TargetSpace: Benchmarking Target-Specific Forecasting Under Partial Observation

Proper-scored target-state forecasting with own-routine baselines, permutation specificity, and evidence ablation

Anonymous Author(s)

Pre-pilot draft (v2.4) — preprint — not peer reviewed — June 2026

Abstract

AI systems increasingly claim to understand specific users, customers, agents, organizations, environments, robots, and biological systems. But a forecast can look impressive while only reflecting *population priors, base rates, or generic routines*: a model can predict that an email will be answered, a project will slip, or a task will succeed without knowing anything specific about *this* person, *this* project, or *this* environment. **TargetSpace** is a benchmark framework that asks whether a model's forecasts are genuinely **target-specific**. It evaluates prospective, sealed, proper-scored forecasts of a target system's future **target-state transitions** under **partial observation**, and certifies target-specificity with a stack of controls: a population-prior baseline (**R1**) the model must beat to exceed base rates; a strong, instance-specific **own-routine baseline (R2)** it must beat to exceed routine mimicry; a **calibration** gate; and a **permutation specificity test** in which a model's forecasts for one target are scored against another target's outcomes — skill that survives permutation was never target-specific. An **evidence-tier ablation** measures which evidence streams add target-specific information, and an **evidence-tier × architecture-class** grid compares large language models, vision-language models, JEPA-style latent predictive models, symbolic and probabilistic systems, multimodal agents, human self-report, and an oracle bound on identical instances. Proper scoring, calibration, sealing, evidence ablation, and architecture-neutral evaluation are adopted from prior work and cited; TargetSpace's contribution is the **conjunction**, anchored on the own-routine baseline and the permutation specificity gate, around one question: *is the forecast about the target, or about the average?* TargetSpace is not a single dataset or one domain-specific benchmark but a **shared evaluation apparatus with multiple application tracks** (personal, health, energy and markets, robotics, enterprise), each instantiating the same spine with domain-specific targets, evidence bands, horizons, validators, and baselines. It is **domain-general in formulation** and **instantiated first in personal world modeling**; only that track has apparatus today and the others are planned or research-tier. This is a pre-pilot proposal; no empirical results are reported, no track other than the synthetic person track is implemented, raw target data is never public, and the apparatus is public.

1 Introduction

Modern AI systems increasingly claim to understand *specific* targets: this user, this customer, this agent, this organization, this robot, this environment, this project, this biological specimen. Recommender systems, personal assistants, agentic copilots, digital twins, and predictive models of markets, cells, and machines are all sold, implicitly or explicitly, on the promise that they have learned something about a *particular* system and can anticipate what it will do. The promise is valuable and increasingly central. It is also, today, almost entirely unaudited.

The problem is that **many forecasts that look like understanding are not target-specific at all.** A model can appear skilled by predicting what *usually* happens: most planned meetings occur, most emails are eventually answered, most projects slip a little, most regenerating tissues grow back. It can appear personalized by replaying a target's *routine*: this person checks email at 9am, this project ships on Fridays, this robot usually completes the grasp. Both are useful, and both can produce a confident, plausible forecast — while revealing nothing about whether the model has captured anything specific to *this* target beyond population base rates and obvious routine. Retrospective explanation does not settle the question either: a model can rationalize what already happened without ever being held to what happens next.

Did the model learn something specific about this target, or is it merely predicting the average system?

There is a second way a forecast can look like understanding without being it. Modern systems increasingly impress because they *generate* plausibly — fluent explanations, convincing simulated behaviour, smooth robot motions. But generation is not

agency. Reliable agency requires forecasting the *consequences* of actions and events — how a specific target's state will transition — under partial observation. A system that produces a plausible next action without predicting the state it leads to cannot plan, cannot be held to an outcome, and cannot be told apart from a confident imitator. TargetSpace evaluates this missing layer: not whether a system can *say* or *do* something plausible, but whether it can forecast what a specific target will *do or become* next.

TargetSpace asks not whether a model can say what usually happens, but whether it can forecast what this target will do or become next, under specified evidence constraints.

The right question, then, is comparative and prospective: can the model forecast this target system better than (i) **the average system** — the population prior — and (ii) **the target's own routine**? Only a model that beats both, stays calibrated, and loses its skill when its forecasts are matched to the *wrong* target has demonstrated something specific to this system rather than to its crowd or its habit. **TargetSpace** operationalizes exactly this. It is a benchmark framework for prospective, sealed, proper-scored forecasting of a target system's future **target-state transitions** under **partial observation**, with the controls that make target-specificity measurable: a population-prior baseline (**R1**), a strong instance-specific own-routine baseline (**R2**), a calibration gate, a permutation specificity gate, and an evidence-tier ablation. We name these the *target-specificity stack* (Section 3).

TargetSpace is not a benchmark for whether a model can predict what usually happens. It is a benchmark for whether a model can predict what this target will do next. Personal world modeling is the motivating first application, because understanding individual people is valuable and is currently underserved: today's personal AI is trained largely on *digital exhaust* — calendars, messages, clicks — which often encodes little more than routine. But the formulation applies to any target system for which repeated observations and resolved outcomes make an own-routine baseline and a permutation test well-defined. We therefore describe TargetSpace as **domain-general in formulation** and **instantiated first in personal world modeling**, and we make no claim of cross-domain empirical validation in v1.

Forecast the target, not the average. Turn understanding claims into accountable forecasts.

1.1 Understanding is the capability; forecasting is the measurement

The capability we care about is understanding a specific system: an accurate, continuously updated grasp of how its situation and dispositions evolve. It is latent and impossible to score directly. Forecasting is its measurement — the observable quantity a system possessing the capability should produce — the role classification played for ImageNet [1]. Across the sciences, predictive success is how a model is judged when the thing modeled cannot be inspected: weather theories by proper-scored forecasts [38,23], market and behavioral models by out-of-sample prediction [18,19]. We do not claim prediction *is* understanding; we claim it is its strongest available operational signal — and, crucially, only when the prediction is shown to be about the *target* and not the crowd.

Understanding is the capability. Forecasting is the measurement.

1.2 The observation bottleneck, and possibility-space resolution

Why is this hard, and why now? The binding constraint is an **observation bottleneck**: we rarely see a system's latent state, only a thin, biased residue of it. For a person that residue is digital exhaust, produced *after* the person already decided what was worth recording; for a cell population it is a handful of assays; for an organization, filings and releases. A model that learns only the surface regularities of this residue learns a routine, not a generator — which is exactly the failure target-specificity is meant to expose. What should a forecast be *about*? A partially observed system at any moment occupies a **possibility space**, a distribution over the states it might next move into, and an adaptive system does not wander it at random: it acts to reach certain **target states**. Understanding the system is operationalized as forecasting how its possibility space **resolves** toward those target states, and especially the **transitions** between them — the moments when *what the system is acting to reach* changes, where routine breaks and target-specific skill is decisive. We use *resolve* (and, sparingly, *collapse*) only as an epistemic metaphor for a distribution concentrating toward a realized outcome; **we make no physical or quantum claim**, and flag the metaphor wherever it appears.

1.3 Contributions

(1) Target-specific forecasting as a well-posed evaluation object. We distinguish three things often conflated — *generation* (predicting surface outputs), *imitation* (predicting plausible actions from demonstrations), and *target-state forecasting* (predicting the consequence, i.e. the latent target-state transition, of actions and events) — and define and score the third, under partial observation, distinguished from predicting population base rates, a target's routine, next-action, or external events (Sections 2, 4–5). **(2) The target-specificity stack.** A stack of controls centered on a strong instance-specific **own-routine baseline (R2)** and a **permutation specificity gate**, while adopting proper scoring, calibration, sealing, and evidence ablation from prior work (Section 3). **(3) An evidence-tier × architecture-class grid** for comparing LLMs, VLMs, JEPA-style models, multimodal agents, symbolic/probabilistic systems, hybrids, human self-report, and oracle bounds on identical sealed instances (Section 6). **(4) A first instantiation in personal world modeling**, where passive first-person evidence is tested against digital exhaust for target-specific predictive lift (Section 8). **(5) Cross-domain extension criteria** for other target systems, stated as formulation rather than as validated empirical breadth (Section 9). The audited novelty is the *conjunction* assembled around one question, not any individual ingredient (Sections 3, 11).

