal experiments, and no new data were collected. All experiments use either procedurally generated videos or publicly available research datasets, in accordance with their licenses and terms of use.

REPRODUCIBILITY STATEMENT

We have taken several measures to ensure the reproducibility of our results. All synthetic keys and answers are generated deterministically from integer seeds and can be regenerated bit-exactly, and the natural-video experiments use publicly available datasets. The anonymized source code, which implements data generation, training and measurement, is provided in the supplementary material.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.3, knowledge capacity scaling laws. *arXiv preprint arXiv:2404.05405*, 2024. ICLR 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Sam Buchanan, Druv Pai, Yi Ma, and Valentin De Bortoli. On the edge of memorization in diffusion models. *arXiv preprint arXiv:2508.17689*, 2025.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 961–970, 2015.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium*, 2019.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *USENIX Security Symposium*, 2021.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy (S&P)*, 2022.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *USENIX Security Symposium*, 2023a.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. Quantifying memorization across neural language models. In *International Conference on Learning Representations (ICLR)*, 2023b.

Chen Chen, Enhuai Liu, Daochang Liu, Mubarak Shah, and Chang Xu. Investigating memorization in video diffusion models. *arXiv preprint arXiv:2410.21669*, 2024. DATA-FM Workshop at ICLR 2025.

Ran Cheng. Context channel capacity: An information-theoretic framework for understanding catastrophic forgetting. *arXiv preprint arXiv:2603.07415*, 2026.

Shibhansh Dohare, J. Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A. Rupam Mahmood, and Richard S. Sutton. Loss of plasticity in deep continual learning. *Nature*, 632:768–774, 2024.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Counting out time: Class agnostic video repetition counting in the wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10387–10396, 2020.

Vitaly Feldman. Does learning require memorization? A short tale about a long tail. In *ACM Symposium on Theory of Computing (STOC)*, 2020.

Jonas Geiping, Alex Stein, Manli Shu, Khalid Saifullah, Yuxin Wen, and Tom Goldstein. Coercing LLMs to do and reveal (almost) anything. *arXiv preprint arXiv:2402.14020*, 2024.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Bargav Jayaraman, Chuan Guo, and Kamalika Chaudhuri. Déjà vu memorization in vision-language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2402.02103.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Juyeop Kim, Songkuk Kim, and Jong-Seok Lee. How diffusion models memorize. *arXiv preprint arXiv:2509.25705*, 2025.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

Jie Lei, Tamara L. Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 11846–11858, 2021.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-LLaVA: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5971–5984, 2024. arXiv:2311.10122.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Pierre Marion and Yu-Han Wu. Understanding diffusion models requires rethinking (again) generalization. *arXiv preprint arXiv:2605.06077*, 2026.

John X. Morris, Chawin Sitawarin, Chuan Guo, Narine Kokhlikyan, G. Edward Suh, Alexander M. Rush, Kamalika Chaudhuri, and Saeed Mahloujifar. How much can language models memorize? In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2505.24832.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations (ICLR)*, 2020.

Charles O'Neill. Can a language model learn facts continually in its weights? *arXiv preprint arXiv:2607.11020*, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

Yicheng Qiu and Keiji Yanai. DST-Mamba: Spatial–temporal adapter with boundary-aware mamba for video temporal action detection. *Big Data and Cognitive Computing*, 10(7):231, 2026a.

Yicheng Qiu and Keiji Yanai. Efficient spatial-temporal focal adapter with SSM for temporal action detection. In *IEEE International Conference on Multimedia and Expo (ICME)*, 2026b. To appear; arXiv:2604.09164.

Yicheng Qiu, Li Niu, and Feng Sha. Multipath 3D-conv encoder and temporal-sequence decision for repetitive-action counting. *Expert Systems with Applications*, 249:123760, 2024a.

Yicheng Qiu, Feng Sha, and Li Niu. Efficient temporal attention with state space model for temporal action localization. In *Neural Information Processing (ICONIP)*, pp. 183–197, 2024b.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Akshay Sasi. Bits and memories: Measuring verbatim extraction across llm quantization. *arXiv preprint arXiv:2607.25451*, 2026.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (S&P)*, 2017.

Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision (ECCV)*, pp. 510–526, 2016.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? Investigating data replication in diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Yiding Song and Hanming Ye. Model capacity determines grokking through competing memorisation and generalisation speeds. *arXiv preprint arXiv:2605.09724*, 2026.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Li Wang, Wenyu Chen, Ning Yu, Zheng Li, and Shanqing Guo. Vidleaks: Membership inference attacks against text-to-video models. *arXiv preprint arXiv:2601.11210*, 2026.

Haoran Wei, Yaofeng Sun, and Yukun Li. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*, 2025.

Yuxin Wen, Yangsibo Huang, Tom Goldstein, Ravi Kumar, Badih Ghazi, and Chiyuan Zhang. Quantifying cross-modality memorization in vision-language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2506.05198.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

Shuxin Zhuang, Zi Liang, Runsheng Yu, Hongzong Li, Rong Feng, Shiqin Tang, and Youzhi Zhang. How much information can a vision token hold? A scaling law for recognition limits in VLMs. *arXiv preprint arXiv:2602.02539*, 2026.

Appendix

A Extended Related Work 14
B The Measurement Protocol in Pseudocode 15
C Protocol and Implementation Details 16
D Additional Results 19
E Beyond One Architecture: A Second Decoder Family and Mixture of Experts 25
F Discussion and Extended Limitations 27

A Extended Related Work

Language-model capacity and the memorization literature. Over-parameterized networks can memorize random labels (Zhang et al., 2017), and some memorization is *necessary* for long-tailed generalization (Feldman, 2020). Measurement work has studied canary exposure (Carlini et al., 2019), verbatim extraction (Carlini et al., 2021), and its dependence on model size, duplication, and context (Carlini et al., 2023b). Morris et al. (2026) measure stored bits as the gap between a known entropy and the model's codelength, reporting per-model means of 3.51 bits/param in bf16 and 3.83 in fp32; the widely cited ~ 3.6 is a fit over their full size range. Our keyless track, a calibration replica of their protocol, falls within their stated ± 0.1 band at mid-range sizes under bf16 autocast, and its fp32 control at 2×128 reads 3.84 against their 3.83. Since our precision setting changes arithmetic, not the width of stored weights (Appendix C.6), the fp32 agreement is a calibration check rather than a replication of their precision experiment. Knowledge-capacity scaling laws (Allen-Zhu & Li, 2024) obtain ~ 2 bits per parameter with a different estimator. Capacity-based accounts of grokking (Power et al., 2022; Song & Ye, 2026) and double descent (Belkin et al., 2019; Nakkiran et al., 2020) treat memorization and generalization as competing for a shared resource, and compute-optimal scaling (Kaplan et al., 2020; Hoffmann et al., 2022) implicitly balances capacity against data. We adopt the same unit and extend it across modalities, enabling the binding-cost comparison of Section 4.4.

Why prior metrics are not directly comparable baselines. Similarity- and attack-based evaluations measure observables other than conditional codelength (Table 4). Extraction success alone does not establish parametric storage, since a model can sometimes be induced to emit strings it did not store (Geiping et al., 2024), and membership inference (Shokri et al., 2017; Carlini et al., 2022) reports population-level attack performance rather than per-sample bits. We instead test whether the bit score predicts a conventional observable; as a loss-threshold membership statistic, it achieves the AUC reported in Appendix D.

Table 4: Families of memorization analyses and their reported quantities. Conditional compression yields a per-sample score in bits with an operationally measured plateau; in our experiments, the same score can also be used as a loss-threshold membership statistic.

