Can a MISL Fly? Analysis and Ingredients for Mutual Information Skill Learning

Chongyi Zheng[*] **Jens Tuyls**[*] **Joanne Peng** **Benjamin Eysenbach**
Princeton University
{chongyiz, jtuyls}@princeton.edu

Abstract

Self-supervised learning has the potential of lifting several of the key challenges in reinforcement learning today, such as exploration, representation learning, and reward design. Recent work (METRA [52]) has effectively argued that moving away from mutual information and instead optimizing a certain Wasserstein distance is important for good performance. In this paper, we argue that the benefits seen in that paper can largely be explained within the existing framework of mutual information skill learning (MISL). Our analysis suggests a new MISL method (contrastive successor features) that retains the excellent performance of METRA with fewer moving parts, and highlights connections between skill learning, contrastive representation learning, and successor features. Finally, through careful ablation studies, we provide further insight into some of the key ingredients for both our method and METRA.[2]

1 Introduction

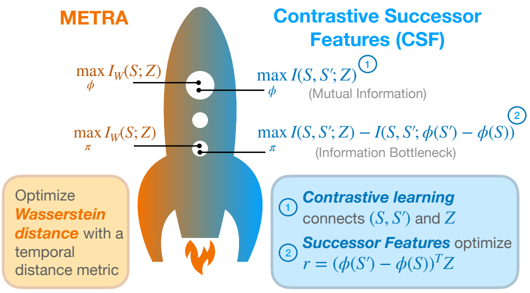

METRA
Contrastive Successor Features (CSF)
$\max_{\phi} I_W(S;Z)$
$\max_{\pi} I_W(S;Z)$
$\max_{\phi} I(S,S';Z)$ 1
(Mutual Information)
$\max_{\pi} I(S,S';Z) - I(S,S';\phi(S') - \phi(S))$ 2
(Information Bottleneck)
Optimize Wasserstein distance with a temporal distance metric
1 Contrastive learning connects (S,S') and Z
2 Successor Features optimize $r = (\phi(S') - \phi(S))^T Z$

Figure 1: **From METRA to MISL. (Left)** METRA argues optimizing a Wasserstein distance is superior to using mutual information. **(Right)** Through careful analysis, we show METRA still bears striking similarities to MISL algorithms, which allows us to develop a new MISL algorithm (CSF) that matches the performance of METRA while retaining the theoretical properties associated with MI maximization.

Self-supervised learning has had a large impact on areas of machine learning ranging from audio processing [46, 47] or computer vision [56, 12] to natural language processing [16, 57, 58, 9]. In the reinforcement learning (RL) domain, the "right" recipe to apply self-supervised learning is not yet clear. Several self-supervised methods for RL directly apply off-the-shelf methods from other domains such as masked autoencoding [38], but have achieved limited success so far. Other methods design self-supervised routines more specifically built for the RL setting [10, 53, 18, 63, 54]. We will focus on the skill learning methods, which aim to learn a set of diverse and distinguishable behaviors (skills) without an external reward function. This objective is typically formulated as maximizing the mutual information between skills and states [24, 18], namely *mutual information skill learning* (MISL). However, some promising recent advances in skill learning methods build on other intuitions such as Lipschitz constraints [49] or transition distances [50]. This paper focuses on determining whether the good performance of those recent methods can still be explained within the well-studied framework of mutual information maximization.

[*] These authors contributed equally.
[2] Website and code: https://princeton-rl.github.io/contrastive-successor-features

Accepted at the Intrinsically Motivated Open-ended Learning workshop at NeurIPS 2024.

METRA [52], one of the strongest prior skill learning methods, proposes maximizing the Wasserstein dependency measure between states and skills as an alternative to the idea of mutual information maximization. The success of this method calls into question the viability of the MISL framework. However, mutual information has a long history dating back to Shannon [62] and gracefully handles stochasticity and continuous states [44]. These appealing properties of mutual information raises the question: *Can we build effective skill learning algorithms within the MISL framework, or is MISL fundamentally flawed?*

We start by carefully studying the components of METRA both theoretically and empirically. For representation learning, METRA maximizes a lower bound on the mutual information, resembling contrastive learning. For policy learning, METRA optimizes a mutual information term *plus an extra exploration term*. These findings provide an interpretation of METRA that does not appeal to Wassertein distances and motivate a simpler algorithm (Fig. 1).

Building upon our new interpretations of METRA, we propose a simpler and competitive MISL algorithm called Contrastive Successor Features (CSF). First, CSF learns state representations by directly optimizing a contrastive lower bound on mutual information, preventing the dual gradient descent procedure adopted by METRA. Second, while any off-the-shelf RL algorithm (e.g. SAC [27]) is applicable, CSF instead learns a policy by leveraging successor features of linear rewards defined by the learned representations. Experiments on six continuous control tasks show that CSF is comparable with METRA, as evaluated on exploration performance and on downstream tasks. Furthermore, ablation studies suggest that rewards derived from the information bottleneck as well as a specific parametrization of representations are key for good performance.

2 Understanding the Prior Method

In this section we reinterpret METRA through the lens of MISL, showing that *(1)* the METRA representation objective is nearly identical to a contrastive loss (which maximizes a lower bound on mutual information, see Sec. 2.1), and *(2)* the METRA actor objective is equivalent to a mutual information lower bound *plus an extra term*. This extra term is related to an information bottleneck [68, 3] and our experiments will show it is important for exploration. See Sec. 2.2. The sections below will provide a brief overview of the theoretical results, while a full and more formal exposition is left to Appendix A.

2.1 Connecting METRA's Representation Objective and Contrastive Learning

Instead of enforcing a temporal distance constraint on each transition pair $(s, s') \in \mathcal{S}_{\text{adj}}^{\beta}$ (see Appendix C), where $\mathcal{S}_{\text{adj}}^{\beta}$ denotes the set of all the adjacent state pairs visited by policy β, the METRA representation objective actually imposes the temporal distance constraint over all transition pairs (s, s') in expectation:

$$\max_{\phi} \mathbb{E}_{p(z)p^{\beta}(s,s'|z)}[(\phi(s') - \phi(s))^{\top} z] \quad \text{s.t. } \mathbb{E}_{p^{\beta}(s,s')}\left[\|\phi(s') - \phi(s)\|_2^2\right] \leq 1.$$

Identifying the actual METRA representation objective allows us to draw a connection with the rank-based contrastive loss (InfoNCE [47, 41]). Specifically, METRA learns state representations ϕ via (approximately) maximizing a contrastive lower bound of mutual information between transition pairs and skills under the behavior policy $I^{\beta}(S, S'; Z)$:

$$I^{\beta}(S, S'; Z) \geq \mathbb{E}_{p^{\beta}(s,s',z)}[f(s, s', z)] - \mathbb{E}_{p^{\beta}(s,s')}\left[\log \mathbb{E}_{p(z)}\left[e^{f(s,s',z)}\right]\right],$$

where $f : \mathcal{S} \times \mathcal{S} \times \mathcal{Z} \mapsto \mathbb{R}$ is the critic function [41, 55, 20, 76]. Furthermore, our empirical studies (Sec. 4) confirm that the representations learned by METRA indeed bear a resemblance to those learned by a contrastive loss.

2.2 Connecting METRA's Actor Objective with an Information Bottleneck

METRA uses different objectives for the representations and the actor. Specifically, the METRA actor objective optimizes the intrinsic reward $(\phi(s') - \phi(s))^{\top} z$ (which misses the λ term from Eq. 10) derived from learned representations, which we show is a lower bound of the information bottleneck $I^{\pi}(S, S'; Z) - I^{\pi}(S, S'; \phi(S') - \phi(S))$ (see Prop. 3). This result implies that simply maximizing the mutual information $I^{\pi}(S, S'; Z)$ may be insufficient for deriving a diverse skill-conditioned policy π.

3 A Simplified Algorithm for MISL via Contrastive Learning

In this section, we derive a simpler unsupervised skill learning method building upon our understanding of METRA (Sec. 2). This method maximizes MI (unlike METRA), while retaining the good performance of METRA (see discussion in Appendix C). We will first use the contrastive lower bound to optimize the state representation ϕ and estimate intrinsic rewards, and then we will learn the policy π using successor features. We name our method **contrastive successor features (CSF)**.

3.1 Learning Representations through Contrastive Learning

Based on our analysis in Sec. 2.1, we use the contrastive lower bound on $I^\beta(S, S'; Z)$ to optimize the state representation directly. Unlike METRA, we obtain this contrastive lower bound *within* the MISL framework (Eq. 7 & 8) by employing a parameterization of the variational distribution $q(z \mid s, s')$ mentioned in prior work [55, 65]. Specifically, using a scaled energy-based model conditioned representations of transition pairs (s, s'), we define the variational distribution as

$$q(z \mid s, s') \triangleq \frac{p(z) e^{(\phi(s') - \phi(s))^\top z}}{\mathbb{E}_{p(z')}[e^{(\phi(s') - \phi(s))^\top z'}]}. \tag{1}$$

Plugging this parameterization into Eq. 7 produces

$$\phi_{k+1} \leftarrow \arg\max_{\phi} \mathbb{E}_{p^\beta(s,s',z)} \left[(\phi(s') - \phi(s))^\top z\right] - \mathbb{E}_{p^\beta(s,s')} \left[\log \mathbb{E}_{p(z')} \left[e^{(\phi(s') - \phi(s))^\top z'}\right]\right], \tag{2}$$

which is exactly the contrastive lower bound on $I^\beta(S, S'; Z)$. This contrastive lower bound allows us to learn the state representation ϕ while getting rid of the dual gradient descent procedure (Eq. 10) adopted by METRA. In practice, we find that adding a fixed coefficient $\xi = 5$ to the second term of Eq. 2 helps boost performance.

In the same way that the METRA actor objective excluded the anti-exploration term (Sec. 2.2), we propose to construct the intrinsic reward by removing the negative term from our representation objective (Eq. 2), resulting in the same RL objective as $J(\pi)$ (Eq. 11):

$$\pi_{k+1} \leftarrow \arg\max_{\pi} \mathbb{E}_{p^\pi(s,s',z)} \left[r_k(s, s', z)\right], r_k(s, s', z) \triangleq (\phi_k(s') - \phi_k(s))^\top z \tag{3}$$

We use this RL objective as the update rule for the skill-conditioned policy π in our algorithm.

