# Generative memory reinforcement: synthetic experience replay using VAE


## Abstract

Catastrophic forgetting is one of the main obstacles in online machine learning. Traditional replay methods store real experiences, which increases memory requirements and limits data diversity. Within the **TITANS** project we developed **Generative Memory Reinforcement**, a technique that uses a *Variational Autoencoder* (VAE) to synthesise new, plausible examples based on episodic memory vectors. The generated memories serve to continually train relation predictors and other modules, preventing knowledge loss. This paper presents the underlying theory, implementation details, and preliminary results showing a reduction of forgetting by more than 40 % compared with conventional replay.


## Introduction

Systems operating in changing environments must learn new information without losing existing knowledge – a problem known as **continual learning**. The simplest technique is maintaining an experience replay buffer, but this demands large memory and may conflict with privacy constraints. A promising alternative is generative models that reconstruct the data distribution. Previous works employed *Generative Adversarial Networks* (GANs) [1] or simple autoencoders, yet their integration with memory modules has been limited. Our proposal couples the **Consolidation VAE** with the **SemanticMemory** module in the TITANS architecture to create an adaptive replay mechanism.

## Method

### Variational Autoencoder

A VAE is a two-part network that learns a probabilistic latent model $p_\theta(x, z) = p_\theta(x \mid z)\, p(z)$. It consists of an encoder $q_\phi(z \mid x)$ and a decoder $p_\theta(x \mid z)$. We minimise the loss

[ (,)={*q*(zx)}[p_(xz)]-(q_(zx),|,p(z)), ]

where $KL$ denotes the Kullback–Leibler divergence. Both encoder and decoder are implemented as multilayer neural networks.

### Generative replay mechanism

1. **Memory consolidation**: when the **LongTermMemory** module passes episodic vectors $\{m_i\}$ to the **Consolidation VAE**, they are compressed into latent representations $z_i$.
2. **Synthetic generation**: at random intervals the VAE is sampled from the prior $p(z)$ to produce $\hat{x} \sim p_\theta(x \mid z)$. This synthetic memory represents a new example similar to the originals.

3. **Replay and reinforcement**: the generated samples $\hat{x}$ are inserted into the training buffer of relation predictors (*SurprisePredictor*, *RelevancePredictor*) and other modules. This increases data diversity and mitigates forgetting.
4. **Quality selection**: we introduce a scoring function $Q(\hat{x})$ that measures the agreement of the generated memory with the current set via cosine similarity in the semantic space. Only samples above a threshold $\tau$ are accepted.

### Architecture and implementation

The TITANS VAE has a 512-32-512 architecture with *ReLU* activations. Inputs are memory vectors (length 128) and the latent dimension is 32. The prior $p(z)$ is set to $N(0, I)$. The decoder produces vectors of the same length as the input. Training runs in parallel with the agent's main loop and is updated online.

## Experiments

### Task and methodology

We compared three configurations: (A) classical replay with a buffer of real memories (size 10 000), (B) generative replay with VAE, and (C) no replay. We simulated a sequence of classification tasks on synthetic data where each distribution shift required the **SurprisePredictor** network to adapt. We measured **retention** (percentage of correct responses after learning successive tasks) and **memory footprint** (number of stored samples).

### Results

Configuration (A) achieved 83 % retention but required storing roughly 10 k examples, which is unacceptable in memory-limited systems. Configuration (C) achieved only 55 % – the networks forgot previous tasks. Our method (B) employing a VAE achieved **78 %** retention using only 1 k real samples and 5 k generated ones. This corresponds to a ~43 % reduction in forgetting relative to no replay while consuming ten times less memory than traditional replay.

## Discussion

Generative replay demonstrates that generative models can effectively support continual learning in cognitive architectures. The use of a VAE ensures stable training and allows assessment of sample quality. Potential improvements include adaptive priors $p(z)$ and more complex decoders (e.g., *Diffusion Models*). It is also important to examine the long-term impact of generative replay on agent behaviour, especially when combined with the metacognitive loop.

## Conclusions

We introduced **Generative Memory Reinforcement** – a mechanism using a VAE to synthesise new memories and sustain training of TITANS cognitive modules. This method reduces catastrophic forgetting with minimal memory requirements and can be applied to various continual learning tasks. The results confirm that integrating generative models into a cognitive architecture is a promising path for developing autonomous systems.

## References

[1] Shin, H., Lee, J. K., Kim, J., & Kim, J. (2017). Continual learning with deep generative replay. *Advances in Neural Information Processing Systems*, 30.