Family	Reports	Granularity	Limit	Video
Similarity / extraction (Carlini et al., 2021; Somepalli et al., 2023; Chen et al., 2024)	replication events	sample	none	diffusion only
Membership inference (Shokri et al., 2017; Carlini et al., 2022; Wang et al., 2026)	attack AUC	population	none	generation only
Knowledge scaling laws (Allen-Zhu & Li, 2024)	facts/param	aggregate	implied	no
Conditional compression (ours)	bits	sample	measured	yes

Visual memorization. Diffusion-model studies detect replication by similarity search (Carlini et al., 2023a; Somepalli et al., 2023), characterize sample-level memorization onset (Buchanan et al., 2025; Kim et al., 2025; Marion & Wu, 2026), and extend the analysis to video diffusion (Chen et al., 2024). For understanding models, déjà-vu probes (Jayaraman et al., 2024) and cross-modality transfer studies (Wen et al., 2025) expose what vision-language models retain, and membership-inference attacks reach text-to-video systems (Wang et al., 2026). All report similarity, attack success, or accuracy rather than bits against a capacity. Our per-sample analysis (Appendix D) links the two vocabularies, since the raw-memorization score is itself a membership-inference statistic whose AUC stays at 1.0 well past capacity.

Tokens as information carriers. Optical-compression systems (Wei et al., 2025) and vision-token scaling laws (Zhuang et al., 2026) ask how much information a vision token can *transmit* through a frozen model at inference time. We study the storage-side complement, the bits held by the *parameters*. The key-token ablation of Section 4.4 links the two, as lengthening the key consumes sequence capacity without storage benefit.

Continual learning. Classical work quantifies forgetting against baselines and mitigations such as regularization (Kirkpatrick et al., 2017; Li & Hoiem, 2018) or episodic memory and rehearsal (Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017; Rolnick et al., 2019). Dohare et al. (2024) document loss of plasticity under long training sequences, which our naive-track collapse reproduces in a pure-memorization setting. Concurrent work finds that facts added to a language model's weights by sequential updates are often *behaviorally* forgotten without being erased (O'Neill, 2026), a distinction that Eq. 1 resolves by reading partial traces in bits rather than through a pass/fail behavioral test. Closest in spirit, context-channel-capacity theory (Cheng, 2026) bounds forgetting information-theoretically from the *context* side, whereas the ledger of Section 3.3 is stated in bits, is conditional on a capacity ceiling, and is compared with measured retention in the onset experiment.

B The Measurement Protocol in Pseudocode

Three procedures define the protocol: dataset construction (index to key and answer), the training loop with its stopping rules (Algorithm 2), and the measurement pass that turns model likelihoods into bits (Algorithm 3). Notation follows Section 3: N samples, answers of S_a tokens over a vocabulary of size V_a, and $K = S_a \log_2 V_a$ bits of known entropy per answer. Algorithm 1 requires no disk access. Keys and answers are pure functions of an integer index and two seeds, so training-time and measurement-time data are bit-identical by construction.

Algorithm 1 Dataset construction: sample index $\rightarrow$ (key, answer).

Require: index $i \in [0, N)$, global offset o (0 for a single dataset, tn for chunk t of a stream), key lattice with axes $k = 1{:}A$ of b_k bits each ($B = \sum_k b_k$), seeds $s_{\text{key}}, s_{\text{data}}$

if $4(o + N) > 2^B$ **then abort** ▷ separation margin over the *global* index range; a crowded lattice could produce a spurious plateau

$p \leftarrow \textsc{Feistel}_4(i + o; B, s_{\text{key}})$ ▷ bijection on $[0, 2^B)$; cycle-walk when B is odd

for $k = 1$ to A **do**

  $\theta_k \leftarrow \ell_k + \Delta_k \cdot \big((p \gg \sum_{j<k} b_j) \bmod 2^{b_k}\big)$ ▷ ℓ_k axis low, Δ_k axis step

$v_i \leftarrow \textsc{Render}(\theta)$ ▷ video tracks: $T{\times}3{\times}H{\times}W$ clip; text track: θ read as tokens; keyless: $\varnothing$

for $l = 0$ to $S_a - 1$ **do**

  $y_i[l] \leftarrow \textsc{TakeBits}\big(\textsc{Hash}((i + o)S_a + l;\ s_{\text{data}}),\ \log_2 V_a\big)$ ▷ V_a a power of two $\Rightarrow$ no modulo bias

return (v_i, y_i)

Two details of Algorithm 2 matter. The probe runs on a *fixed* subset and is scaled to N only to steer stopping; every reported number comes from a full pass over all N samples. The two exits also differ. The ceiling exit fires in the data-limited regime, where the model has stored everything available, and the patience exit fires in the capacity-limited regime, where the curve has turned over and further training only erodes stored bits.

Reading Figure 1. The drawing is schematic. The protocol's keys are procedurally generated clips and random token prefixes, not the natural examples shown, and the inset line is illustrative with axes not to scale; Figure 3 gives the measured curve over 0.19–27.9M parameters. Prior studies, in the left panel, offer no unit, scale or limit. In Figures 1 and 2, $C(\theta)$ and $C(\hat{\theta})$ denote capacity generically, whereas the main text distinguishes the ceiling $\mathrm{C}^{\star}$, the finite-grid estimate $\widehat{\mathrm{C}}$ and the stream-specific ceiling $\mathrm{C}_{\text{stream}}$. In the experiments, loss begins at $\mathrm{C}_{\text{stream}}$, below $\widehat{\mathrm{C}}$, rather than exactly where panel C3 draws it.

Reading Figure 2. The overview figure is schematic, and three panels combine elements that the protocol keeps separate. Panel A3 shows four appearance keys side by side; the object and its attributes are fixed within a clip, so color and shape vary across keys, not frames. Panels B2 and C1 show appearance and motion examples together, although each sample belongs to exactly one track.

Algorithm 2 Training with peak tracking. The capacity reading is the running maximum, not the final value (Appendix C).

Require: model f_θ, N, steps $T_{\max}$, probe period P, fixed probe subset $\mathcal{P} \subset [0, N)$, patience Π, ceiling fraction ρ=0.995

SELFCHECK() ▷ bit accounting on constructed inputs; a failure aborts before any result is written
$\theta \sim$ init; opt $\leftarrow \mathrm{AdamW}(\eta,\ \beta{=}(0.9, 0.95),\ \lambda{=}0)$; $M^\star \leftarrow -\infty$
$H_{\mathrm{reach}} \leftarrow NK$; **if** track is keyless **then** $H_{\mathrm{reach}} \leftarrow H_{\mathrm{reach}} - N \log_2 V_a$ ▷ first token is unpredictable in principle
for $t = 1$ to $T_{\max}$ **do**
    $\eta_t \leftarrow \eta \cdot \min(1, t/T_{\mathrm{warm}})$; accumulate $2048/B_\mu$ micro-batches of size B_μ sampled i.i.d. from $[0, N)$
    loss $\leftarrow$ token cross-entropy on the answer span only, computed in fp32 under a bf16 autocast
    clip gradients to norm 1; opt.STEP()
    if $t \bmod P = 0$ **then**
        $M_t \leftarrow (N/|\mathcal{P}|) \cdot \textsc{Measure}(f_\theta,\ \mathcal{P})$ ▷ MEASURE returns a sum, so the factor makes M_t a full-dataset estimate; the fixed subset keeps successive probes comparable
        if $M_t > M^\star$ **then** $M^\star \leftarrow M_t$; $t^\star \leftarrow t$; checkpoint θ
        if $M_t \geq \rho H_{\mathrm{reach}}$ **then break** ▷ dataset exhausted: nothing left to store
        else if $t - t^\star \geq \Pi$ **then break** ▷ no new record for Π steps: past the peak
$C_{\mathrm{final}} \leftarrow \textsc{Measure}(f_\theta,\ \text{all } N)$; $C_{\mathrm{peak}} \leftarrow \textsc{Measure}(\theta^{(t^\star)},\ \text{all } N)$
return $\widehat{\mathrm{C}}(N) \leftarrow \max(C_{\mathrm{final}}, C_{\mathrm{peak}})$ ▷ full-dataset passes; the probe only steers stopping

Algorithm 3 Measurement and capacity estimation.

function MEASURE(f_θ, index set $\mathcal{I}$)
    for each batch of indices $\mathcal{B} \subseteq \mathcal{I}$ **do**
        regenerate (v_i, y_i) by Algorithm 1; $z \leftarrow f_\theta(v_i, y_i)$ under teacher forcing
        $\mathrm{cl}_i \leftarrow -\sum_{l=0}^{S_a-1} \log_2 p_\theta(y_i[l] \mid v_i, y_i[{<}l])$ ▷ log-softmax reduced in fp32; answer span only
        $\mathrm{mem}_U(i) \leftarrow \max(0,\ K - \mathrm{cl}_i)$ ▷ coding worse than uniform stores nothing, not negative bits
    return $\sum_{i \in \mathcal{I}} \mathrm{mem}_U(i)$

$\widehat{\mathrm{C}} \leftarrow \max_{N \in \mathcal{N}} \widehat{\mathrm{C}}(N)$ over converged runs ▷ plateau reading; non-converged points excluded
$\alpha \leftarrow \arg\min_a \sum_m \left(\widehat{\mathrm{C}}_m - a\, P_m\right)^2$ over model sizes m ▷ through-origin slope; also reported as the mean of $\widehat{\mathrm{C}}_m / P_m$

Each answer in panel B1 carries $K = S_a \log_2 V_a$ bits. Panel C3 illustrates an oldest-first loss pattern similar to sequential fine-tuning. The bound constrains how much must be lost after a valid capacity ceiling is crossed, not when loss begins relative to a lower-bound estimate or which items are lost first; in our union-training runs, more recent chunks are lost first (Figure 7).