3.2 Learning a Policy with Successor Features

To optimize the policy (Eq. 3), we will use an actor-critic method. Most skill learning methods use an off-the-shelf RL algorithm (e.g., TD3 [22], SAC [27]) to fit the critic. However, by noting that the intrinsic reward function $r(s, s', z)$ [3] is a linear combination between basis $\phi(s') - \phi(s) \in \mathbb{R}^d$ and weights $z \in \mathcal{Z} \subset \mathbb{R}^d$, we can borrow ideas from successor representations to learn a vector-valued critic. We learn the successor features $\psi^\pi : \mathcal{S} \times \mathcal{A} \times \mathcal{Z} \mapsto \mathbb{R}^d$:

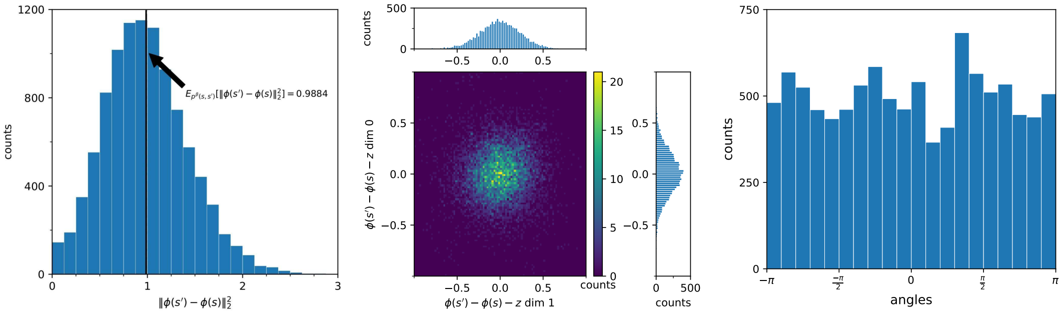
$$\psi^\pi(s, a, z) \triangleq \mathbb{E}_{s \sim p^\pi(s_+ = s \mid z), s' \sim p(s' \mid s, a)} \left[\phi(s') - \phi(s)\right],$$

with the corresponding skill-conditioned policy π in an actor-critic style:

$$\begin{gathered}
\psi_{k+1}(s, a, z) \leftarrow \arg\min_{\psi} \mathbb{E}_{(s,a,z) \sim p^\beta(s,a,s',z), a' \sim \pi(a' \mid s', z)} \left[\left(\psi(s, a, z) - \hat{\psi}_k(s, s', a', z)\right)^2\right], \\
\text{where} \quad \hat{\psi}_k(s, s', a', z) \triangleq \phi_k(s') - \phi_k(s) + \gamma \bar{\psi}_k(s', a', z), \\
\pi_{k+1} \leftarrow \arg\max_{\pi} \mathbb{E}_{(s,z) \sim p^\beta(s,z), a \sim \pi(a \mid s, z)} \left[\psi_k(s, a, z)^\top z\right],
\end{gathered}$$

where ψ is an estimation of ψ^π. In practice, we optimize ψ and π for one gradient step iteratively. In Appendix D, we summarize our algorithm and show the pseudo-code of CSF (Alg. 1).

[3] We ignore the iteration k for notation simplicity.



(a) Expected constraint (b) Gaussianity (c) Uniformity

Figure 2: **Histograms of METRA representations.** *(a)* The expected distance of representations converges to 1.0, helping to explain what objective METRA's representations are optimizing. *(b)* Given a latent skill, the conditional difference in representations $(\phi(s') - \phi(s) \mid z)$ converges to an isotropic Gaussian distribution. *(c)* Taking the marginal over latent skills, the normalized difference in representations $\left(\frac{(\phi(s')-\phi(s))}{\|\phi(s')-\phi(s)\|_2}\right)$ converges to a $\textsc{Unif}(\mathbb{S}^{d-1})$. These observations are consistent with our theoretical analysis (Cor. 1) suggesting that METRA is performing a form of contrastive learning.

4 Experiments

The aims of our experiments are *(1)* verifying the theoretical analysis in Sec. 2 experimentally, and *(2)* comparing our simplified algorithm CSF to prior work. Our experiments will use standard benchmarks introduced by prior work on skill learning. All experiments show means and standard deviations across ten random seeds. In Appendix F, we also include experiments identifying several ingredients that are key to making MISL algorithms work well more broadly.

4.1 METRA Constrains Representations in Expectation

Prop. 1 predicts that the optimal METRA representation satisfies its constraint $\mathbb{E}_p^\beta(s, s')\left[\|\phi(s') - \phi(s)\|_2^2\right] = 1$ strictly. We study whether this condition holds after training the algorithm for a long time. To answer this question, we conduct didactic experiments with the state-based `Ant` from METRA [52] navigating in an open space. We set the dimension of ϕ to $d = 2$ such that visualizing the learned representations becomes easier. After training the METRA algorithm for 20M environment steps (50K gradient steps), we analyze the norm of the difference in representations $\|\phi(s') - \phi(s)\|_2^2$.

We plot the histogram of $\|\phi(s') - \phi(s)\|_2^2$ over 10K transitions randomly sampled from the replay buffer (Fig. 2a). The observation that the empirical average of $\|\phi(s') - \phi(s)\|_2^2$ converges to 0.9884 suggests that the learned representations are feasible. Stochastic gradient descent methods typically find globally optimal solutions on over-parameterized neural networks [17], making us conjecture that the learned representations are nearly optimal (Prop. 1). Furthermore, the spreading of the value of $\|\phi(s') - \phi(s)\|_2^2$ implies that maximizing the METRA representation objective will *not* learn state representations ϕ that satisfy $\|\phi(s') - \phi(s)\|_2^2 \leq 1$ for every $(s, s') \in \mathcal{S}_{\text{adj}}^\beta$. These results help to explain what objective METRA's representations are optimizing.

4.2 METRA Learns Contrastive Representations

We next study connections between representations learned by METRA and those learned by contrastive learning empirically. Our analysis in Sec. 2.1 reveals that the representation objective of METRA corresponds to the contrastive lower bound on $I^\beta(S, S'; Z)$. This analysis raises the question of whether representations learned by METRA share similar structures to representations learned by contrastive losses [25, 41, 71].

To answer this question, we reuse the trained algorithm in Sec. 4.1 and visualize two important statistics: *(1)* the conditional differences in representations $\phi(s') - \phi(s) - z$ and *(2)* the normalized marginal differences in representations $(\phi(s') - \phi(s))/\|\phi(s') - \phi(s)\|_2$. The resulting histograms (Fig. 2b & 2c) indicate that the conditional differences in representations $\phi(s') - \phi(s) - z$ converge to an isotropic Gaussian in distribution while the normalized marginal differences in representations $(\phi(s') - \phi(s))/\|\phi(s') - \phi(s)\|_2$ converge to a uniform distribution on the d-dimensional unit hypersphere $\mathbb{S}^{d-1}$ in distribution. Prior work [71] has shown that representations derived from contrastive learning preserve properties similar to these observations. We conjecture that maximizing

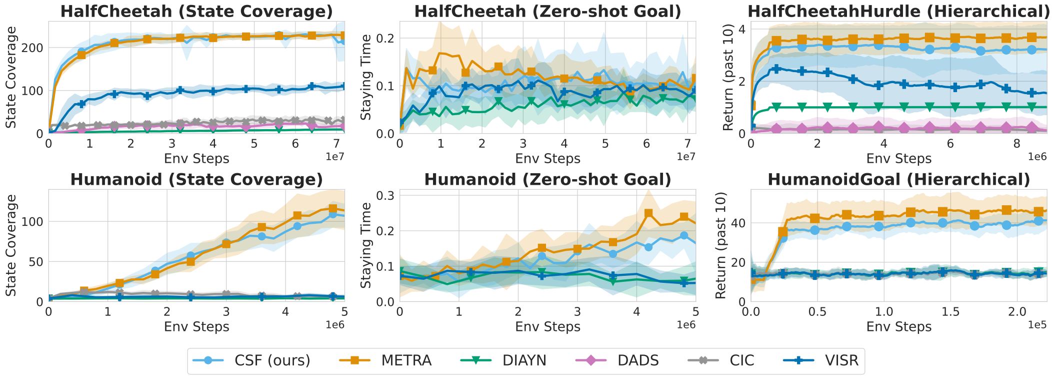


Figure 3: **CSF performs on par with METRA.** We compare CSF with baselines on state coverage *(left)*, zero-shot goal reaching *(middle)*, and hierarchical control *(right)*. We generally find CSF to perform roughly on par with METRA and outperform all other baselines in most settings. Shaded areas indicate one standard deviation. Appendix Fig. 5, 6& 7 show the learning curves for all tasks.

the contrastive lower bound on $I^\beta(S, S'; Z)$ directly has the same effect as maximizing the METRA representation objective. See Appendix G.6 for formal claims and connections.

4.3 CSF Matches SOTA for both Exploration and Downstream performance

Our final set of experiments compare CSF to prior MISL algorithms, measuring performance on both unsupervised exploration and solving downstream tasks.

Experimental Setup. We evaluate on the same five tasks as those used in Park et al. [52] plus `Robobin` from LEXA [42], though we will only focus on `HalfCheetah` and `Humanoid` in the main text. For baselines, we also use a subset from Park et al. [52] (METRA [52], CIC [34], DIAYN [18], and DADS [63]) along with VISR [28]. See Appendix H.2 for details.

Exploration performance. To measure the inherent exploration capabilities of each method without considering any particular downstream task, we compute the state coverage by counting the unique number of (x, y) coordinates visited by the agent. Fig. 3 *(left)* shows CSF matches METRA on both `HalfCheetah` and `Humanoid`. For the full set of exploration results, please see Appendix H.3.

Zero-shot goal reaching. In this setting the agent infers the right skill given a goal without further training on the environment. We evaluate on the same set of six tasks and defer both the goal sampling and skill inference strategies to Appendix H.4. We report the *staying time fraction*, which is the number of time steps that the agent stays at the goal divided by the horizon length. In Fig. 3 *(middle)*, we find all methods to perform similarly on `HalfCheetah`, while METRA and CSF perform best on `Humanoid`, with METRA performing slightly better on the latter. For the full set of zero-shot goal reaching results, please see Appendix H.4.

Hierarchical control. We train a hierarchical controller $\pi_h(z \mid s)$ that outputs latent skills z as actions for every fixed number of time steps to maximize the discounted return in two downstream tasks from Park et al. [52], one of which requires to reach a specified goal (`HumanoidGoal`) and one requires jumping over hurdles (`HalfCheetahHurdle`). The results in Fig. 3 *(right)* show CSF and METRA are the best performing methods, showing mostly similar performance. For further details as well as the full set of results on all tasks, please see Appendix H.5.

5 Conclusion

In this paper, we show how one of the current strongest unsupervised skill discovery algorithms can be understood through the lens of mutual information skill learning. Our analysis allowed the development of our new method CSF, which we showed to perform on par with METRA in most settings. More broadly, our work provides evidence that mutual information maximization can still be effective to build high performing skill discovery algorithms.