2 Why Target-Specificity Matters

This section makes the central problem concrete, because it is what TargetSpace exists to measure and what most current evaluations cannot distinguish. **Generic prediction can masquerade as understanding.** Two cheap strategies produce convincing forecasts without any target-specific knowledge. The first is learning **base rates**: a model that knows the population statistics — how often meetings occur, emails get answered, projects slip, grasps succeed — will look calibrated and often correct, while knowing nothing about any individual target. The second is learning a **routine**: a model that replays a particular target's recent regularities will look *personalized* while having captured only habit. A benchmark that scores raw accuracy or agreement cannot tell either strategy apart from genuine understanding.

Target-specific understanding is what remains after both are subtracted. Operationally it means three things together. First, the model must **beat a strong own-routine baseline** (R2): predicting a target's habit is exactly what R2 already does, so skill over R2 is skill the routine does not contain. Second, the skill must be **calibrated**: a confident lucky streak is not understanding. Third, and most distinctively, the skill must **depend on the correct target-instance pairing**: if we permute outcomes across targets — scoring the model's forecasts for target *i* against the realized outcomes of a different target *j* — genuine target-specific skill should *collapse*. If it survives permutation, the model was modeling task or population structure, not the target.

A model earns target-specific credit only by beating the average, beating the routine, staying calibrated, and failing under permutation.

Four examples make the distinction tangible. **Person.** Predicting that someone will eventually answer an email is base-rate prediction; the target-specific question is whether the model knew *this* person would ignore *this* message because their attention had shifted to another project this week. **Organization.** Predicting that a project will slip is routine; the target-specific question is whether the model knew *this* project would slip *beyond this organization's normal delay pattern*. **Robot.** Predicting task success from population success rates is generic; the target-specific question is whether the model captured *this* environment's or instance's particular constraints. **Biological tissue.** Predicting species-level regeneration is a base rate; the target-specific question is whether the model inferred *this* specimen's latent target-state response to a specific perturbation. In every case the gap between the generic and the target-specific answer is exactly what R2 and the permutation gate are built to expose.

3 The Target-Specificity Stack

We present the controls not as a loose conjunction of desirable properties but as a **stack**: each layer removes a way a forecast can look like understanding without being target-specific. TargetSpace adopts most of these tools from prior work (Section 11); its contribution is assembling them around a single question — *is the forecast genuinely about the target system?* — and adding the two layers that directly test it.

Layer	What it removes / certifies	Status
1. Prospective sealing	forecasts sealed and timestamped before outcomes exist — removes hindsight rationalization and contamination	adopted [2,47]
2. Proper scoring	log / Brier scoring of the whole distribution — rewards calibrated probabilities, not cherry-picked accuracy	adopted [18,19,23]

Layer	What it removes / certifies	Status
3. R1 population-prior baseline	skill must exceed population base rates — filters out generic base-rate prediction	adopted
4. R2 own-routine baseline	skill must exceed the target's *own* strong routine — filters out routine mimicry	**anchor (this work)**
5. Calibration gate	ECE threshold — filters out overconfident lucky prediction	adopted [†HELM]
6. Permutation specificity gate	skill must collapse when forecasts are matched to the wrong target — tests dependence on the correct instance pairing	**anchor (this work)**
7. Evidence ablation	skill as a function of evidence tier — measures which streams add target-specific information	adopted (e.g. active/passive sensing)

Table 1: The target-specificity stack. Layers 1-3, 5, and 7 are established tools we adopt and cite; layers 4 and 6 — the own-routine baseline and the permutation specificity gate — are the anchors that make the stack a test of target-specificity rather than of generic predictive quality.

The stack composes into a single decision rule: a forecast earns *target-specific* credit only if it beats R1 (better than the average), beats a strong R2 (better than the routine), passes calibration (honestly uncertain), and *fails* under permutation (specific to this target). Removing any one layer reopens a way to score well without target-specific understanding, which is why we present them together rather than as separable contributions.

4 From Goals to Target States

The object the stack scores is a target system's **target-state transitions**. The intuitive version — forecast a system's *goals* — smuggles in conscious intent, which excludes most adaptive systems. We therefore speak of **target states**: configurations a system behaves as if acting to reach and to restore when perturbed, whether or not any mind intends them — the minimal generalization that lets one apparatus span persons and non-persons.

That target states need not presuppose conscious goals is supported, carefully, by adaptive biological systems. Levin's work on basal cognition and morphogenesis [44] documents collectives — canonically regenerating planaria — that behave as if storing a *target morphology*: cut them and they rebuild toward the correct anatomy; perturb the right control signal and they stabilize a different but coherent target form. These are target states with *no conscious goal behind them*, which is precisely why they license a benchmark about target states that does not presuppose a mind. **We take from this only a bounded, operational hypothesis — that target state is a privileged, forecastable object of measurement in adaptive systems across scales — and no metaphysics.** The benchmark stands if Levin's framework is wrong and if tissues have no goals in any rich sense. (We keep two researchers distinct: *Michael* Levin, developmental biology, motivates target states beyond conscious intent; *Sergey* Levine [27], reinforcement learning, contributes to the control-as-inference lineage of Section 11.) *Transitions* matter because the moments when what a system is acting to reach changes are where non-routine behavior originates and where a routine baseline fails — the regime in which target-specific skill is decisive.

5 What TargetSpace Is, and Is Not

TargetSpace is not a model, a dataset, or an architecture. It is a benchmark framework — a task definition, a scoring methodology, standardized baselines, an evaluation grid, and integrity and governance protocols — organized as a shared apparatus with multiple application tracks (Section 9). Many datasets may instantiate it and many architectures may compete on it, the sense in which ImageNet is distinct from any network trained on it and SWE-bench from any agent that solves its issues.

5.1 Definitions and the forecast unit

A **target-system instance** is a specific partially observed adaptive system tracked over time (a person, animal, robot, tissue, organization, project, software agent, market). Its **current abstract state** is the latent configuration generating its behaviour at time t; a **latent target state** is a configuration it acts to reach or maintain; a **transition path** is the sequence of target states it moves through. A benchmark instance is the tuple $\boldsymbol{(i, E_{\leq t}, q, A, r)}$: instance i; evidence available up to time t; a query q about a future state with discrete answer space A; and resolution time $r > t$ with a deterministic **resolution rule**. The system outputs a distribution over A. The **unit of analysis is the resolved forecast**, so power scales with the number of forecasts, not the number of instances. Three assumptions are explicit: **A1** the future is partially predictable, bounded above [17]; **A2** latent target states are evaluated only through observable consequences; **A3** the instance changes, so the benchmark rewards adaptation. The measurement is agnostic to which latent structure produces the forecasts — the property that makes the grid of Section 6

architecture-neutral. A **forecast horizon** is part of the object: forecasts are specified at multiple horizons and abstraction levels — *short-horizon* concrete next states, *medium-horizon* routine deviations, and *long-horizon* abstract target-state transitions — reflecting that detailed prediction is reliable only at short horizons while long-horizon prediction requires abstraction. The four tracks (Section 8.2) form exactly this horizon–abstraction ladder, scored throughout against the routine baseline (R2) and the permutation specificity gate.