C Protocol and Implementation Details

C.1 Learning rate

The published text-only protocol specifies Adam, a batch size of 2048 and 10^6 steps but no learning rate. We keep the optimizer and batch size (with gradient accumulation where memory requires), cap training at 4×10^5 steps, and sweep the rate on the keyless track at the calibration point of 6.7×10^5 parameters (Table 5). Across an order of magnitude all four rates lie within 6.1%, the ranking is stable, and the two highest are indistinguishable. We adopt 2×10^{-3}, the more seed-robust of the two (3.347 and 3.368 on two seeds, a 0.6% spread). At width 512 this rate suppresses capacity and 5×10^{-4} recovers it: keyless readings rise from 3.08 to 3.38 bits/param at 13.7M parameters and from 2.68 to 3.15 at 26.3M (counts exclude the stem, which keyless models lack). Every width-512 run in Table 1 therefore uses the lower rate on all four tracks, which the aggregation script enforces

from the rate recorded in each run's probe log. Matched 2×10^{-3} controls with the same model and dataset sizes on the video and text tracks at both width-512 shapes confirm the rule beyond the keyless track. All six read lower, by 5.9% and 6.4% (appearance and motion at 4×512), 17.9% (text at 8×512), 50.9% and 55.9% (motion and appearance at 8×512) and 85.6% (text at 4×512). The *instability* follows the key type. Video controls peak early, after 42–46k steps at 4×512 and 6–8k at 8×512, then lose most of their peak, by factors of 23 to 2×10^3, and one diverges outright. Text controls peak late, at 104k and 292k steps, and give up 1.8% and 1.2% afterwards. The *magnitude* of the suppression follows neither key type nor shape; the six span 5.9–85.6% with no clean ordering, and text supplies both the worst cell and, at 8×512, the mildest. The lower rate thus improves stability on the video tracks and capacity on all tracks, by an amount that key type and shape alone do not predict. Two caveats apply. The width-512 readings rest on a thinner dataset-size sweep, with one capacity-limited N at 4×512 (a seed replicate reads 3.385 against 3.383) and two at 8×512, so the maximum over N does less work at these sizes, although both maxima fall at the same N as their 2×10^{-3} counterparts. The rate is also a property of this grid, not a width law. Appendix D finds the optimum back at 5×10^{-4} for one 8×768 shape and higher for another, so depth and track matter alongside width.

Table 5: Learning-rate sweep at the calibration point (keyless track, 2×128, N=16,384, weight decay 0). Peak capacity and the step at which it occurs.

Learning rate	3×10^{-4}	1×10^{-3}	2×10^{-3}	3×10^{-3}
Peak α (bits/param)	3.17	3.28	**3.35**	3.36
Peak step	1.4×10^5	7.2×10^4	6.6×10^4	4.6×10^4

C.2 Weight decay erases stored bits

Our first sweeps used a conventional λ=0.01 and gave memorization curves that peak and then decline. The decline varies systematically with learning rate (Table 6), consistent with decoupled weight decay, which shrinks weights at a rate proportional to $\mathrm{lr} \times \lambda$ and can gradually reduce the stored-bit score once the cross-entropy gradient on memorized data becomes small. Over 3×10^5 steps the two highest keyless rates lose 8.7% and 11.6% of their peak and the text run at 1×10^{-3} loses 12.7%, against 2.3% for each of the two lowest keyless rates. Setting λ=0 removes most of this decline and raises every peak, so we use no weight decay. End-of-training measurement would have underestimated these peaks by up to 13% for an optimization-related reason, not a demonstrated change in intrinsic capacity.

Table 6: Capacity erosion under weight decay (λ=0.01, 3×10^5 steps), by track and learning rate. Decline is measured from each run's own peak.

	keyless				text
lr	2×10^{-3}	1×10^{-3}	3×10^{-4}	1×10^{-4}	1×10^{-3}
Peak α (bits/param)	3.24	3.21	3.12	3.20	3.11
Final α (bits/param)	2.96	2.84	3.05	3.12	2.72
Decline	-8.7%	-11.6%	-2.3%	-2.3%	-12.7%

C.3 Stopping rules and the capacity reading

Even at λ=0 a few percent of late drift remains, so capacity is read at the running maximum. Each record-setting probe writes a checkpoint, and we report the larger of the full-dataset measurements of that checkpoint and of the final weights; the latter wins in about 8% of runs, by at most 0.65%. Training stops after 4×10^4 steps without a record (8×10^4 on the video tracks) or when the probe reaches 99.5% of the dataset's *reachable* entropy. The largest observed record-to-record gaps, 4.0×10^4 and 7.8×10^4, sit at and just inside these thresholds; since the rule censors every gap at its run's threshold, this does not show that the thresholds are generous. The limit has a cost: a text 8×256 run recovers from a mid-training collapse after exactly 4×10^4 idle steps and peaks 1.6×10^5 steps later, 56% above its previous record; a motion 4×256 run closes a 7.8×10^4 gap one probe inside the limit and peaks at 2.2×10^5 steps. Likewise, six of the 119 patience-stopped keyless and text grid runs set their next record on the exact probe at which patience would have expired, and two of the 64 video runs come within one probe of it. We therefore test the threshold with three patience-doubled controls, which yield no gain at 58M, $+0.8\%$ at 52M (Appendix D) and $+5.1\%$

on a slow-starting modern-decoder seed (Table 3); such truncation can only lower a reading. On the keyless track the reachable entropy lies $\log_2 V_a$ bits per sample below $H(\mathcal{D})$, since without a key the first answer token is unpredictable in principle, and a rule based on raw $H(\mathcal{D})$ would never fire. A run that reaches the step cap counts toward capacity only if, at some probe, its record grew by less than 0.1% per 10^4 steps over the five preceding probe intervals. The four cells of Table 1 that rest on such runs (appearance 1×64; motion 1×64, 2×64 and 1×128) peak at 3.84–3.98×10^5 of the 4×10^5 steps and still grew by 0.2–1.1% over the last 10^5 steps, so they are lower bounds even by the protocol's own standard. Capacity for a size is the maximum over an eight-point log grid in N spanning $64\times$, with design spacing $64^{1/7} = 1.81$ and realized spacing $1.8\times$ at every capacity-defining step (1.7–$1.8\times$ nearby). Like longer patience, a finer grid could only raise the estimates, consistent with every constant in this paper being a lower bound.

