TWINGITAL INSTITUTE · TECHNICAL NOTE · 2026

Beyond the next token: three architectures for three shortcomings of the LLM paradigm

World models, memory and JEPA as partial responses to dynamics, persistence and the prediction space

Jérôme Vetillard, VP R&D and Engineering, CPO, Twingital Institute

1. The founding shortcoming

The dominant paradigm of artificial intelligence in 2026 remains the autoregressive model trained by maximum likelihood on a corpus of symbols. Its fundamental task is simple to state: predict the next element of a sequence given the preceding elements. The empirical success of this task is considerable. It has produced systems capable of summarizing, translating, programming, reasoning locally, conversing, reformulating knowledge and manipulating symbolic structures with an efficiency that would have seemed implausible ten years earlier.

But this success must not be misnamed. What in a large language model passes for understanding, memory or simulation of the world is not directly optimized as such. It is an emergent property of sequential learning on symbolic traces. The model learns to produce plausible continuations in the space of tokens. By that operation alone, it does not learn an explicit model of world dynamics, a persistent biographical memory[1], or a prediction space suited to non-linguistic phenomena.

The architectural diagnosis starts here. The standard autoregressive model presents three distinct blind spots.

1. The first is dynamics. The model knows how to estimate what is likely to come after a sequence; by construction, it does not know what would happen if an action were performed in an environment. The difference between passive prediction and prediction conditioned on action is fundamental for planning, robotics, embodied cognition, clinical simulation or any system subject to interventions. An LLM can describe a consequence; it has not necessarily learned the causal dynamics that produce it.
2. The second is persistence. The context window is a computational device, not memory in the strong sense. It can be very long, sometimes up to a million tokens, but it remains essentially flat: it does not natively distinguish reference information, dated recollection, current state, transient instruction, stable preference or singular event. It does not naturally persist beyond a session and does not, by itself, constitute a biography. A contemporary LLM has a larger scratchpad than a 2022 LLM; that is not enough to say it possesses memory.
3. The third is the prediction space. Predicting the next token imposes a target space: that of tokenization. Yet much of what we seek to predict (physical states, biological trajectories, spatial configurations, clinical evolution, the response of a system under intervention) does not properly reduce to a symbolic sequence. The relevant question is no longer only: what is the next token? It becomes: in which space must we predict in order to learn the useful invariants?

Three families of architectures respond, each partially, to these gaps:

1. World models attack dynamics.
2. Memory models attack persistence.
3. JEPA-type architectures attack the prediction space.

This term-by-term correspondence is, as we shall see, an initial posture rather than an equilibrium state: the boundaries deform as soon as one looks at the most recent architectures.

These lineages are often presented as competing alternatives to LLMs or to each other. Such presentation is convenient for marketing, hence naturally intellectually suspect. Above all, it is architecturally false. The three families do not properly substitute for one another. They respond to different deficits. Their probable trajectory is therefore not mutual elimination but the composition of these architectures. This composition, however, does not automatically solve the problem: it relocates it to module coordination, training stability, action governance and output validation. A hybrid architecture is not a magical synthesis; it is a stack of better localized problems.

This note does not address products or markets. Nor does it settle the still-speculative question of whether any of these lineages constitutes a path toward general intelligence. It proposes a minimal technical cartography in order to

avoid three confusions: calling memory what is merely a long context, calling a world model any system that appears to understand the world, and calling JEPA a general alternative to LLMs when it is first of all another prediction target.

2. Operational definitions and cuts

A useful definition is not a seductive one. It is a definition that allows one to decide, to take a position. The three terms that follow are used too broadly in public debate. They must therefore be restricted, without claiming to abolish all their other uses.

A *world model*, in the strict sense, is a model that learns the dynamics of an environment from observations, generally coupled with actions, and that allows the prediction of a future state conditionally on a sequence of actions. This strict definition emphasizes dynamic projection: what becomes of the environment if the agent does this rather than that? It applies clearly to the Ha and Schmidhuber lineage, then Dreamer, where an agent learns to plan in a latent space rather than by repeated trials in the real environment.

There is, however, a broader use of the term. Generative video models such as Sora, or interactive systems such as Genie, can be described as *implicit world models* when they learn temporal, physical or spatial regularities without necessarily having explicitly annotated actions. In this case, the dynamics are learned, but the action may be latent, induced or reconstructed. The distinction matters: a strict world model is conditioned on action; an implicit world model learns world dynamics without that conditionality always being explicit. Conflating the two makes for fine announcements and poor architectural choices, which is by now a well-established industrial tradition.