Limitations. It is unclear how far CSF can *scale* to increasingly complex environments with potentially an increased number of interactive objects, partial observability, environment stochasticity, and discrete action spaces. We leave investigating these empirical scaling limits to future work.

Acknowledgements

We thank the National Science Foundation (Grant No. 2239363) for providing funding for this work. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. We thank Seohong Park for providing code for several of the baselines in the paper. In addition, we thank Qinghua Liu for providing feedback on early drafts of this paper. We also thank Princeton Research Computing. Finally, we thank the anonymous reviewers for their valuable feedback and time.

References

[1] Abramowitz, M. and Stegun, I. A. (1968). *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, volume 55. US Government printing office.

[2] Ahmad, I. and Lin, P.-E. (1976). A nonparametric estimation of the entropy for absolutely continuous distributions (corresp.). *IEEE Transactions on Information Theory*, 22(3):372–375.

[3] Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. (2017). Deep variational information bottleneck. In *International Conference on Learning Representations*.

[4] Barber, D. and Agakov, F. (2004). The im algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201.

[5] Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H. P., and Silver, D. (2017). Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30.

[6] Baumli, K., Warde-Farley, D., Hansen, S., and Mnih, V. (2021). Relative variational intrinsic control. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6732–6740.

[7] Borsa, D., Barreto, A., Quan, J., Mankowitz, D. J., van Hasselt, H., Munos, R., Silver, D., and Schaul, T. (2019). Universal successor features approximators. In *International Conference on Learning Representations*.

[8] Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. (2016). Openai gym. *arXiv preprint arXiv:1606.01540*.

[9] Brown, T. B. (2020). Language models are few-shot learners. *arXiv preprint ArXiv:2005.14165*.

[10] Burda, Y., Edwards, H., Storkey, A., and Klimov, O. (2019). Exploration by random network distillation. In *Seventh International Conference on Learning Representations*, pages 1–17.

[11] Campos, V., Trott, A., Xiong, C., Socher, R., Giró-i Nieto, X., and Torres, J. (2020). Explore, discover and learn: Unsupervised discovery of state-covering skills. In *International Conference on Machine Learning*, pages 1317–1327. PMLR.

[12] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

[13] Choi, J., Sharma, A., Lee, H., Levine, S., and Gu, S. S. (2021). Variational empowerment as representation learning for goal-based reinforcement learning. *arXiv preprint arXiv:2106.01404*.

[14] Chopra, S., Hadsell, R., and LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546 vol. 1.

[15] Dayan, P. (1993). Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624.

[16] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*.

[17] Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. (2019). Gradient descent finds global minima of deep neural networks. In *International conference on machine learning*, pages 1675–1685. PMLR.

[18] Eysenbach, B., Gupta, A., Ibarz, J., and Levine, S. (2019). Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*.

[19] Eysenbach, B., Salakhutdinov, R., and Levine, S. (2020). C-learning: Learning to achieve goals via recursive classification. *arXiv preprint arXiv:2011.08909*.

[20] Eysenbach, B., Zhang, T., Levine, S., and Salakhutdinov, R. R. (2022). Contrastive learning as goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 35:35603–35620.

[21] Florensa, C., Duan, Y., and Abbeel, P. (2016). Stochastic neural networks for hierarchical reinforcement learning. In *International Conference on Learning Representations*.

[22] Fujimoto, S., Hoof, H., and Meger, D. (2018). Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR.

[23] Gao, T., Yao, X., and Chen, D. (2021). Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

[24] Gregor, K., Rezende, D. J., and Wierstra, D. (2016). Variational intrinsic control. *arXiv preprint arXiv:1611.07507*.

[25] Gutmann, M. and Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings.

[26] Gutmann, M. and Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Teh, Y. W. and Titterington, M., editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 297–304, Chia Laguna Resort, Sardinia, Italy. PMLR.

[27] Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR.

[28] Hansen, S., Dabney, W., Barreto, A., Warde-Farley, D., de Wiele, T. V., and Mnih, V. (2020). Fast task inference with variational intrinsic successor features. In *International Conference on Learning Representations*.

[29] He, S., Jiang, Y., Zhang, H., Shao, J., and Ji, X. (2022). Wasserstein unsupervised reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6884–6892.

[30] Henaff, O. (2020). Data-efficient image recognition with contrastive predictive coding. In *International conference on machine learning*, pages 4182–4192. PMLR.

[31] Ho, J. and Ermon, S. (2016). Generative adversarial imitation learning. *Advances in neural information processing systems*, 29.

[32] Inc., W. R. (2024). Mathematica, Version 14.0. Champaign, IL, 2024.

[33] Kulkarni, T. D., Saeedi, A., Gautam, S., and Gershman, S. J. (2016). Deep successor reinforcement learning. *arXiv preprint arXiv:1606.02396*.

[34] Laskin, M., Liu, H., Peng, X. B., Yarats, D., Rajeswaran, A., and Abbeel, P. (2022). Cic: Contrastive intrinsic control for unsupervised skill discovery. *arXiv preprint arXiv:2202.00161*.

[35] Laskin, M., Srinivas, A., and Abbeel, P. (2020). Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pages 5639–5650. PMLR.

[36] Lee, L., Eysenbach, B., Parisotto, E., Xing, E., Levine, S., and Salakhutdinov, R. (2019). Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*.

[37] Li, M., Zhao, X., Lee, J. H., Weber, C., and Wermter, S. (2023). Internally rewarded reinforcement learning. In *International Conference on Machine Learning*, pages 20556–20574. PMLR.

[38] Liu, F., Liu, H., Grover, A., and Abbeel, P. (2022). Masked autoencoding for scalable and generalizable decision making. *Advances in Neural Information Processing Systems*, 35:12608–12618.

[39] Liu, H. and Abbeel, P. (2021). Aps: Active pretraining with successor features. In *International Conference on Machine Learning*, pages 6736–6747. PMLR.

[40] Ma, Y. J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., and Zhang, A. (2023). Vip: Towards universal visual reward and representation via value-implicit pre-training. In *The Eleventh International Conference on Learning Representations*.

[41] Ma, Z. and Collins, M. (2018). Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3698–3707.

[42] Mendonca, R., Rybkin, O., Daniilidis, K., Hafner, D., and Pathak, D. (2021). Discovering and achieving goals via world models. *Advances in Neural Information Processing Systems*, 34:24379–24391.

[43] Mohamed, S. and Jimenez Rezende, D. (2015). Variational information maximisation for intrinsically motivated reinforcement learning. *Advances in neural information processing systems*, 28.

[44] Myers, V., Zheng, C., Dragan, A., Levine, S., and Eysenbach, B. (2024). Learning temporal distances: Contrastive successor features can provide a metric structure for decision-making. *arXiv preprint arXiv:2406.17098*.

[45] Nachum, O., Chow, Y., Dai, B., and Li, L. (2019). Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. *Advances in neural information processing systems*, 32.

[46] Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W., and Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. In *Speech Synthesis Workshop*.

[47] Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

[48] Ozair, S., Lynch, C., Bengio, Y., van den Oord, A., Levine, S., and Sermanet, P. (2019). Wasserstein dependency measure for representation learning. *Advances in Neural Information Processing Systems*, 32.

[49] Park, S., Choi, J., Kim, J., Lee, H., and Kim, G. (2022). Lipschitz-constrained unsupervised skill discovery. In *International Conference on Learning Representations*.

[50] Park, S., Lee, K., Lee, Y., and Abbeel, P. (2023). Controllability-aware unsupervised skill discovery. In *International Conference on Machine Learning*, pages 27225–27245. PMLR.

[51] Park, S. and Levine, S. (2023). Predictable mdp abstraction for unsupervised model-based rl. In *International Conference on Machine Learning*, pages 27246–27268. PMLR.

[52] Park, S., Rybkin, O., and Levine, S. (2024). METRA: Scalable unsupervised RL with metric-aware abstraction. In *The Twelfth International Conference on Learning Representations*.

[53] Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. (2017). Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR.

[54] Pong, V. H., Dalal, M., Lin, S., Nair, A., Bahl, S., and Levine, S. (2020). Skew-fit: state-covering self-supervised reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, pages 7783–7792.

[55] Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., and Tucker, G. (2019). On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR.

[56] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

[57] Radford, A. and Narasimhan, K. (2018). Improving language understanding by generative pre-training. unpublished.

[58] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. unpublished.

[59] Schaul, T., Horgan, D., Gregor, K., and Silver, D. (2015). Universal value function approximators. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1312–1320, Lille, France. PMLR.

[60] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

[61] Shafiullah, N. M. M. and Pinto, L. (2022). One after another: Learning incremental skills for a changing world. In *International Conference on Learning Representations*.

[62] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423.

[63] Sharma, A., Gu, S., Levine, S., Kumar, V., and Hausman, K. (2020). Dynamics-aware unsupervised discovery of skills. In *International Conference on Learning Representations*.

[64] Sohn, K. (2016). Improved deep metric learning with multi-class n-pair loss objective. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

[65] Song, Y. and Kingma, D. P. (2021). How to train your energy-based models. *arXiv preprint arXiv:2101.03288*.

[66] Strouse, D., Baumli, K., Warde-Farley, D., Mnih, V., and Hansen, S. S. (2022). Learning more skills through optimistic exploration. In *International Conference on Learning Representations*.

[67] Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D. d. L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., et al. (2018). Deepmind control suite. *arXiv preprint arXiv:1801.00690*.

[68] Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. *arXiv preprint physics/0004057*.

[69] Todorov, E., Erez, T., and Tassa, Y. (2012). Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033.

[70] Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. (2020). On mutual information maximization for representation learning. In *International Conference on Learning Representations*.

[71] Wang, T. and Isola, P. (2020). Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

[72] Warde-Farley, D., Van de Wiele, T., Kulkarni, T., Ionescu, C., Hansen, S., and Mnih, V. (2019). Unsupervised control through non-parametric discriminative rewards. In *International Conference on Learning Representations*.

[73] Wikipedia (2024). Von Mises–Fisher distribution — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Von%20Mises%E2%80%93Fisher%20distribution&oldid=1230057889`. [Online; accessed 10-July-2024].

[74] Yarats, D., Kostrikov, I., and Fergus, R. (2021). Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International Conference on Learning Representations*.

[75] Zhang, J., Springenberg, J. T., Boedecker, J., and Burgard, W. (2017). Deep reinforcement learning with successor features for navigation across similar environments. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2371–2378. IEEE.

[76] Zheng, C., Salakhutdinov, R., and Eysenbach, B. (2024). Contrastive difference predictive coding. In *The Twelfth International Conference on Learning Representations*.