C.4 Task-stream protocol

A stream partitions one index space into disjoint chunks of n samples. For the 2×128, 2×256 and 4×256 appearance models of Figures 5–7, $n = 1024$, 3200 and 4224, so each chunk carries about a third of the model's grid capacity ($\widehat{\mathrm{C}}/3.1$, $\widehat{\mathrm{C}}/3.0$ and $\widehat{\mathrm{C}}/3.9$); the chunk-entropy-ratio streams on the 2×128 model use $n = 320$, 1024 and 1600. Phase t trains on chunk t alone under naive fine-tuning; on a uniform draw from chunks $0, \ldots, t$ under union training, which is therefore replay at ratio $t/(t{+}1)$; and, under replay at ratio r, on a draw from all earlier samples with probability r and from chunk t otherwise, with no buffer limit. EWC and LwF use naive sampling plus their penalty. Weights carry over between phases; AdamW is re-initialized and its 200-step warmup re-applied at every boundary, with the grid's learning rate 2×10^{-3}, batch 2048 and zero weight decay. A phase stops when its probe reaches 99.5% of the reachable entropy of the probed set, after 2×10^4 steps without a record, or at 10^5 steps. Every 10^3 steps the probe scores the first 4096 samples of the probed set, which is the current chunk under naive fine-tuning, replay, EWC and LwF, and the union under union training. The union's first indices lie in the oldest chunks (at 4×256, chunk 0 alone), so union phases stop on the oldest material's record. Retention is measured on every chunk seen, over its full dataset, at the end of each phase rather than at its peak.

C.5 Key spaces

Appearance keys are drawn from a quantized lattice over shape identity, color, texture frequency and phase. Motion keys fix appearance and vary velocity, acceleration and angular velocity. Their first frame is byte-identical across the track and carries no identity information, although later frames, and even unordered sets of positions, may reveal trajectory-derived identity, as the frame-shuffling result of Section 4.7 shows. All three pathways in Figure 8 bind to answers from the same distribution, which makes their capacity slopes comparable. A Feistel bijection maps key index i to a lattice point, so distinct keys occupy distinct points whose minimum separation equals the lattice spacing however many keys a run uses. Key crowding under rejection sampling could otherwise create a plateau difficult to distinguish from a capacity plateau. A run aborts if the requested N exceeds one quarter of the lattice size.

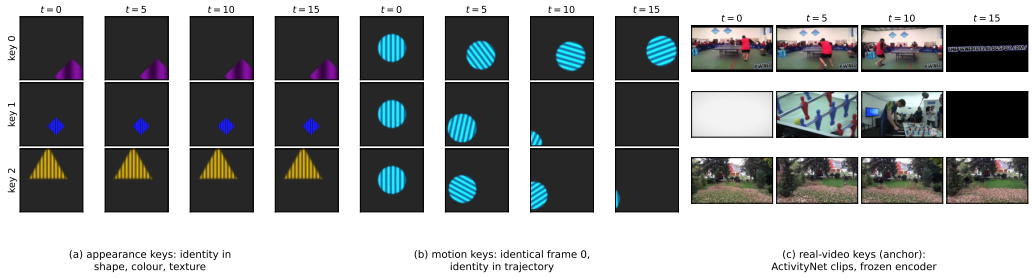

Figure 8: Key pathways. **(a)** Appearance keys carry identity in shape, color, and texture, readable from a single frame. **(b)** Motion keys are byte-identical at $t{=}0$ and differ only in trajectory. **(c)** Real-video keys of the anchor experiments: ActivityNet clips through a frozen pretrained encoder. All pathways bind to the same answer distribution.

Answers are drawn i.i.d. uniform from a counter-mode generator keyed independently of the clip content. In principle, compressing the index-to-answer map below its entropy therefore requires

inverting the generator from N input–output pairs, which we take to be beyond a decoder of $\leq 10^8$ parameters trained by gradient descent. On held-out indices adjacent to the training range, measured memorization is exactly 0.0 bits per sample on every checkpoint tested; the raw codelength is 85–120 bits *worse* than uniform, as the model commits probability mass to its trained patterns.

C.6 Measurement pool and testing

Training may run on any GPU, but every likelihood entering a reported bit count is evaluated on RTX PRO 6000 (sm_120) with a pinned torch 2.8.0/cu128 stack in bf16, and the aggregation script refuses to pool results across measurement fingerprints. Runs trained off the pool keep only their checkpointed peak, and only its pool re-measurement is reported, never a training-GPU measurement of the final weights. Across the 76 runs whose re-measurement is finite, this moves the reading by a median of 0.0004% and at most 1.4%, and the single grid point among them by 0.002%. Evaluation precision is a sub-2% effect with a conservative sign. Six peak checkpoints (all four tracks, two sizes) measure $+0.5\%$ to $+1.5\%$ higher under fp32 than under bf16 autocast, so bf16 evaluation slightly *understates* every constant in this paper. Here bf16 always denotes autocast, with parameters and optimizer state kept in fp32, so the fp32 training controls of Section 4.7 change arithmetic precision, not the width of the stored weights. At matched dataset sizes these controls raise α by 11.3%, 13.3%, 14.5% and 10.7% on the keyless, text, appearance and motion tracks at 2×128, and by 3.8% and 3.6% on keyless and appearance at 1×64, while γ at 2×128 moves only from 1.53 to 1.54. Unit tests check bit accounting against hand-computable cases and assert causal answer scoring (no position sees its own target), a clip cache bit-identical to live rendering, motion keys unreadable and appearance keys readable from a single frame, and approximately zero memorization for an untrained model. The pinning is conservative: one peak checkpoint re-measured on an Ampere card and on a Blackwell card from the pool differs by 0.001%, three orders of magnitude below the seed spread. Further tests cover the path from a correct bit count to a reported constant. The pooled and per-model fits must recover a known slope from synthetic points, take capacity as the maximum over dataset sizes rather than the value at the largest, and refuse a fit with fewer than two sizes; every numeric row of every table in this paper is regenerated from its script and compared against the manuscript, so a hand-edited cell fails the suite.

D Additional Results

Notes on Table 1. Keyless and text models are smaller than video-keyed ones by the stem. Trained parameters range from 0.19M (keyless 1×64, no stem) to 27.9M (video-keyed 8×512); dataset sizes, from $N{=}128$ to 1.07M. Cells marked † either lack larger-N points, which exceed the key-lattice and cache limits (video tracks), or are the one size run at this width's learning rate (keyless; Appendix C). Deep-narrow shapes ($8\times$) have the largest cross-run spread, as in the original text-only grid.

Notes on Figure 4. The panels show the 251 grid points used to fit the constants, drawn from 258 converged runs. Where a later re-run of seven points disagrees, we keep the original grid run rather than the larger reading so that a repeat cannot raise a cell; this lowers two motion cells, by 0.2% and 0.6%. Points on the grey dashed line are fully memorized; points below it are capacity-limited. Three effects appear here and in no table: the plateau is reached and held at every mid-grid size; beyond it, a larger dataset can store slightly *less*, most sharply for text at 4×256 and 8×256; and in the two width-512 columns the keyless and video rows rest on one or two dataset sizes instead of the eight at smaller sizes, so three of those cells are marked as lower bounds.

Numerical detail behind the main-text results. This paragraph gives the statistics behind the qualitative results of Section 4. *Capacity (Section 4.3).* Over ten sizes the origin-through fit has $R^2 = 0.9995$ (appearance), 0.9996 (motion) and $R^2 = 0.998$ (text); bootstrapping over sizes gives $[2.43, 2.56]$ for video and $[3.04, 3.42]$ for text. The per-model means of Table 1 are lower ($\alpha_{\text{video}} = 2.28 \pm 0.29$ over the two tracks' twenty readings; binding cost 1.44 ± 0.17) because both quantities drift with size and the fit gives the two largest sizes about 92% of its leverage. The keyless replica reaches 3.43–3.48 bits/param at mid-range sizes, 0.7–2.2% below the published 3.51 ± 0.1 (Morris et al., 2026). *Binding (Section 4.4).* The motion-over-appearance premium is 1.00; above $\sim$1.5M parameters the video tracks agree within 1% at four of five sizes, the exception being 2×256 (5.5%). The two video keys differ by 17% at the smallest size, where motion

