Learning to Adapt: Self-Supervised Representations for Robust Contextual Bandits

János Horváth [1]

Abstract

We propose a new *self-supervised domain adaptation* framework for **contextual bandits**, addressing both abrupt and gradual environment shifts. Our method pretrains a compact representation on unlabeled data, then integrates it into both classical (e.g., `LinUCB`, `TS`) and neural bandit algorithms. Empirically, we show that our approach dramatically reduces regret and speeds adaptation across eight distinct domains, outperforming standard non-adaptive baselines and simpler autoencoder methods in final performance.

1. Introduction

Contextual bandits drive personalized decisions in *recommendation* (Li et al., 2010a), *healthcare* (Shortreed et al., 2011), and *robotics* (Kober et al., 2013), repeatedly observing contexts, choosing actions, and receiving rewards. While the standard goal is to minimize regret through adaptation, real-world environments often *shift*, invalidating static assumptions and degrading performance. *Domain adaptation* addresses such shifts by leveraging knowledge from earlier domains. Here, we propose **self-supervised representation learning** to extract robust features from unlabeled data, mitigating nonstationarity in contextual bandits.

Contributions. We develop a **self-supervised domain adaptation** pipeline:

1. *Offline pretraining*: Learn a compressed embedding via *autoencoder* or *contrastive* objectives on unlabeled samples.
2. *Online integration*: Use the learned representation in classical or neural bandit algorithms for a robust, low-dimensional feature space.
3. *Multi-domain shifts*: Evaluate abrupt and gradual changes (Domains A–H), showing reduced regret and improved final performance.

We compare baselines like `NoRep`, `VanillaAE`, `MaskedAE`, and bandits like `EpsGreedy`, `LinUCB`, `TS`, `NeuralEG`, and `NeuralTS`. Our results demonstrate **faster adaptation**, **lower cumulative regret**, and near-Oracle outcomes.

2. Related Work

Contextual Bandits. Classic contextual bandits (Langford & Zhang, 2007; Li et al., 2010a) optimize decisions given context features. Approaches such as `LinUCB` (Li et al., 2010b) and `ThompsonSampling` (Chapelle & Li, 2011) have been widely applied to recommendation and personalization. Extensions include *Neural Bandits* (Riquelme et al., 2018; Zhang et al., 2022) that replace linear reward models with neural networks.

Nonstationary and Multi-Domain Bandits. To handle changing data distributions, prior works propose `sliding window` or `restart` heuristics (Garivier & Moulines, 2011; Cheung & Mannor, 2019), or weighted regression approaches (Lu et al., 2021), but these typically do not harness explicit *domain adaptation* principles. Multi-domain or multi-stage bandits (Bouneffouf et al., 2020) consider a scenario with piecewise-constant transitions, but they often retrain from scratch. By contrast, our method *transfers* knowledge via a common representation that is robust to domain shifts.

Representation Learning for Bandits. Recent efforts incorporate unsupervised or self-supervised features into bandits to improve sample efficiency and handle distribution changes (Pascual & Agmon, 2020; Mitton et al., 2022). For instance, Zhang et al. (2023b) examine *identifiability guarantees* for causal disentanglement under soft interventions, a perspective that can inform domain-invariant embedding strategies in bandits. Moreover, Zhang et al. (2023a) propose an *active learning* approach for intervention design in causal models, shedding light on structured policy updates relevant to multi-domain bandit scenarios. Ap-

[*]Equal contribution [1]Visionary Tech & Event Solutions, USA. Correspondence to: János Horváth <horvath_janos@visionarytecheventsolutions.com>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

proaches may use `autoencoders` to reduce dimensionality or `contrastive` objectives (e.g., InfoNCE) to learn domain-invariant embeddings. We specifically focus on *eight domains* with abrupt or gradual transitions, extensively comparing `NoRep` (baseline), `VanillaAE`, `MaskedAE`, and `Contrastive`.

Abrupt vs. gradual domain shift. We define a shift as *abrupt* when the total-variation distance between successive context distributions crosses 0.30 in a single time step. A shift is *gradual* when the same cumulative distance is accrued over at least 50 steps. Unless otherwise noted, domain shift acts only on the context distribution $P(x)$; the conditional reward model $P(r \mid x, a)$ is kept fixed. We report the average ΔTV for every benchmark in Appendix A.2 so readers can map quantitative severity to our qualitative labels.