5.2 The walk-forward sealed protocol

A retrospective benchmark replays logged histories, so apparent forecasting becomes partly retrieval — the failure that motivated contamination-resistant designs [20,21]. TargetSpace privileges a prospective, sealed mode: forecasts are timestamped and sealed (SHA-256) before outcomes exist and resolved automatically. We enforce strict walk-forward (prequential) evaluation — only evidence timestamped $\leq t$, indices rebuilt per slice; random cross-validation is prohibited. **Federation** is a core choice: the harness runs where the data lives, only sealed forecasts and resolved outcomes leave the client, reporting is aggregate-only — solving the dataset problem (most targets cannot be centralized) and the privacy problem at once. Two properties follow. First, target-specific forecasting is inherently **longitudinal rather than IID**: a target's earlier history may become relevant to later forecasts, so the evaluation preserves chronology rather than randomizing examples away from their temporal context. Second, a forecast is **scored against sealed external outcomes**, never for internal self-consistency: a system that produces confident rollouts inside its own latent simulator earns no credit until those rollouts resolve against what the target actually did, which guards against a model that is competent only inside its own dream.

5.3 What TargetSpace is not

Not user modeling. A user model predicts a target's *outputs* (clicks, ratings, responses) to tailor a system; a target-state model forecasts the *evolving latent state that generates those outputs*. **Not event-outcome forecasting.** Sealed proper-scored forecasting of external events is established [47,66,68]; TargetSpace scores transitions in the latent target state of a *tracked instance* with instance-specific controls (R2, permutation) those benchmarks do not use. **Not a competitor to JEPA** or to latent-prediction research (Section 11): it evaluates such systems rather than replacing them.

5.4 Why target-state forecasting differs from generation and imitation

Three capabilities are easy to conflate and worth separating, because TargetSpace scores only the third. **Generation** predicts surface outputs — the next token, the next frame, a fluent answer; it is judged by the plausibility or fidelity of what is produced. **Imitation** (behaviour cloning) predicts a plausible action from demonstrations — given an observation and instruction, emit what a demonstrator would do. **Target-state forecasting** predicts a *consequence*: given a state and the actions or events that follow, what latent target state the system will transition into. Only the third asks whether a model can be held to an outcome, and it is the capability a planner needs — inference becomes search over predicted futures rather than a single plausible output. This aligns TargetSpace with the evaluation of world models, but at the measurement layer: it scores the forecast of the transition, not the reconstruction of the full future, and not the production of a plausible act. Crucially, it does so **without assuming any architecture** — JEPA-style latent predictors, LLMs, VLMs, symbolic/probabilistic models, and imitation-trained policies are all candidate systems, scored identically by whether their forecasts are calibrated and target-specific.

The part of the future that matters. TargetSpace does not ask a system to predict every ripple in the river — every pixel, leaf, token, or surface detail, much of which is irrelevant or intrinsically unpredictable. It asks whether the system identifies and scores the *target-relevant* transition: the part of the future that matters for the specified target, horizon, and evidence tier. This is the same intuition that motivates predicting in a learned representation rather than reconstructing raw observations (Section 11.2), lifted to the measurement layer. It also fixes the benchmark's object precisely as a world-model question made architecture-neutral: given a **target**, **partial evidence**, and a **horizon** (optionally a hypothetical intervention), what distribution does the system assign to the target's **next state**? A system that can only produce a plausible surface, or reconstruct a full but mostly-irrelevant future, has not answered it.

A robotics analogy, stated neutrally. A vision-language-action (VLA) policy [85,86] maps an instruction and observation directly to a plausible next action; a world-model planner instead predicts the *consequence* of a hypothetical action — the next state — and searches over actions to reach a goal. Both are legitimate and both are advancing rapidly; we take no side on which will prevail, and note that action-conditioned world-model planning is at present earlier and more limited than VLA in robotics.

TargetSpace abstracts the distinction to the measurement layer: it does not ask whether a system imitates a plausible action, but whether the system can forecast the target-specific state transition that follows — a question that applies equally to a VLA policy wrapped to emit outcome distributions, a latent world model, or an LLM. The benchmark measures the capability; it does not adjudicate the architecture debate.

6 The Evaluation Grid: Evidence Tier × Architecture Class

Holding the forecast unit and scoring fixed, TargetSpace varies *what a system may observe* (evidence tier) and *what kind of system it is* (architecture class). Every cell is comparable because every system, whatever its internals, emits a distribution over the same A on the same sealed instance. We did not invent architecture-neutral evaluation — representation-agnostic world-model evaluation was introduced by AutumnBench/WorldTest [69], and WorldPrediction [70] already scores latent-predictive and language models on shared instances. TargetSpace's addition to the grid is the target-specificity stack: calibrated, proper-scored, sealed evaluation with R1/R2 baselines and the permutation gate.

6.1 Architecture classes

Architecture class	What it is
LLM-only	text-in, distribution-out language model (semantic / reasoning component)
VLM	vision-language model over image / video evidence
Multimodal agent	tool-using agent integrating heterogeneous evidence, retrieval, and memory
JEPA-style / latent predictive (incl. world models)	learns latent dynamics and predicts in representation space [54–57,72] (latent-dynamics component)
Symbolic / probabilistic	explicit structured or Bayesian model (HMMs, Bayesian filters, routine models)
Hybrid	any combination (neuro-symbolic; LLM reasoning + latent-dynamics prediction)
Human self-report *(baseline)*	the target, or an expert, predicts itself
Oracle upper bound *(baseline)*	a privileged predictor giving an approximate, possibly loose ceiling

Table 2: Architecture classes. LLMs and JEPA-style models are complementary components, not rivals: TargetSpace asks whether any of them produces calibrated, target-specific forecasts. Human self-report and the oracle anchor the achievable range and are not ranked.

6.2 The evidence ladder

The evidence ladder is the benchmark's scientific instrument. Tiers are cumulative; we give the canonical *person-domain* ladder and other domains substitute their own sensor ladders (Section 9), preserving the ordering from already-interpreted residue toward pre-interpretive observation.

Tier	Evidence (person domain)	Sensitivity
L0	calendar + communications metadata, routine, digital exhaust	social graph / rhythms
L1	text traces (chat, notes, documents)	high — third-party content
L2	audio / transcripts	very high — bystanders
L3	passive multimodal observation (egocentric / scene video, ambient audio, screen)	extreme
L4	location / behavioral / mobility traces	extreme — re-identifying
L5	physiological / specialized sensors	extreme — health-revealing

Table 3: The person-domain evidence ladder. By reporting target-specific skill (over R2) as a function of tier, TargetSpace turns 'does richer observation add target-specific information?' into a measured quantity rather than an argument.

The ladder also encodes a distinction richness alone does not. Lower tiers (L0–L1) are records the target already produced by deciding what to write, send, or schedule — they begin *after* relevance was assigned, and largely encode routine. Higher passive tiers are **pre-interpretive**: captured before the target fixed what would matter, they practise **delayed relevance assignment** and carry **retrospective option value** — the same recording can be re-scored under questions chosen later. Pre-interpretive capture is not thereby 'objective'; it is, more precisely, *independent from retrospective human interpretation*. The target-specific hypothesis is sharp: **digital exhaust may predict routine; passive evidence may reveal deviations from routine** — exactly the transitions where skill over R2 is earned. The grid is the cross-product: each leaderboard cell names an (evidence tier,

architecture class) pair, with R1, R2, human self-report, and the oracle spanning all tiers.