keys require fewer parameters, and by 0.1% at the largest, as perception amortizes. The per-size γ in Table 1 declines from 1.76 to 1.26, though not monotonically (8×256 reads 1.19, below both neighbours). Excluding the perceptual stem puts the video-to-text capacity ratio within $[0.81, 1.26]$ across the grid, from 1.07 (appearance) and 1.25 (motion) at the smallest size to 0.84 at the largest. Enlarging the patch to give video keys the same 8 tokens as text keys closes the residual (3.41 vs. 3.41 bits/param excluding the stem). *Onset (Section 4.5).* Under naive fine-tuning a chunk's retention drops to 0.00 once the next chunk trains; under union training it stays near-lossless (≥ 0.96) while cumulative entropy remains below the ledger's ceiling. Realized capacity (1.63, 1.83, 2.08 bits/param) is a consistent 78–80% of the grid values (2.09, 2.33, 2.59), reproduced by a seed replicate. Phases stop on patience (2×10^4 steps without a record) or a 10^5-step cap (Appendix C.4), short of the 1.64×10^5 and 1.40×10^5 steps the joint runs need to peak at the two smaller sizes. At 4×256, where the joint run peaks at 5.8×10^4 steps, 13 of the 16 phases after the second stop earlier (2.1–5.3×10^4 steps) and the union probe tracks chunk 0 alone. Limited per-phase optimization may therefore explain the gap at all three sizes. On a fixed model with chunk entropy $\widehat{\mathrm{C}}/10$, $\widehat{\mathrm{C}}/3$ and $\widehat{\mathrm{C}}/2$, the first lossy chunk is 8, 3 and 2 against grid predictions of 10, 4 and 2; the grid rule is two chunks late at $\widehat{\mathrm{C}}/10$ and one at $\widehat{\mathrm{C}}/3$, and the two rules coincide at $\widehat{\mathrm{C}}/2$. In Figure 5, onset falls at the third chunk, where cumulative entropy reaches the four fifths of grid capacity that the stream realizes; total retention then stays within 8% over the remaining seven chunks while written entropy more than triples. In Figure 7, naive fine-tuning keeps only the diagonal and loses even that in the final rows, where plasticity collapse blocks acquisition of the current chunk; replay widens the retained region in proportion to its budget; union training fills the triangle to its plateau, beyond which an exposure gradient favours older chunks, seen in more phases, over recent ones. Under naive fine-tuning the steps to each chunk's peak rise while chunks are still acquired, reaching 3.5×10^4, 8×10^3 and 1.5×10^4 at the three sizes, before the chunk's own retention falls to zero at chunk 8, 5 and 9, respectively. *Audit (Section 4.6).* At the audit point of Figure 6 (Table 7), the union-training ceiling is 2.94 chunk-equivalents, replay saturates near 1.7, and cumulative written entropy is $3.1\times$ that ceiling and $2.3\times$ the grid capacity. *Ablations (Section 4.7).* The answer-length sweep (S_a=4→256) tracks $\min(\widehat{\mathrm{C}}, H(\mathcal{D}))$ with mean absolute error 1.1%. On the real-video anchor, four capacity-limited points spanning $3.9\times$ in N share a plateau of 2.73–2.87 bits/param, the binding cost falls to $\gamma_{\text{real}} \approx 1.16$, and the constant moves by 3.3% across $3.6\times$ in trainable parameters, with a fifth size (4.23M) bounding it from below. The four corpora agree within 7% across domains and seeds (5.7% at seed 0), and seed replicates move each estimate by at most 1.2%. Across the five encoders, capacity per trained parameter varies by 54–80%, more for smaller decoders. The projection's width tracks the feature dimension (1024–4096); with it excluded from the denominator, the five towers agree within 4–5% at each of three decoder sizes spanning $3.6\times$ in decoder parameters, and a tower that has never seen language agrees with an 8B generative one within 3.1%. The sixteen replicated points carry 2–4 seeds each, with spreads of 0.01–1.04% (median 0.40%). Membership-inference AUC stays at 1.00 across the anchor sweep. Shuffling frame order moves either video track's capacity by at most 0.9% at three sizes. Training six grid points in fp32 raises the keyless calibration point from 3.45 to 3.84 bits/param, within 0.3% of the 3.83 reported for fp32 language models (Morris et al., 2026), and the appearance track from 2.09 to 2.39; over the six points, fp32 raises the reading by 3.6–14.5% and leaves γ at 2×128 at 1.54 against 1.53.

Table 7: Continual-learning audit at the nine-chunk point, in chunk-equivalents (2.97×10^6 bits each). Distance: gap to the union-training ceiling. Replay 30% is read from its run log at T=9 (run terminated in chunk 10).

Method	Retention (chunk-equiv.)	Union ceiling	Distance
Naive fine-tuning	0.00	2.94	2.94
Replay 10%	1.02	2.94	1.92
Replay 30%	1.26	2.94	1.68
Replay 50%	1.79	2.94	1.15
Replay 70%	1.72	2.94	1.22
Union training	2.94	2.94	0.00

Beyond the grid. We swept learning rate and dataset size at four shapes beyond the grid's 28M ceiling, 4×1024 (52M), 8×768 (58M), 8×1024 (103M) and 12×1024 (155M), taking the largest converged reading as elsewhere. Keyless and text agree within 3.1% at each ($3.16/3.07$, $2.92/2.89$, $2.94/2.89$ and $2.85/2.81$ bits/param). All four fall below the mid-grid constants (3.45 keyless and 3.41 bits/param text, the mean over the four mid-grid sizes), by 8–10% at 52M and 15–18% at the three larger sizes, and the deficit does not grow with scale among them.

The data do not support two candidate explanations. The first is the learning rate. At 4×1024 and N=401,408, lowering it from 5×10^{-4} to 2×10^{-4} raises the reading from 2.77 to 3.14 bits/param, and 1×10^{-4} overshoots back to 2.95, but the optimum is not a simple function of width. For text at 8×768, a three-point bracket at fixed N puts it at 5×10^{-4} (2.52 at 1×10^{-3}, 2.82 at 5×10^{-4}, 2.51 at 3×10^{-4}), the width-512 rate rather than the $1/\text{width}$ prediction of roughly 3×10^{-4}. Keyless at this shape peaks at 10^{-3}, bracketed on both sides at that N (2.74, 2.78 and 2.92 bits/param at 3×10^{-4}, 5×10^{-4} and 10^{-3}), and does not train at 2×10^{-3}, where neither of two seeds exceeds 0.25 bits per sample in its first 1.2×10^{5} steps, a thirtieth to a hundred-and-fiftieth of the 6.8–37 that healthy runs of this shape reach by their third probe, a failure of descent rather than a capacity reading. Depth and track shift the optimum alongside width, so no single rate follows from width alone. Width instead tracks the *ceiling*, the largest rate that still trains, which falls as the model widens; 2×10^{-3} carries the grid to width 256, already suppresses capacity at 512 (Appendix C) and does not train at 768. Across our sweeps the best rate is worth 2–13%, well short of the 8–18% gap. The second is the step budget. Only the 52M text run hit the step cap, still setting small records (the last at 394k of 400k), so its 3.07 is a lower bound, likely within a few percent; all other beyond-grid runs stopped on patience. The 52M keyless reading comes from a patience-doubled run that gained one further record and $+0.8\%$ over the best standard-patience run; the stopping rule thus accounts for about 0.8 points of the gap there, already included in the quoted reading, and the same doubling at 58M recovers nothing. At the other beyond-grid points, record-gap statistics are censored by the patience itself and cannot alone rule out truncation.

The four readings instead follow shape rather than scale. The one wide-shallow model, 4×1024, reads highest despite being the smallest; the three deep-narrow ones agree within 3.4% (keyless) and 2.8% (text) across a $2.7\times$ range of parameters. This extends the deep-narrow penalty inside the grid, where 8×256 and 8×512 are the low points of Table 1 (for keyless the 8×512 low point partly reflects the width-512 rate; for text it does not), to shapes the grid could not reach. We therefore state the linear law over the measured range and report these four as converged lower bounds at shapes outside the grid, consistent with the shape effect the grid already shows, not as evidence that capacity per parameter decays with size.

Frame order does not carry the motion identity (a negative result). We expected shuffling frame order in training and measurement to destroy motion keys but spare appearance keys. Instead, across three sizes it moves capacity by at most 0.9% on either video track (e.g. 2.29 vs. 2.29 bits/param for motion at 1.07M). As Figure 8 shows, even the unordered *set* of object positions encodes speed and acceleration, so trajectory identity is invariant to permutation. The motion track thus measures motion-derived visual statistics rather than temporal order itself, consistent with the scale-convergence of the two video tracks; none of our keys, and to our knowledge no existing memorization probe, isolates strictly order-dependent storage.