A A Complete Understanding of The Prior Method

A.1 Connecting METRA's Representation Objective and Contrastive Learning

Our understanding of METRA starts by interpreting the representation objective of METRA as a contrastive loss. This interpretation proceeds by two steps. First, we focus on understanding the *actual* representation objective of METRA, aiming to predict the convergent behavior of the learned representations. Second, based on the actual representation objective, we draw a connection between METRA and contrastive learning. In Sec. 4, we conduct experiments to verify that METRA learns optimal representations in practice and that they bear resemblance to contrastive representations.

Appendix C mentioned that the Lagrangian $L(\phi, \lambda)$ used as the METRA representation objective does not correspond to the constrained optimization problem in Eq. 9, raising the following question: *What is the actual METRA representation objective?* To answer this question, we note that, rather than using distinct dual variables for each pair of $(s, s') \in \mathcal{S}^{\beta}_{\text{adj}}$, $L(\phi, \lambda)$ employs a single dual variable, imposing an *expected* temporal distance constraint over all pairs of (s, s') under the historical transition distribution $p^{\beta}(s, s')$. This observation suggests that METRA's representations are optimized with the following objective

$$\max_{\phi} \mathbb{E}_{p(z)p^{\beta}(s,s'|z)}[(\phi(s') - \phi(s))^{\top} z] \quad \text{s.t. } \mathbb{E}_{p^{\beta}(s,s')}\left[\|\phi(s') - \phi(s)\|_2^2\right] \leq 1. \tag{4}$$

Applying KKT conditions to $L(\phi, \lambda)$, we claim that

Proposition 1. *The optimal state representation $\phi^{\star}$ of the actual METRA representation objective (Eq. 4) satisfies*

$$\mathbb{E}_{p^{\beta}(s,s')}\left[\|\phi^{\star}(s') - \phi^{\star}(s)\|_2^2\right] = 1.$$

The proof is in Appendix G.2. Constraining the representations of consecutive states in expectation not only clarifies the actual METRA representation objective, but also means that we can predict the value of this expectation for optimal ϕ. Sec. 4.1 includes experiments studying whether the optimal representation satisfies this proposition in practice. Importantly, identifying the actual METRA representation objective allows us to draw a connection with the rank-based contrastive loss (InfoNCE [47, 41]), which we discuss next.

We relate the actual METRA representation objective to a contrastive loss, which we will specify first and then provide some intuitions for what it is optimizing. This loss is a lower bound on the mutual information $I^{\beta}(S, S'; Z)$ and a variant of the InfoNCE objective [30, 41, 76]. Starting from the standard variational lower bound [4, 55], prior work derived an unnormalized variational lower bound on $I^{\beta}(S, S'; Z)$ (I_{UBA} in [55]),

$$I^{\beta}(S, S'; Z) \geq \mathbb{E}_{p^{\beta}(s,s',z)}[f(s, s', z)] - \mathbb{E}_{p^{\beta}(s,s')}\left[\log \mathbb{E}_{p(z')}\left[e^{f(s,s',z')}\right]\right],$$

where $f : \mathcal{S} \times \mathcal{S} \times \mathcal{Z} \mapsto \mathbb{R}$ is the critic function [41, 55, 20, 76]. Since the critic function f takes arbitrary functional form, one can choose to parameterize f as the inner product between the difference of transition representations and the latent skill, i.e. $f(s, s', z) = (\phi(s') - \phi(s))^{\top} z$. This yields a specific lower bound:

$$I^{\beta}(S, S'; Z) \geq \underbrace{\mathbb{E}_{p^{\beta}(s,s',z)}[(\phi(s') - \phi(s))^{\top} z]}_{\text{LB}^{\beta}_{+}(\phi)} \underbrace{- \mathbb{E}_{p^{\beta}(s,s')}\left[\log \mathbb{E}_{p(z')}\left[e^{(\phi(s') - \phi(s))^{\top} z'}\right]\right]}_{\text{LB}^{\beta}_{-}(\phi)} \triangleq \text{LB}^{\beta}(\phi). \tag{5}$$

Intuitively, $\text{LB}^{\beta}_{+}(\phi)$ pushes together the difference of transition representations $\phi(s') - \phi(s)$ and the latent skill z sampled from the same trajectory (positive pairs), while $\text{LB}^{\beta}_{-}(\phi)$ pushes away $\phi(s') - \phi(s)$ and z sampled from different trajectories (negative pairs). This intuition is similar to the effects of the contrastive loss, and we note that Eq. 5 only differs from the standard InfoNCE loss in excluding the positive pair in $\text{LB}^{\beta}_{-}(\phi)$. We will call this lower bound on the mutual information the *contrastive lower bound*.

We now connect the contrastive lower bound $\text{LB}^{\beta}(\phi)$ (Eq. 5) to the actual METRA representation loss $L(\phi, \lambda)$ (Eq. 10). While both of these optimization problems share the positive pair term ($\text{LB}^{\beta}_{+}(\phi)$), they vary in the way they handle randomly sampled (s, s', z) pairs (negatives): METRA constrains the expected L2 representation distances $\lambda\left(1 - \mathbb{E}_{p^{\beta}(s,s')}\left[\|\phi(s') - \phi(s)\|_2^2\right]\right)$, while the contrastive lower bound minimizes the log-expected-exp score ($\text{LB}^{\beta}_{-}(\phi)$). However, we bridge this difference by viewing the expected L2 distance as a quadratic approximation of the log-expected-exp score:

Proposition 2. *There exists a $\lambda_0(d)$ depending on the dimension d of the state representation ϕ such that the following second-order Taylor approximation holds*

$$\lambda_0(d)(1 - \mathbb{E}_{p^{\beta}}\left[\|\phi(s') - \phi(s)\|_2^2\right]) \approx \textit{LB}^{\beta}_{-}(\phi).$$

See Appendix G.3 for a proof. This approximation shows that the constraint in the actual METRA representation loss has effects similar to $\text{LB}^{\beta}_{-}(\phi)$, namely pushing $\phi(s') - \phi(s)$ away from randomly sampled skills. Furthermore, this proposition allows us to spell out the (approximate) equivalence between representation learning in METRA and the contrastive lower bound on $I^{\beta}(S, S'; Z)$:

Corollary 1. *The METRA representation objective is equivalent to a second-order Taylor approximation of $I^{\beta}(S, S'; Z)$, i.e., $L(\phi, \lambda_0(d)) \approx LB^{\beta}(\phi)$.*

The METRA representation objective can be interpreted as a contrastive loss, allowing us to predict that the optimal state representations $\phi^{\star}$ (Prop. 1) have properties similar to those learned via contrastive learning. In Appendix I.1, we include experiments studying whether the approximation in Prop. 2 is reasonable in practice. In Sec. 4.2, we empirically compare METRA's representations to those learned by the contrastive loss. Appendix F studies whether replacing the METRA representation objective with a contrastive objective retains similar performance.

A.2 Connecting METRA's Actor Objective with an Information Bottleneck

This section discusses the actor objective used in METRA. We first clarify the distinction between the actor objective of METRA and those used in prior methods, helping to identify a term that discourages exploration. Removing this anti-exploration term results in covering a larger proposition of the state space while learning distinguishable skills. We then relate this anti-exploration term to estimating another mutual information, drawing a connection between the entire METRA actor objective and a variant of the information bottleneck [68, 3].

While prior work [18, 24, 63, 28, 11] usually uses the same functional form of the lower bound on the mutual information $I(S, S'; Z)$ to learn both representations and skill-conditioned policies (Eq. 7 & 8), METRA uses different objectives for the representation and the actor. Specifically, the actor objective of METRA $J(\pi)$ (Eq. 11) only encourages the similarity between the difference of transition representations $\phi(s') - \phi(s)$ and their skill z (positive pairs), while ignoring the dissimilarity between $\phi(s') - \phi(s)$ and a random skill z (negative pairs):

$$J(\pi) = \text{LB}^{\pi}_{+}(\phi) = \text{LB}^{\pi}(\phi) - \text{LB}^{\pi}_{-}(\phi),$$

where $\text{LB}^{\pi}(\phi)$, $\text{LB}^{\pi}_{+}(\phi)$, and $\text{LB}^{\pi}_{-}(\phi)$ are under the target policy π instead of the behavioral policy β. The SOTA performance of METRA and the divergence between the functional form of the actor objective (positive term) and the representation objective (positive and negative terms) suggests that $\text{LB}^{\pi}_{-}(\phi)$ may be a term discouraging exploration. Intuitively, removing this anti-exploration term boosts the learning of diverse skills. We empirically study the effect of the anti-exploration term in Appendix F and provide theoretical interpretations next.

Our understanding of the anti-exploration term $\text{LB}^{\pi}_{-}(\phi)$ relates it to a resubstituion estimation of the differential entropy $h^{\pi}(\phi(S') - \phi(S))$ in the representation space (see Appendix G.4 for details), i.e., $\text{LB}^{\pi}_{-}(\phi) = \hat{h}^{\pi}(\phi(S') - \phi(S))$. Note that this entropy is different from the entropy of states $h^{\pi}(S)$, indicating that we want to *minimize* the entropy of difference of representations $\phi(s') - \phi(s)$ to encourage exploration. There are two underlying reasons for this (seemly counterintuitive) purpose: METRA aims to *(1)* constrain the expected L2 distance of difference of representations $\phi(s') - \phi(s)$ (Eq. 10) and *(2)* push difference of representations $\phi(s') - \phi(s)$ towards skills z sampled from $\textsc{Unif}(\mathbb{S}^{d-1})$. Nonetheless, this relationship allows us to further rewrite the anti-exploration term $\text{LB}^{\pi}_{-}(\phi)$ as an estimation of the mutual information $I^{\pi}(\phi(S') - \phi(S'); S, S')$, connecting the METRA actor objective to an information bottleneck:

Proposition 3. *The METRA actor objective is a lower bound on the information bottleneck $I^{\pi}(S, S'; Z) - I^{\pi}(S, S'; \phi(S') - \phi(S))$, i.e., $J(\pi) \leq I^{\pi}(S, S'; Z) - I^{\pi}(S, S'; \phi(S') = \phi(S))$.*

See Appendix G.4 for a proof and further discussions. Maximizing the information bottleneck $I^{\pi}(S, S'; Z) - I^{\pi}(S, S'; \phi(S') - \phi(S))$ compresses the information in transitions (s, s') into difference in representations $\phi(s') - \phi(s)$ while relating these representations to the latent skills z [3, 68]. This result implies that simply maximizing the mutual information $I^{\pi}(S, S'; Z)$ may be insufficient for deriving a diverse skill-conditioned policy π, and removing the anti-exploration $\text{LB}^{\pi}_{2}(\phi)$ may be a key ingredient for the actor objective. In Appendix G.5, we propose a general MISL framework based on Prop. 3.