The ladder is more than a convenience stratification; it reflects different **bands of access** to the target, each partial and mediated. Textual records are *translated* traces — experience already rendered into language; logs and digital exhaust are *behavioural* traces; audio and video are *temporal observational* traces; and action-conditioned or interventional evidence exposes how the target responds when the world pushes back. No band is reality, and a higher tier is not automatically better — whether it pays is the empirical question the ablation answers. Because the relevance of an observation may only become clear retrospectively, TargetSpace treats **raw-evidence preservation, provenance, and ablation as first-class evaluation concerns**: the benchmark can ask not only whether a system used evidence but *which* evidence was necessary to improve a target-specific forecast. Preservation is always bounded by the consent, federation, aggregate-only, and retention limits of Section 12; the benchmark scores evidence value, it does not license indiscriminate retention.

7 Metrics

The scoring foundation is established practice we **adopt and cite**; we then add metrics for possibility-space resolution. ‘Collapse’ is the flagged epistemic metaphor of Section 1.2 — a distribution concentrating — with no physical meaning.

7.1 Preserved foundation (adopted)

Forecasts are scored with strictly proper rules: the **log score** (primary) and **Brier score** (secondary) [18,19,23]. The headline quantity is **Skill**, in bits: *Skill* = (mean log-score$_{system}$ − mean log-score$_{reference}$) / ln 2, a paired comparison on identical sealed instances, equal in expectation to an information gain — **not a novel metric**. It is reported against **R1** (population prior; the entry condition) and **R2** (the target's recency-weighted, walk-forward, instance- and task-specific routine; the target-specific condition). **Calibration** (top-label ECE: ≤0.10 pass / ≤0.20 warn / >0.20 fail) gates the result, in the spirit of holistic evaluation that treats calibration as first-class [†HELM]. The **permutation specificity gate** scores a system's model for *i* against another instance's outcomes; skill that does not collapse was not target-specific. Uncertainty is reported with a **day-blocked bootstrap**. Of this stack, the **R2 own-routine baseline and the permutation gate** are the unoccupied elements; R1, proper scoring, calibration, and sealing are adopted from forecasting practice [47,66,†HELM,18,19,23]. A deliberate consequence of scoring a distribution over a defined answer space is that **exact surface form is not the success criterion**: a system is judged on proper score, calibration, rank/order, and transition correctness, not on reproducing a reference phrasing. This mirrors the observation from the JEPA literature (Section 11.2) that predicting the *embedding* of a target answer rather than its exact wording avoids penalizing answers that are correct but phrased differently; we adopt the principle at the level of scoring.

7.2 Metrics for possibility-space resolution

Predictive lift by evidence tier is Skill computed as a function of the evidence rung — the evidence-ablation read-out; no new mathematics. **Top-*k* target-state recall** complements distribution-level Skill for large answer spaces. **Transition-path likelihood** is the length-normalized likelihood of the realized path of target-state transitions, in bits vs R1/R2 — the natural metric for the transition track. **Time-to-correct-collapse** and **false-collapse rate** measure *when* a model resolves the possibility space: respectively, how early its distribution concentrates correctly on the realized target, and how often it concentrates confidently and early on the *wrong* one. We are explicit that ‘how early can a target be identified’ is **not** a new idea: it descends directly from goal-recognition design and worst-case distinctiveness [48] and from online goal recognition (recognition time, false-positive rate). Our contribution is to operationalize it as a *proper-scored, calibrated, prospective* quantity on tracked instances, paired so that a model must achieve early correct collapse *and* low false collapse together. We freeze and unit-test these definitions before claiming them and report none as results pre-pilot.

Metric	Status	Nearest prior art (cite, do not claim)
Skill / lift / entropy reduction	adopted (information gain)	skill score; Gneiting & Raftery [19]
Log + Brier, ECE gate, sealing	adopted	[18,23,47]; calibration-as-axis [†HELM]
R2 own-routine + permutation gate	**anchor (this work)**	own-history baselines & self-consistency [7] as cousins
Top-*k* recall; transition-path likelihood	adapted	retrieval; cell-fate / trajectory likelihoods [49,50]
Time-to-correct-collapse	operationalization, not new idea	goal-recognition design / WCD [48]; online goal recognition
False-collapse rate	new metric (concept has precedent)	premature-commitment / false-stop / FPR in goal recognition

Table 4: Metric provenance. The honesty rule: claim the conjunction and the R2 + permutation anchors; concede the rest as adopted, adapted, or (for collapse-timing) a new operationalization of an existing idea.

8 First Instantiation: Personal World Modeling

Personal world modeling is the instantiation we carry furthest, because understanding individual people is valuable and the observation bottleneck bites hardest there. The target-system instance is a consenting individual; the target states are evolving attention, commitments, concerns, priority shifts, and avoidance patterns; the evidence ladder is Table 3. The personal-AI question is sharpened by target-specificity: **not merely 'does passive capture help?' but 'does passive first-person evidence help a model forecast *this person's* future target-state transitions beyond population priors and beyond this person's own routine?'**

Concretely, the forecasts concern **attention allocation, commitment follow-through, response behaviour, meeting and event realization, task drift, avoidance, priority shift, social context, and emotional salience**. The target-specific hypothesis is the same throughout: digital exhaust (L0–L1) largely encodes this person's routine, which a strong R2 already captures; passive evidence (L2–L3) may reveal the *deviations* from routine — the shifted attention, the commitment about to slip — that are exactly where skill over R2 is earned and where the permutation gate confirms the skill is about this person and not the population.

The generation / imitation / forecasting distinction (Section 5.4) maps cleanly onto the personal case. The task is not to *summarize* a person (generation) or to predict *what people usually do* (a base rate, or the imitation of typical behaviour); it is to forecast *this* person's deviations and next-state transitions from partial evidence, while beating that person's own routine baseline. That is the layer a useful personal model must reach and the one current personal AI, trained on routine-laden digital exhaust, rarely demonstrates.

8.1 Attention as a target-specific signal

Attention is the highest-bandwidth observable consequence of a person's latent target states: how someone spends a fundamentally limited resource is a revealed signal of what they are trying to do [43]. Where a discrete action reveals a latent cause once, the trajectory of attention reveals it continuously, often before any action resolves — a leading indicator of target-state transitions. Attention is treated as *evidence*, not ground truth. (In non-person domains the analogue is whatever scarce resource the instance allocates — compute for a software agent, capital for an organization.)

8.2 Tasks and a planned validation study

Version 1 uses auto-scorable, high-frequency targets organized into four domain-neutral tracks instantiated for persons: **TS-R** short-horizon realization (next contact; event realization; response behaviour); **TS-A** attention/allocation; **TS-D** decision/commitment (commitment follow-through); **TS-X** target-state transition/drift. This paper reports no results. The first instantiation, an audio-first personal pilot, is deliberately one narrow instantiation that tests the *apparatus*, not the full cross-domain or cross-architecture thesis: five consenting participants, thirty days, sealed daily calibrated predictions, systems differing only in evidence (A population prior; B digital exhaust; C +text/transcript; D +continuous first-person stream; model and prompts fixed across B–D). *Hypotheses:* B–D beat R1 and R2 (H1); skill is non-decreasing in evidence, the open question being the C→D increment (H2); skill collapses under permutation (H3); null: no system beats R2 (H0). The primary test is the paired log-score difference against R2 per instance with day-blocked intervals; attention allocation is the primary endpoint and commitment follow-through co-primary. **Abandonment criteria:** no system beats R1 → noise; none beats R2 → target-specific forecasting not demonstrated; skill survives permutation → not target-specific.

9 Tracks: A Multi-Track Apparatus

TargetSpace is not a single dataset and not one domain-specific benchmark. It is a shared evaluation apparatus with multiple application tracks. Each track defines domain-specific *targets, evidence sources, horizons, baselines, and outcome validators*, while preserving a common scoring spine. This mirrors how disciplined evaluations are organized — a fixed metric spine applied across scenarios (as in holistic LLM evaluation [†HELM]) or across domain areas (as in systems benchmarking) — rather than as one monolithic task.

