VAULT: TRUSTWORTHY FINITE MEMORY FOR CONTAMINATED CONTINUAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In contaminated continual learning, replay can turn transient labeling errors into persistent supervision. A finite memory must therefore determine which observations are reliable enough to influence future tasks. We introduce VAULT, a framework that separates participation in current-task learning from admission to long-term memory. VAULT constructs hierarchical evidence in a bootstrapped semantic space, combining granular support and stacked zentropy to assess candidate reliability. Classifier and prototype agreement then certify candidates for storage, while conflict-aware replay preserves the knowledge supported by the resulting memory. Experiments on CIFAR-100 show that VAULT achieves higher final accuracy than the compared replay baselines at the smallest evaluated memory budget across task partitions, together with compact, high-purity storage. Ablations and sample-level analyses connect this advantage to the complementary roles of evidence-based admission and replay preservation. Together, the results show how evidence-based admission builds reliable historical supervision within finite replay memory.

1 INTRODUCTION

Continual learning (CL) requires models to acquire new knowledge while preserving earlier capabilities (Kirkpatrick et al., 2017; Lopez-Paz & Ranzato, 2017). Experience replay addresses this challenge by repeatedly revisiting a finite set of historical observations (Rebuffi et al., 2017; Buzzega et al., 2020). Under contamination, however, the same persistence can propagate labeling errors across tasks: a stored observation continues to influence optimization long after its original task has ended. Memory construction thus determines both which knowledge survives and which errors become recurring supervision. This raises a central question: *what evidence should an observation provide before it enters persistent replay memory?*

Prior work improves replay through gradient selection, interference-aware retrieval, coresets, and compression (Aljundi et al., 2019b;a; Borsos et al., 2020; Yoon et al., 2022; Tiwari et al., 2022; Wang et al., 2022). Noise-aware approaches assess sample reliability through purification, diversity, forgetting dynamics, and prototypes (Kim et al., 2021; Bang et al., 2022; Millunzi et al., 2024; 2026; Li et al., 2025; Mucllari et al., 2025). Persistent replay makes reliability an admission question: an observation can support current-task learning while requiring stronger evidence for long-term storage. Once admitted, it becomes recurring supervision for future tasks. VAULT therefore separates learning participation from storage eligibility. Nominal capacity defines the storage ceiling; occupancy, purity, and predictive performance characterize the memory constructed within it.

We introduce **VAULT (Verified and Adaptive Uncertainty-guided Long-Term Memory)**, which couples evidence-based admission with the preservation of certified history. A bootstrapped semantic encoder supplies granular-ball representations that capture local structural support (Xia et al., 2019; 2023; Cao et al., 2024). Stacked zentropy integrates uncertainty across classes, core support, and fine sub-balls (Yuan et al., 2024; 2025a;b; 2026). Together with semantic agreement, this evidence routes observations into trusted, unresolved, or quarantined states. Trusted candidates qualify for persistent storage through structural, classifier, and prototype checks; quality and coverage then guide their selection within the memory budget.

VAULT pairs reliable admission with historical preservation. Early-learning regularization (Liu et al., 2020) supports current-task learning, while feature-, logit-, and relation-level constraints pre-

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.

serve historical anchors. Conflict-aware projection and bounded restoration regulate their influence on subsequent updates. Training and admission use observed data and model evidence; contamination indicators are reserved for post-hoc auditing.

Contributions.

- **Evidence-controlled memory admission.** We formulate persistent replay as an admission problem, separating an observation's participation in current learning from its eligibility for long-term storage. This distinction connects sample reliability to the construction of finite memory.
- **Coupled admission and preservation.** VAULT combines hierarchical structural evidence, stacked zentropy, and semantic certification with conflict-aware replay. The resulting memory supplies historical constraints to subsequent updates, linking reliable selection to knowledge retention.
- **Higher accuracy with compact, high-purity memory.** At a 500-sample cap and 50% cross-task contamination, VAULT leads the compared replay baselines in FAA on CIFAR-100 with 5, 10, and 20 tasks. On five tasks, it gains 2.65 FAA points over EARL with 7.16% fewer stored samples and 98.66% purity.

2 Related Work

Replay-based continual learning and memory efficiency. Replay-based continual learning preserves historical knowledge by revisiting a limited set of observed examples. GEM and A-GEM use episodic samples to constrain parameter updates, iCaRL uses exemplar rehearsal, and DER and DER++ preserve historical inputs and predictions (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2019; Rebuffi et al., 2017; Buzzega et al., 2020). Subsequent approaches improve representation transfer or combine episodic and semantic memory (Cha et al., 2021; Arani et al., 2022). Gradient-based selection and retrieval identify samples that represent historical optimization directions or are susceptible to interference (Aljundi et al., 2019b;a). Bilevel and online coreset methods optimize representative subsets (Borsos et al., 2020; Yoon et al., 2022; Tiwari et al., 2022), while MRDC increases the number of retained examples through compression (Wang et al., 2022). VAULT complements these approaches by linking memory admission to evidence of sample reliability under contamination.

Noisy and contaminated continual learning. Noise-aware continual learning considers the reliability of replayed data. Self-Purified Replay combines rehearsal with centrality-based purification (Kim et al., 2021), whereas PuriDivER jointly considers memory purity and diversity in contaminated streams with blurry task boundaries (Bang et al., 2022). AER exploits forgetting dynamics and asymmetric balanced sampling, and EARL further develops amnesic replay (Millunzi et al., 2024; 2026). PAA uses prototypes and similarity-based denoising, while SANU examines the contribution of noisy samples to robust continual learning (Li et al., 2025; Nie et al., 2025). Noise-Tolerant Coreset-Based CIL connects robustness to noisy labels with coreset construction (Mucllari et al., 2025). Building on these approaches, VAULT combines structural and semantic evidence to determine eligibility for persistent replay.

Coreset selection and trustworthy memory construction. Coreset methods seek compact subsets that preserve the utility of a larger history. Selection criteria include gradient coverage, adversarial sample value, representativeness, informativeness, diversity, and difficulty (Aljundi et al., 2019b; Shim et al., 2021; Borsos et al., 2020; Yoon et al., 2022; Tiwari et al., 2022; Maharana et al., 2024). Example-forgetting analysis and dataset cartography characterize samples through temporal confidence statistics (Toneva et al., 2019; Swayamdipta et al., 2020). Reducible Loss, ELFS, and Laplace Sample Information examine loss dynamics, proxy training signals, and Bayesian informativeness (Tong et al., 2025; Zheng et al., 2025; Kaiser et al., 2025). These criteria capture complementary aspects of sample utility. VAULT combines reliability, coverage, semantic agreement, and local structure to guide persistent memory allocation.

Structural, semantic, and multi-granularity knowledge preservation. Historical memory also carries semantic and geometric structure. iCaRL organizes exemplars around class means, LUCIR

constrains feature drift and inter-class separation, and PODNet preserves spatial representations through distillation and class proxies (Rebuffi et al., 2017; Hou et al., 2019; Douillard et al., 2020). Co2L and CLS-ER emphasize transferable representations and interactions between episodic and semantic knowledge (Cha et al., 2021; Arani et al., 2022), while recent work uses semantic relationships and higher-order topology for knowledge retention (Sarfraz et al., 2025; Zang et al., 2026). Granular-ball computing provides adaptive local structure (Xia et al., 2019; 2023; Cao et al., 2024), and zentropy characterizes uncertainty at multiple granularities (Yuan et al., 2024; 2025a;b; 2026). VAULT uses these structural and uncertainty signals as evidence for trust routing and memory certification, connecting representation structure to persistent sample selection.

3 METHOD

VAULT separates current-task participation from persistent admission. Figure 1 presents dual encoding, semantic evidence construction, route-aware training, and memory certification with prediction. The bootstrapped semantic encoder provides cross-fold structural evidence; the plastic encoder learns from routed observations and certified replay. Task-end certification and consolidation prepare historical anchors for subsequent tasks, while agreement-gated fusion combines the two readouts at inference.

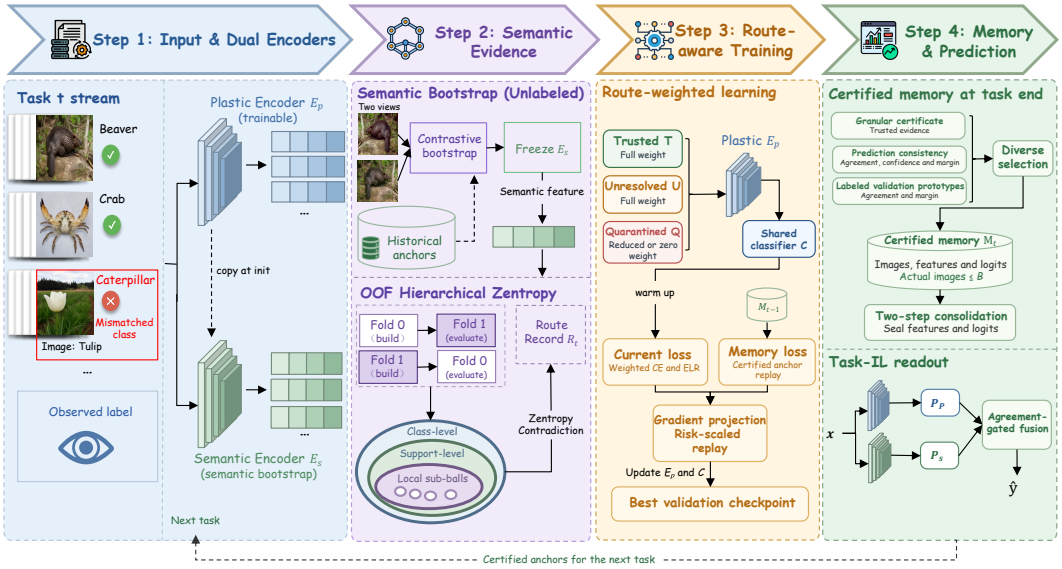


Figure 1: VAULT overview: semantic evidence guides training, memory admission, and historical preservation.