A *memory model* is an architecture that distinguishes current computation from persistent, compressed or re-addressable storage. The criterion is not the length of the context. The criterion is the differentiation between immediate processing and conservation. An indexed external base, a compressed recurrent state and a learned memory module are three very different mechanisms; they share only the idea that part of the information must survive immediate inference or be reused over a long sequence.

JEPA, for Joint Embedding Predictive Architecture, designates a family of architectures that predict in the space of representations rather than in the raw space of observations. A context view and a target view are encoded; a predictor learns to approach the representation of the target from the representation of the context. The loss is computed in the latent space, not on pixel-by-pixel reconstruction. The central proposition is therefore simple: to learn what matters, one must avoid forcing the model to reconstruct what is visible but not relevant.

These definitions produce three cuts.

1. First cut: predicting observations or predicting representations. It separates generative world models, which may reconstruct pixels, from JEPA architectures, which learn predictive representations without explicit reconstruction of the raw observation.
2. Second cut: context or memory. It separates long-context LLMs from memory models in the architectural sense. A long context grants access to more information during an inference; a memory imposes a structure of conservation, writing, recall, forgetting or compression.
3. Third cut: passive prediction or prediction conditioned on action. It separates linguistic continuation from a dynamic model. An LLM can produce a sentence about the consequences of an action; a world model aims to simulate the effect of that action on a latent or observable state.

These cuts are not absolute boundaries. They are instruments of analysis. Their function is not to freeze the landscape but to prevent the confusion of levels.

3. Generative world models

The family of generative world models is the oldest of the three lineages discussed here. It is also the most directly linked to planning and control.

Ha and Schmidhuber published in 2018 a paper explicitly titled *World Models*. The architecture there is simple: a perceptual module, often a variational autoencoder[3], compresses the observation into a latent vector; a recurrent dynamic model learns to predict future latent states; a controller chooses actions from that latent state. The important idea is not only compression. It is that the controller can be trained inside the model's *dream*, that is, inside the internal simulation learned by the system.

The Dreamer lineage generalizes this intuition. DreamerV1 introduces a Recurrent State Space Model combining a

deterministic state and a stochastic state, in order to model both temporal continuity and uncertainty. DreamerV2 shows that planning in a latent space[4] can rival reinforcement methods more directly anchored in the environment. DreamerV3 reinforces the stability and scope of application of the paradigm. DayDreamer transposes this logic to physical robots, where learning by imagination reduces the cost and risk of trial-and-error in the real world.

At another scale, recent generative video models can be read as implicit world models. When a system learns to produce coherent video sequences, it must internalize certain regularities: object persistence, occlusions, spatial continuity, apparent gravity, plausible trajectories. This does not mean it possesses an explicit physics of the world. It means that part of the visible dynamics is captured in its representations. The ambiguity begins precisely here: between learned regularity and intervention model, between visual coherence and controllable simulation.

The common mechanism of the strict family is nonetheless clear. The agent does not plan directly in the environment; it plans in an internal approximation of the environment. The control loop evaluates future trajectories in a latent space, then selects an action as a function of those imagined trajectories. This is the fundamental shift: learn less in the real world, learn more in the model of the world.

The limits are structural.

1. The first is computational. When supervision passes through the reconstruction of rich observations, a considerable share of capacity is spent modeling details of little decisional relevance: textures, noise, visual micro-variations, contingent elements. Pixel reconstruction can become a poor teacher: very demanding, hardly selective, spending the model’s capacity on details with no decisional bearing.
2. The second is temporal. Prediction errors compound. Over a short horizon, a latent rollout may be useful; over a long horizon, the accumulation of errors progressively distorts the trajectory. This problem is not an optimization detail. It affects the very reliability of planning.
3. The third is distributional. A world model learns dynamics local to the training distribution. If the real environment diverges structurally, the internal simulation becomes misleading. This is the classic sim-to-real transfer problem, but here it takes a more general form: any dynamic model is reliable within a validity domain, not in the world as such.

These limits do not invalidate world models. They define their domain of use. They also explain why the question of the prediction space becomes central.

4. JEPA and the representational break

JEPA responds to a precise weakness of generative models: the obligation to predict in the raw space of observations. Its proposition is not to reconstruct better. It is not to reconstruct what does not deserve to be reconstructed.