The shared scoring spine is invariant across every track: a tracked target object; a partial evidence bundle; a forecast horizon; a sealed, walk-forward prediction; proper scoring; a calibration gate; an own-routine/own-system baseline (R2) and a population baseline (R1); a permutation specificity test; an evidence-ablation ladder; and provenance with deterministic outcome validation (Sections 3, 5, 7). A track is admitted only when it requires a *distinct* validator, evidence band, and horizon profile **and** admits a strong R2 — otherwise it is a regime within an existing track, or not yet a track. We cap the top level at a small number to avoid the dilution that afflicts many-track benchmarks, and we tier each track honestly by readiness.

Track *(status)*	Target object	Example evidence bands	Horizon	Example target states	Validator / outcome
TS-Personal *(current)*	a consenting individual	metadata, text, audio, passive multimodal, location, physiology	hours–weeks	attention, commitments, routine deviation, priority shift	observed action / confirmation
TS-Health *(planned)*	a patient	vitals, labs, CGM, wearables, notes	minutes–days	glycemic excursion, deterioration onset, care-state transition	clinical onset labels / sensor thresholds
TS-Energy *(planned)*	a series / asset	history, weather, calendar, exogenous covariates	hours–days	load / renewable level band, price regime	realized value / market settlement
TS-Robotics *(research)*	embodied agent + scene	proprioception, sensor stream, action log	sub-second–minutes	goal-conditioned configuration, subgoal transition	achieved configuration
TS-Enterprise *(research)*	a project / team / workflow	trackers, commits, comms metadata, releases	days–quarters	milestone state, scope / priority drift	observed milestone / outcome

*Table 5: The five TargetSpace tracks, one shared spine. Status — **current**: apparatus exists (synthetic, pre-pilot) for the person track only; **planned**: strong sealed precedent and a natural own-routine baseline, not yet implemented (energy/grid: GEFCom [87], M4/M5 [88]; physiology: PhysioNet/CinC [89], OhioT1DM/BGLP [90]; personal: GLOBEM [91], HuMob [92]); **research**: a distinct regime exists but a proper-scored forecasting protocol and/or a strong R2 are not yet established. We make **no claim that any track other than the synthetic person track is implemented**, and no claim that TargetSpace solves robotics, healthcare, energy, or enterprise forecasting.*

Adding a track means instantiating the spine, not changing it: define the target objects, the evidence bands, the forecast horizons, the deterministic validators, the R1/R2 baselines, and the standard leaderboard metrics (Section 10). The formulation extends in principle to further substrates — animal tracking, organizational regimes, software agents — as candidate instantiations rather than launch tracks. We **deliberately exclude** one tempting domain: scientific perturbation forecasting (e.g. cell-fate or perturbation-response prediction) is *one-shot, cross-sectional* prediction whose strong baseline is a cross-sectional mean rather than an own-routine R2, and whose target is not tracked over time; it fails the spine and we cite it only as a contrast for what TargetSpace is not. Energy and markets are *one* track with two regimes (physical load/renewables, where seasonal-naive R2 is strong, versus price/financial, where a near-efficient regime makes R2 hard), not two tracks.

10 How to Use TargetSpace

TargetSpace is meant to be run, not only read. A researcher choosing to evaluate a system specifies, in order: **(1)** the target-system instances; **(2)** the forecast questions and their deterministic resolution rules; **(3)** the evidence tiers the system may access; **(4)** the architecture class(es) under test; **(5)** the R1 population-prior and R2 own-routine baselines; **(6)** the sealed forecast times and horizons; **(7)** the scoring metrics; and **(8)** the permutation test. The federated harness then runs where the data lives and reports a standardized row: **Skill vs R1**, **Skill vs R2**, **calibration** (pass/warn/fail), **permutation result** (collapses / survives), **predictive lift by evidence tier**, the **number of resolved forecasts**, the **horizon**, and the **target domain**. The reporting contract is the point: every row discloses, simultaneously, whether the system beat the average, beat the routine, stayed calibrated, and depended on the correct target — so a reader can tell target-specific understanding from generic prediction at a glance.

11 Related Work

11.1 What prior work already owns (and we adopt, not claim)

Credibility requires being explicit about what TargetSpace does *not* invent. We adopt and cite, rather than claim, each of the following.

Ingredient	Owned / established by	TargetSpace stance
Representation-agnostic / architecture-neutral world-model evaluation	AutumnBench / WorldTest [69]	adopt the framing; add calibration + R1/R2 + permutation
Cross-architecture comparison on shared instances	WorldPrediction [70] (ran JEPA + LLMs)	adopt; add proper scoring + sealing + specificity
Latent predictive representation learning	CPC [72]; JEPA [54–57]	precedent for the latent-predictive class; we evaluate it
Proper scoring & calibration as a measured axis	Good/Brier/Gneiting&Raftery [18,23,19]; HELM [†]	adopt directly
Sealed / prospective / leakage-free forecasting	ForecastBench [47]; Prophet Arena [66]; SWE-bench [2]	adopt
Evidence ablation (e.g. active vs passive sensing)	digital-phenotyping forecasting literature	adopt; report as lift over R2
Goal recognition & time-to-identification / collapse timing	goal-recognition design / WCD [48]; online goal recognition	adopt; re-cast as proper-scored, prospective, on tracked instances

Table 6: What TargetSpace adopts rather than claims. Its remaining contribution is target-specific forecasting under strong controls — the R2 own-routine baseline and permutation specificity gate — assembled around one evaluation question.

11.2 World models, JEPA, and predictive learning

A long lineage predicts the *latent* state of the world rather than its surface — predictive coding [61,24]; world-model agents that learn latent dynamics and plan by imagining rollouts [16,58,59,60]; the joint-embedding, non-contrastive methods (Barlow Twins, VICReg [73,74]) and contrastive predictive coding [72] that learn representations by prediction rather than reconstruction; and LeCun's Joint Embedding Predictive Architecture (JEPA) program [54] with its image, video, and video-plus-planning instantiations [55,56,57]. **This lineage is long, and we flag it to avoid over-crediting any single architecture.** Information-theoretic precursors include Barlow's redundancy-reduction principle [80] and Linsker's Infomax [81]; in neural networks, Schmidhuber and Prelinger's *Discovering Predictable Classifications* [76] learned latent classifications predictable across networks while constrained to remain informative (avoiding collapse), and Schmidhuber's recurrent world-model and artificial-curiosity work [77,78] trained models to predict future observations — anticipating later joint-embedding predictive architectures, though predating the JEPA terminology. (A related but distinct objective, predictability *minimization* [79], instead removes redundancy to learn factorial codes; we keep the two separate.) The modern self-supervised line — language-supervised CLIP [82] and the joint-embedding DINO family [83,84] — continues it. JEPA is best understood not as a model but as a *training framework*: encode an input and a target, and train a predictor to predict the target's *embedding* — optionally conditioned on an action — rather than reconstructing raw pixels or tokens. The motivation bears directly on our object of evaluation: generative next-frame or next-token prediction spends capacity on unpredictable, irrelevant detail and, in video, collapses to blurry averages of possible futures, whereas predicting in embedding space can keep the salient structure and discard what cannot be predicted — the same reason TargetSpace scores *transitions in a latent target state* rather than reconstruction of a full future. Conditioned on actions, a JEPA predictor becomes a world model usable for planning by search over predicted futures [57,75], the capability LeCun argues agentic systems require. Uniformly, however, this work delivers *architectures, methods, and planning procedures*, evaluated by representation quality, reward, or task success — not by calibrated, proper-scored forecasting of target-state transitions with instance-specific controls. **We treat JEPA-style models as a natural architecture class for TargetSpace and as precedent, not as a competitor**: they are candidate *ways to produce* latent predictions, and TargetSpace is a way to *measure* whether those predictions are calibrated and target-specific. We claim no superiority for them — action-conditioned JEPA planning is early, limited in horizon, and behind vision-language-action (VLA) policies in current robotics demonstrations [57,75] — and we take no side in the debate over which architecture will prevail. Even V-JEPA 2 [57], the closest case, is scored by robot task-success and action-anticipation accuracy, not by calibrated target-specific forecasting, so it is a system TargetSpace can *evaluate*, not an instance of TargetSpace. (We do not conflate JEPA's prediction in *representation* space with forecasting in *target/outcome* space, and we do not claim to validate world models in the full sense of LeCun's program.)