3.1 PROBLEM FORMULATION AND DESIGN OBJECTIVE

Tasks arrive as $\{\mathcal{D}_t\}_{t=1}^{T}$ with observed labels $\tilde{y}_i$ that may be contaminated. VAULT constructs persistent sample memory under the constraint

$$N_t = |\mathcal{M}_t| \leq B, \qquad t = 1, \ldots, T. \tag{1}$$

Here B bounds stored images. Admission uses observable evidence; contamination indicators are reserved for the post-training audit in Section 4. Memory purity and predictive performance jointly assess the resulting allocation.

3.2 EVIDENCE-GROUNDED TRUST ESTIMATION

The semantic encoder E_s is initialized as a copy of the plastic encoder E_p. For each task, label-free contrastive learning adapts E_s using two augmented views and historical anchors. The semantic encoder is then frozen during evidence construction and supervised training, providing a fixed reference space as the plastic encoder and shared classifier are optimized. The structural question is

whether an observation agrees with a supported region of its observed class. Let h_i be its normalized frozen semantic feature and define $\nu(v) = v/\max(\|v\|_2, \epsilon)$, with $\epsilon > 0$. Routing uses two sample-disjoint folds: each candidate is evaluated against observed-class reference regions constructed from the opposite fold. For a nonempty reference region G, its center and descriptive radius are

$$\mu_G = \nu\left(\frac{1}{|G|}\sum_{i\in G} h_i\right), \qquad r_G = Q_{0.90}\big(\{1 - h_i^\top \mu_G : i \in G\}\big). \tag{2}$$

Q_p denotes the empirical p-quantile. The radius summarizes cosine-distance spread; the routing threshold below is calibrated from core similarities. For $|G| \geq s_{\min} \geq 2$, let $k_G = \min(k, |G| - 1)$, where $k \geq 1$, and let $\mathcal{N}_{k_G}(i)$ contain the most similar other members of G. Density-ranked indices define the core:

$$\begin{cases} d_i = \dfrac{1}{k_G}\displaystyle\sum_{j\in\mathcal{N}_{k_G}(i)} h_i^\top h_j, \\ m_G = \min\{|G|, \max(s_{\min}, \lceil \alpha|G| \rceil)\}, \\ C_G = \arg\operatorname{top}^{m_G}_{i\in G} d_i. \end{cases} \tag{3}$$

Here $0 < \alpha \leq 1$. The core center and support floor are

$$\mu_{C_G} = \nu\left(\frac{1}{|C_G|}\sum_{i\in C_G} h_i\right), \qquad s_G = Q_{0.10}\big(\{h_i^\top \mu_{C_G} : i \in C_G\}\big). \tag{4}$$

The core is divided deterministically by sample identifiers into two fine sub-balls. Both require at least two members and sufficient center agreement. For candidate i, $\mathcal{C}_i$ contains the available stable reference classes, and μ_c, s_c denote their core centers and support floors. Evidence is valid only if $\tilde{y}_i \in \mathcal{C}_i$ and $|\mathcal{C}_i| \geq 2$; otherwise the candidate remains unresolved. Fold superscripts are suppressed for readability.

Uncertainty and trust routing. Structural support is combined with uncertainty over valid reference classes. With temperature $\tau > 0$, the class distribution is

$$p_i(c) = \frac{\exp(h_i^\top \mu_c/\tau)}{\sum_{c'\in\mathcal{C}_i}\exp(h_i^\top \mu_{c'}/\tau)}, \qquad c \in \mathcal{C}_i. \tag{5}$$

For a probability vector q over K outcomes, normalized entropy is

$$\mathcal{H}(q) = \begin{cases} -\dfrac{\sum_{k:q_k>0} q_k \log q_k}{\log K}, & K > 1, \\ 0, & K = 1. \end{cases} \tag{6}$$

This definition lies in $[0, 1]$ and uses $0 \log 0 = 0$; numerical evaluation clips probabilities at 10^{-12}. For the observed class, support and local probabilities are

$$\begin{cases} a_i = \sigma\big((h_i^\top \mu_{\tilde{y}_i} - s_{\tilde{y}_i})/\tau\big), \quad q_i = (a_i, 1 - a_i), \\ \ell_i(b) = \dfrac{\exp(h_i^\top \mu_{\tilde{y}_i,b}/\tau)}{\sum_{b'=1}^{2}\exp(h_i^\top \mu_{\tilde{y}_i,b'}/\tau)}, \quad b \in \{1, 2\}. \end{cases} \tag{7}$$

Here σ is the logistic function and $\mu_{c,b}$ denotes a fine sub-ball center. Stacked zentropy combines the class, support, and local distributions through a probability-weighted recursion:

$$Z_i = \frac{1}{3}\left[\mathcal{H}(p_i) + p_i(\tilde{y}_i)\left(\mathcal{H}(q_i) + a_i\mathcal{H}(\ell_i)\right)\right]. \tag{8}$$

The factor $1/3$ normalizes the score to $[0, 1]$. Each entropy component is weighted by the probability supporting its level, so uncertainty is evaluated along the candidate's evidence hierarchy. Entropy measures concentration; semantic agreement identifies whether that concentration favors the observed label. VAULT therefore pairs stacked zentropy with a signed semantic margin and a structural contradiction score:

$$\begin{cases} m_i = h_i^\top \mu_{\tilde{y}_i} - \displaystyle\max_{c\in\mathcal{C}_i\setminus\{\tilde{y}_i\}} h_i^\top \mu_c, \\ \delta_i = \max\{0, -m_i, 2(s_{\tilde{y}_i} - h_i^\top \mu_{\tilde{y}_i})\}. \end{cases} \tag{9}$$

Valid candidates with $\delta_i \geq \delta_{\mathrm{q}}$ enter quarantine. Otherwise, $Z_i \leq z_{\mathrm{t}}$, $m_i \geq m_{\mathrm{t}}$, and $h_i^\top \mu_{\tilde{y}_i} \geq s_{\tilde{y}_i}$ jointly define the trusted route; the last condition is equivalent to $a_i \geq 1/2$. The remaining candidates are unresolved. A per-class participation floor returns selected low-contradiction quarantined samples to the unresolved state when needed. This maintains current-task participation while leaving persistent admission subject to certification.

3.3 Certified memory and replay updates

Trust routing identifies candidates for certification. The admission criterion then combines structural support with agreement from the trainable classifier and validation prototypes. Let u_i and $\hat{y}_i$ denote the classifier's maximum softmax probability and predicted label, and let Δ_i^{cls} denote its top-two probability gap. The corresponding prototype prediction, maximum cosine similarity, and similarity gap are $\hat{y}_i^{\mathrm{p}}$, v_i, and Δ_i^{p}. The full model admits candidates satisfying

$$\mathrm{Cert}(i) = 1 \iff \begin{cases} \text{evidence valid}, \quad \mathrm{route}(i) = \mathrm{trusted}, \\ Z_i \leq z_{\max}, \quad a_i \geq a_{\min}, \\ p_i(\tilde{y}_i) \geq p_{\min}, \quad \delta_i \leq \delta_{\max}, \\ \hat{y}_i = \tilde{y}_i, \quad u_i \geq u_{\min}, \quad \Delta_i^{\mathrm{cls}} \geq \Delta_{\min}^{\mathrm{cls}}, \\ \hat{y}_i^{\mathrm{p}} = \tilde{y}_i, \quad v_i \geq v_{\min}, \quad \Delta_i^{\mathrm{p}} \geq \Delta_{\min}^{\mathrm{p}}. \end{cases} \tag{10}$$

Certification is performed after restoring the best validation checkpoint. Labeled validation prototypes provide reference features, with their source images kept in the validation split. Within each class, uncertainty, quality, and coverage guide diverse selection under budget B. Two memory-consolidation updates precede storage of the resulting reference features and logits. At inference, an available semantic anchor readout is fused with the plastic classifier through an agreement gate: agreeing predictions are blended, and disagreements retain the plastic prediction.

Conflict-aware updates. After warm-up, trusted and unresolved samples retain full training weight; quarantined samples receive reduced or zero weight according to contradiction evidence. The current loss $\mathcal{L}_c$ combines route-weighted cross-entropy, early-learning regularization, and configured semantic alignment. Certified historical memory supplies

$$\mathcal{L}_m = \lambda_{\mathrm{ce}} \mathcal{L}_{\mathrm{ce}}(\mathcal{M}_{t-1}) + \lambda_f \mathcal{L}_{\mathrm{feat}} + \lambda_l \mathcal{L}_{\mathrm{logit}} + \lambda_r \mathcal{L}_{\mathrm{rel}}. \tag{11}$$

The first three terms use normalized sample weights for anchor-label cross-entropy, cosine feature preservation, and logit preservation via $D_{\mathrm{KL}}(p_{\mathrm{stored}} \| p_\theta)$. The relation term measures the mean squared difference between current and stored feature-similarity matrices. With $g_c = \nabla_\theta \mathcal{L}_c$ and $g_m = \nabla_\theta \mathcal{L}_m$,

$$\begin{cases} \beta = \dfrac{\min(0, g_c^\top g_m)}{\max(\|g_m\|_2^2, \epsilon)}, \\ \rho = \min\left\{ \lambda_{\max}, \max\left(0, \dfrac{\mathcal{L}_m - s}{r}\right) \right\}, \\ g_{\mathrm{safe}} = g_c - \beta g_m + \rho g_m, \\ \theta^+ = \mathrm{OptimizerStep}(\theta, g_{\mathrm{safe}}). \end{cases} \tag{12}$$

Here $s \geq 0$ is the memory-loss slack, $r > 0$ scales restoration, and $\lambda_{\max} \geq 0$ bounds its magnitude. The projection term counteracts a negative inner product between current and memory gradients. Restoration adds a memory-loss descent direction when replay loss exceeds the slack. Projection acts on parameters with current-task gradients, treating missing memory components as zero; without replay or when $g_m = 0$, the rule reduces to g_c. This rule targets first-order gradient conflicts; finite-step behavior depends on numerical stabilization, the optimizer, and local loss geometry.