In Yann LeCun’s positioning text, the argument is direct. A large part of the observable world is unpredictable in its fine detail and useless for action. The exact position of every leaf in a tree, the texture of a wall, the background noise of an image or the micro-variation of lighting are not necessarily relevant variables for learning the structure of the world. A model that devotes a large part of its capacity to reconstructing these details wastes a resource that could be used to learn invariants.

JEPA therefore shifts the target. It does not predict the observation; it predicts the representation of the observation. Part of the input serves as context. Another part serves as target. Two encoders produce representations. A predictor learns to transform the context representation so as to approach that of the target. The loss is computed in the representation space. No pixel reconstruction is required.

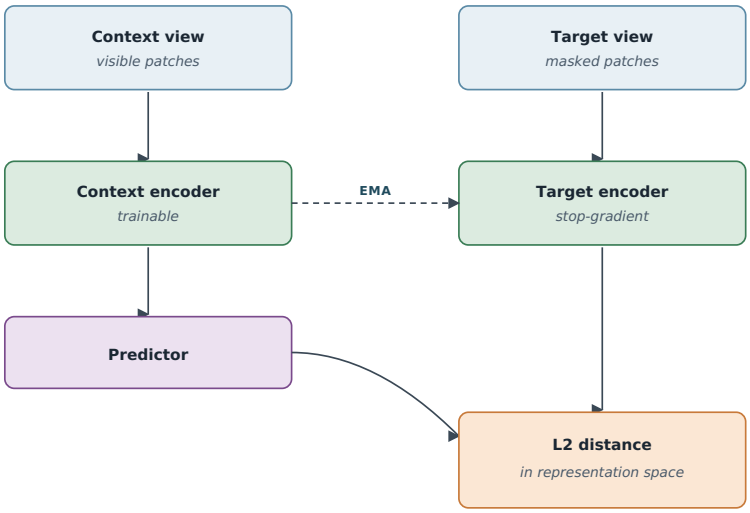

Figure 1. JEPA architecture. The context (visible patches of an image or a video) and the target (masked patches) are processed by two distinct encoders. The target encoder is not trained by backpropagation: its weights follow those of the context encoder via exponential moving average. A predictor transforms the context representation so that it approaches that of the target, and the loss is computed (this is the central theoretical point) in representation space, never in pixel space.

This shift is deeper than it appears. The model no longer learns to produce a plausible image. It learns to produce a representation compatible with what should be present. The objective is therefore not sensory fidelity but abstract predictability. **JEPA can therefore be understood (in an "approximate" sense) as an attempt to learn the constraints of the world rather than its appearances.**

The training mechanism must, however, avoid trivial collapse. If the two encoders learned to produce a constant, the loss could be minimized with no useful information. To prevent this, the target encoder is not directly updated by backpropagation; its weights follow those of the context encoder via exponential moving average[5]. This self-distillation mechanism stabilizes learning and prevents the system from solving the task by suppressing information.

I-JEPA applies this principle to images. V-JEPA extends it to video. V-JEPA-2 adds an agentic dimension: the model learns to predict future representations conditionally on actions. This last point changes JEPA's status. As long as it predicts masked representations, it remains principally a predictive representation architecture. As soon as it predicts future states under action conditioning, it becomes a world model in the strict sense, but a non-generative one.

This evolution is important because it shows that JEPA is not a separate family once and for all. It is a response to the question of the prediction space. When coupled with action, it joins the question of dynamics. The boundary between JEPA and world model does not disappear; it is reformulated. **The relevant distinction becomes: a generative world model in observation space, or a predictive world model in representation space.**

The limits of JEPA must be stated without indulgence.

1. The first concerns scale. JEPA architectures have not, to date, demonstrated a scaling law comparable to that observed for large language models. The difficulty is not only empirical. It is also formal. Next-token prediction provides text with a universal, massive, homogeneous and naturally available task. At this stage, JEPA does not have an objective that is as universal, homogeneous and industrially exploitable: the choice of views, masking, latent space, temporal horizons and losses remains domain-dependent.
2. The second concerns the absence of a mechanism equivalent to in-context learning. LLMs do not merely store knowledge in their parameters; they learn to use the context as a local program. JEPA learns predictive representations, but it has not yet demonstrated a general capacity comparable to reconfiguring its behavior from a sequence of arbitrary examples.
3. The third concerns modal scope. JEPA is particularly natural for ("visual") perception: image, video, possibly robotics. It does not replace a general language model. Presenting it as a global alternative to LLMs is a rhetorical convenience. JEPA addresses another blind spot: representational prediction. It does not solve, on its own, memory, symbolic reasoning, action governance or linguistic generation.