Evaluation metrics. We use two standard measures: (i) **Cumulative reward** $\sum_{t=1}^{T} r_t$, the total payoff over horizon T (larger is better); and (ii) **Normalized cumulative regret** where $r_t^\star$ is the reward of an oracle that always pulls the best arm in hindsight. The oracle operates with knowledge of future rewards and therefore serves solely as an optimistic ceiling. It is included to visualise remaining headroom, not as a deployable baseline.

Masked and Contrastive Pretraining. Inspired by `BERT` (Devlin et al., 2019) in NLP and `MAE` (He et al., 2022) in vision, masked autoencoders reconstruct missing tokens or patches. Contrastive methods (Chen et al., 2020) learn feature invariances via augmented pairs. Moreover, recent fairness research (Kusner et al., 2017) illustrates the importance of handling unobserved confounders in shifting settings, a challenge mirrored in multi-domain bandits. Although these ideas are prevalent in vision/language tasks, their use in contextual bandits remains relatively understudied.

3. Methodology

In this section, we describe our overall *domain adaptation* approach for contextual bandits, focusing on two key components: *(i)* self-supervised representation learning (to embed raw contexts into a lower-dimensional, domain-invariant space) and *(ii)* adaptable bandit strategies (that learn to select arms given these representations). We assume a setting where domain shifts occur over time—either abruptly or gradually.

3.1. Problem Formulation

We consider a **contextual bandit** scenario with K arms, where at each time step t we observe a *context* vector $\mathbf{x}_t \in \mathbb{R}^d$ and must choose an arm $a_t \in \{1, \ldots, K\}$. We then observe a reward r_t drawn from an unknown function $r_t = R(a_t, \mathbf{x}_t)$ subject to noise.

In this paper, the *context distribution* (and corresponding reward mapping) changes over **multiple domains**, labeled A, B, C, etc. Domain transitions can be:

- **Abrupt:** the environment instantly shifts from domain i to j,
- **Gradual:** the environment slowly interpolates from one distribution to another.

Our **goal** is to learn a policy $\pi(\mathbf{x}_t)$ that selects high-reward arms consistently, even when shifting between domains.

3.2. Self-Supervised Representation Learning

To help the bandit adapt across domains, we first learn an *embedding* function

$$f : \mathbb{R}^d \to \mathbb{R}^m$$

that maps raw contexts $\mathbf{x}_t$ into a lower-dimensional representation $\mathbf{z}_t = f(\mathbf{x}_t)$. We train f **offline** (before the bandit interaction) on unlabeled data from at least one domain (here, Domain A). This pretraining step encourages domain-invariant or robust features. We consider **four** approaches:

NoRep (No Representation). As a baseline, we use the raw context $\mathbf{x}_t$ directly; i.e., f is the identity map. This helps quantify improvements due to representation learning.

VanillaAE (Autoencoder). A standard autoencoder comprises an encoder $E(\cdot)$ and a decoder $D(\cdot)$:

$$\mathbf{z} = E(\mathbf{x}), \quad \widehat{\mathbf{x}} = D(\mathbf{z}).$$

We train it by minimizing the mean-squared error $\|\mathbf{x} - \widehat{\mathbf{x}}\|^2$, thus encouraging E to learn a *compact* representation of $\mathbf{x}$. We retain only $E(\cdot)$ at test time for the bandit.

MaskedAE (Toy Masked Autoencoder). Inspired by masked image/language modeling, we randomly zero out features in $\mathbf{x}$ during training (with some ratio ρ) and task the autoencoder to reconstruct the full context. This can encourage robustness if domain changes selectively corrupt or shift subsets of features.

Contrastive (Toy Contrastive Encoder). We apply a simplified InfoNCE-like loss: each sample $\mathbf{x}$ is augmented randomly twice, producing $\mathbf{x}_1^+, \mathbf{x}_2^+$, and the encoder $E(\cdot)$ aims to map them close in latent space while pushing away other samples. This can foster *domain-invariant* embeddings if augmentations approximate domain perturbations.

After training, we freeze $E(\cdot)$ and pass each new context $\mathbf{x}_t$ through it: $\mathbf{z}_t = E(\mathbf{x}_t)$.