11.3 Forecasting, person modeling, and goal recognition

Calibrated event forecasting is established — ForecastBench [47], Prophet Arena [66], and others — but scores external public events, not target-state transitions of a tracked instance, and uses no own-routine or permutation control. Person-modeling

benchmarks are the nearest in framing: Park et al. [7] replicate individuals' survey answers (their own-consistency reference is the closest cousin to R2, though used as a ceiling, not a baseline); KnowMe-Bench [45] formalizes person understanding but as *retrospective* comprehension, not prospective forecasting; EgoToM [46] infers a camera-wearer's goals and future actions from egocentric video, but per clip, by accuracy, without calibration or a persistent target; PersonaMem [9] tracks evolving preferences. Person-specific multi-horizon forecasting with active/passive evidence comparison already exists in digital phenotyping — we concede this and differentiate on proper scoring, calibration, and the permutation gate. Goal- and plan-recognition [48] infer latent target/goal posteriors under partial observability and define how early a target is identifiable; we adopt that idea for the collapse-timing metrics and re-cast it as a calibrated, prospective, instance-tracked measurement.

11.4 Cross-scale target-state systems

Beyond persons, the target-state idea recurs: active inference and preferred states [53,24]; cell-fate methods estimating terminal states [49,50]; structure prediction treating the native fold as an attractor [51]; and Levin's morphogenesis [44] motivating target states beyond conscious intent (Section 4). Each forecasts or infers a target state from evidence; we cite them as lineage and sibling domains, as methods rather than calibrated benchmarks, not as competitors.

11.5 The conjunction

Each ingredient above is occupied somewhere; the conjunction is not. We state the supported claim narrowly: **to our knowledge no existing benchmark scores prospective, calibrated, proper-scored forecasts of a tracked target system's latent target-state transitions under a strong instance-specific own-routine baseline (R2) and an instance-permutation specificity gate, with an evidence-tier ablation, across architecture classes.** We do *not* claim to be first at architecture-neutral evaluation [69], cross-architecture comparison [70], calibrated sealed forecasting [47], evidence ablation, or collapse-timing [48]. The contribution is their assembly around one measurable question.

Most benchmarks cannot tell target-specific understanding from base-rate exploitation. TargetSpace is built to make that distinction measurable.

12 Discussion

What the measurement stands in for. The capability at stake is understanding the specific target; target-specific forecasting is its measurable proxy. A model that cannot anticipate a target beyond its crowd and its routine does not, operationally, understand it. TargetSpace quantifies the proxy without asserting that a high score is understanding in a rich sense.

Forecasting is not the end goal. It is the ruler — and the ruler is calibrated to the target, not the average.

Relation to predictive architectures. TargetSpace and JEPA-style models are complementary: the latter are candidate *ways to do* latent prediction; TargetSpace is a *way to measure* whether any such system forecasts target-state transitions, calibrated and target-specific. LLMs can serve as semantic/reasoning components and latent-dynamics models as predictive components within systems under test; TargetSpace takes no side on architecture. **Limitations.** (L1) Latent targets are evaluated only through observable proxies. (L2) The federated/prospective design makes reproduction harder than a downloadable dataset; we mitigate with a versioned protocol and shared harness. (L3) Single-instance and small-cohort evaluation limits external validity. (L4) Predictability is bounded [17] and heterogeneous, so reporting is per-instance. (L5) Pre-interpretive capture is bounded by current sensing — engineering limits, not the long-term boundary. **(L6) Domain-generality is a property of the formulation, demonstrated in one domain; the cross-domain claim is specified, not established. Ethics and privacy.** The data are sensitive and a forecaster of behaviour could be used to influence it; we specify informed revocable consent, federation so raw data never leaves the target's control, aggregate-only reporting, institutional review for human-subjects deployment, respect for recording law, and a prohibition on evaluating systems that act on the target during the window — a benchmark that rewarded changing the target would measure influence, not understanding. Manipulation, surveillance normalization, and third-party consent are open problems we do not consider solved.

Target-context collapse (a named failure mode). A target state is not meaningful in isolation; it is defined relative to a target system, a reference frame, and an evaluation context. We call **target-context collapse** the failure in which a forecast or plan optimizes or predicts a state while erasing the context that gives the state its meaning — for example, optimizing engagement while degrading user well-being, optimizing safety by eliminating autonomy, or, in personal world modeling, predicting a generic idealized behaviour instead of this person's live trajectory. We raise this as an evaluation/failure-mode concept, not a

sweeping safety theorem: TargetSpace's instance-permutation gate and its observe-not-intervene rule are partial guards (a forecast that fits a generic profile rather than the specific target tends to survive permutation and to lose skill over R2), but detecting context collapse in general remains open.

Balanced view of latent world models. Because JEPA-style and other latent predictors are a natural architecture class here (Section 11.2), it is worth stating their open evaluation challenges plainly and without partisanship. Latent state can be hard to ground outside the model's own geometry; deterministic latent prediction may under-represent uncertainty; long-horizon rollouts can drift; multi-agent settings require branching futures and belief/policy modeling rather than averaged outcomes; energy or latent compatibility is not the same as value or reward; and physical-world competence currently lags symbolic and digital competence. These are reasons TargetSpace scores calibrated distributions against sealed external outcomes rather than rewarding internal latent compatibility — and reasons we present current latent-world-model results as promising but not decisive, for any architecture.

13 Open Research Questions

Each is a controlled comparison of target-specific Skill. **1. Evidence sufficiency:** which tier first beats R2, per task and domain? **2. Decay:** how fast does target-specific skill decay once evidence stops? **3. Sparsity:** can sparse evidence substitute for continuous observation, at what loss? **4. Modality value:** which modalities add marginal skill over R2? **5. Refresh rate:** how often must a model update to stay calibrated? **6. Transition anticipation:** can target-state transitions be predicted before they are explicit (time-to-correct-collapse)? **7. Observation vs. self-report:** can passive observation beat self-report, and where? **8. Cross-architecture and cross-domain transfer:** do the architectures that earn target-specific skill in the person domain do so elsewhere? **9. Active evidence acquisition (a proposed extension, distinct from the TS-A track):** given partial evidence, which missing modality or time window would most reduce uncertainty about the target's next state, does a system know when it lacks enough evidence to forecast, and can it request useful evidence without overreaching? Such an extension would score expected information gain, uncertainty reduction, forecast improvement after a requested observation, cost-adjusted evidence value, and consent-aware requesting — turning evidence acquisition itself into a measured, privacy-bounded capability rather than a fixed input.

14 Conclusion

Many systems now claim to understand specific targets, and almost none can show their forecasts are target-specific rather than reflections of base rates and routine. TargetSpace makes that distinction measurable: prospective, sealed, proper-scored forecasting of target-state transitions under partial observation, certified by an own-routine baseline, a calibration gate, and a permutation specificity test, with an evidence-tier ablation and an architecture-neutral grid. Its novelty is the conjunction assembled around one question — *is the forecast about the target, or about the average?* — not any single ingredient, each of which it adopts and cites. Personal world modeling is its first and highest-value instantiation; the formulation extends, by stated criteria, to other tracked target systems.