4 Experiments

Evaluation metrics. We report final average accuracy (FAA), average forgetting (Forget), actual memory occupancy (Actual), and memory purity (Purity). Let $a_{t,k} \in [0, 100]$ be test accuracy on task k after learning task t, where $1 \leq k \leq t \leq T$. FAA averages final performance; Forget

measures the decline from each old task's highest attained accuracy:

$$\begin{cases} \mathrm{FAA} = \dfrac{1}{T}\displaystyle\sum_{k=1}^{T} a_{T,k}, \\ \mathrm{Forget} = \dfrac{1}{T-1}\displaystyle\sum_{k=1}^{T-1}\left(\max_{k\le t\le T} a_{t,k} - a_{T,k}\right), \quad T > 1. \end{cases} \tag{13}$$

FAA is reported in percent and Forget in percentage points. Including the final evaluation in the historical maximum makes Forget nonnegative; the last task is excluded because it has no subsequent update. For $T = 1$, the implementation reports Forget as zero.

Let $\mathcal{M}_T$ be the final persistent memory, with occupancy $N_T = |\mathcal{M}_T| \le B$ under nominal budget B. The audit indicator c_i equals zero for an uncontaminated sample and one for a contaminated sample. Memory metrics are

$$\begin{cases} \mathrm{Actual} = N_T, \qquad \mathrm{Purity} = 100P_T, \\ P_T = \dfrac{1}{N_T}\displaystyle\sum_{i\in\mathcal{M}_T} \mathbf{1}\{c_i = 0\}, \qquad N_T > 0. \end{cases} \tag{14}$$

Actual counts retained samples; Purity measures their uncontaminated proportion and is defined for nonempty memories. Audit indicators are used after training. We interpret $\mathrm{Actual}\downarrow$ jointly with FAA, Forget, and $\mathrm{Purity}\uparrow$. Effective clean capacity, $C_{\mathrm{eff},T} = \sum_{i\in\mathcal{M}_T} \mathbf{1}\{c_i = 0\}$, equals $N_T P_T$ for $N_T > 0$ and zero otherwise. This quantity measures the audited uncontaminated samples available for replay. Metrics are averaged equally across runs, so mean Actual can be fractional; aggregation details appear in Appendix B.

4.1 Experimental setup and main results

The main experiments use CIFAR-100 with five tasks of 20 classes under Task-IL, 50% cross-task contamination, and sample caps of 500, 1000, and 2000. Methods share the data and training protocol (Appendix B). Longer sequences and additional-dataset results appear in Appendices C and D.

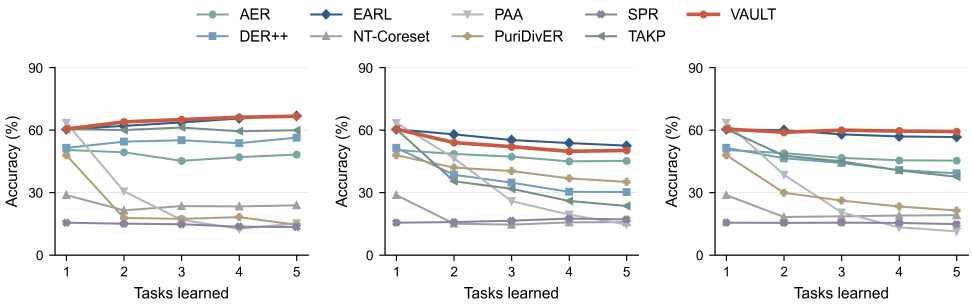
Main comparison. Table 1 compares VAULT with eight replay baselines. The capacity sweep tests accurate prediction under constrained storage, reporting FAA and Forget alongside the occupancy and purity of the memory that supports replay.

Table 1: CIFAR-100 with five tasks: main results under 50% cross-task contamination.

Method	Cap 500				Cap 1000				Cap 2000			
	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑
AER [19]	500	45.30	4.70	94.44	1000	49.35	3.16	95.54	2000	55.67	2.46	96.54
DER++ [6]	500	39.21	18.73	50.60	1000	40.44	16.15	49.30	2000	44.25	11.39	49.56
EARL [20]	500	56.63	8.80	96.32	1000	59.20	7.50	96.58	2000	**62.84**	3.26	96.61
NT-Coreset [21]	500	19.12	6.31	57.16	1000	21.53	6.46	51.66	2000	25.97	5.58	48.65
PAA [15]	396.2	11.36	20.48	53.35	928.8	11.49	23.67	51.34	1852.4	17.40	19.03	52.19
PuriDivER [4]	500	21.35	3.99	58.00	1000	24.27	3.11	53.48	2000	31.85	5.13	54.14
SPR [13]	**192.8**	14.80	**0.77**	72.98	**586.2**	23.06	**0.11**	80.64	1887.6	31.29	**0.59**	78.34
TAKP [38]	500	37.53	28.32	49.52	1000	39.76	24.56	50.24	2000	45.28	19.06	50.01
VAULT	464.2	**59.28**	6.45	**98.66**	871	**61.58**	4.07	**97.92**	**1521.2**	61.86	3.93	**97.60**

VAULT's strongest accuracy advantage appears at the 500-sample cap: it reaches 59.28% FAA, exceeding EARL by 2.65 percentage points, with 6.45 points of forgetting and 98.66% purity. Raising the cap to 1000 and 2000 yields 61.58% and 61.86% FAA, respectively, with purity of 97.92% and 97.60%. VAULT has the highest memory purity among the compared methods at all three capacities. The capacity sweep therefore supports two complementary findings: reliable admission yields a clear accuracy advantage under tight storage, and the retained memory remains highly purified as the nominal budget grows.

Figure 2 separates acquisition and retention through new-task, first-task, and all-learned-task accuracy at the 500-sample cap. Curves and shaded bands denote means and sample standard deviations across runs.



(a) New-task accuracy. (b) First-task retention. (c) All learned tasks.

Figure 2: Learning and retention across the CIFAR-100 task sequence.

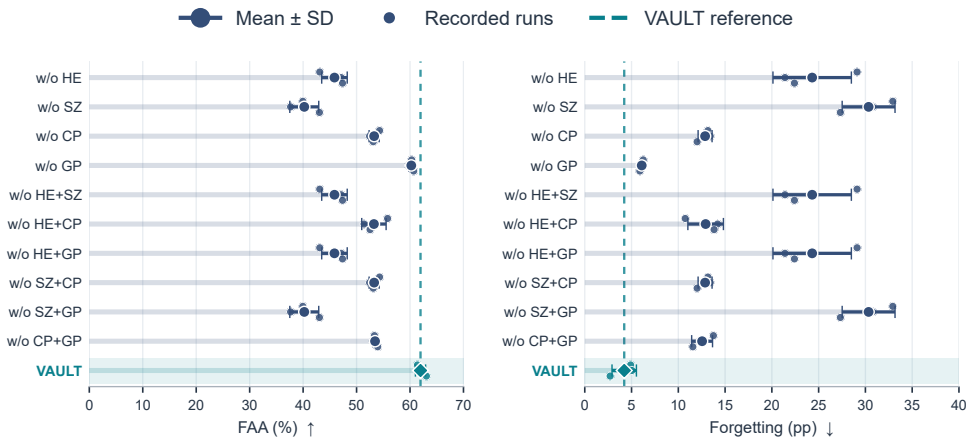
4.2 Mechanism attribution: 5-task ablations

The ablation study evaluates four mechanisms: hierarchical evidence (HE), stacked zentropy (SZ), certified purification (CP), and gradient projection (GP). In Table 2, w/o HE replaces hierarchical routing with flat routing, w/o SZ replaces stacked zentropy with ordinary entropy, w/o CP uses unfiltered balanced selection, and w/o GP disables projection. Paired variants combine these changes. All variants use CIFAR-100 with five tasks of 20 classes, a 1000-sample cap, and 50% cross-task contamination.

Table 2: Component ablations on CIFAR-100.

Variant	FAA ↑	Forget ↓
w/o HE	45.86	24.31
w/o SZ	40.22	30.34
w/o CP	53.29	12.86
w/o GP	60.24	6.09
w/o HE+SZ	45.86	24.31
w/o HE+CP	53.25	12.92
w/o HE+GP	45.86	24.31
w/o SZ+CP	53.29	12.86
w/o SZ+GP	40.22	30.34
w/o CP+GP	53.46	12.54
VAULT (full)	**61.96**	**4.22**

The full model achieves 61.96% FAA with 4.22 points of forgetting. Replacing HE or SZ reduces FAA by 16.10 or 21.74 percentage points, respectively. Certified purification adds 8.67 FAA points over unfiltered selection. With admission intact, projection adds 1.72 FAA points and reduces forgetting by 1.87 points. Figure 3 shows all variants: large markers denote means, error bars indicate sample standard deviations, and small points represent recorded runs. Paired ablations examine how preservation depends on admission.



(a) Final accuracy.

(b) Historical retention.

Figure 3: Component contributions to accuracy and retention.

The paired ablations identify how admission enables preservation. Replacing hierarchical routing or stacked zentropy produces no certified replay memory, and additionally removing GP leaves the corresponding scores unchanged. With certified anchors available, projection improves accuracy and retention. Evidence construction thus determines which historical samples can constrain learning, while projection regulates the updates supported by those samples. Appendix G connects these effects to the resulting memory composition.