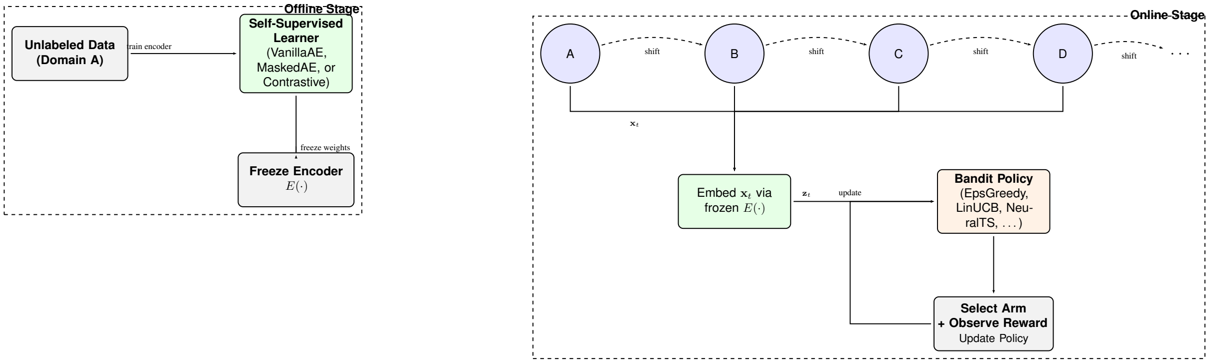
Offline Stage
Unlabeled Data (Domain A)
train encoder
Self-Supervised Learner (VanillaAE, MaskedAE, or Contrastive)
freeze weights
Freeze Encoder $E(\cdot)$
Online Stage
A
B
C
D
shift
shift
shift
shift
. . .
$\mathbf{x}_t$
Embed $\mathbf{x}_t$ via frozen $E(\cdot)$
$\mathbf{z}_t$
update
Bandit Policy (EpsGreedy, LinUCB, NeuralTS, . . .)
Select Arm
+ Observe Reward
Update Policy

Figure 1. **Self-Supervised Domain Adaptation Pipeline for Robust Contextual Bandits.** Left: In the *offline stage*, unlabeled data from Domain A trains an encoder E, which is then frozen. Right: In the *online stage*, streaming domains (A, B, C, . . .) undergo abrupt or gradual *shifts*. Each incoming context $\mathbf{x}_t$ is embedded to $\mathbf{z}_t$ via the frozen E, fed into a bandit policy, an arm is selected, reward observed, and the policy is updated continually.

3.3. Bandit Algorithms

Given the embedding $\mathbf{z}_t$, a bandit algorithm selects an arm a_t. We explore both *classical* (linear or tabular) and *neural* strategies:

EpsilonGreedy (`EpsGreedy`). A simple policy that chooses a random arm with probability ϵ, and otherwise selects the arm $\arg\max_a \widehat{Q}(a, \mathbf{z}_t)$, where $\widehat{Q}$ is an estimated linear or tabular value function. We maintain a weight vector $\mathbf{w}_a$ for each arm and update it via gradient steps.

LinUCB (`LinUCB`). A confidence-bound approach that models the reward as $r_t \approx \mathbf{w}_a^\top \mathbf{z}_t$. For each arm a, it keeps $\mathbf{A}_a = \mathbf{I}$, $\mathbf{b}_a = \mathbf{0}$ and updates:

$$\mathbf{A}_a \leftarrow \mathbf{A}_a + \mathbf{z}_t \mathbf{z}_t^\top, \quad \mathbf{b}_a \leftarrow \mathbf{b}_a + r_t \mathbf{z}_t.$$

The policy selects

$$a_t = \arg\max_a \Big(\hat{\theta}_a^\top \mathbf{z}_t \; + \; \alpha \sqrt{\mathbf{z}_t^\top \mathbf{A}_a^{-1} \mathbf{z}_t} \Big),$$

where $\hat{\theta}_a = \mathbf{A}_a^{-1} \mathbf{b}_a$ and α is an exploration parameter.

ThompsonSampling (`TS`). A Bayesian approach that samples a parameter vector $\tilde{\theta}_a$ from the posterior of each arm a at every step, then selects $a_t = \arg\max_a \tilde{\theta}_a^\top \mathbf{z}_t$. We update the posterior using the linear model assumption $r_t = \theta_a^\top \mathbf{z}_t + \epsilon$.

Neural EpsilonGreedy (`NeuralEG`) and Neural TS (`NeuralTS`). We also test neural versions where the reward predictor is a small neural network $g_a(\cdot)$ for each arm (or a shared network with multiple outputs). For `NeuralEG`, we do ϵ-greedy on $g_a(\mathbf{z}_t)$. For `NeuralTS`, we approximate parameter uncertainty with dropout or added noise in the network weights, sampling from it each step to pick the arm.

3.4. Handling Domain Shifts

Our experiment iterates through domains A, B, C, . . . , H in succession (either abruptly switching or gradually interpolating the distribution). The bandit algorithm does not know *when* or *how* these changes occur but must continually adapt. The learned embedding $E(\cdot)$ from Domain A aims to capture robust features, reducing the complexity of adaptation in later domains.