Forecast the target, not the average.

References

Citations marked ‡ are 2025–2026 works whose arXiv identifiers were checked but should be re-verified at camera-ready; † marks a work cited by name pending final bibliographic formatting. Identifiers without a marker were verified against primary sources during preparation. No citation was invented.

A. Benchmarks and evaluation methodology

[1] J. Deng et al. ImageNet: A Large-Scale Hierarchical Image Database. CVPR, 2009.

[2] C. E. Jimenez et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? ICLR, 2024. arXiv:2310.06770.

[20] C. White et al. LiveBench: A Contamination-Limited LLM Benchmark. 2024. arXiv:2406.19314.

[21] SWE-bench-Live: Contamination-Resistant Evaluation for Software Agents. 2025. arXiv:2505.23419.

[34] A. Wang et al. GLUE: A Multi-Task Benchmark for Natural Language Understanding. ICLR, 2019. arXiv:1804.07461.

[35] A. Wang et al. SuperGLUE. NeurIPS, 2019. arXiv:1905.00537.

[36] D. Hendrycks et al. Measuring Massive Multitask Language Understanding (MMLU). ICLR, 2021. arXiv:2009.03300.

[37] M. Chen et al. Evaluating Large Language Models Trained on Code (HumanEval). 2021. arXiv:2107.03374.

[71] P. Liang et al. Holistic Evaluation of Language Models (HELM). TMLR, 2023. arXiv:2211.09110. (Calibration as a first-class measured axis.)

B. Forecasting, calibration, scoring, goal recognition

[17] A. Bellot, J. Richens, T. Everitt. The Limits of Predicting Agents from Behaviour. ICML, 2025. arXiv:2506.02923.

[18] I. J. Good. Rational Decisions. J. Royal Statistical Society B, 14(1), 1952.

[19] T. Gneiting, A. E. Raftery. Strictly Proper Scoring Rules, Prediction, and Estimation. JASA, 102(477), 2007.

[23] G. W. Brier. Verification of Forecasts Expressed in Terms of Probability. Monthly Weather Review, 78(1), 1950.

[38] A. H. Murphy. What Is a Good Forecast? Weather and Forecasting, 8(2), 1993.

[47] E. Karger et al. ForecastBench: A Dynamic Benchmark of AI Forecasting Capabilities. ICLR, 2025. arXiv:2409.19839.

[48] S. Keren, A. Gal, E. Karpas. Goal Recognition Design (worst-case distinctiveness). ICAPS, 2014.

[66] Q. Yang et al. Prophet Arena (LLM-as-a-Prophet): a live prediction-market forecasting benchmark. 2025. arXiv:2510.17638.

[67] L. Nel. KalshiBench: Epistemic Calibration via Prediction Markets. 2025. arXiv:2512.16030. (recent preprint)‡

[68] Goel et al. FutureSim: Replaying World Events to Evaluate Adaptive Agents. 2026. arXiv:2605.15188.‡

C. World models, JEPA, predictive learning

[16] D. Ha, J. Schmidhuber. World Models. NeurIPS, 2018. arXiv:1803.10122.

[24] K. Friston. The Free-Energy Principle: A Unified Brain Theory? Nat. Rev. Neuroscience, 11, 2010.

[27] S. Levine. Reinforcement Learning and Control as Probabilistic Inference. 2018. arXiv:1805.00909.

[53] C. Heins et al. pymdp: A Python library for active inference. JOSS, 2022. arXiv:2201.03904.

[54] Y. LeCun. A Path Towards Autonomous Machine Intelligence. v0.9.2, 2022. OpenReview BZ5a1r-kVsf. (Position paper; not arXiv.)

[55] M. Assran et al. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture (I-JEPA). CVPR, 2023. arXiv:2301.08243.

[56] A. Bardes et al. Revisiting Feature Prediction for Learning Visual Representations from Video (V-JEPA). 2024. arXiv:2404.08471.

[57] M. Assran et al. V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning. 2025. arXiv:2506.09985.

[58] D. Hafner et al. Learning Latent Dynamics for Planning from Pixels (PlaNet). ICML, 2019. arXiv:1811.04551.

[59] D. Hafner et al. Mastering Diverse Domains through World Models (DreamerV3). 2023. arXiv:2301.04104.

[60] J. Schrittwieser et al. Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model (MuZero). Nature 588, 2020. arXiv:1911.08265.

[61] R. P. N. Rao, D. H. Ballard. Predictive Coding in the Visual Cortex. Nature Neuroscience, 2(1), 1999.

[69] A. Warrier et al. Benchmarking World-Model Learning with Environment-Level Queries (AutumnBench / WorldTest). 2025. arXiv:2510.19788.

[70] D. Chen et al. WorldPrediction: A Benchmark for High-level World Modeling and Long-horizon Procedural Planning. 2025. arXiv:2506.04363.

[72] A. van den Oord, Y. Li, O. Vinyals. Representation Learning with Contrastive Predictive Coding (CPC). 2018. arXiv:1807.03748.

[73] A. Bardes, J. Ponce, Y. LeCun. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning. ICLR, 2022. arXiv:2105.04906.

[74] J. Zbontar, L. Jing, I. Misra, Y. LeCun, S. Deny. Barlow Twins: Self-Supervised Learning via Redundancy Reduction. ICML, 2021. arXiv:2103.03230.

[75] G. Zhou, H. Pan, Y. LeCun, L. Pinto. DINO-WM: World Models on Pre-trained Visual Features enable Zero-shot Planning. 2024. arXiv:2411.04983.

[76] J. Schmidhuber, D. Prelinger. Discovering Predictable Classifications. Neural Computation, 5(4):625–635, 1993.

[77] J. Schmidhuber. Making the World Differentiable: On Using Fully Recurrent Self-Supervised Neural Networks for Dynamic Reinforcement Learning and Planning. Tech. Rep. FKI-126-90, TU Munich, 1990.

[78] J. Schmidhuber. A Possibility for Implementing Curiosity and Boredom in Model-Building Neural Controllers. Proc. Simulation of Adaptive Behavior (SAB), 1991.

[79] J. Schmidhuber. Learning Factorial Codes by Predictability Minimization. Neural Computation, 4(6):863–879, 1992.

[80] H. B. Barlow. Possible Principles Underlying the Transformation of Sensory Messages. In Sensory Communication (W. Rosenblith, ed.), MIT Press, 1961.

[81] R. Linsker. Self-Organization in a Perceptual Network (Infomax). Computer (IEEE), 21(3), 1988.

[82] A. Radford et al. Learning Transferable Visual Models From Natural Language Supervision (CLIP). ICML, 2021. arXiv:2103.00020.

[83] M. Caron et al. Emerging Properties in Self-Supervised Vision Transformers (DINO). ICCV, 2021. arXiv:2104.14294.

[84] O. Siméoni et al. DINOv3. 2025. arXiv:2508.10104.

[85] A. Brohan et al. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. 2023. arXiv:2307.15818.

[86] K. Black et al. (Physical Intelligence). π0: A Vision-Language-Action Flow Model for General Robot Control. 2024. arXiv:2410.24164.

D. Person modeling, theory of mind, personalization, observation

[3] A. Salemi et al. LaMP: When Large Language Models Meet Personalization. SIGIR, 2024. arXiv:2304.11406.

[6] J. S. Park et al. Generative Agents: Interactive Simulacra of Human Behavior. UIST, 2023. arXiv:2304.03442.