4.3 Robustness across contamination levels

We evaluate robustness as cross-task contamination increases from 0% to 75%. Table 3 compares four replay methods on CIFAR-100 with five tasks of 20 classes and a 500-sample cap. Mean results are reported under matched data and training protocols at each contamination level.

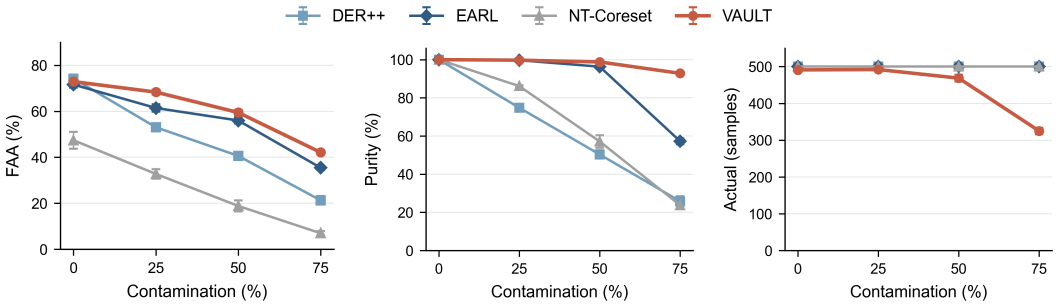
Table 3: CIFAR-100 with five tasks: robustness to cross-task contamination.

Method	0% contamination				25% contamination				75% contamination			
	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑
DER++ [6]	500	**74.24**	15.34	**100.00**	500	53.05	22.69	74.80	500	21.29	15.34	25.80
EARL [20]	500	71.62	14.50	**100.00**	500	61.44	16.22	99.67	500	35.45	6.52	57.27
NT-Coreset [21]	500	47.41	15.52	**100.00**	500	32.76	11.18	86.13	500	6.98	**1.64**	23.73
VAULT	**491**	72.91	**3.15**	**100.00**	**492**	**68.35**	**6.49**	**99.73**	**325**	**42.15**	5.48	**92.82**

VAULT combines strong predictive performance with high-purity memory across the contamination sweep. On the clean stream, it retains 491 samples at 100% purity and achieves 72.91% FAA with 3.15 points of forgetting. At 25% contamination, it reaches 68.35% FAA, exceeding EARL by 6.91 percentage points, with 99.73% purity and 6.49 points of forgetting. At 75% contamination, its 325 retained anchors have 92.82% purity and support 42.15% FAA, exceeding EARL by 35.55 purity points and 6.70 accuracy points. VAULT leads the four compared methods in FAA at both nonzero contamination levels in Table 3.

Figure 4 relates accuracy, purity, and occupancy across contamination levels. All points follow the repetition protocol in Appendix B; shaded bands denote sample standard deviations.

From 25% to 75% contamination, occupancy decreases from 492 to 325 samples, with purity of 99.73% and 92.82%, respectively. At the higher contamination level, the smaller memory supports the highest FAA among the four methods. The following analysis traces how admission constructs this selective historical supervision.



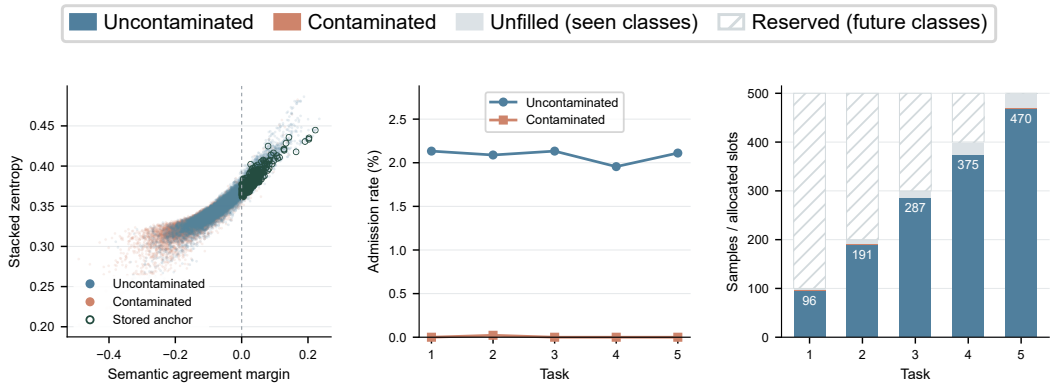
(a) Predictive robustness. (b) Memory purity. (c) Retained samples.

Figure 4: Accuracy and memory composition across contamination levels.

4.4 Memory purity and effective capacity analysis

Figure 5 traces evidence and memory formation across all 45,000 candidates in a five-task stream with 50% contamination and a 500-sample cap.

Panel (a) relates semantic agreement to stacked zentropy, highlighting admitted anchors. Certification also uses structural, classifier, and prototype evidence. Panel (b) compares admission rates: 469 uncontaminated samples and one contaminated sample are retained from equally sized pools of 22,500 candidates.



(a) Candidate evidence. (b) Selective admission. (c) Persistent memory.

Figure 5: Evidence-controlled memory construction.

Panel (c) shows occupancy growing from 96 to 470 samples, ending with 469 uncontaminated entries and 99.79% purity. Grey segments mark unfilled slots for seen classes; hatched segments mark capacity reserved for future classes. This run reaches 61.64% FAA. Across repeated runs at the same cap, VAULT uses 7.16% fewer samples than EARL, with 2.65 points higher FAA and 2.34 points higher purity. The trace explains memory formation, while the aggregate comparison establishes predictive performance. Further diagnostics and memory analyses appear in Appendices F and G.

5 Conclusion

VAULT makes reliable admission the basis of persistent replay. Structural support, stacked zentropy, and semantic certification determine which observations enter memory; conflict-aware preservation carries their supervision into subsequent tasks. On CIFAR-100, this design delivers higher accuracy at the smallest evaluated budget across task partitions while retaining compact, high-purity memory. Ablations connect this advantage to the interaction between admission and preservation, and recorded trajectories reveal how reliable anchors accumulate over tasks. Together, these results support allocating finite memory according to evidence of reliability.

REPRODUCIBILITY STATEMENT

An anonymized implementation of VAULT is available at https://anonymous.4open.science/r/VAULT-3646/. The release provides the CIFAR-100 five-task configuration with 50% cross-task contamination and a 500-sample memory budget. It includes fixed task and sample assignments, the contamination manifest, model and optimization settings, installation and training instructions, and routines for exporting accuracy matrices and auditing memory occupancy and purity. CIFAR-100 is obtained from its official source through the provided data loader. Corruption annotations are isolated from the training data view and used for post-training auditing. The appendix complements the implementation with algorithmic details and experimental protocols.

AI USE STATEMENT

Large language models (LLMs) were used solely for language refinement and polishing of the manuscript. They did not contribute to the conception of the research, design of methods, execution of experiments, or analysis of results.

REFERENCES

Rahaf Aljundi, Lucas Caccia, Eugene Belilovsky, Massimo Caccia, Min Lin, Laurent Charlin, and Tinne Tuytelaars. Online continual learning with maximally interfered retrieval. In *Advances in Neural Information Processing Systems*, volume 32, pp. 11849–11860, 2019a. URL https://proceedings.neurips.cc/paper/2019/hash/15825aee15eb335cc13f9b559f166ee8-Abstract.html.

Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems*, volume 32, pp. 11816–11825, 2019b. URL https://proceedings.neurips.cc/paper/2019/hash/e562cd9c0768d5464b64cf61da7fc6bb-Abstract.html.

Elahe Arani, Fahad Sarfraz, and Bahram Zonooz. Learning fast, learning slow: A general continual learning method based on complementary learning system. In *International Conference on Learning Representations*, 2022.

Jihwan Bang, Hyunseo Koh, Seulki Park, Hwanjun Song, Jung-Woo Ha, and Jonghyun Choi. Online continual learning on a contaminated data stream with blurry task boundaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9275–9284, 2022. doi: 10.1109/CVPR52688.2022.00906. URL https://openaccess.thecvf.com/content/CVPR2022/html/Bang_Online_Continual_Learning_on_a_Contaminated_Data_Stream_With_Blurry_CVPR_2022_paper.html.

Zalán Borsos, Mojmir Mutny, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. In *Advances in Neural Information Processing Systems*, volume 33, pp. 14879–14890, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/aa2a77371374094fe9e0bc1de3f94ed9-Abstract.html.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: A strong, simple baseline. In *Advances in Neural Information Processing Systems*, volume 33, pp. 15920–15930, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/b704ea2c39778f07c617f6b7ce480e9e-Abstract.html.

Xuemei Cao, Xin Yang, Shuyin Xia, Guoyin Wang, and Tianrui Li. Open continual feature selection via granular-ball knowledge transfer. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):8967–8980, 2024. doi: 10.1109/TKDE.2024.3428485. URL https://doi.org/10.1109/TKDE.2024.3428485.

Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2L: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9516–9525, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Cha_Co2L_Contrastive_Continual_Learning_ICCV_2021_paper.html.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019.

Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. PODNet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision*, pp. 86–102. Springer, 2020. doi: 10.1007/978-3-030-58565-5_6. URL https://doi.org/10.1007/978-3-030-58565-5_6.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 831–839, 2019. doi: 10.1109/CVPR.2019.00092. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Hou_Learning_a_Unified_Classifier_Incrementally_via_Rebalancing_CVPR_2019_paper.html.

Johannes Kaiser, Kristian Schwethelm, Daniel Rueckert, and Georgios Kaissis. Laplace sample information: Data informativeness through a bayesian lens. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/ab05dc8bf36a9f66edbff6992ec86f56-Abstract-Conference.html.

Chris Dongjoo Kim, Jinseo Jeong, Sangwoo Moon, and Gunhee Kim. Continual learning on noisy data streams via self-purified replay. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 537–547, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Kim_Continual_Learning_on_Noisy_Data_Streams_via_Self-Purified_Replay_ICCV_2021_paper.html.