Offline-to-Online Pipeline. Our pipeline thus consists of:

1. **Self-Supervised Pretraining:** Collect unlabeled samples $\{\mathbf{x}\}$ from Domain A; train $E(\cdot)$ (`VanillaAE`, `MaskedAE`, or `Contrastive`) or use the identity map (`NoRep`).

2. **Online Bandit:** For $t = 1, \ldots, T$, observe $\mathbf{x}_t$ from some domain, compute $\mathbf{z}_t = E(\mathbf{x}_t)$, choose a_t, observe reward r_t, and update the bandit parameters (e.g., $\mathbf{w}_a$, $\mathbf{A}_a$, $\mathbf{b}_a$, or neural network weights).

3. **Domain Transitions:** At certain time steps, the environment distribution changes from domain i to j (abrupt) or gradually interpolates. The agent automatically continues receiving $(\mathbf{x}_t, r_t)$ tuples and must adapt without explicit domain labels.

Implementation Details. We use an embedding dimension $m \leq d$ (often $m = 4$ or $m = 16$). In `MaskedAE`, the mask ratio is around 30%; for `Contrastive`, we apply random noise augmentations. We tune hyperparameters for each bandit (e.g., $\epsilon = 0.1$, $\alpha = 2.0$ for `LinUCB`, small neural networks of size 64 units for `NeuralEG` or `NeuralTS`). We maintain a small buffer of past steps for online updates in neural bandits, performing a few gradient steps per observation.

The next section (§4) describes how these methods are evaluated under multiple domain shifts, comparing both classical and neural bandits with each representation scheme.

4. Experiments

We evaluate our self-supervised domain adaptation on eight domains (A–H), each with distinct context distributions and reward parameters. These domains include abrupt shifts (C → D, G → H) and gradual transitions (D → E, E → F), forcing bandit policies to adapt under evolving conditions. We compare several **bandit strategies** (`EpsGreedy`, `LinUCB`, `TS`, `NeuralEG`, `NeuralTS`, `Random`) with four **representations** (`NoRep`, `VanillaAE`, `MaskedAE`, `Contrastive`).

We run multiple seeds and track: *(i)* Cumulative Reward, *(ii)* Cumulative Regret, *(iii)* Per-step (Incremental) Reward, *(iv)* Time to $\alpha\%$ of Oracle, *(v)* AURC, and *(vi)* Final-Domain Reward. This section focuses on Cumulative Reward/Regret, while the Appendix covers the remaining metrics.

4.1. Experimental Setup

Domains and Contexts. Domains evolve from i.i.d. Gaussians (A) through mean/variance shifts (B, C) to hybrid draws (H), requiring frequent policy updates to handle domain changes.

Self-Supervised Pretraining. We collect unlabeled data from Domain A to train `VanillaAE`, `MaskedAE`, or `Contrastive`; `NoRep` uses raw contexts directly.

Metrics.

Evaluation metrics. We report two standard quantities. (i) **Cumulative reward** $\sum_{t=1}^{T} r_t$: total payoff over the horizon (higher = better). (ii) **Normalized cumulative regret**

$$\frac{1}{T}\sum_{t=1}^{T}\left(r_t^\star - r_t\right),$$

where $r_t^\star$ is the reward of an *oracle* that always pulls the best arm in hindsight. The oracle knows future outcomes and therefore serves only as an optimistic ceiling—useful for visualising headroom but not deployable in practice. We measure Cumulative Reward & Regret (Figures 2 and 3). The Appendix details Time to 80% of Oracle (Figure 4), AURC (Figure 5), and final-domain performance (Figure 6).

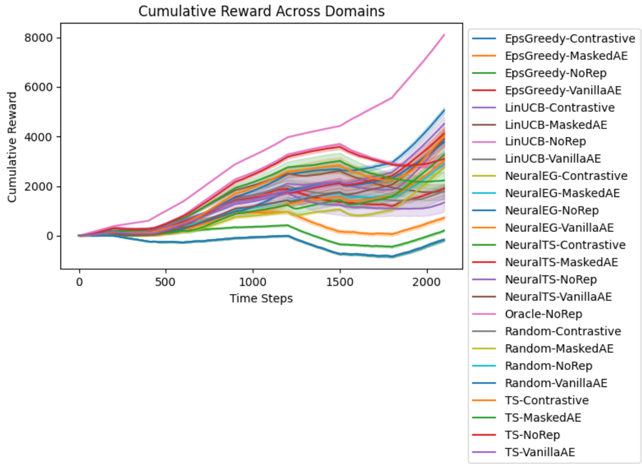

Figure 2. Cumulative Reward across eight domains.