Onset at three chunk-entropy ratios. Section 4.5 predicts the first lossy chunk from the capacity a stream realizes rather than from the grid capacity. For the fixed 1.07M appearance model with per-chunk entropy $\widehat{\mathrm{C}}/10$, $\widehat{\mathrm{C}}/3$ and $\widehat{\mathrm{C}}/2$ of its grid capacity, the first lossy chunk is 8, 3 and 2 under both criteria (total retained below 0.96 of total written, or any earlier chunk losing more than 4% of its own), whereas the grid capacity would predict 10, 4 and 2. Because the realized capacity is read from the union run of the same stream, agreement with it is a consistency check, not an out-of-sample prediction; the grid capacity, the independent predictor, places onset up to two chunks late because these streams realize only 0.75–0.80 of it. Integer samples per chunk (320 and 1600) put the $\widehat{\mathrm{C}}/10$ and $\widehat{\mathrm{C}}/2$ streams 0.85% above their nominal $\widehat{\mathrm{C}}/k$ ratios, moving each prediction one chunk earlier. Only the $\widehat{\mathrm{C}}/3$ stream separates the two rules with margin, its crossing chunk clearing the grid capacity by 29% against 0.85% for the other two, a margin comparable to the 0.72% seed spread at this grid point (Table 12); at $\widehat{\mathrm{C}}/2$ both rules name the same chunk. The realization rate is 0.75, 0.78 and 0.80 of the grid value at the three ratios, falling as capacity spreads over more phases. The 10^{5}-step phase cap is reached only in the last phase of the $\widehat{\mathrm{C}}/10$ stream, in no phase of the $\widehat{\mathrm{C}}/2$ stream, and at the onset chunk of the $\widehat{\mathrm{C}}/3$ stream. A capped phase can only lower retention and so bias onset earlier; at that chunk the reading falls from 0.995 to 0.805, far past the 0.96 threshold, so the placement is robust to a large correction. At a chunk entropy equal to C, the first chunk itself saturates (0.92 of its entropy stored) and the two criteria split between chunks 1 and 2; we report this case as the regime boundary rather than as a test.

Table 8: Regularization pilot on a three-chunk appearance stream at 4×256, below capacity. Retention of each chunk after the third phase and in total, in chunk equivalents; one seed.

Method	λ	Chunk 0	Chunk 1	Chunk 2	Total
union	—	0.98	0.97	0.95	2.91
naive	—	0.00	0.00	1.00	1.00
EWC	10^2	0.00	0.00	1.00	1.00
EWC	10^4	0.00	0.00	0.77	0.77
EWC	10^5	0.00	0.00	0.00	0.00
EWC	10^6	0.00	0.00	0.00	0.00
L2	0.01	0.00	0.00	0.75	0.75
L2	0.1	0.00	0.00	0.00	0.01
L2	0.3	0.00	0.00	0.00	0.00
L2	1	0.00	0.00	0.00	0.00
L2	3	0.00	0.00	0.00	0.00
L2	10	0.00	0.00	0.00	0.00
L2	30	0.00	0.00	0.00	0.00
L2	10^2	0.99	0.00	0.00	0.99
LwF	0.5	0.00	0.00	0.92	0.92
LwF	2	0.00	0.00	0.76	0.76
LwF	8	0.00	0.03	0.26	0.29

Table 9: Sanity check: measured memorization against the parameter-free prediction $\min(\widehat{\mathrm{C}}, H(\mathcal{D}))$ under an answer-length sweep (appearance track, 2×128, N=4096; bits $\times10^6$).

S_a	4	16	128	256
Params	1.061M	1.063M	1.077M	1.093M
$H(\mathcal{D})$	0.180	0.721	5.767	11.534
Regime	data-limited	data-limited	capacity-limited	capacity-limited
Predicted	0.180	0.721	2.251	2.285
Measured	0.180	0.719	2.249	2.374
Error	0.0%	0.3%	0.1%	3.9%

Regularization baselines (pilot). Before the ten-chunk audit, we piloted EWC (Kirkpatrick et al., 2017), LwF (Li & Hoiem, 2018) and an L2 pull towards the previous phase's weights (EWC with an identity Fisher) on a shorter stream of three appearance chunks of 4224 samples on the 4×256 model, totalling 0.77 of its grid capacity (below the ledger's threshold), with one seed, a 2.5×10^4-step phase cap and 6×10^3-step patience. Table 8 lists retention after the third phase. No regularizer retains more in total than naive fine-tuning, and none keeps an earlier chunk except L2 at $\lambda=10^2$, which holds chunk 0 by failing to learn chunks 1 and 2. A larger phase budget moves no total by more than 0.03 chunk equivalents. As the stream stays below capacity, these losses are not forced by the ledger, consistent with the audit's reading that forgetting below the ceiling is attributable to optimization. A plausible cause of the LwF result is that answers are independent of keys, so the old model's outputs on new keys carry no information about earlier pairs. Being short and single-seed, the pilot serves as context for the audit rather than as a benchmark of these methods.

The constant is defined at the protocol's answer format. All results pack each answer as S_a=64 tokens over V_a=2048, the format in which the keyless track calibrates against the published text protocol. That protocol uses the same S=64 and V=2048 and scores against the same undeducted $NS\log_2 V$ (Morris et al., 2026), so the calibration is like-for-like at 704 bits per sequence; the keyless track's unpredictable first token costs 11 of those 704 bits on both sides, and neither side is corrected. At the 2×128 keyless calibration point, the same 704 bits read lower in a smaller alphabet. At each vocabulary's best rate, V_a=256 (S_a=88) reads at most 10.2% below the protocol value, V_a=16 (S_a=176) at most 24.3% below, and V_a=4 (S_a=352, rate bracketed on both sides) 32% below (2.34 against 3.45 bits/param); the ordering is monotone across the tested grid. Small alphabets are also harder to train. Of seven runs at V_a=4, three at 10^{-3}–2×10^{-3} agree within 0.15%, one at 5×10^{-4} reads 4.3% lower, one at 4×10^{-3} plateaus 21% lower, one at 8×10^{-3} diverges, and one seed at 2×10^{-3} makes no progress from its first probe, reading 0.97; we list every run and quote the best. Answer length alone is not the cause, since at V_a=2048 the constant moves by under 4% from S_a=128 to 256, the two capacity-limited rows of Table 9. As every comparison holds the format fixed, the ratios (γ, the sparse-expert efficiencies and the cross-family differences) are unaffected; the absolute constants are format-specific.

Notes on Table 9. Parameter count shifts slightly with S_a through the position embedding and is accounted for in the prediction.

Membership inference and per-sample storage. Thresholding the raw per-sample memorization scores of the anchor measurements yields a membership-inference attack. Across the full anchor sweep (N=1,024 to 19,008; $H(\mathcal{D})/\mathrm{C}$ from 0.3 to 5) the attack is *perfect* (AUC 1.000 at every N), even at the deepest capacity-limited point, where the model stores on average only 20% of each answer. Storage thins with N (the best-remembered clip holds 242 of 704 bits, the worst about 64), but no trained clip's trace reaches zero, whereas never-trained clips score 85–120 bits *below* uniform. Partial memorization therefore confers no membership privacy; a model far over its capacity still marks every sample it has seen.



Figure 9: What the model finds memorable: frames of the four most- and four least-memorized ActivityNet clips at N=19,008 (four uniformly spaced frames each; per-clip stored bits at left).

Notes on Figure 9. The frames are real. The most-memorized clips contain hard cuts, title cards and scene changes, distinctive and rare features to bind against; the least-memorized are static single-scene clips whose frames resemble each other and many other videos. Memorability tracks visual distinctiveness, the video analogue of rare-token memorability in language models (Morris et al., 2026) and the per-sample counterpart of the corpus-level ordering in Table 13.

Cross-domain anchor. The four corpora differ in camera work, scene diversity and editing: web activity video (ActivityNet (Caba Heilbron et al., 2015)), static-camera indoor households (Charades (Sigurdsson et al., 2016)), edited YouTube vlogs (QVHighlights (Lei et al., 2021); 12,562 segments deduplicated to 3,515 distinct sources, since two clips of one source are not independent

Table 10: Real-video anchor on ActivityNet (Caba Heilbron et al., 2015): frozen Qwen2.5-VL encoder (Bai et al., 2025b), trainable projection and decoder (0.93M parameters). Two dataset sizes fall on the diagonal and four agree on the capacity plateau.