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. doi: 10.1073/pnas.1611835114. URL https://doi.org/10.1073/pnas.1611835114.

Shao-Yuan Li, Yu-Xiang Zheng, Sheng-Jun Huang, Songcan Chen, and Kangkan Wang. Prototypes as anchors: Tackling unseen noise for online continual learning. *Neural Networks*, 190: 107634, 2025. doi: 10.1016/j.neunet.2025.107634. URL https://doi.org/10.1016/j.neunet.2025.107634.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/ea89621bee7c88b2c5be6681c8ef4906-Abstract.html.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6467–6476, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/f87522788a2be2d171666752f97ddebb-Abstract.html.

Adyasha Maharana, Prateek Yadav, and Mohit Bansal. $\mathbb{D}^2$ pruning: Message passing for balancing diversity & difficulty in data pruning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f646a4c970799f789e031b63161d018d-Abstract-Conference.html.

Monica Millunzi, Lorenzo Bonicelli, Angelo Porrello, Jacopo Credi, Petter N. Kolm, and Simone Calderara. May the forgetting be with you: Alternate replay for learning with noisy labels. In *British Machine Vision Conference*. BMVA, 2024. URL https://bmvc2024.org/proceedings/680/.

Monica Millunzi, Lorenzo Bonicelli, Angelo Porrello, Jacopo Credi, Petter N. Kolm, and Simone Calderara. EARL: Embracing amnesic replay for learning with noisy labels. *Pattern Recognition*, 179:113514, 2026. doi: 10.1016/j.patcog.2026.113514. URL https://doi.org/10.1016/j.patcog.2026.113514.

Edison Mucllari, Aswin Raghavan, and Zachary Alan Daniels. Noise-tolerant coreset-based class incremental continual learning. *arXiv preprint arXiv:2504.16763*, 2025. URL https://arxiv.org/abs/2504.16763. Preprint.

Hongyi Nie, Shiqi Fan, Yang Liu, Quanming Yao, and Zhen Wang. Using samples with label noise for robust continual learning. *Neural Networks*, 188:107422, 2025. doi: 10.1016/j.neunet.2025.107422. URL https://doi.org/10.1016/j.neunet.2025.107422.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Rebuffi_iCaRL_Incremental_Classifier_CVPR_2017_paper.html.

Fahad Sarfraz, Elahe Arani, and Bahram Zonooz. Semantic aware representation learning for lifelong learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/93b11b5128ced940120f41ce9b216f39-Abstract-Conference.html.

Dongsub Shim, Zheda Mai, Jihwan Jeong, Scott Sanner, Hyunwoo Kim, and Jongseong Jang. Online class-incremental continual learning with adversarial shapley value. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 9630–9638, 2021. doi: 10.1609/aaai.v35i11.17159. URL https://doi.org/10.1609/aaai.v35i11.17159.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 9275–9293, 2020. doi: 10.18653/v1/2020.emnlp-main.746. URL https://aclanthology.org/2020.emnlp-main.746/.

Rishabh Tiwari, Krishnateja Killamsetty, Rishabh Iyer, and Pradeep Shenoy. GCR: Gradient coreset based replay buffer selection for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 99–108, 2022. doi: 10.1109/CVPR52688.2022.00020. URL https://openaccess.thecvf.com/content/CVPR2022/html/Tiwari_GCR_Gradient_Coreset_Based_Replay_Buffer_Selection_for_Continual_Learning_CVPR_2022_paper.html.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019.

Ruilin Tong, Yuhang Liu, Javen Qinfeng Shi, and Dong Gong. Coreset selection via reducible loss in continual learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/24a8d40f6656e542f3fd43bac678e71b-Abstract-Conference.html.

Liyuan Wang, Xingxing Zhang, Kuo Yang, Longhui Yu, Chongxuan Li, Lanqing Hong, Shifeng Zhang, Zhenguo Li, Yi Zhong, and Jun Zhu. Memory replay with data compression for continual learning. In *International Conference on Learning Representations*, 2022.

Shuyin Xia, Yunsheng Liu, Xin Ding, Guoyin Wang, Hong Yu, and Yuoguo Luo. Granular ball computing classifiers for efficient, scalable and robust learning. *Information Sciences*, 483:136–152, 2019. doi: 10.1016/j.ins.2019.01.010. URL https://doi.org/10.1016/j.ins.2019.01.010.

Shuyin Xia, Shaoyuan Zheng, Guoyin Wang, Xinbo Gao, and Binggui Wang. Granular ball sampling for noisy label classification or imbalanced classification. *IEEE Transactions on Neural Networks and Learning Systems*, 34(4):2144–2155, 2023. doi: 10.1109/TNNLS.2021.3105984. URL https://doi.org/10.1109/TNNLS.2021.3105984.

Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. In *International Conference on Learning Representations*, 2022.

Kehua Yuan, Duoqian Miao, Yiyu Yao, Hongyun Zhang, and Xuerong Zhao. Feature selection using zentropy-based uncertainty measure. *IEEE Transactions on Fuzzy Systems*, 32(4):2246–2260, 2024. doi: 10.1109/TFUZZ.2023.3347757. URL https://doi.org/10.1109/TFUZZ.2023.3347757.

Kehua Yuan, Duoqian Miao, Witold Pedrycz, Hongyun Zhang, and Liang Hu. Multigranularity data analysis with zentropy uncertainty measure for efficient and robust feature selection. *IEEE Transactions on Cybernetics*, 55(2):740–752, 2025a. doi: 10.1109/TCYB.2024.3499952. URL https://doi.org/10.1109/TCYB.2024.3499952.

Kehua Yuan, Duoqian Miao, Hongyun Zhang, and Witold Pedrycz. An efficient and robust feature selection approach based on zentropy measure and neighborhood-aware model. *IEEE Transactions on Neural Networks and Learning Systems*, 36(9):16351–16365, 2025b. doi: 10.1109/TNNLS.2025.3565320. URL https://doi.org/10.1109/TNNLS.2025.3565320.

Kehua Yuan, Duoqian Miao, Witold Pedrycz, and Yiyu Yao. Zentropy-enhanced multigranularity knowledge modeling for robust feature selection. *IEEE Transactions on Cybernetics*, 56(7):4096–4107, 2026. doi: 10.1109/TCYB.2026.3665802. URL https://doi.org/10.1109/TCYB.2026.3665802.

Han Zang, Yongfeng Dong, Linhao Li, Liang Yang, and Yu Wang. Topology-aware knowledge preservation for class-incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28059–28067, 2026. doi: 10.1609/aaai.v40i33.40031. URL https://doi.org/10.1609/aaai.v40i33.40031.

Haizhong Zheng, Elisa Tsai, Yifu Lu, Jiachen Sun, Brian R. Bartoldson, Bhavya Kailkhura, and Atul Prakash. ELFS: Label-free coreset selection with proxy training dynamics. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/9eda77f505efbb89462970d739143f73-Abstract-Conference.html.

A VAULT PSEUDOCODE

Algorithm 1 presents VAULT's evidence routing, certified admission, and replay update with the key computations written explicitly. Here, $\mathcal{H}(q) = -\sum_{k:q_k>0} q_k \log q_k / \log K$ for a distribution over $K > 1$ entries, and $\mathcal{H}(q) = 0$ for $K = 1$. Corruption indicators are used for post-hoc auditing, never for training or memory admission.

Algorithm 1 VAULT for contaminated continual learning.

Require: Tasks $\{\mathcal{D}_t\}_{t=1}^{T}$, budget B, semantic encoder, learner θ, validation split;
fixed thresholds, temperature τ, loss weights, restoration parameters $s \geq 0$, $r > 0$, $\lambda_{\max} \geq 0$, $\epsilon > 0$.
Ensure: Trained learner θ_T and certified memory $\mathcal{M}_T$.