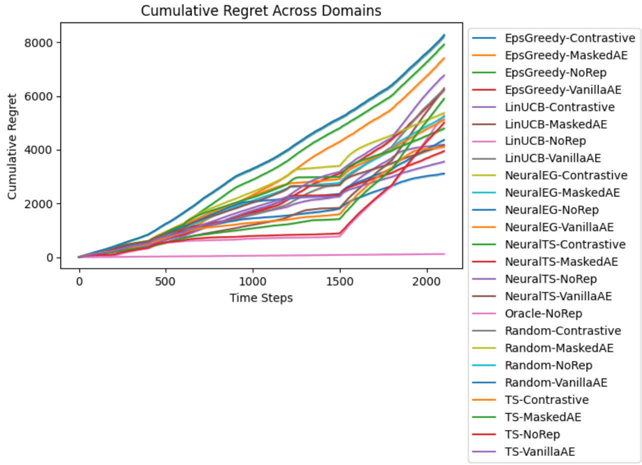

Figure 3. Cumulative Regret across eight domains.

4.2. Results and Discussion

Performance Summary. `Contrastive` and `VanillaAE` outperform `NoRep`, with **EpsGreedy-Contrastive** leading in reward and regret. `NeuralTS` achieves high returns but with higher variance. Additional metrics confirm faster adaptation in these setups.

Key Insights. (1) Representations improve adaptation. (2) `Contrastive` and `VanillaAE` are most effective. (3) Neural methods need careful tuning for early performance.

5. Conclusion

We introduced a *self-supervised domain adaptation* framework for contextual bandits, combining offline representation learning with online adaptation. Pretraining compact embeddings (e.g., `VanillaAE`, `MaskedAE`, `Contrastive`) on unlabeled data improves adaptation to domain shifts, leading to faster learning and higher rewards. Experiments across eight domains show that **representation-based** bandits outperform `NoRep` by reducing regret and boosting final performance. Future work will explore continual learning and domain-aware contrastive augmentations.

References

Bouneffouf, D., Ghisu, E., Cho, S., and Rish, I. Multi-domain lifelong learning with application to nonstationary bandits. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1913–1919, 2020.

Chapelle, O. and Li, L. An empirical evaluation of thompson sampling. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2249–2257, 2011.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 1597–1607, 2020.

Cheung, P. M. and Mannor, S. Nonstationary Bandits: Regret bounds and efficient algorithms. In *International Conference on Machine Learning (ICML)*, pp. 1106–1115, 2019.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pp. 4171–4186, Minneapolis, MN, 2019. Association for Computational Linguistics.

Garivier, A. and Moulines, É. On Upper-Confidence bound policies for nonstationary bandit problems. In *Conference on Algorithmic Learning Theory (ALT)*, pp. 174–188, 2011.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, 2022.

Kober, J., Bagnell, J. A., and Peters, J. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

Kusner, M., Loftus, J., Russell, C., and Silva, R. Counterfactual fairness. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 4066–4076, 2017.

Langford, J. and Zhang, T. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 817–824, 2007.

Li, L., Chu, W., Langford, J., and Schapire, R. E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pp. 661–670, 2010a.

Li, L., Chu, W., Langford, J., and Wang, X. LinUCB: A contextual bandit algorithm for internet advertising. In *Proceedings of the 21st International Conference on World Wide Web (WWW)*, pp. 447–456, 2010b.

Lu, X., Yang, J., and Li, L. Weighted regression for nonstationary contextual bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 14539–14550, 2021.

Mitton, L., Reinert, S., and Tipnis, K. Self-supervised context embeddings in multi-domain bandits. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 138–148, 2022.

Pascual, S. and Agmon, N. Unsupervised representation learning for contextual bandits via autoencoder pretraining. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2736–2744, 2020.

Riquelme, C., Tucker, G., and Snoek, J. Deep Bayesian Bandits Showdown: A comparison of Bayesian Deep Networks for Thompson Sampling. In *International Conference on Learning Representations (ICLR)*, 2018.

Shortreed, S. M., Laber, E., Lizotte, D. J., Strém, A. K., Pineau, J., and Murphy, S. A. Informing sequential clinical decision-making through reinforcement learning: An empirical study. *Machine Learning*, 84(1-2):109–136, 2011.