JEPA is therefore a strong but partial proposition. Its value does not lie in the promise of replacing LLMs. It lies in the possibility of taking predictive learning out of the narrow space of the token and pixel reconstruction.

5. Memory models

Memory is probably the most mistreated term in the contemporary AI debate. A long context is called memory. A document base is called memory. A hidden state is called memory. A persisted user preference is called memory. At this stage, the word covers so many different objects that it has ceased to designate anything precise.

A minimal taxonomy distinguishes three sub-families.

1. The first is indexed external memory. The main model generally remains an autoregressive LLM. An external system retrieves relevant documents, passages, fragments or events, then injects them into the context. This is the RAG principle. MemGPT adds a more explicit management layer: the system decides what to save, what to recall, what to summarize. This sub-family has a major advantage: it is governable. The contents can be inspected, deleted, versioned, traced, subjected to access rules. Its limit is symmetrical: it assumes that useful memory can be converted into indexable objects, often textual. A continuous physiological state, a probabilistic clinical trajectory or a system dynamics fits poorly within it.
2. The second is compressed recurrent state memory. Mamba, RWKV and more broadly state space models[2] maintain a fixed-size hidden state updated along the sequence. The memory is not external. It is in the state. The advantage is computational: the cost can grow linearly with length, where classical transformer attention becomes costly. The limit is informational: to compress is to choose; to choose is to forget. What is not retained in the current state cannot be recovered later by simply returning to the context. The memory is continuous but lossy.
3. The third is learned long-term memory. Titans illustrates this direction: a neural memory module learns what to write, when to write, how to forget and how to reuse. The architecture distinguishes working memory, long-term memory and sometimes persistent memory. Memory ceases to be only external storage or an implicit state; it becomes a trained component.

This tripartition can be related to categories drawn from cognitive psychology (working memory, reference memory, episodic memory), but the analogy must remain strictly limited. Current architectures do not implement a human memory. They implement mechanisms of conservation, retrieval or compression of information.

An episodic memory, in the minimal architectural sense, presupposes three conditions: a singular event indexed temporally, a recall oriented by the present situation, and a contextual update that is neither simple overwriting nor erasure. None of the three current industrial sub-families fully satisfies these three conditions. Some modules come close; none deserves the conflation of persistent storage with memory in the proper sense.

The architectural point is therefore precise. Memory models do not solve the world problem. They solve part of the persistence problem. They allow the conservation or compression of traces, not necessarily their understanding, hierarchization or causal use.

6. Comparative synthesis

The three families can now be read side by side. The matrix below formalizes this comparison along six architectural axes. It is not a decision guide: it is a mapping tool, whose function is to avoid conceptual confusion, not to arbitrate an industrial choice.

Criterion	LLM (reference)	Generative world models	JEPA	Memory models
Predicted object	Next token	Pixel / observation	Representation	Next token
Target space	Symbolic	Latent + reconstruction	Latent only	Symbolic
Action	No	Central	V-JEPA-2 only	No
Memory	Flat context	Recurrent state	None dedicated	Index, state, learned
Evaluation	Perplexity, benchmarks	Reconstruction, RL return	Linear probing	Recall, retrieval
Cost	Quadratic in length	Pixel reconstruction	Self-distillation	Sub-family dependent

Figure 2. Comparative mapping. Four families, six criteria. The leftmost column lists the criteria; the LLM column is set as a reference, not as a member of the taxonomy.

The usual reading of such a matrix is positive: for each family, one asks what it can do. The useful reading is the reverse. What illuminates the debate is not the list of claimed capacities but the list of structurally absent capacities, absent not through engineering default but by architectural construction.

The autoregressive LLM cannot learn the dynamics of an environment from its training task alone. No volume of text, however large, exposes the model to the conditionality of an intervention. This absence is not a temporary gap that additional parameters would fill; it is consubstantial with the learning objective. Text describes actions and their consequences, but it does not make the model act in an environment where its own choices would modify subsequent observations.