N (clips)	1,024	2,240	4,864	10,560	16,064	19,008
Memorized / $H(\mathcal{D})$	1.000	0.995	0.742	0.351	0.235	0.200
α (bits/param)	0.77*	1.69*	**2.73**	**2.80**	**2.86**	**2.87**

*data-limited, not capacity-limited: these rows lower-bound α by construction.

Table 11: The real-video anchor across decoder sizes and dataset sizes (ActivityNet, same frozen tower). Parameters exclude the frozen tower and include the projection. †: data-limited run; a lower bound, excluded from the reported spread.

Decoder	Params	N	α_{real} (bits/param)
1×128	0.73M	10,560	2.87
2×128	0.93M	10,560	2.80
2×128 (seed 1)	0.93M	10,560	2.79
4×128	1.33M	10,560	2.97
2×256	2.65M	19,008	2.92
4×256	4.23M	19,008	$\geq 2.96^{\dagger}$
2×128	0.93M	16,064	2.86
2×128	0.93M	19,008	2.87

Table 12: Seed stability at the sixteen mid-grid points with replicates. Capacity per trained parameter for each seed at the listed N, and the max–min spread as a percentage of the mean. Values come from dedicated replicate runs at one dataset size and can differ slightly from the Table 1 cells, which are the maximum over the seed-0 grid runs.

Track	Size	N	α per seed (bits/param)	Spread
keyless	2×128	26,816	3.449 3.443 3.441	0.23%
keyless	2×256	47,360	3.416 3.417	0.01%
keyless	4×128	42,880	3.473 3.509	1.04%
keyless	4×256	45,760	3.442 3.464	0.66%
text	2×128	26,816	3.348 3.359 3.328	0.93%
text	2×256	85,824	3.423 3.413	0.27%
text	4×128	42,880	3.427 3.448	0.62%
text	4×256	25,280	3.439 3.441	0.05%
appearance	2×128	26,816	2.095 2.080 2.087 2.087	0.72%
appearance	4×128	42,880	2.415 2.421	0.28%
appearance	2×256	26,176	2.309 2.320	0.46%
appearance	4×256	25,280	2.599 2.624 2.619	0.95%
motion	2×128	14,784	2.267 2.263 2.271 2.270	0.34%
motion	4×128	23,680	2.460 2.463	0.13%
motion	2×256	26,176	2.447 2.453	0.26%
motion	4×256	25,280	2.609 2.620 2.625	0.61%

keys) and Kinetics-derived actions (Countix (Kay et al., 2017; Dwibedi et al., 2020)). The ordering is stable, with heterogeneous corpora reading slightly higher than the single-scene indoor corpus, whose visually similar keys are harder to distinguish. The spread narrows as the decoder grows, from 6.9% across the four corpora at 0.32M parameters to 2.6% across the three reaching the larger N at 0.93M, so the domain effect is a small-model effect and the constant is, if anything, more domain-stable at the sizes used elsewhere in the paper.

Notes on Table 11. Using the capacity-limited reading at each size, the constant moves by 3.3% over a $3.6\times$ range of trainable parameters (2.87, 2.87, 2.97, 2.92 bits/param) and by 5% over a $3.9\times$ range of N at fixed size; a seed replicate moves it by 0.2%. In the row marked †, the largest corpus we hold (19,994 ActivityNet clips) leaves the 4.23M run 93% data-limited, so its reading is a lower bound, excluded from the spread but consistent with the four capacity-limited sizes.

Notes on Table 12. Replicates cover the four mid-grid sizes on all four tracks; the deepest and largest sizes have one seed each in Table 1. Four further pairs at 8×256, one per track, spread by 0.2% (keyless), 2.4% (text), 1.8% (appearance) and 1.7% (motion); the three keyed pairs lie above the mid-grid median of 0.40%, consistent with the deep-narrow remark in the notes on Table 1.

Notes on Table 2. The five frozen towers are DINOv2-L (Oquab et al., 2024), SigLIP2-so400m (Tschannen et al., 2025), and the vision towers of Qwen2.5-VL-3B (Bai et al., 2025b),

Table 13: Cross-domain anchor at two decoder sizes; entries are capacity in bits/param at the capacity-limited point. QVHighlights deduplicates to 3,515 distinct sources, so it cannot reach the larger block's N and is omitted there rather than run data-limited.

	ActivityNet	QVHighlights	Countix (Kinetics)	Charades
0.32*M parameters,* N=3,456*, two seeds*				
seed 0	2.44	2.42	2.37	2.31
seed 1	2.43	2.44	2.38	2.28
0.93*M parameters,* N=4,864				
seed 0	2.73	—	2.68	2.66

Qwen3-VL-4B and Qwen3-VL-8B (Bai et al., 2025a). All rows use the same 12,000 clips and the same deep capacity-limited N regime, with each tower pooled to one vector per frame so that keys are 16 tokens in every row. Rows are emitted by the released tooling from the run outputs. At 1×64 the projection is 26–58% of the trained parameters and stores bits of its own, inflating the excluded-projection level there; the claim is the cross-tower spread, not the level.

Encoder sweep. Table 2 underlies the encoder-stability claim of Section 4.7 and also makes a methodological point. In any two-stage model the projection's size is set by the frozen tower's feature width, which is arbitrary with respect to storage, and a bits-per-parameter figure that keeps the projection in the denominator inherits that arbitrariness. The 2×64 and 1×64 blocks repeat the sweep with 236K and 186K trainable decoder parameters, against 669K for the 2×128 block, and reproduce the picture. The excluded-projection level drifts upward as the decoder shrinks (3.7–3.9, 3.8–4.0, 4.0–4.1 bits/param), reflecting the limit of the decomposition rather than a property of the towers, since at 1×64 the projection is 26–58% of the trained parameters and necessarily stores some information. The cross-tower spread, which locates the constant in the decoder, stays at 4–5% throughout. The Qwen2.5-VL row also re-extracts the anchor at 224px instead of 140px and lands within 1% of Table 10, so the anchor numbers do not depend on frame resolution.

Per-seed capacities. Table 12 lists per-seed capacities at the sixteen replicated points (two to four seeds each; spread 0.01–1.04%), generated directly from the run outputs by the included code; no seed reverses any ordering reported in the main text.

Key-token ablation detail. The patch ablation of Section 4.4 enlarges the spatial patch from 16×16 to 32×32 (temporal extents 4 and 8), giving 16- and 8-token keys. Total bits/param falls sharply in these configurations (1.37, 0.86 at 2×256) because the stem grows with patch volume while storing almost no information, the same accounting identity that motivates the non-stem normalization.

Front-end normalization. Section 4.4 removes the perceptual stem from the video denominators but keeps the text-key embedding in the text denominator; Table 14 reports both choices. Excluding both front ends leaves γ between 1.15 and 1.60, so symmetric normalization does not close the binding cost. The asymmetric choice rests on the last three columns, which compare each keyed model with the keyless model of the same decoder. Each key-embedding parameter adds at least 2.9 stored bits at nine of the ten sizes (the exception, 8×256, is the grid's low text cell), whereas each stem parameter adds less than one bit, and from 4×128 upward the video models store fewer total bits than the keyless model. The binding cost therefore reflects a front end that adds parameters without adding storage, not a lower storage rate in the decoder; training with a frozen stem would test this directly and is left to future work.

E Beyond One Architecture: A Second Decoder Family and Mixture of Experts

The grid of Table 1 uses one architecture family, the GPT-2-style decoder of the text-only protocol, chosen so that the keyless track could be calibrated against published numbers; two experiments test how much the result depends on this choice.

A second decoder family. We repeat three tracks with a modern decoder (RMSNorm, rotary positions, SwiGLU, no learned position table), as in current open models such as the Qwen family, holding everything else fixed. This family saturates roughly six to eight times more slowly, peaking at the calibration point after 3.9–5.5 $\times$ 10^5 steps against 0.7 $\times$ 10^5 for GPT-2. All runs therefore

Table 14: Front-end normalization on the grid of Table 1. α with each track's front end excluded, the binding cost γ_{sym} with both front ends excluded, and stored bits added per front-end parameter relative to the keyless model of the same decoder.