Initialize $\mathcal{M}_0 \leftarrow \emptyset$ and copy plastic encoder E_p to semantic encoder E_s.
for $t = 1, \ldots, T$ **do**
  Contrastively bootstrap E_s without labels using two views and historical anchors; freeze E_s.
  Normalize $h_i \leftarrow E_s(x_i)/\max(\|E_s(x_i)\|_2, \epsilon)$; build opposite-fold cores, floors s_c, and sub-balls.
  for each current candidate i **do**
    if its reference evidence is invalid **then**
      Route i to unresolved; **continue**.
    end if
    $p_i \leftarrow \mathrm{softmax}_{c\in\mathcal{C}_i}(h_i^\top \mu_c/\tau)$.
    $a_i \leftarrow \sigma((h_i^\top \mu_{\tilde{y}_i} - s_{\tilde{y}_i})/\tau)$; $q_i \leftarrow (a_i, 1-a_i)$.
    $\ell_i \leftarrow \mathrm{softmax}_{b\in\{1,2\}}(h_i^\top \mu_{\tilde{y}_i,b}/\tau)$.
    $Z_i \leftarrow [\mathcal{H}(p_i) + p_i(\tilde{y}_i)(\mathcal{H}(q_i) + a_i\mathcal{H}(\ell_i))]/3$.
    $m_i \leftarrow h_i^\top \mu_{\tilde{y}_i} - \max_{c\in\mathcal{C}_i\setminus\{\tilde{y}_i\}} h_i^\top \mu_c$.
    $\delta_i \leftarrow \max\{0, -m_i, 2(s_{\tilde{y}_i} - h_i^\top \mu_{\tilde{y}_i})\}$.
    if $\delta_i \geq \delta_{\mathrm{q}}$ **then**
      Route i to quarantine.
    else if $Z_i \leq z_{\mathrm{t}}$ and $m_i \geq m_{\mathrm{t}}$ and $a_i \geq 1/2$ **then**
      Route i to trusted.
    else
      Route i to unresolved.
    end if
  end for
  Enforce the per-class active-sample floor by returning selected quarantined items to unresolved.
  for each training minibatch **do**
    Compute route-weighted CE, ELR, and alignment loss $\mathcal{L}_c$; $g_c \leftarrow \nabla_\theta \mathcal{L}_c$; $g \leftarrow g_c$.
    if historical replay is available from $\mathcal{M}_{t-1}$ **then**
      $\mathcal{L}_m \leftarrow \lambda_{\mathrm{ce}}\mathcal{L}_{\mathrm{ce}} + \lambda_f\mathcal{L}_{\mathrm{feat}} + \lambda_l\mathcal{L}_{\mathrm{logit}} + \lambda_r\mathcal{L}_{\mathrm{rel}}$; $g_m \leftarrow \nabla_\theta \mathcal{L}_m$.
      $\beta \leftarrow \min(0, g_c^\top g_m)/\max(\|g_m\|_2^2, \epsilon)$.
      $\rho \leftarrow \min\{\lambda_{\max}, \max(0, (\mathcal{L}_m - s)/r)\}$; $g \leftarrow g_c - \beta g_m + \rho g_m$.
    end if
    $\theta \leftarrow \mathrm{OptimizerStep}(\theta, g)$.
  end for
  Restore the best validation checkpoint; compute classifier and prototype evidence.
  Keep valid, trusted candidates satisfying $Z_i \leq z_{\max}$, $a_i \geq a_{\min}$, $p_i(\tilde{y}_i) \geq p_{\min}$, $\delta_i \leq \delta_{\max}$.
  Retain those with $\hat{y}_i = \tilde{y}_i$, $u_i \geq u_{\min}$, $\Delta_i^{\mathrm{cls}} \geq \Delta_{\min}^{\mathrm{cls}}$, and $\hat{y}_i^{\mathrm{p}} = \tilde{y}_i$, $v_i \geq v_{\min}$, $\Delta_i^{\mathrm{p}} \geq \Delta_{\min}^{\mathrm{p}}$.
  Select certified anchors with quality/coverage ranking under $|\mathcal{M}_t| \leq B$.
  Take two memory-loss optimizer steps; store post-consolidation features and logits.
end for
return $\theta_T, \mathcal{M}_T$.

The algorithm distinguishes temporary participation from persistent admission. Cross-fold structural evidence and stacked zentropy determine routing, while classifier and validation-prototype agreement certify storage eligibility. Certified replay then supplies the historical gradient used for conflict removal and bounded restoration. Task-end consolidation updates the retained reference features and logits for the next task.

The ordering makes certification the boundary between transient observations and persistent supervision. Current-task learning uses routed observations, whereas future-task replay draws on stored

anchors. The memory passed to the next task is therefore an explicit algorithmic output, with admission and preservation performing distinct operations.

B Experimental Protocols

The evaluation separates a common data protocol from method-specific learning rules. Shared assignments define the observations available at each stage, while each method determines how to learn from them and construct its memory. This distinction supports comparisons of admission and preservation under matched data exposure.

B.1 CIFAR-100 configuration and aggregation

Baseline implementation and configuration. Baseline implementations follow the source papers and, where available, documented details of the corresponding official code. Method-specific optimizers and learning rates retain these source configurations. The shared protocol standardizes the classification backbone, task partition, batch sizes, maximum training epochs per task, data visibility, and contamination assignments. These adaptations preserve each method's learning and memory mechanisms while enabling comparison under the same contaminated Task-IL setting.

The main CIFAR-100 comparison uses PaperResNet18 and five independent runs with seeds 3–7. Methods share the contamination manifest, task order, data visibility, training budget, and nominal sample caps. The main capacity tables and test-accuracy trajectories aggregate these five runs. Ablations and contamination sweeps use seeds 3–5; the 50% point in the contamination-response figure uses the same three-seed subset of the main comparison. The recorded process diagnostics use seed 3 and are reported separately from aggregate benchmark results.

Each CIFAR-100 benchmark metric is computed within a run before averaging. Actual is the mean retained sample count and can be fractional; Purity is the equally weighted mean of per-run proportions. Error bars and shaded bands denote sample standard deviations with an $S - 1$ denominator for S runs. Process trajectories display recorded task-end and epoch-level measurements.

Memory composition is aggregated at the run level. In particular, effective clean capacity is counted within each memory before averaging, preserving the association between occupancy and purity. Multiplying the separately averaged occupancy and purity does not generally recover this quantity. Reported memory metrics therefore retain the same run-level accounting as the predictive metrics.

C Longer Task Sequences

This appendix tests the same finite-memory design over longer task sequences. The 10-task and 20-task comparisons retain the main table's metrics and column layout. Within each partition, the capacity sweep relates predictive accuracy and forgetting to the occupancy and purity of the retained memory.

C.1 Ten-task capacity comparison

The ten-task partition allocates ten classes to each task. Holding the nominal budget fixed while increasing the number of tasks distributes the same storage allowance across more task boundaries. The comparison therefore examines how persistent supervision supports learning over an extended sequence. At the 500-sample cap, VAULT achieves 63.67% FAA, exceeding EARL by 2.49 percentage points while reducing forgetting from 10.37 to 6.86 points (Table 4). It retains 497 samples with 97.99% purity. At caps of 1000 and 2000, purity is 97.74% and 96.75%, respectively. These results extend the accuracy advantage at the smallest budget to ten tasks and show that memory purity remains high as capacity increases.

Table 4: CIFAR-100 with ten tasks: results under 50% cross-task contamination.

Method	Cap 500				Cap 1000				Cap 2000			
	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑
AER [19]	500	49.38	7.45	94.12	1000	55.97	5.11	94.70	2000	59.52	4.53	94.89
DER++ [6]	500	42.74	24.34	50.28	1000	46.07	20.01	50.48	2000	51.47	16.22	51.36
EARL [20]	500	61.18	10.37	94.00	1000	**66.02**	6.92	94.94	2000	**70.26**	3.64	94.85
NT-Coreset [21]	500	28.79	10.10	59.80	1000	29.22	5.20	53.96	2000	32.17	6.80	51.45
PAA [15]	449	13.67	16.71	51.85	981	14.44	17.31	50.80	1926.6	18.34	17.34	52.49
PuriDivER [4]	500	29.45	4.32	52.48	1000	35.42	6.67	59.20	2000	38.33	6.49	52.01
SPR [13]	**328.2**	24.50	**1.45**	82.97	1000	39.22	**1.78**	81.38	2000	44.10	**2.29**	82.17
TAKP [38]	500	38.91	29.72	49.32	1000	42.95	25.40	49.58	2000	45.42	23.02	49.28
VAULT	497	**63.67**	6.86	**97.99**	**971.4**	65.54	4.94	**97.74**	**1858.2**	68.10	2.54	**96.75**

C.2 TWENTY-TASK CAPACITY COMPARISON

The twenty-task partition assigns five classes to each task, extending the sequence while preserving the total class count. Each table row jointly reports prediction, forgetting, occupancy, and purity. This joint view relates historical retention to the predictive capability supported by the stored observations. With 20 tasks, VAULT achieves 69.48% FAA at the 500-sample cap, exceeding EARL by 1.52 percentage points, with 6.71 points of forgetting and 96.41% purity (Table 5). At the 2000-sample cap, it achieves 73.07% FAA, 3.76 points of forgetting, and 95.32% purity. Across the three sequence lengths, the smallest budget consistently combines strong prediction with high-purity memory. FAA comparisons are made within each Task-IL partition because the number of candidate classes per task differs across partitions.

Table 5: CIFAR-100 with twenty tasks: results under 50% cross-task contamination.

Method	Cap 500				Cap 1000				Cap 2000			
	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑	Actual ↓	FAA ↑	Forget ↓	Purity ↑
AER [19]	500	56.93	10.11	89.80	1000	62.36	8.78	90.82	2000	70.76	5.24	92.32
DER++ [6]	500	49.77	21.62	48.68	1000	53.85	18.87	50.18	2000	59.05	14.87	50.25
EARL [20]	500	67.96	10.23	90.36	1000	**71.72**	7.22	91.10	2000	**74.91**	5.53	91.82
NT-Coreset [21]	500	41.90	11.18	62.76	1000	42.78	10.67	59.50	2000	44.72	8.71	54.68
PAA [15]	**486.4**	21.93	14.41	51.68	**912.8**	23.64	14.86	50.68	**1888.6**	23.94	18.35	50.78
PuriDivER [4]	500	39.00	6.43	48.36	1000	48.60	7.35	57.04	2000	50.23	7.20	52.18
SPR [13]	500	45.87	**4.07**	81.16	1000	53.25	**4.37**	85.66	2000	56.74	4.50	85.40
TAKP [38]	500	44.74	30.17	50.28	1000	48.25	26.77	50.28	2000	51.23	23.58	50.23
VAULT	495.6	**69.48**	6.71	**96.41**	989.8	71.52	4.70	**96.26**	1969	73.07	**3.76**	**95.32**

D ADDITIONAL-DATASET RESULTS

We evaluate VAULT on ImageNet100 and WebVision to examine finite-memory continual learning under controlled contamination and naturally noisy web supervision. ImageNet100 comprises five tasks with 20 classes each and 50% cross-task contamination. WebVision uses a 20-class subset partitioned into five tasks with four classes each, retaining its native web-label noise without additional synthetic corruption. Both evaluations use a 500-sample memory cap. Table 6 jointly reports final accuracy, forgetting, realized storage, and elapsed training time, with memory purity additionally reported for ImageNet100.

Table 6: Additional-dataset results under a 500-sample memory cap.