B Related Work

Through careful theoretical and experimental analysis, we develop a new mutual information skill learning method that builds upon contrastive learning and successor features.

Unsupervised skill discovery. Our work builds upon prior methods that perform unsupervised skill discovery. Prior work has achieved this aim by maximizing lower bounds [70, 55] of different mutual information formulations, including diverse and distinguishable skill-conditioned trajectories [37, 18, 28, 34, 66], intrinsic empowerment [43, 13], distinguishable termination states [24, 72, 6], entropy bonus [21, 36, 61], predictable transitions [63, 11], etc. Among those prior methods, perhaps the most related works are CIC [34] and VISR [28]. Another line of unsupervised skill learning methods utilize ideas other than mutual information maximization, such as Lipschitz constraints [49], MDP abstraction [51], model learning [50], and Wasserstein distance [52, 29]. Our work will analyze the state-of-the-art method named METRA [52] that builds on the Wasserstein dependency measure [48], provide an alternative explanation under the well-studied MISL framework, and ultimately develop a simpler method.

Contrastive learning. Contrastive learning has achieved great success for representation learning in natural language processing and computer vision [56, 12, 23, 64, 14, 47, 26, 41, 70]. These methods aim to push together the representations of positive pairs drawn from the joint distribution, while pushing away the representations of negative pairs drawn from the marginals [41, 47]. In the domain of RL, contrastive learning has been used to define auxiliary representation learning objective for control [35, 74], solve goal-conditioned RL problems [76, 20, 19, 40], and derive representations for skill discovery [34]. Prior work has also provided theoretical analysis for these methods from the perspective of mutual information maximization [55, 70] and the geometry of learned representations [71]. Our work will combine insights from both angles to analyze METRA and show its relationship to contrastive learning, resulting in a new skill learning method.

Successor features. Our work builds on successor representations [15], which encode the discounted state occupancy measure of policies. Prior work has shown these representations can be learned on high-dimensional tasks [33, 75] and help with transfer learning [5]. When combined with universal value function approximators [59], these representations generalize to *universal* successor features, which estimates a value function for any reward under any policy [7]. While prior methods have combined successor feature learning with mutual information skill discovery for fast task inference [28, 39], we instead use successor features to replace Q estimation after learning state representations (Sec. 3).

C Preliminaries

Mutual information skill learning. The MISL problem typically involves two steps: *(1)* unsupervised pretraining and *(2)* downstream control. For the first step, we consider a Markov decision process (MDP) *without* reward function defined by states $s \in \mathcal{S}$, actions $a \in \mathcal{A}$, initial state distribution $p_0 \in \Delta(\mathcal{S})$, discount factor $\gamma \in (0, 1]$, and dynamics $p : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})$, where $\Delta(\cdot)$ denotes the probability simplex. The goal of unsupervised pretraining is to learn a skill-conditioned policy $\pi : \mathcal{S} \times \mathcal{Z} \mapsto \Delta(\mathcal{A})$ that conducts diverse and discriminable behaviors, where $\mathcal{Z}$ is a latent skill space. We use $\beta : \mathcal{S} \times \mathcal{Z} \mapsto \Delta(\mathcal{A})$ to denote the behavioral policy. We define the prior distribution of skills as a uniform distribution over the d-dimensional unit hypersphere $p(z) = \textsc{Unif}(\mathbb{S}^{d-1})$ (a uniform von Mises–Fisher distribution [73]) and will use this prior throughout our discussions.

Given a latent skill space $\mathcal{Z}$, prior methods [18, 63, 34, 24, 28] maximizes the MI between skills and states $I^\pi(S; Z)$ or the MI between skills and transitions $I^\pi(S, S'; Z)$ under the target policy. We will focus on $I^\pi(S, S'; Z)$ but our discussion generalizes to $I^\pi(S; Z)$. Specifically, maximizing the MI between skills and transitions can be written as

$$\max_\pi I^\pi(S, S'; Z) \overset{\text{const.}}{=} \max_\pi \mathbb{E}_{\substack{z \sim p(z), s \sim p^\pi(s_+ = s \mid z) \\ s' \sim p^\pi(s' \mid s, z)}}[\log p^\pi(z \mid s, s')], \tag{6}$$

where $p^\pi(s_+ = s \mid z)$ is the discounted state occupancy measure [31, 45, 20, 76] of policy π conditioned on skill z, and $p^\pi(s' \mid s, z)$ is the state transition probability given policy π and skill z. This optimization problem can be casted into an iterative min-max optimization problem by first choosing a variational distribution $q(z \mid s, s')$ to fit the historical posterior $p^\beta(z \mid s, s')$, which is an approximation of $p^\pi(z \mid s, s')$, and then choosing policy π to maximize discounted return defined by the intrinsic reward $\log q(z \mid s, s')$:

$$q_{k+1} \leftarrow \arg\max_q \mathbb{E}_{p^\beta(s,s',z)}[\log q(z \mid s, s')]. \tag{7}$$

$$\pi_{k+1} \leftarrow \arg\max_\pi \mathbb{E}_{p^\pi(s,s',z)}[\log q_k(z \mid s, s')], \tag{8}$$

Algorithm 1 Contrastive Successor Features

```
Input state representations $\phi_\theta$, successor features $\psi_\omega$, skill-conditioned policy $\pi_\eta$, and target successor feature $\psi_{\bar{\omega}}$.
for each iteration do
    Collect trajectory $\tau$ with $z \sim p(z)$ and $a \sim \pi_\eta(a \mid s, z)$, and then add $\tau$ to the replay buffer.
    Sample $\{(s, a, s', z)\} \sim$ replay buffer, $\{a'\} \sim \pi_\eta(a' \mid s', z)$, and $\{z'\} \sim p(z')$.
    $\mathcal{L}(\theta) \leftarrow -\mathbb{E}_{(s,s',z)}\left[(\phi_\theta(s') - \phi_\theta(s))^\top z\right] + \mathbb{E}_{(s,s')}\left[\log \sum_{z'} e^{(\phi_\theta(s') - \phi_\theta(s))^\top z'}\right]$.
    $\mathcal{L}(\omega) \leftarrow \mathbb{E}_{(s,a,s',a',z)}\left[(\psi_\omega(s, a, z) - (\phi_\theta(s') - \phi_\theta(s) + \gamma\psi_{\bar{\omega}}(s', a', z)))^2\right]$.
    $\mathcal{L}(\eta) \leftarrow -\mathbb{E}_{(s,z), a\sim\pi_\eta(a|s,z)}\left[\psi_\omega(s, a, z)^\top z\right]$.
    Update $\theta$, $\omega$, and $\eta$ by taking gradients of $\mathcal{L}(\theta)$, $\mathcal{L}(\omega)$, and $\mathcal{L}(\eta)$.
    Update $\bar{\omega}$ using exponential moving averages.
Return $\phi_\theta$, $\psi_\omega$, and $\pi_\eta$.
```

where k indicates the number of updates. See Appendix G.1 for detailed discussion.

For the second step, given a regular MDP (*with* reward function), we reuse the skill-conditioned policy π to solve a downstream task. Prior methods achieved this aim by *(1)* reaching goals in a zero-shot manner [49, 50, 52], *(2)* learning a hierarchical policy $\pi_h : \mathcal{S} \mapsto \Delta(\mathcal{Z})$ that outputs skills instead of actions [18, 34, 24], or *(3)* planning in the latent space with a learned dynamics model [63].

METRA. Maximizing the mutual information between states and latent skills $I(S; Z)$ only encourages an agent to find discriminable skills, while the algorithm might fail to prioritize state space coverage [52, 49]. A prior state-of-the-art method METRA [52] proposes to solve this problem by learning representations of states $\phi : \mathcal{S} \mapsto \mathbb{R}^d$ via maximizing the Wasserstein dependency measure (WDM) [48] between states and skills $I_{\mathcal{W}}(S; Z)$. Specifically, METRA chooses to enforce the 1-Lipschitz continuity of ϕ under the temporal distance metric, resulting in a constrained optimization problem for ϕ:

$$\max_\phi \mathbb{E}_{p(z)p^\beta(s,s'|z)}[(\phi(s') - \phi(s))^\top z] \quad \text{s.t. } \|\phi(s') - \phi(s)\|_2^2 \le 1 \ \forall (s, s') \in \mathcal{S}^\beta_{\text{adj}}, \tag{9}$$

where $p^\beta(s, s' \mid z)$ denotes the probability of first sampling s from the discounted state occupancy measure $p^\beta(s_+ = s \mid z)$ and then transiting to s' by following the behavioral policy β, and $\mathcal{S}^\beta_{\text{adj}}$ denotes the set of all the adjacent state pairs visited by β. In practice, METRA uses dual gradient descent to solve Eq. 9, resulting in an iterative optimization problem[4]

$$\min_{\lambda \ge 0} \max_\phi \ L(\phi, \lambda)$$
$$L(\phi, \lambda) \triangleq \mathbb{E}_{p(z)p^\beta(s,s'|z)}[(\phi(s') - \phi(s))^\top z] + \lambda\left(1 - \mathbb{E}_{p^\beta(s,s')}\left[\|\phi(s') - \phi(s)\|_2^2\right]\right), \tag{10}$$

Importantly, $L(\phi, \lambda)$ is *not* the Lagrangian of Eq. 9 because $L(\phi, \lambda)$ does not contain a dual variable for every $(s, s') \in \mathcal{S}^\beta_{\text{adj}}$. We discuss the actual METRA representation objective and the behavior of convergent representations in Sec. 2.

After learning the state representation ϕ, METRA finds its skill-conditioned policy π via maximizing the RL objective with intrinsic reward $(\phi(s') - \phi(s))^\top z$:

$$\max_\pi J(\pi), J(\pi) \triangleq \mathbb{E}_{\substack{z\sim p(z), s\sim p^\pi(s_+=s|z) \\ s'\sim p^\pi(s'|s,z)}}\left[(\phi(s') - \phi(s))^\top z\right]. \tag{11}$$

D Algorithm summary

We summarize our new algorithm, CSF, in Alg. 1 and the code is available online [5]. Starting from an existing MISL algorithm (e.g., DIAYN [18]), implementing our algorithm requires making three simple changes: *(1)* learning state representations ϕ_θ by minimizing an InfoNCE loss (excluding positive pairs in the denominator) between pairs of (s, s') and z, *(2)* using a critic ψ_ω with d-dimensional outputs and replacing the scalar reward with the vector $\phi_\theta(s') - \phi_\theta(s)$, *(3)* sampling the action a from the policy ϕ_η to maximize the inner product $\psi_\omega(s, a, z)^\top z$.