[7] J. S. Park et al. Generative Agent Simulations of 1,000 People. 2024. arXiv:2411.10109.

[8] N. C. Rabinowitz et al. Machine Theory of Mind (ToMnet). ICML, 2018. arXiv:1802.07740.

[9] B. Jiang et al. Know Me, Respond to Me: Benchmarking LLMs for Dynamic User Profiling (PersonaMem). COLM, 2025. arXiv:2504.14225.

[10] S. Zhao et al. Do LLMs Recognize Your Preferences? (PrefEval). ICLR, 2025. arXiv:2502.09597.

[11] Twin-2K-500: Digital Twins of over 2,000 People. 2025. arXiv:2505.17479.

[12] Z. Wang, Y. Lu et al. OPeRA. 2025. arXiv:2506.05606.

[13] J. Li et al. How Far are LLMs from Being Our Digital Twins? (BehaviorChain). Findings of ACL, 2025. arXiv:2502.14642.

[15] M. Binz, E. Schulz et al. A Foundation Model to Predict and Capture Human Cognition (Centaur). Nature, 2025. arXiv:2410.20268.

[45] T. Wu et al. KnowMe-Bench: Evaluating Person Understanding in Language Models. 2026. arXiv:2601.04745.‡

[46] EgoToM: Theory of Mind from Egocentric Video. Meta, 2025. arXiv:2503.22152.

[52] MMToM-QA: Multimodal Theory of Mind Question Answering. ACL, 2024. arXiv:2401.08743.

[40] R. E. Nisbett, T. D. Wilson. Telling More Than We Can Know. Psychological Review, 84(3), 1977.

[43] H. A. Simon. Designing Organizations for an Information-Rich World, in M. Greenberger (ed.), 1971. (Attention as the scarce resource.)

E. Cross-scale target-state systems

[44] M. Levin. Technological Approach to Mind Everywhere (TAME). Frontiers in Systems Neuroscience, 16:768201, 2022.

[49] M. Lange et al. CellRank for directed single-cell fate mapping. Nature Methods, 19, 2022.

[50] W. Saelens et al. A comparison of single-cell trajectory inference methods. Nature Biotechnology, 37, 2019.

[51] J. Jumper et al. Highly accurate protein structure prediction with AlphaFold. Nature, 596, 2021. (CASP14.)

F. Domain forecasting precedents (track validation regimes)

[87] T. Hong et al. Probabilistic Energy Forecasting: Global Energy Forecasting Competition 2014 (GEFCom2014). Int. J. Forecasting, 32(3), 2016.

[88] S. Makridakis, E. Spiliotis, V. Assimakopoulos. The M4 / M5 Competitions: results and findings. Int. J. Forecasting, 2020 / 2022.

[89] M. A. Reyna et al. Early Prediction of Sepsis (PhysioNet/Computing in Cardiology Challenge 2019). Critical Care Medicine, 2020.

[90] C. Marling, R. Bunescu. The OhioT1DM Dataset and the Blood Glucose Level Prediction (BGLP) Challenge. KDH/KHD, 2018/2020.

[91] X. Xu et al. GLOBEM: Generalization of Longitudinal Behavior Modeling. NeurIPS Datasets & Benchmarks, 2022.

[92] HuMob Challenge: Human Mobility Prediction. ACM SIGSPATIAL, 2023/2024.

TargetSpace pre-pilot draft v2.4 (June 2026). Benchmark proposal; no experimental results. ***Reframe (v2.0 → v2.1):*** *retitled to 'Benchmarking Target-Specific Forecasting Under Partial Observation' and reorganized around one evaluation question — is a forecast about the target or the average? Added 'Why Target-Specificity Matters' (§2) and 'The Target-Specificity Stack' (§3); recast the five-part conjunction as a stack anchored on the R2 own-routine baseline and the permutation specificity gate; added a 'What Prior Work Already Owns' concessions table (§11.1) and a 'How to Use TargetSpace' section (§10). Per the JEPA/prior-art audit, removed all standalone 'first' claims (architecture-neutral, cross-architecture, evidence-ablation, collapse-timing) and conceded them to AutumnBench/WorldTest [69], WorldPrediction [70], digital-phenotyping forecasting, and goal-recognition design [48]; added HELM [71] (calibration-as-axis) and CPC [72] (latent-future-prediction ancestor); kept JEPA as precedent and an evaluable architecture class, not a competitor. Honesty posture retained: pre-pilot, synthetic-only where applicable, no results; raw data private / apparatus public; passive capture 'independent from retrospective human interpretation,' not 'objective'; 'collapse' a flagged epistemic metaphor with no physical claim; nothing anti-LLM. Personal world modeling remains the first and highest-value instantiation; 'domain-general in formulation,' not empirically validated across domains.* ***Update (v2.1 → v2.2):*** *sharpened the generation / imitation / target-state-forecasting distinction (contribution 1; new §5.4 with a neutral VLA-vs-world-model robotics example); framed JEPA as a training framework (predict embeddings, not pixels/tokens; optionally action-conditioned) and added the generation-vs-embedding motivation, with no superiority claim and an explicit note that action-conditioned JEPA planning is early and behind VLA today (§11.2); added a forecast-horizon / abstraction hierarchy to the benchmark object (§5.1) and a scoring note that exact surface form is not the success criterion (§7.1); added verified references VICReg [73], Barlow Twins [74], DINO-WM [75]. The Welch Labs JEPA video informed this framing as an interpretive source only and is not cited; CLIP/DINO/DINOv3/VL-JEPA/RT-2/π0 remain on a citation-verification TODO (docs/targetspace/12) and are not cited until verified.* ***Update (v2.2 → v2.3):*** *incorporated a further world-model / Schmidhuber interpretive transcript set (docs/targetspace/13–14) as framing only (not cited). Added target-relevance ('do not predict every ripple') and the architecture-neutral world-model framing (§5.4); evidence bands plus preservation/provenance/ablation as first-class (§6.2); longitudinal non-IID evaluation and sealed-external-outcome scoring against simulator self-consistency (§5.2);* ***target-context collapse*** *as a named failure mode and a balanced latent-world-model limitations paragraph (§12); an* ***active evidence acquisition*** *future extension (§13). Added a credit-neutral historical-lineage passage (§11.2) with primary citations verified before inclusion — Barlow [80], Linsker [81], Schmidhuber & Prelinger 1993 [76], Schmidhuber 1990/1991 [77,78], predictability minimization [79] (kept distinct), CLIP [82], DINO/DINOv3 [83,84] — and RT-2 [85] and π0 [86] as VLA exemplars. No priority/credit claim is made; no transcript is cited; JEPA is neither asserted superior nor obsolete; the paper remains architecture-neutral and centered on calibrated target-specific state-transition forecasting under partial observation.* ***Update (v2.3 → v2.4):*** *reframed TargetSpace as a* ***multi-track benchmark apparatus*** *rather than a single-domain benchmark, grounded in a taxonomy research pass (docs/targetspace/15–16). Rewrote §9 into five status-tiered tracks on one shared scoring*

spine — TS-Personal (current, synthetic), TS-Health and TS-Energy (planned), TS-Robotics and TS-Enterprise (research) — replacing the prior substrate table; made the invariant spine explicit; added apparatus language to the abstract and §5; deliberately excluded scientific/perturbation forecasting (one-shot cross-sectional, fails the own-routine spine) and merged energy+markets into one two-regime track. Added verified domain-precedent references for planned tracks: GEFCom [87], M4/M5 [88], PhysioNet/CinC sepsis [89], OhioT1DM/BGLP [90], GLOBEM [91], HuMob [92]. No claim that any track other than the synthetic person track is implemented; no claim TargetSpace solves robotics, health, energy, or enterprise forecasting.