	ImageNet100					WebVision (20 classes)			
Method	FAA (%) ↑	Forget (pp) ↓	Purity (%) ↑	Actual ↓	Time (h) ↓	FAA (%) ↑	Forget (pp) ↓	Actual ↓	Time (h) ↓
DER++ [6]	45.84	24.85	48.40	500	16.57	**75.30**	11.63	**500**	14.01
EARL [20]	50.60	27.68	97.60	500	42.97	72.90	15.75	**500**	23.99
NT-Coreset [21]	13.16	**7.28**	51.00	500	**9.47**	60.20	6.63	**500**	**1.02**
VAULT	**64.26**	19.33	**99.58**	**480**	15.36	72.60	**4.63**	**500**	3.05

On ImageNet100, VAULT achieves 64.26% FAA, exceeding EARL by 13.66 percentage points and DER++ by 18.42 points. Its memory reaches 99.58% purity with 480 retained samples, compared with EARL's 97.60% purity and 500 samples. This combination extends the accuracy and memory-quality advantage to higher-resolution images. VAULT completes the recorded task sequence in 15.36 hours, compared with 16.57 hours for DER++ and 42.97 hours for EARL.

Under native WebVision noise, VAULT achieves 72.60% FAA with 500 stored samples and a recorded elapsed time of 3.05 hours. EARL achieves 72.90% FAA in 23.99 hours, and DER++ achieves 75.30% in 14.01 hours under the same sample cap. NT-Coreset reaches 60.20% FAA in 1.02 hours. VAULT thus combines an FAA within 0.30 percentage points of EARL with a shorter recorded task-sequence time. Together, the two datasets extend the evaluation from controlled cross-task contamination to native web-label noise, highlighting memory quality and the accuracy–time trade-off, respectively.

Both datasets use the same shared training configuration: PaperResNet18 initialized from scratch, 224×224 inputs, and training and test batch sizes of 32 and 128. Each method runs on one NVIDIA RTX 5090 GPU with 32 GB of memory. Within each dataset, methods share task order, data assignments, evaluation splits, and the nominal memory cap. Method-specific optimization settings are retained. DER++, EARL, and VAULT use a 100-epoch task ceiling with validation-based early stopping; NT-Coreset uses its 40-epoch and 20-epoch training phases. Time denotes recorded elapsed task-sequence time, including training, validation, task-end evaluation, and memory operations; it measures end-to-end execution under these protocols.

Actual counts the final retained images. ImageNet100 purity is computed from the known synthetic-contamination assignments. WebVision is evaluated using its native labels; synthetic-corruption bookkeeping does not identify errors in these labels. We therefore characterize WebVision through FAA, realized storage, and elapsed time.

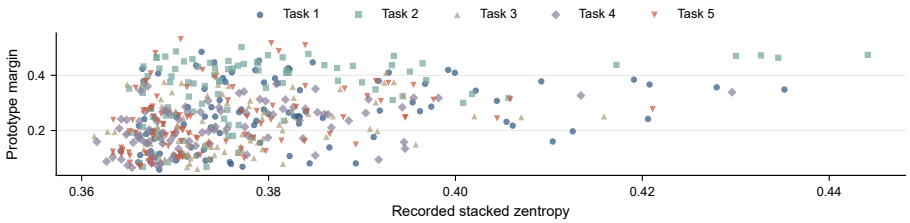
Forgetting is computed from each old task's highest attained accuracy to its final accuracy, using the definition in Section 4. VAULT records 19.33 percentage points of forgetting on ImageNet100 and 4.63 on WebVision. On ImageNet100, it combines higher FAA with 5.53 and 8.35 points less forgetting than DER++ and EARL, respectively. On WebVision, it has the lowest forgetting among the four compared methods, improving on DER++ by 7.00 points and EARL by 11.13 points. These comparisons evaluate retention jointly with acquired predictive capability.

E Recorded Anchor Evidence

The aggregate memory metrics describe how much supervision is retained and its audited composition. The recorded-anchor analysis adds the sample-level view: which evidence accompanies the retained observations, and what images constitute the resulting memory. This links the admission mechanism to the concrete material available for replay.

Figure 6 examines memory from an independent CIFAR-100 checkpoint: 472 anchors under a 500-sample cap, with 99.36% audited purity. Panel (a) plots stacked zentropy against prototype margin, colored by task identity. This checkpoint-level view characterizes the evidence associated with retained observations, complementing the full candidate-to-memory trajectory in Figure 5 and the repeated-run main comparison.

Panel (b) displays one stored anchor per task, selected deterministically by the smallest recorded sample identifier. Each image is recovered by reversing the recorded input normalization. The paired panels connect retained examples with their recorded decision evidence, providing a sample-level account of the memory available for subsequent learning.



(a) Evidence distribution of retained anchors.



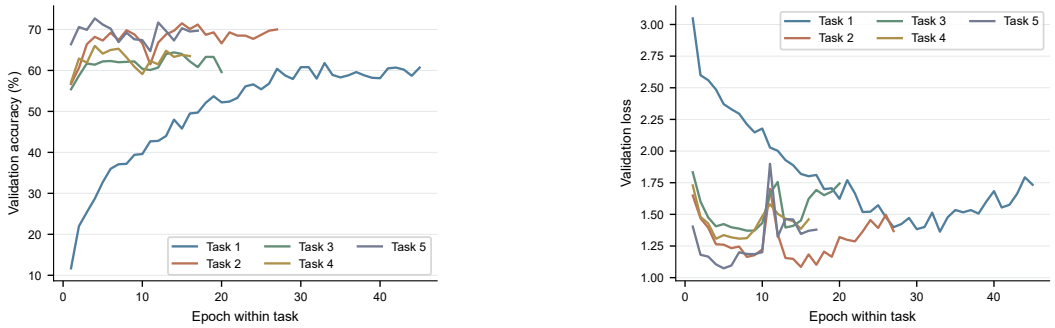
(b) Stored examples from the five tasks.

Figure 6: Recorded anchor evidence and stored examples.

F Process Diagnostics

F.1 Training trajectories

The validation curves describe optimization within each task, whereas the task-end test matrix evaluates preservation across the learned sequence. Epoch indices restart for each task, so curve lengths reflect the task-specific training duration under the stopping rule. These complementary views distinguish adaptation to incoming data from the historical performance sustained after memory updates. Figure 7 traces validation accuracy and loss across five tasks under validation-based early stopping. Together with Figure 5, it links within-task optimization to task-end memory admission. After checkpoint selection, certification, and consolidation, the resulting model achieves 61.64% test FAA.



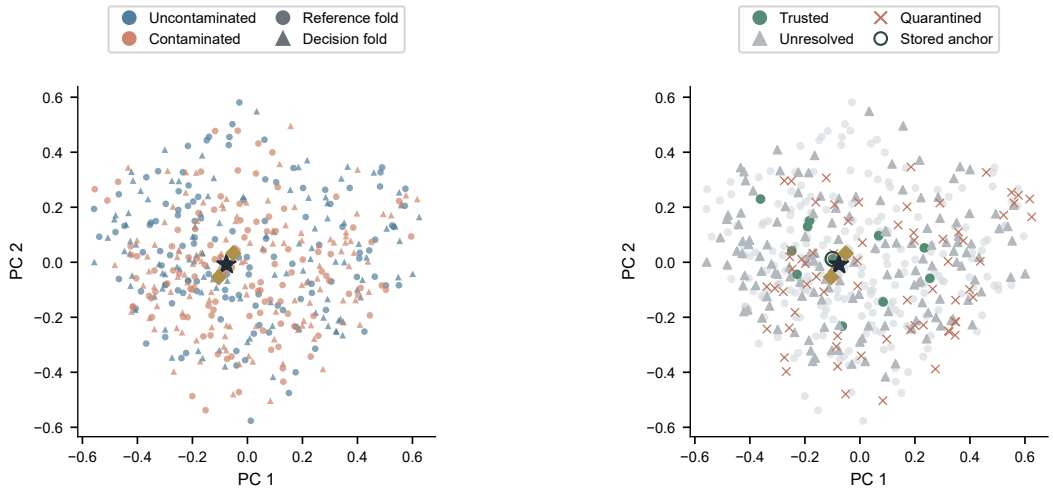
(a) Validation accuracy.

(b) Validation loss.

Figure 7: Within-task learning trajectories.

F.2 Recorded granular structure and admission

The structural diagnostic follows the information flow from reference evidence to candidate routing and storage. Its paired views use common display coordinates to relate local support to the resulting admission decisions. The recorded reference centers and routing states provide the link between the method's structural computations and the retained examples. The structural case in Figure 8 uses a fixed selection rule: Task 1, the smallest observed class index, and decision fold 0. Both panels share a deterministic PCA projection of the same observed-class features. Reference-fold samples construct the evidence used to evaluate the disjoint decision fold. Stars mark recorded reference-core centers and diamonds mark fine-granule centers; routing states and outlined stored anchors expose the resulting selection. The centers and decisions come from the recorded high-dimensional computation, while PCA provides a shared display coordinate system.



(a) Reference evidence.

(b) Routing and admission.

Figure 8: Recorded granular evidence and memory admission.

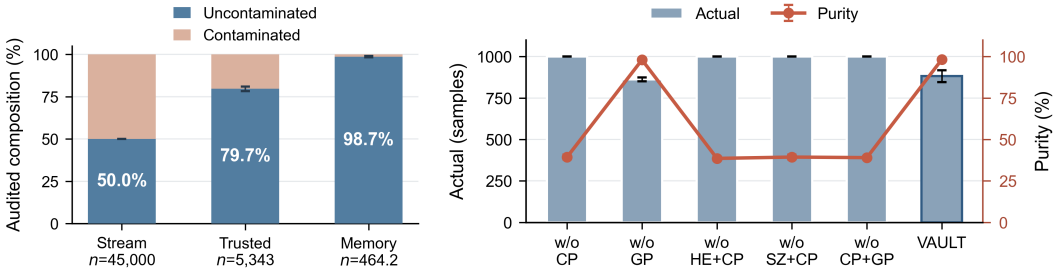
Process diagnostics link recorded evidence to memory admission. Candidate identifiers are matched to the fixed contamination manifest for offline auditing. All 45,000 candidates have valid recorded evidence and are included in the analysis. Memory composition is measured at task boundaries, when admission occurs. Task-specific admission rates use separate denominators of 4,500 uncontaminated and 4,500 contaminated candidates.