[4] We ignore the slack variable ϵ in Park et al. [52] because it takes a fairly small value $\epsilon = 10^{-3} \ll 1$.

[5] https://github.com/Princeton-RL/contrastive-successor-features

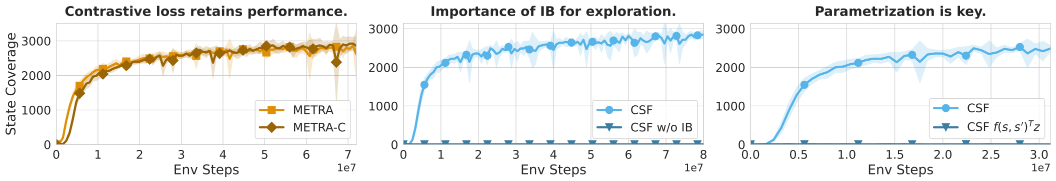


Figure 4: **Ablation studies.** *(Left)* Replacing the METRA representation loss with a contrastive loss retains performance. *(Center)* Using an information bottleneck to define the intrinsic reward is important for MISL. *(Right)* Choosing the right parametrization is crucial for good performance. Shaded areas indicate 1 std. dev.

Unlike CIC [34], our method does not use the standard InfoNCE loss and instead employs a variant of it. Unlike VISR [28], our method does not train the state representation ϕ using a skill discriminator. Unlike METRA, our method learns representations using the contrastive lower bound directly, avoids the Wasserstein distance and dual gradient descent optimization, and results in three fewer hyperparameters (see Appendix E for details).

E Hyperparameter Comparison to METRA

Compared to METRA, CSF has three fewer hyperparameters, as it gets rid of *(1)* the ϵ slack variable in Park et al. [52], *(2)* the dual gradient descent learning rate, and *(3)* the dual gradient descent optimizer. Reducing the number of hyperparameters in a method is important when thinking about *scaling* these methods to more complex domains where training runs may take several days or more. Extensive hyperparameter tuning in these settings will be impossible.

F Ablation Studies

We now study various design decisions of both METRA and our simplified method, aiming to identify some key factors that boost these MISL algorithms. We will conduct ablation studies on `Ant` again, comparing coverage of (x, y) coordinates of different variants.

(1) Contrastive learning recovers METRA's representation objective. Our analysis (Sec. 2.1) and experiments (Sec. 4.2) have shown that METRA learns contrastive representations. We now test whether we can retain the performance of METRA by simply replacing its representation objective with the contrastive lower bound (Eq. 5). Results in Fig. 4 *(Left)* suggest that using the contrastive loss (METRA-C) fully recovers the original performance, circumventing explanations building upon the Wasserstein dependency measure.

(2) Maximizing the information bottleneck is important. In Sec. 2.2, we interpret the intrinsic reward in METRA as a lower bound on an information bottleneck. We conduct ablation experiments to study the effect of maximizing this information bottleneck over maximizing the mutual information directly, a strategy typically used by prior methods [18, 42, 28]. Results in Fig. 4 *(Center)* show that CSF failed to discover skills when only maximizing the mutual information (i.e. including the anti-exploration term). These results indicate that using the information bottleneck as the intrinsic reward may be important for MISL algorithms.

(3) Parametrization is key for CSF. When optimizing a lower bound on the mutual information $I^\pi(S, S'; Z)$ using a variational distribution, there are many ways to parametrize the dependence on S, S', and Z. In Eq. 1, we chose the parametrization $(\phi(s') - \phi(s))^\top z$, but there are many other choices. Testing the sensitivity of this choice of parametrization allows us to determine whether a *specific form* of the lower bound is important. In Fig. 4, we study a variant of CSF that uses MLP$(s, s')^\top z$ instead of $(\phi(s') - \phi(s))^\top z$ in Eq. 1. We find using the alternative parametrization is catastrophic for performance, suggesting that practitioners may need to be careful about the parametrization when optimizing a lower bound on mutual information.

G Theoretical Analysis

G.1 Mutual Information Maximization as a Min-Max Optimization Problem

Maximizing the mutual information $I^\pi(S, S'; Z)$ (Eq. 6) is more challenging than standard RL because the reward function $\log p^\pi(z \mid s, s')$ depends on the policy itself. To break this cyclic dependency, we introduce a variational distribution $q(z \mid s, s') \in Q \triangleq \{q(z \mid s, s')\}$ to approximate the posterior $p^\pi(z \mid s, s')$, where we assume that the variational family Q is expressive enough to cover the ground true distribution under any π:

Assumption 1. *For any skill-conditioned policy $\pi : \mathcal{S} \times \mathcal{Z} \mapsto \Delta(\mathcal{A})$, there exists $q^\star(z \mid s, s') \in Q$ such that $q^\star(z \mid s, s') = p^\pi(z \mid s, s')$.*

This assumption allows us to rewrite Eq. 6 as

$$\max_\pi \mathbb{E}_{p^\pi(s,s',z)}[\log p^\pi(z \mid s, s')] - \min_{q \in Q} \mathbb{E}_{p^\pi(s,s')}\left[D_{\mathrm{KL}}\left(p^\pi(\cdot \mid s, s') \parallel q(\cdot \mid s, s')\right)\right],$$

where $D_{\mathrm{KL}}\left(p^\pi(\cdot \mid s, s') \parallel q(\cdot \mid s, s')\right)$ is the KL divergence between distributions p^π and q and it satisfies $D_{\mathrm{KL}}(p^\pi(\cdot \mid s, s') \parallel q(\cdot \mid s, s')) = 0 \iff p^\pi(z \mid s, s') = q(z \mid s, s')$. The new max-min optimization problem can be solved iteratively by first choosing variational distribution $q(z \mid s, s')$ to fit the ground truth $p^\pi(z \mid s, s')$ and then choosing policy π to maximize discounted return defined by the intrinsic reward $q(z \mid s, s')$:

$$q_{k+1} \leftarrow \arg\max_{q \in Q} \mathbb{E}_{p^{\pi_k}(s,s',z)}\left[\log q(z \mid s, s')\right],$$
$$\pi_{k+1} \leftarrow \arg\max_{\pi} \mathbb{E}_{p^{\pi}(s,s',z)}[\log q_k(z \mid s, s')],$$

where k indicates the number of updates. In practice, the data used to update q are uniformly sampled from a replay buffer typically containing trajectories from historical policies. Thus, the behavioral policy is exactly the average of historical policies $\beta = \frac{1}{k}\sum_{i=1}^{k} \pi_i(a \mid s, z)$ and the update rule for q becomes

$$q_{k+1} \leftarrow \arg\max_{q \in Q} \mathbb{E}_{p^\beta(s,s',z)}[\log q(z \mid s, s')].$$

G.2 Proof of Proposition 1

Proposition 1. *The optimal state representation $\phi^\star$ of the actual METRA representation objective (Eq. 4) satisfies*

$$\mathbb{E}_{p^\beta(s,s')}\left[\|\phi^\star(s') - \phi^\star(s)\|_2^2\right] = 1.$$

Proof. Suppose that the optimal $\phi^\star$ satisfies

$$0 \le \mathbb{E}_{p^\beta(s,s')}\left[\|\phi^\star(s') - \phi^\star(s)\|_2^2\right] = \alpha^2 < 1, \tag{12}$$

where $0 \le \alpha < 1$. Then, there exists a $1/\alpha > 1$ that scales the expectation in Eq. 12 to exactly 1:

$$\frac{1}{\alpha^2}\mathbb{E}_{p^\beta(s,s')}\left[\|\phi^\star(s') - \phi^\star(s)\|_2^2\right] = 1.$$

Note that the $\phi^\star/\alpha$ will also scale the objective to a larger number

$$\frac{1}{\alpha}\mathbb{E}_{p(z)p^\beta(s,s'|z)}\left[(\phi^\star(s') - \phi^\star(s))^\top z\right] > \mathbb{E}_{p(z)p^\beta(s,s'|z)}\left[(\phi^\star(s') - \phi^\star(s))^\top z\right],$$

which contradicts the assumption that $\phi^\star$ is optimal. Therefore, we conclude that the optimal $\phi^\star$ must satisfy $\mathbb{E}_{p^\beta(s,s')}\left[\|\phi^\star(s') - \phi^\star(s)\|_2^2\right] = 1$. □

G.3 Proof of Proposition 2

Proposition 2. *There exists a $\lambda_0(d)$ depending on the dimension d of the state representation ϕ such that the following second-order Taylor approximation holds*

$$\lambda_0(d)(1 - \mathbb{E}_{p^\beta}\left[\|\phi(s') - \phi(s)\|_2^2\right]) \approx \mathit{LB}_-^\beta(\phi).$$

Proof. We first compute $\log \mathbb{E}_p(z) \left[e^{(\phi(s')-\phi(s))^\top z} \right]$ analytically,

$$\begin{aligned}
\log \mathbb{E}_p(z) \left[e^{(\phi(s')-\phi(s))^\top z} \right] &= \log C_d(0) \int e^{(\phi(s')-\phi(s))^\top z} dz \\
&= \log \frac{C_d(0)}{C_d(\|\phi(s')-\phi(s)\|_2)} \\
&\quad + \log \int C_d(\|\phi(s')-\phi(s)\|_2) e^{\|\phi(s')-\phi(s)\|_2 \frac{(\phi(s')-\phi(s))^\top z}{\|\phi(s')-\phi(s)\|_2}} dz \\
&\overset{(a)}{=} \log \frac{C_d(0)}{C_d(\|\phi(s')-\phi(s)\|_2)} \\
&= \log \frac{\Gamma(d/2)(2\pi)^{d/2} \mathcal{I}_{d/2-1}(\|\phi(s')-\phi(s)\|_2)}{2\pi^{d/2} \|\phi(s')-\phi(s)\|_2^{d/2-1}} \\
&= \log \frac{\Gamma(d/2) 2^{d/2-1} \mathcal{I}_{d/2-1}(\|\phi(s')-\phi(s)\|_2)}{\|\phi(s')-\phi(s)\|_2^{d/2-1}},
\end{aligned} \tag{13}$$

where $\Gamma(\cdot)$ is the Gamma function, $\mathcal{I}_v(\cdot)$ denotes the modified Bessel function of the first kind at order v, and in *(a)* we use the definition of the density of von Mises-Fisher distributions. Applying Taylor expansion [1] to Eq. 13 around $\|\phi(s')-\phi(s)\|_2 = 0$ by using Mathematica [32] gives us a polynomial approximation

$$\log \frac{\Gamma(d/2) 2^{d/2-1} \mathcal{I}_{d/2-1}(\|\phi(s')-\phi(s)\|_2)}{\|\phi(s')-\phi(s)\|_2^{d/2-1}} = \frac{1}{2d} \|\phi(s')-\phi(s)\|_2^2 + O(\|\phi(s')-\phi(s)\|_2^3)$$