The generative world model cannot, by itself, exploit its internal simulation for arbitrarily long sequences. Error composition over the horizon is a mathematical property of predictive chaining, not a calibration defect. Every useful horizon is a bounded horizon. This bound is not an obstacle to be lifted but a characteristic to be respected in the design of the system that calls the world model, typically through frequent replanning rather than through prolonged trust in the rollout.

JEPA cannot, in its current state, predict in the symbolic space with the flexibility of next-token prediction. Its central proposition, predicting in representation space, excludes by construction the production of explicit tokens. The passage from representation to linguistic action remains to be invented, and it is not certain that it can be invented without reintroducing a generative decoder. As long as this articulation is not resolved, JEPA addresses principally perception and non-textual dynamics.

A memory model does not guarantee, by its architecture alone, that what it conserves will be relevant, legitimate or reusable in the appropriate context. It may learn storage or recall policies, but relevance remains dependent on the objective, the domain and the governance regime.

This inverted reading reveals what the matrix cannot say: none of the four families is defective in the sense that it would fail at its task. Each does what it was designed to do. The architectural question is therefore not to measure their individual performance; it is to understand what they refuse by nature, in order to compose what none brings alone.

Three further axes escape the matrix and must be reintroduced explicitly before any industrial use. The first is ecosystem maturity, profoundly asymmetric:

- LLMs industrialized,
- world models in advanced R&D, mainly academic,
- JEPA at the stage of representational proof of concept,
- memory models in heterogeneous industrialization by sub-family.

The second is the multi-module integration cost, which grows faster than the number of modules, each interface being itself an object of validation.

The third is the set of deployment, governance, auditability and compliance constraints, which belong to the context

of use and not to the bare architecture. An elegant matrix can lead to an unmanageable system.

The question “which one will win?” is therefore badly posed. It assumes a global competition where there are different functions. It turns an architectural decision into a tribal bet, which is admittedly a popular method of governance, though rarely a productive one. The correct reading proceeds by deficit addressed: to language and symbolic manipulation responds the LLM; to dynamic projection under action responds the world model; to sober representational prediction responds JEPA; to persistence, recall and temporal compression responds the memory model. This list is not enough to choose a solution; it is enough to stop being mistaken about the problem.

7. Ongoing convergences

The three lineages do not remain separate. They converge. This convergence is real, but it should not be confused with an already accomplished synthesis.

1. First movement: JEPA becomes agentic. With V-JEPA-2, representational prediction is no longer only tied to perceptual masking. It becomes conditioned on actions. JEPA then enters the territory of strict world models, but by a non-generative path: it does not necessarily simulate future pixels; it predicts useful future representations.
2. Second movement: transformers become memorial. Titans, MemGPT and various hybrid architectures signal the same pressure: context is not enough. One must distinguish what is being manipulated now, what must be conserved, what must be recalled, what must be forgotten. Memory becomes an architectural component, not a mere increase of sequence length.
3. Third movement: world models integrate memories and more abstract representations. Dreamer already has a recurrent latent state that can be read as working memory. The open question is that of the coupling between a latent dynamic model, a learned long-term memory and a predictive representation encoder. This is probably one of the most important directions for action-oriented architectures.

But this convergence relocates the difficulty. Composing an LLM, a world model, a memory and a predictive encoder does not automatically produce a superior system. **It produces a system more difficult to train, to interpret, to validate and to govern.**

Four problems appear immediately.

1. The first is error propagation. An approximate representation feeds a partial memory, which feeds an uncertain planning, which produces an action whose consequences return into the system. In a composite architecture, error does not remain local. It circulates.
2. The second is objective coordination. One module may optimize representational fidelity, another planning performance, a third recall relevance, a fourth linguistic generation. Nothing guarantees that these objectives are aligned, and in practice they are not spontaneously aligned.
3. The third is validation. A modular architecture requires validating components, their interfaces and their interactions. The test surface grows faster than the number of modules. This is one of the many places where architectural enthusiasm dies, stabbed by quality assurance.
4. The fourth is governance. Who decides that a representation is reliable enough to feed an action? Which memory may be mobilized? Which event must be logged? Which action must be refused? Above what threshold must the human take back control? These questions are not peripheral. They become central as soon as a composite architecture acts in a real environment.

Convergence is therefore indeed the likely direction. But it must not be told as the resolution of the problem. It is the opening of a higher-order problem.