Size	α excl. front end			γ_{sym}		bits per front-end parameter		
	text	appear.	motion	appear.	motion	key emb.	stem (app.)	stem (mot.)
1×64	5.42	3.39	3.99	1.60	1.36	3.04	0.18	0.75
2×64	4.78	3.23	3.67	1.48	1.30	3.01	0.21	0.75
1×128	4.85	3.44	3.90	1.41	1.24	2.91	0.31	0.87
2×128	4.65	3.31	3.62	1.41	1.29	3.08	−0.17	0.36
4×128	4.31	3.31	3.35	1.30	1.29	3.36	−0.41	−0.29
2×256	4.25	3.18	3.36	1.34	1.27	3.32	−0.61	−0.13
4×256	3.93	3.14	3.16	1.25	1.24	3.46	−1.35	−1.23
8×256	3.12	2.72	2.70	1.15	1.16	−1.06	−4.20	−4.33
4×512	3.68	2.80	2.79	1.31	1.32	3.87	−5.01	−5.13
8×512	3.34	2.71	2.71	1.23	1.23	4.92	−7.21	−7.28

use a 1.5×10^6-step budget, so the patience rule, not the cap, decides when to stop. The learning rate (3×10^{-3}) comes from a sweep designed as in Appendix C, whose top three rates agree to 1%. As Table 3 shows, the modern family stores *more* per parameter on every track and at every size (5–23% on keyless, 8–44% on text, 5–32% on appearance), but less uniformly. Over the same eight sizes, its per-size constants scatter by 8–9% on the keyless and text tracks against the GPT-2 family's 4–6%, and by 21% against 16% on appearance. We therefore report the per-size mean and spread rather than the through-origin slope, whose R^2 of 0.999 is carried by the largest point and would overstate the constant's stability. The binding cost is stable, with γ at 2×128 of 1.58 under the modern decoder and 1.60 under GPT-2. The smallest model reads low on all three tracks, but this size dependence is not a position-encoding effect. Adding a learned position table to the modern decoder changes its keyless capacity by −6% at 1×64 and +0.3% at 2×128, and its text capacity by +1.6% and −0.3%, with no consistent sign across eight points. The unit and the modality cost thus transfer across families while the number does not, and we state the headline constant as a property of the calibration family.

Notes on Table 3. The modern family has no position table and slightly more parameters per block, so the parameter columns differ. The two seeds shown in the cell marked ‡ are matched at N=84,992 and agree to 0.06%. A third seed exists only at N=153,920, where the three seeds read 3.74, 3.38 and 3.92; the 3.38 run made no progress from its first probe (0.05 against 0.6–0.9 for the other two), the small-alphabet failure mode of Appendix D. Re-run at doubled patience, it again makes no progress (0.08) and reads 3.56, about 5% higher but leaving most of its gap to the other two seeds.

Mixture of experts: capacity follows total parameters, not active ones. The dense law ties capacity to the parameters a model *contains*. A mixture-of-experts layer holds E experts but routes each token through one, separating total from active parameters. We replace the MLP of the 2×128 keyless model with $E \in \{2, 4, 8, 16, 32\}$ experts under two routers, a position hash (balanced by construction, with no router parameters) and learned top-1 routing with a load-balance term. Active parameters per token thus stay at 0.67M while total parameters grow to 8.8M, thirteen times the active count. In Table 15, capacity per *active* parameter rises from 3.45 to 38.9, so capacity is not a property of the active path. Capacity per *total* parameter drifts from 3.45 to 2.73; sparse experts store bits at 79–99% of the dense rate, declining slowly with no sign of a floor over a sixteen-fold range of E. The hash router stores 2.8–9.3% more than the learned one at every E under either normalization, with the gap widening from E=2 to E=16 and not beyond, so learned routing decisions themselves consume capacity that the hash router does not require. The reading is not specific to one model, decoder family or track. A second base size (4×256, E=4) gives 96% of its dense rate. Under the modern decoder, E=4 and E=8 hash experts store 97% and 94% of that family's dense rate (3.65 and 3.57 against the same seed's dense reading of 3.78 bits/param; MoE runs use seed 0 only), two to four points above the GPT-2 family at the same E. On the appearance track, capacity per total parameter of the same 2×128 base *rises* with E, from 2.09 dense to 2.29, 2.49 and 2.69 at E=2, 4, 8 (hash), because each added expert is pure storage while the 0.39M perceptual stem is a fixed cost that stores almost no information (§4.4). Excluding the stem from the denominator, as that section does, gives 3.25, 3.16 and 3.11 bits/param per non-stem parameter against 3.31 dense (98, 96 and 94%), the same slow decline as on the keyless track. The dense law therefore extends to sparse models with a mild, measured efficiency factor.

Table 15: Mixture of experts on the keyless track, 2×128 base. Capacity in Mbit at the capacity-limited point; the two ratios divide it by total and by active parameters. Cells with two values are seeds 0 and 1; dense row from Table 1.

Config	Total (M)	Active (M)	C (Mbit)	C / total	C / active
dense	0.67	0.67	2.30	3.45	3.45
E=2, hash	0.93	0.67	3.14 / 3.17	3.38 / 3.41	4.71 / 4.74
E=2, learned	0.93	0.67	3.08	3.31	4.61
E=4, hash	1.45	0.67	4.72 / 4.70	3.25 / 3.24	7.07 / 7.04
E=4, learned	1.45	0.67	4.48	3.08	6.71
E=8, hash	2.50	0.67	7.78 / 7.75	3.11 / 3.10	11.7 / 11.6
E=8, learned	2.50	0.67	7.21	2.88	10.8
E=16, hash	4.60	0.67	13.8 / 13.8	3.01 / 2.99	20.8 / 20.6
E=16, learned	4.60	0.67	12.7	2.76	19.0
E=32, hash	8.79	0.67	25.9	2.95	38.9
E=32, learned	8.80	0.67	24.0	2.73	36.0

Notes on Table 15. Every row activates 0.67M parameters per token, one expert the size of the dense MLP it replaces.

F Discussion and Extended Limitations

What the constant supports, and what it does not. A common unit lets token-compression methods report preserved bits, privacy analyses compare corpus information with an operational capacity estimate, and pruning or low-rank adaptation (Hu et al., 2022) be evaluated on the same scale. For continual learning, a valid capacity upper bound would imply an algorithm-independent retention ceiling, whereas the present experiments provide a lower-bound capacity estimate and a stream-specific empirical ceiling from union training. The reported constants are lower bounds under gradient descent and a fixed budget, measured through the answer channel on synthetic keys with one answer format. The anchor experiments support transfer of the constants to natural video under a frozen pretrained encoder, the second decoder family shows that the unit transfers while the numerical coefficient is architecture-specific (Appendix E), and precision below bf16 is unmeasured.

Synthetic keys give exact entropy accounting at the cost of visual realism, and end-to-end training on natural video entangles perception, generalization, and storage in ways our decomposition only brackets. In the two-stage regime the frozen encoder receives no gradient and stores nothing about the bound answers, which justifies excluding its parameters from the denominator; a *finetuned* encoder would require a joint accounting that we do not attempt. Precision is likewise bracketed rather than covered. We measure bf16 and fp32, where the additional arithmetic precision yields only a modest gain, but not sub-bf16 storage; concurrent extraction work finds memorized sequences surviving down to four-bit weights (Sasi, 2026), suggesting that the capacity curve below bf16 is worth measuring rather than assuming flat. Appendix E measures mixture-of-experts routing with up to 32 experts on the keyless track and up to 8 on appearance, but not larger expert counts, top-k>1 routing, or the text and motion tracks under sparsity. The step budget is a further limitation. The modern decoder family of Appendix E needs six to eight times the steps of the calibration family to peak, and every constant here is a lower bound at the stated budget. Finally, the ledger bounds *how much* must be forgotten once cumulative entropy exceeds a valid capacity ceiling, not which items are lost first; the exposure gradient in Figure 7 suggests that recency and repetition shape the selection, which we leave open.