Now we can simply set $\lambda_0(d) = \frac{1}{2d}$ to get

$$\lambda_0(d)(1 - \|\phi(s')-\phi(s)\|_2^2) \approx -\log \mathbb{E}_p(z) \left[e^{(\phi(s')-\phi(s))^\top z} \right] + \text{const.}.$$

Hence, we conclude that $\lambda_0(d)(1 - \mathbb{E}_{p^\beta(s,s')} \left[\|\phi(s')-\phi(s)\|_2^2 \right])$ is a second-order Taylor approximation of $\text{LB}_-^\beta(\phi) = -\mathbb{E}_{p^\beta(s,s')} \left[\log \mathbb{E}_p(z) \left[e^{(\phi(s')-\phi(s))^\top z} \right] \right]$ around $\|\phi(s')-\phi(s)\|_2^2 = 0$ up to a constant factor of $\lambda_0(d)$. □

G.4 Proof of Proposition 3

Proposition 3. *The METRA actor objective is a lower bound on the information bottleneck $I^\pi(S, S'; Z) - I^\pi(S, S'; \phi(S') - \phi(S))$, i.e., $J(\pi) \leq I^\pi(S, S'; Z) - I^\pi(S, S'; \phi(S') = \phi(S))$.*

Proof. We consider the mutual information between transition pairs and skills under the target policy $I^\pi(S, S'; Z)$. The standard variational lower bound [4, 55] of $I^\pi(S, S'; Z)$ can we written as:

$$I^\pi(S, S'; Z) \geq h(Z) + \mathbb{E}_{p^\pi(s,s',z)}[\log q^\pi(z \mid s, s')],$$

where $q^\pi(z \mid s, s')$ is an arbitrary variational approximation of $p^\pi(z \mid s, s')$. We can set $\log q^\pi(z \mid s, s')$ to be

$$\log q^\pi(z \mid s, s') = f(s, s', z) + \log p(z) - \log \mathbb{E}_{p(z)} \left[e^{f(s,s',z)} \right],$$

resulting in a lower bound:

$$I^\pi(S, S'; Z) \geq \underbrace{\mathbb{E}_{p^\pi(s,s',z)}[(\phi(s')-\phi(s))^\top z]}_{\text{LB}_+^\pi(\phi)} \underbrace{-\mathbb{E}_{p^\pi(s,s')} \left[\log \mathbb{E}_{p(z)} \left[e^{(\phi(s')-\phi(s))^\top z} \right] \right]}_{\text{LB}_-^\pi(\phi)},$$

where $\text{LB}_+^\pi(\phi)$ is exactly the same as the RL objective $J(\pi)$ (Eq. 11). This lower bound is similar to Eq. 5 but it is under the target policy π instead.

Equivalently, we can write the RL objective as

$$J(\pi) = \mathbb{E}_{p(z)p^{\pi}(s_+=s|z)p^{\pi}(s'|s,z)}\left[(\phi(s') - \phi(s))^{\top} z - \log \mathbb{E}_{p(z')}\left[e^{(\phi(s')-\phi(s))^{\top} z'}\right]\right.$$
$$\left. + \log \mathbb{E}_{p(z')}\left[e^{(\phi(s')-\phi(s))^{\top} z'}\right]\right] = \text{LB}^{\pi}(\phi) - \text{LB}^{\pi}_{-}(\phi)$$

where the two log-expected-exps cancel with each other. We next focus on the additional $\text{LB}^{\pi}_{-}(\phi) = -\mathbb{E}_{p(z)p^{\pi}(s_+=s|z)p^{\pi}(s'|s,z)}\left[\log \mathbb{E}_{p(z')}\left[e^{(\phi(s')-\phi(s))^{\top} z'}\right]\right]$, which can be interpreted as a resubstitution entropy estimator of $\phi(s') - \phi(s)$ [71, 2]:

$$\begin{aligned}
\text{LB}^{\pi}_{-}(\phi) &= -\mathbb{E}_{p^{\pi}(s,s')}\left[\log \mathbb{E}_{p(z)}\left[e^{(\phi(s')-\phi(s))^{\top} z}\right]\right] \\
&\overset{(a)}{=} -\frac{1}{N}\sum_{i=1}^{N}\left[\log\left(\frac{1}{N}\sum_{j=1}^{N} C_d(\|\phi(s_i') - \phi(s_i)\|_2) e^{\|\phi(s_i')-\phi(s_i)\|_2 \frac{(\phi(s_i')-\phi(s_i))^{\top} z_j}{\|\phi(s_i')-\phi(s_i)\|_2}}\right)\right. \\
&\qquad\qquad \left. + \log \frac{1}{C_d(\|\phi(s_i') - \phi(s_i)\|_2)}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left(\log \hat{p}_{\text{vMF-KDE}}(\phi(s_i') - \phi(s_i)) + \log \frac{1}{C_d(\|\phi(s_i') - \phi(s_i)\|_2)}\right) \\
&= \hat{h}^{\pi}(\phi(S') - \phi(S)) - \mathbb{E}_{p^{\pi}(s,s')}\left[\log \frac{1}{C_d(\|\phi(s_i') - \phi(s_i)\|_2)}\right] \\
&\overset{(b)}{=} \hat{I}^{\pi}(S, S'; \phi(S') - \phi(S)) - \mathbb{E}_{p^{\pi}(s,s')}\left[\log \frac{1}{C_d(\|\phi(s_i') - \phi(s_i)\|_2)}\right] \\
&\overset{(c)}{\approx} \hat{I}^{\pi}(S, S'; \phi(S') - \phi(S)) - \lambda_0(d)\mathbb{E}_{p^{\pi}(s,s')}\left[\|\phi(s') - \phi(s)\|_2^2\right] + \text{const.} \\
&\overset{(d)}{\approx} \hat{I}^{\pi}(S, S'; \phi(S') - \phi(S)) + \text{const.},
\end{aligned}$$

where in *(a)* we use Monte Carlo estimator with N transitions and skills $\{(s_i, s_i', z_i)\}_{i=1}^{N}$ to rewrite the expectation, in *(b)* we replace the entropy estimator $\hat{h}^{\pi}$ with the mutual information estimator $\hat{I}^{\pi}$ since $\phi(s') - \phi(s)$ is a deterministic function of (s, s'), in (c) we apply the same approximation in Prop. 2, and in (d) the expected squared norm is replaced by 1.0 given the constraint in Eq. 4. Taken together, we conclude that maximizing the RL objective $J(\pi)$ is approximately equivalent to maximizing a lower bound on the information bottleneck $I^{\pi}(S, S'; Z) - I^{\pi}(S, S'; \phi(S') - \phi(S))$. □

G.5 A General Mutual Information Skill Learning Framework

The general mutual information skill learning algorithm alternates between *(1)* collecting data, *(2)* learning state representation ϕ by maximizing a lower bound on the mutual information $I^{\beta}(S, S'; Z)$ under the behavioral policy β, *(3)* relabeling the intrinsic reward as a lower bound on the information bottleneck $I^{\pi}(S, S'; Z) - I^{\pi}(S, S'; \phi(S') - \phi(S))$, and finally *(4)* using an off-the-shelf off policy RL algorithm to learning the skill-conditioned policy π. We show the pseudo-code of this algorithm in Alg. 2.

G.6 Connection Between Representations Learned by METRA and Contrastive Representations

In our experiments (Sec. 4.2), we sample 10K pairs of (s, s', z) from the replay buffer and use them to visualize the histograms of conditional differences in representations $\phi(s') - \phi(s) - z$ and normalized marginal differences in representations $(\phi(s') - \phi(s))/\|\phi(s') - \phi(s)\|_2$. The resulting histograms (Fig. 2 *(Center)* & *(Right)*) indicate two intriguing properties of representations learned by METRA. First, given a set of skills $\{z\}$, the differences in representations subtracting the corresponding skills $\phi(s') - \phi(s) - z$ converges to an isotropic Gaussian distribution:

Claim 1. *The state representations ϕ learned by METRA satisfies that $\phi(s') - \phi(s) - z \xrightarrow{d} \mathcal{N}(0, \sigma_{\phi}^2 I)$, or, equivalently, $\phi(s') - \phi(s) \mid z \xrightarrow{d} \mathcal{N}(z, \sigma_{\phi}^2 I)$, where $\xrightarrow{d}$ denotes convergence in distribution and σ_{ϕ} is the standard deviation of the isotropic Gaussian.*

Algorithm 2 A General Mutual Information Skill Learning Framework

```
Input state representations $\phi : \mathcal{S} \mapsto \mathbb{R}^d$, latent skill distribution $p(z)$, and skill-conditioned policy $\pi : \mathcal{S} \times \mathcal{Z} \mapsto \Delta(\mathcal{A})$.
for each iteration do
    Collect trajectory $\tau$ with $z \sim p(z)$ and $a \sim \pi(a \mid s, z)$, and then add $\tau$ to the replay buffer.
    Sample $B = \{(s, s', z)\}$ from the replay buffer.
    Update $\phi$ by maximizing a lower bound on $I^\beta(S, S'; Z)$ constructed using $B$.
    Relabel the intrinsic reward as a lower bound on $I^\pi(S, S'; Z) - I^\pi(S, S'; \phi(S') - \phi(S))$.
    Update $\pi$ using an off-policy RL algorithm with $B \cup \{r(s, s', z)\}$.
Return $\phi^\star$ and $\pi^\star$.
```

Second, taking the marginal over all possible skills, the normalized difference in representations $(\phi(s') - \phi(s))/\|\phi(s') - \phi(s)\|_2$ converges to a uniform distribution on the d-dimensional unit hypersphere $\mathbb{S}^{d-1}$:

Claim 2. *The state representations ϕ learned by METRA also satisfy $\frac{\phi(s')-\phi(s)}{\|\phi(s')-\phi(s)\|_2} \xrightarrow{d} \textsc{Unif}(\mathbb{S}^{d-1})$.*

We next propose a Lemma that relates a isotropic Gaussian distribution to a von Mises–Fisher distribution [73] and then draw the connection between Claim 1 and Claim 2.