Zhang, J., Cammarata, L., Squires, C., Sapsis, T. P., and Uhler, C. Active learning for optimal intervention design in causal models. *Nature Machine Intelligence*, 5(10): 1066–1075, 2023a.

Zhang, J., Greenewald, K., Squires, C., Srivastava, A., Shanmugam, K., and Uhler, C. Identifiability guarantees for causal disentanglement from soft interventions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 50254–50292, 2023b.

Zhang, Y., Szepesvári, C., and Li, L. Neural contextual bandits revisited. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 2541–2551, 2022.

Table 1. Summary of key metrics across evaluated bandit algorithms and representation methods. **time_to_80pct_oracle**: the average number of steps required for each algorithm to reach 80% of the cumulative reward achieved by the Oracle (lower values indicate faster learning). **AURC (Area Under Regret Curve)**: the cumulative sum of regrets across all steps, reflecting how closely the algorithm performs compared to the ideal choice at every step (lower is better). **Final-Domain Incremental Reward**: average reward per step obtained specifically in the last evaluated domain (”H”), highlighting how effectively each algorithm adapts to complex distributional shifts (higher is better). *All values are averaged over multiple random seeds to ensure robustness; values shown are truncated for brevity.*

Method	**time_to_80pct**	**AURC**	**Final-Domain Reward**
EpsGreedy-Contrastive	1.75	3109.70	7.03
EpsGreedy-MaskedAE	8.33	7402.35	2.23
EpsGreedy-NoRep	28.20	7915.18	2.20
EpsGreedy-VanillaAE	0.33	6231.32	2.35
LinUCB-Contrastive	0.00	4187.24	6.79
LinUCB-MaskedAE	388.00	6285.64	-0.56
LinUCB-NoRep	257.28	5272.62	-0.37
LinUCB-VanillaAE	0.00	6259.54	1.23
NeuralEG-Contrastive	0.25	5366.07	5.77
NeuralEG-MaskedAE	1.00	5225.75	4.47
NeuralEG-NoRep	0.66	4367.52	4.91
NeuralEG-VanillaAE	-1.00	5116.90	4.74
NeuralTS-Contrastive	0.00	4788.22	5.63
NeuralTS-MaskedAE	0.33	3951.62	5.67
NeuralTS-NoRep	3.00	3557.09	6.50
NeuralTS-VanillaAE	0.50	4169.20	6.12
Oracle-NoRep	9.00	119.14	8.46
Random-Contrastive	0.00	8258.82	2.23
Random-MaskedAE	0.00	8258.82	2.23
Random-NoRep	0.00	8258.82	2.23
Random-VanillaAE	0.00	8258.82	2.23
TS-Contrastive	0.00	4110.29	5.93
TS-MaskedAE	298.75	5896.91	-0.26
TS-NoRep	704.60	5005.87	0.74
TS-VanillaAE	-1.00	6764.66	0.76

A. Additional Figures and Tables

In this appendix, we provide the remaining plots (outlined in §4) for completeness:

As discussed in the main text, these metrics reinforce the conclusion that integrating self-supervised representations (especially `Contrastive` or `VanillaAE`) provides (i) faster adaptation to shifting distributions, (ii) reduced total regret, and (iii) strong final-domain performance close to that of the Oracle.