On terrains such as clinical digital twins, this conclusion is not theoretical. A system that reconstructs or projects patient trajectories must combine a dynamic model, a memory of clinical history, an abstract representation of unobserved states and a governance layer. None of the three families is sufficient alone. But their composition is acceptable only if the interfaces, hypotheses and validity limits are explicit; failing this, the system inherits the blind spots of each module without inheriting their guarantees.

8. Authentic limits

This cartography itself has limits. Making them explicit is necessary, not out of decorative prudence, but because a map that does not state what it excludes quickly becomes an ideology.

1. The first limit is the absence of inter-family benchmarks. LLMs are evaluated by perplexity, linguistic benchmarks, symbolic reasoning or programming tasks. World models are evaluated by reinforcement return, prediction error or control performance. JEPA is evaluated by linear probing, k-NN, representation transfer or downstream performance. Memory systems are evaluated by recall, contextual accuracy or ability to exploit long sequences. These protocols do not measure the same thing. Direct comparisons are therefore rarely scientific. They are often editorial, including, on a smaller scale, the one in this note.
2. The second limit is the absence of a demonstrated scaling law for JEPA at the level of large LLMs. This point does not disqualify the approach, but it forbids treating it as an already proven alternative. The difference between a promising direction and a dominant paradigm is called quantitative proof. It is tedious, but reality sometimes has that bad taste.
3. The third limit is the fragility of sim-to-real transfer for world models. Even when learning in simulation performs well, the passage to a real physical, clinical or organizational environment introduces distribution gaps, unobserved variables and action constraints that cannot be absorbed by a mere data increase.
4. The fourth limit is the ambiguity of the term *world model*. It designates sometimes a dynamic model conditioned on action, sometimes a rich perceptual representation, sometimes an impressive generative system, sometimes a cognitive metaphor. This polysemy is useful for selling a vision; it is dangerous for designing an architecture.
5. The fifth limit is more rarely formulated: none of these families natively integrates complete governance. An LLM does not naturally distinguish recommendation from action. A world model does not naturally bound its validity domain. JEPA does not by itself provide causal traceability of its representations. A memory module does not guarantee that what it recalls is legitimate, current or authorized. Governance must therefore be architected around the model, and sometimes within the model, but it does not emerge automatically from performance.

This point is decisive in regulated environments. A composite system that simulates, recalls, predicts and acts must expose its hypotheses, its events, its refusals, its thresholds, its uncertainties and its responsibilities. Failing this, one obtains an architecture that is technically seductive and regulatorily unusable.

9. Conclusion

The debate on post-LLM architectures is often framed as a succession of replacements. LLMs would have replaced symbolic models. World models would replace LLMs. JEPA would replace generative models. Memory models would repair the weakness of context. This reading is too simple.

The right diagnosis is functional. The autoregressive paradigm has revealed a remarkable power in manipulating symbolic sequences, but it leaves three problems open: world dynamics, information persistence and the choice of prediction space. World models, memory models and JEPA respond to these three problems, each partially, each with its limits.

An LLM speaks about the world. A world model projects possible states of the world. JEPA learns predictive representations of the world. A memory model conserves or recalls traces of the world. None constitutes, on its own, a complete architecture.

The strategic question is therefore not: which paradigm will win? It is: what minimal combination of capacities is necessary for the use case under consideration, under what hypotheses, with what validity domain, what cost, what governance and what proof?

This reformulation changes the nature of the decision. It forbids conflating laboratory announcement, product promise and operable architecture. It requires reasoning by function, by interface and by validation. That is less spectacular than a prophecy. It is, above all, the only level at which an architectural decision ceases to be a belief and becomes defensible.

Footnotes

---

1. On the distinction between biographical memory and informational storage, see J. Vetillard, *Encodage, transduction et modèles du monde*, Twingital Institute, 2025. ↩
2. State Space Models. Family of architectures that maintain a fixed-size hidden state updated at each time step. Unlike the classical transformer, their cost can grow linearly with sequence length. ↩

3. Variational Autoencoder. Network that learns to compress an observation, for example an image, into a reduced-dimension vector belonging to the latent space, while preserving the information useful for reconstruction. ↩
4. The latent space is an auxiliary representation space, learned to make certain operations (reconstruction, prediction, control) simpler. It is neither a subspace of the real, nor a miniature copy of the world. It is a useful internal coordination, with a largely arbitrary geometry and only partial alignment with the structure of the data. ↩
5. EMA, or exponential moving average: a technique by which the weights of the target encoder follow those of the context encoder with a lag, in order to stabilize training and avoid trivial collapse. ↩↩