G Memory Composition and Mechanism Analysis

This analysis separates the size of the candidate stream from the size and composition of persistent memory. The admission audit follows the selection stages, while the component comparison examines the memory produced by alternative mechanisms. Together, the panels explain which observations remain available to support subsequent learning.

Figure 9(a) traces purification from a 50%-pure stream to memory with 98.66% mean purity and 464.2 stored samples at the 500-sample cap. Stream and candidate counts cover the task sequence; memory counts describe the final buffer. Error bars denote sample standard deviations, with aggregation specified in Appendix B.

Panel (b) links the ablations in Table 2 to memory composition at the 1000-sample cap, reporting means and sample standard deviations. The variants w/o HE, w/o SZ, w/o HE+SZ, w/o HE+GP, and w/o SZ+GP yield empty memories: Actual is zero and Purity is undefined. The panel displays nonempty memories; Table 2 retains all variants. Effective clean capacity is counted within each run before averaging.



(a) Admission audit ($B = 500$).

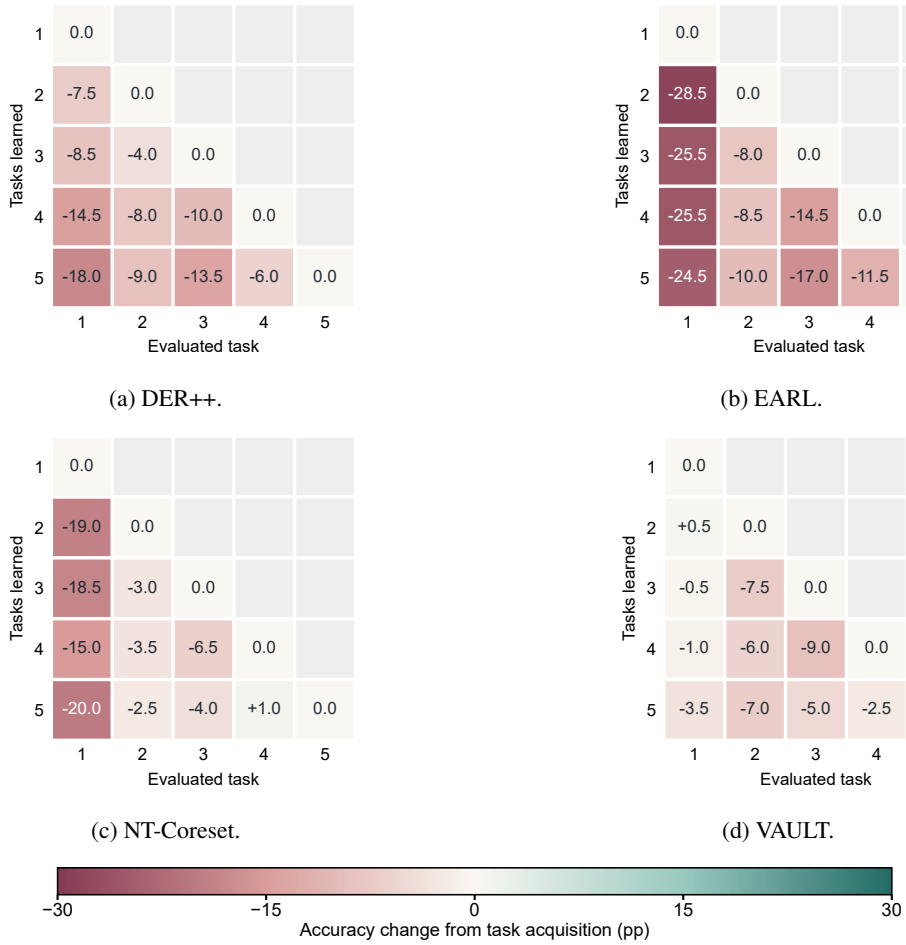
(b) Memory mechanisms ($B = 1000$).

Figure 9: Memory admission, purity, and realized storage.

H TASK-WISE RETENTION ON WEBVISION

Final average accuracy combines task acquisition with subsequent retention. Figure 10 separates these effects using the recorded WebVision accuracy matrices. Each cell reports $a_{t,k} - a_{k,k}$, the accuracy change on task k after learning task t, relative to acquisition. All four methods use the same color scale; grey cells correspond to tasks not yet learned. These changes describe retention relative to each method's own acquisition accuracy, complementing the absolute FAA comparison in Table 6.

Each column follows a task as subsequent tasks are learned. VAULT's first-task accuracy follows the sequence 75.0, 75.5, 74.5, 74.0, and 71.5%, exposing the trajectory behind its final retention. The final heatmap entry is -3.5 points relative to acquisition, whereas this task contributes 4.0 points to peak-based forgetting. The difference reflects the intermediate improvement to 75.5%. Reading the columns alongside Table 6 connects stagewise learning dynamics to aggregate retention while distinguishing acquisition-referenced changes from historical-peak forgetting.



(a) DER++.

(b) EARL.

(c) NT-Coreset.

(d) VAULT.

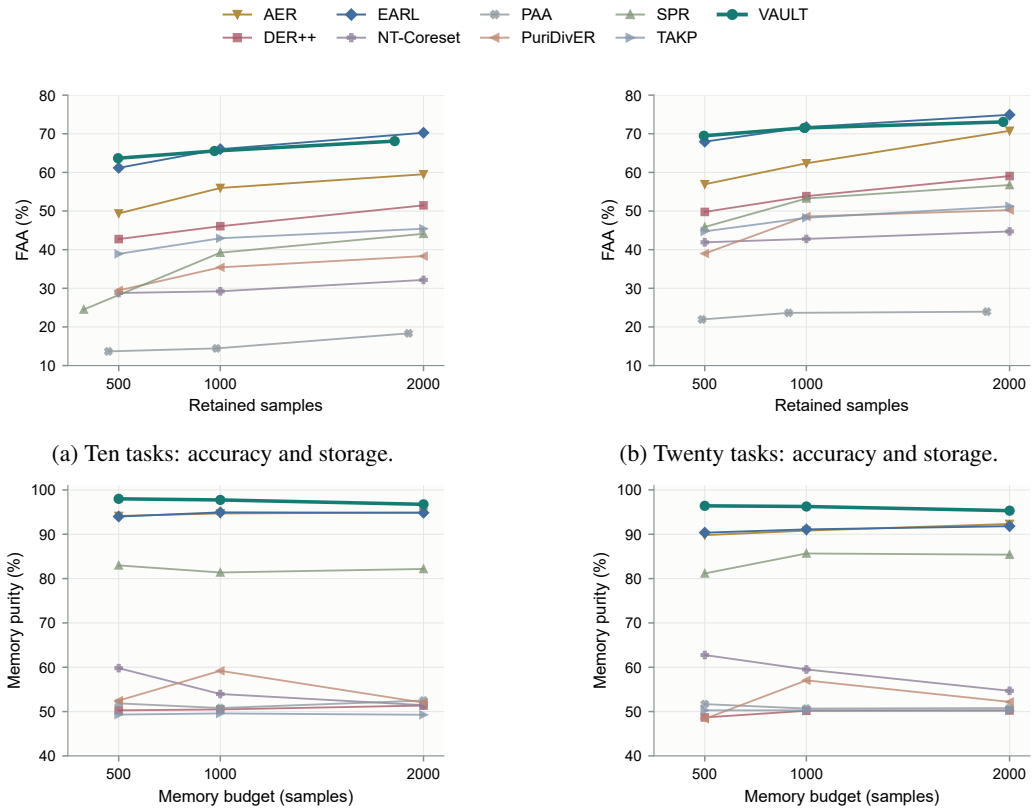
Figure 10: Task-wise accuracy changes on WebVision.

VAULT preserves first-task accuracy from 75.0% at acquisition to 71.5% after all five tasks, a change of -3.5 percentage points. The corresponding changes are -18.0 for DER++, -24.5 for EARL, and -20.0 for NT-Coreset. For VAULT, the final changes on the other previously learned tasks are -7.0, -5.0, and -2.5 points. The complete matrix reveals how task-level changes produce the aggregate retention results, extending the historical-preservation analysis to native web-label noise.

I CAPACITY RESPONSE OVER LONGER SEQUENCES

Figure 11 visualizes the capacity sweeps in Tables 4 and 5. The accuracy panels use realized storage on the horizontal axis, connecting the three evaluated memory budgets for each method. The purity panels use nominal budgets and include all compared methods. Comparisons are made within each task partition; lines connect the three measured budget settings.

The largest budget clarifies how realized storage and memory quality interact. With ten tasks, VAULT retains 1858.2 samples at 96.75% purity; with twenty tasks, it retains 1969 samples at 95.32% purity. At this budget, its purity exceeds EARL's by 1.90 and 3.50 percentage points, respectively. The upper panels place predictive performance against actual storage, while the lower panels show the reliability of the retained observations. Together, they distinguish the availability of additional slots from the quality of the supervision selected to occupy them.



(a) Ten tasks: accuracy and storage.

(b) Twenty tasks: accuracy and storage.

(c) Ten tasks: memory purity.

(d) Twenty tasks: memory purity.

Figure 11: Capacity response on longer CIFAR-100 task sequences.

VAULT maintains the highest memory purity across all displayed budgets in both partitions. From a budget of 500 to 2000 samples, its FAA increases from 63.67% to 68.10% over ten tasks and from 69.48% to 73.07% over twenty tasks. The paired views distinguish increasing storage from maintaining reliable memory composition. They connect the compact-budget accuracy advantage to the persistence of high-purity supervision as the budget expands.