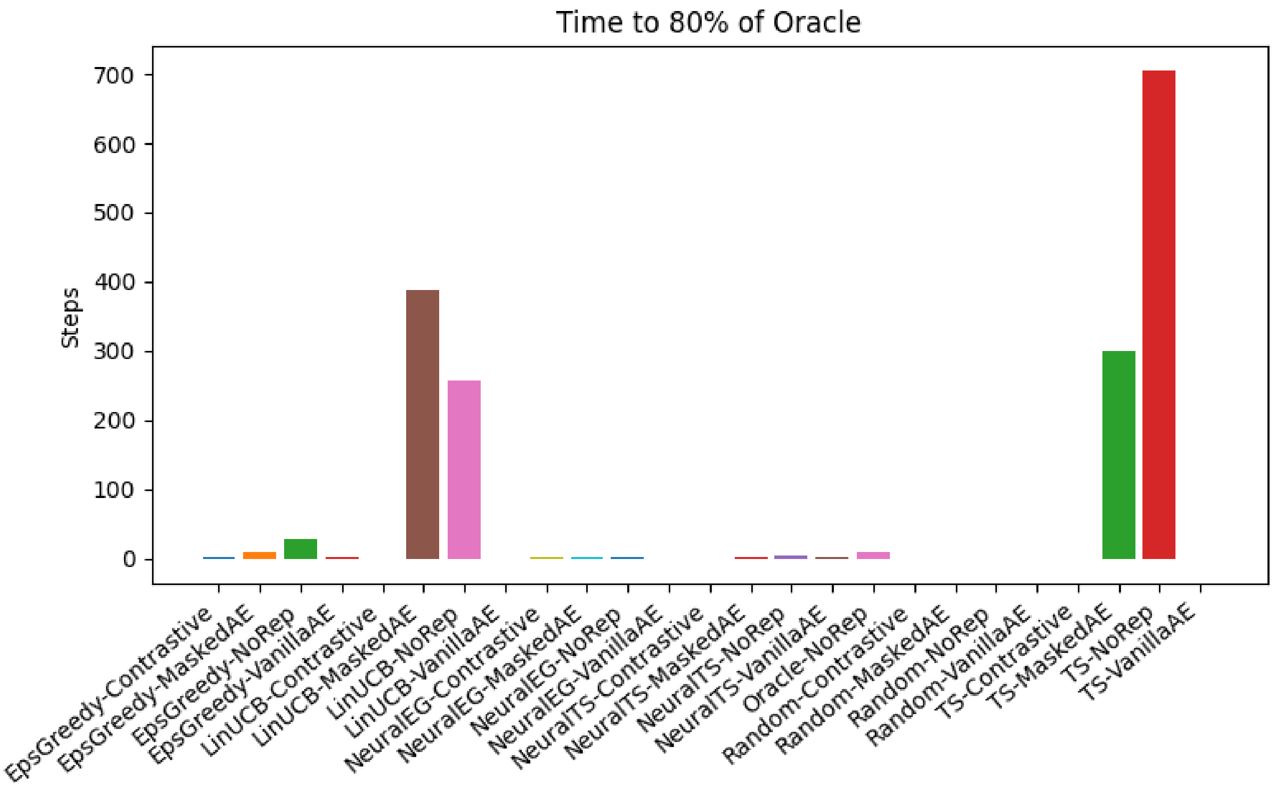

Figure 4. Time (in steps) to reach 80% of Oracle's performance

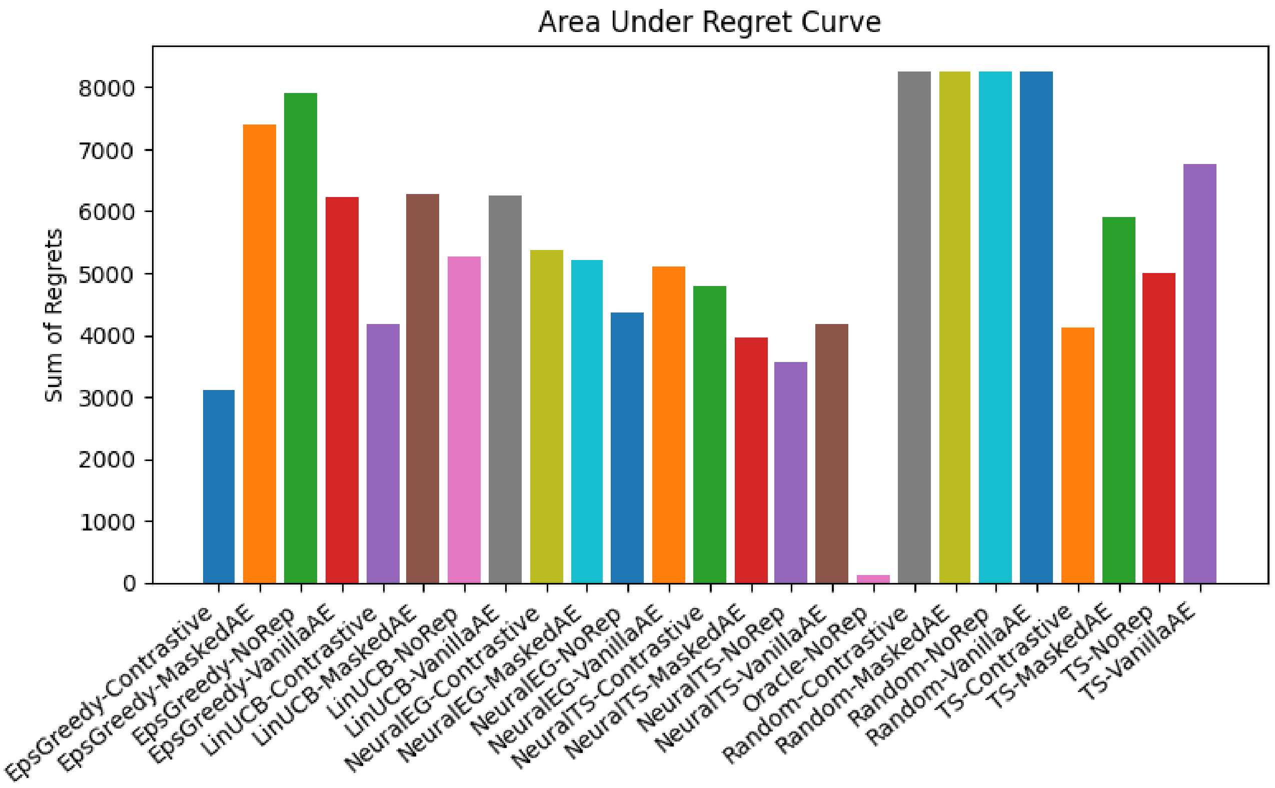

Figure 5. Area Under Regret Curve (AURC).

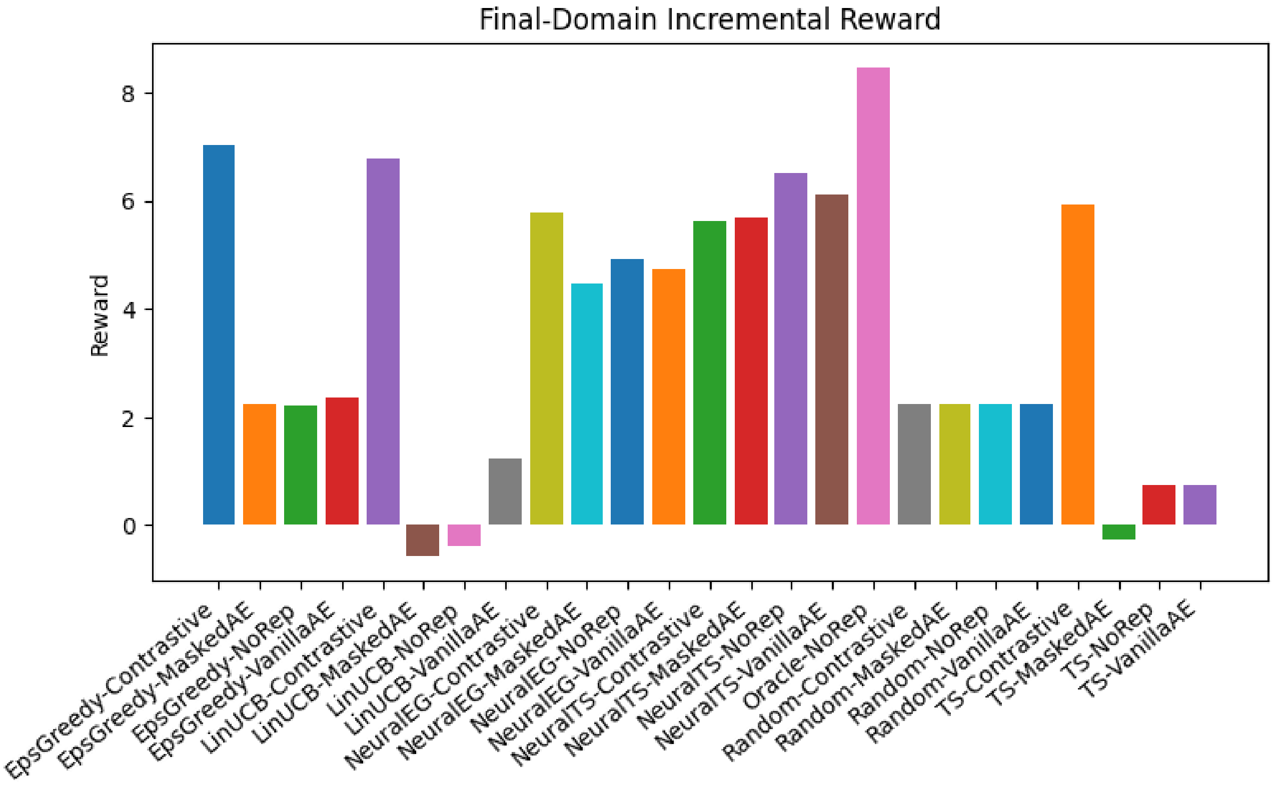

Figure 6. **Final-Domain Incremental Reward**. Higher bars indicate better long-horizon performance in Domain H.