Lemma 1. *Given an n-dimensional isotropic Gaussian distribution $\mathcal{N}(\mu, \sigma^2 I)$ with $\|\mu\|_2 = r_\mu$, a von Mises–Fisher distribution* vMF $(\mu/r_\mu, r_\mu/\sigma^2)$ *can be obtained by restricting the support to be a hypersphere with radius r_μ, i.e., $\{x : \|x\|_2 = r_\mu\}$.*

Proof. The probability density function of $\mathcal{N}(\mu, \sigma^2 I)$ be written as

$$\begin{aligned}p(x) &= \frac{1}{(2\pi\sigma)^{\frac{n}{2}}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^\top(x-\mu)\right)\\ &= \frac{1}{(2\pi\sigma)^{\frac{n}{2}}} \exp\left(-\frac{1}{2\sigma^2}\left(\|x\|_2^2 - 2x^\top\mu + \|\mu\|_2^2\right)\right)\end{aligned}$$

When conditioning on $\|x\|_2 = r_\mu$, we have

$$\begin{aligned}\frac{1}{(2\pi\sigma)^{\frac{n}{2}}} \exp\left(-\frac{1}{2\sigma^2}\left(\|x\|_2^2 - 2x^\top\mu + \|\mu\|_2^2\right)\right) &= \frac{1}{(2\pi\sigma)^{\frac{n}{2}}} \exp\left(-\frac{1}{2\sigma^2}\left(2r_\mu^2 - 2x^\top\mu\right)\right)\\ &= \frac{1}{(2\pi\sigma)^{\frac{n}{2}}} \exp\left(\frac{r_\mu}{\sigma^2} \cdot \frac{\mu^\top x}{r_\mu} - \frac{r_\mu^2}{\sigma^2}\right)\\ &\propto \exp\left(\frac{r_\mu}{\sigma^2} \cdot \frac{\mu^\top x}{r_\mu}\right).\end{aligned}$$

After recomputing the normalizing constant, we recover the probability density function of the von Mises-Fisher distribution vMF $(\mu/r_\mu, r_\mu/\sigma^2)$. □

Since the didactic experiments in Sec. 4.2 have shown that $\phi(s') - \phi(s) \mid z$ converges to a Gaussian distribution $\mathcal{N}(z, \sigma_\phi^2 I)$ (Claim 1) and note that $\|z\|_2 = 1$, by applying Lemma G.6, we conjecture that restricting $\phi(s') - \phi(s)$ within $\{\|\phi(s') - \phi(s)\|_2 = 1\}$ produces a von Mises-Fisher distribution, i.e., $\frac{\phi(s')-\phi(s)}{\|\phi(s')-\phi(s)\|_2}\Big| z \xrightarrow{d} \text{vMF}(z, 1/\sigma_\phi^2)$. Furthermore, we can derive the marginal density of $\frac{\phi(s')-\phi(s)}{\|\phi(s')-\phi(s)\|_2}$,

$$\begin{aligned}p\left(\frac{\phi(s') - \phi(s)}{\|\phi(s') - \phi(s)\|_2}\right) &= \int p(z) p\left(\frac{\phi(s') - \phi(s)}{\|\phi(s') - \phi(s)\|_2}\middle| z\right) dz\\ &\overset{(a)}{=} C_d(0) \int C_d\left(\frac{1}{\sigma_\phi^2}\right) \exp\left(\frac{1}{\sigma_\phi^2} \cdot \frac{(\phi(s') - \phi(s))^\top z}{\|\phi(s') - \phi(s)\|_2}\right) dz\\ &= C_d(0),\end{aligned}$$

where in *(a)* we use the symmetric property of the density function of von Mises-Fisher distributions. Crucially, the marginal density indicates that $\frac{\phi(s')-\phi(s)}{\|\phi(s')-\phi(s)\|_2}$ follows a uniform distribution $\textsc{Unif}(\mathbb{S}^{d-1})$, which is exactly the observation in our experiments (Claim 2).

G.7 Intuition for zero-shot goal inference

In zero-shot goal reaching setting, we want to figure out the corresponding latent skill z when given a goal state or image g; a problem which can be cast as inferring the $z \in \mathcal{Z}$ that maximizes the posterior $p^\pi(z \mid s, g)$. Since the ground truth posterior $p^\pi(z \mid s, g)$ is unknown, a typical workaround is first estimating a variational approximation of $p^\pi(z \mid s, g)$ and then maximizing the variational posterior $q(z \mid s, g)$. We provide an intuition for zero-shot goal inference by specifying the variational posterior as

$$q(z \mid s, g) \triangleq \frac{p(z) e^{(\phi(g)-\phi(s))^\top z}}{\mathbb{E}_{p(z)}\left[e^{(\phi(g)-\phi(s))^\top z'}\right]}$$

and solving the optimization problem in the latent skill space $\mathcal{Z}$

$$\arg\max_{z \in \mathcal{Z}} \; \log q(z \mid s, g),$$

or equivalently,

$$\arg\max_{z} \; (\phi(g) - \phi(s))^\top z \quad \text{s.t. } \|z\|_2^2 = 1.$$

Taking derivative of the Lagrangian and setting it to zero, the analytical solution is exactly $z^\star = \frac{\phi(g)-\phi(s)}{\|\phi(g)-\phi(s)\|_2}$, suggesting that the heuristic used by prior methods and our algorithm can be understood as a maximum a posteriori (MAP) estimation.

H Experimental Details

All experiments were run on a combination of GPUs consisting of NVIDIA GeForce RTX 2080 Ti, NVIDIA RTX A5000, NVIDIA RTX A6000, and NVIDIA A100. All experiments took at most 1 day to run to completion.

H.1 Hyperparameter Comparison to METRA

Compared to METRA, CSF has five fewer hyperparameters, as it gets rid of *(1)* the ϵ slack variable in Park et al. [52], *(2)* the norm constraint value in Eq. 9, *(3)* the dual gradient descent learning rate, *(4)* the dual gradient descent optimizer, and *(5)* the choice of discrete or continuous skills z. Reducing the number of hyperparameters in a method is important when thinking about *scaling* these methods to more complex domains where training runs may take several days or more. Extensive hyperparameter tuning in these settings will be impossible.

H.2 Experimental Setup

Environments. We choose to evaluate on the following six tasks: `Ant` and `HalfCheetah` from Gym [69, 8], `Quadruped` and `Humanoid` from DeepMind Control (DMC) Suite [67], and `Kitchen` and `Robobin` from LEXA [42]. We choose these six tasks to be consistent with the original METRA work [52]. In addition, we added `Robobin` as another manipulation task since the original five tasks are all navigation tasks except for `Kitchen`. The observations are state-based in `Ant` and `HalfCheetah` and 64×64 RGB images of the scene in all other tasks.

Baselines. We consider five baselines. **(1) METRA [52]** is the state-of-the-art approach which provides the motivation for deriving CSF. **(2) CIC [34]** uses a rank-based contrastive loss (InfoNCE) to learn representations of transitions and then maximizes a state entropy estimate constructed using these representations. **(3) DIAYN [18]** represents a broad category of methods that first learn a parametric discriminator $q(z \mid s, s')$ (or $q(z \mid s)$) to predict latent skills from transitions and then construct the reverse variational lower bound on mutual information [11] as an intrinsic

Table 1: **CSF hyperparameters for unsupervised pretraining.**

Hyperparameter	**Value**
Learning rate	0.0001
Horizon	200, except for 50 in Kitchen
Parallel workers	8, except for 10 in Robobin
State normalizer	used in state-based environments only
Replay buffer batch size	256
Gradient updates per trajectory collection round	50 (Ant, Cheetah), 200 (Humanoid, Quadruped), 100 (Kitchen, Robobin)
Frame stack	3 for image-based, n/a for state-based
Trajectories per data collection round	8, except for 10 in Robobin
Automatic entropy tuning	yes
ξ (scales second term in Eq. 2)	5
Number of negative zs to compute $\text{LB}_{-}(\phi)$	256 (in-batch negatives)
EMA τ (target network)	$5e^{-3}$
ϕ, π, ψ network hidden dimension	1024
ϕ, π, ψ network number of layers	1 input, 1 hidden, 1 output
ϕ, π, ψ network nonlinearity	relu (ϕ), tanh (π), relu (ψ)

reward. **(4) DADS [63]** builds upon the forward variational lower bound on mutual information [11] which requires maximizing the state entropy $h(S)$ to encourage state coverage while minimizing the conditional state entropy $h(S \mid Z)$ to distinguish different skills. There is a family of methods studying variational approximations of $h(S)$ and $h(S \mid Z)$ [11, 39, 34, 36, 63] of which DADS is a representative. **(5) VISR [28]** is similar to DIAYN in that it also trains the representations ϕ by learning a discriminator to maximize the likelihood of a skill given a state, though the discriminator is parametrized as a vMF distribution. In addition, VISR learns successor features that allow it to perform GPI as well as fast task adaptation after unsupervised pretraining. Note that our version of VISR does not include GPI since we evaluate on continuous control environments.

H.3 Exploration performance

Please see Fig. 5 for the full set of exploration results. We can see that CSF continues to perform on par with METRA, while sometimes outperforming METRA (Robobin) and sometimes underperforming METRA (Quadruped).

For CSF, all tasks were trained with continuous z sampled from a uniform vMF distribution and $\lambda = 5$. METRA also uses a continuous z sampled from a uniform vMF distribution for all environments except for HalfCheetah and Kitchen, where we used a one-hot discrete z, consistent with the original work [52]. CIC uses a continuous z sampled from a standard Gaussian for all environments. DIAYN uses a one-hot discrete z for all environments. DADS uses a continuous z sampled from a uniform distribution on $[-1, 1]$ for all environments. Finally, VISR uses a continuous z sampled from a uniform vMF distribution for all environments. Please refer to Table 2 for a full overview of skill dimensions per method and environment. A table with all relevant hyperparameters for the unsupervised training phase can be found in Table 1.

H.4 Zero-shot goal reaching

Please see Fig. 6 for the full set of goal reaching results. We find CSF to generally perform closely to METRA, though slightly underperforming in Quadruped, Humanoid, and Kitchen. In Ant however, CSF outperforms METRA.

Goal sampling. We closely follow the setup in Park et al. [52]. For all baselines, 50 goals are randomly sampled from $[-50, 50]$ in Ant, $[100, 100]$ in HalfCheetah, $[-15, 15]$ in Quadruped, and $[-10, 10]$ in Humanoid. In Kitchen, we sample 50 times at random from the following built-in tasks: BottomBurner, LightSwitch, SlideCabinet, HingeCabinet, Microwave, and Kettle. In Robobin, we sample 50 times at random from the following built-in tasks: ReachLeft, ReachRight, PushFront, and PushBack.

Table 2: **Skill dimensions per method and environment.** We list the skill dimension for all methods and environments reported in the paper.

	Ant	HalfCheetah	Quadruped	Humanoid	Kitchen	Robobin
CSF	2	2	4	8	4	9
METRA	2	16	4	2	24	9
DIAYN	50	50	50	50	50	50
DADS	3	3	-	-	-	-
CIC	64	64	64	64	64	64
VISR	5	5	5	5	5	5

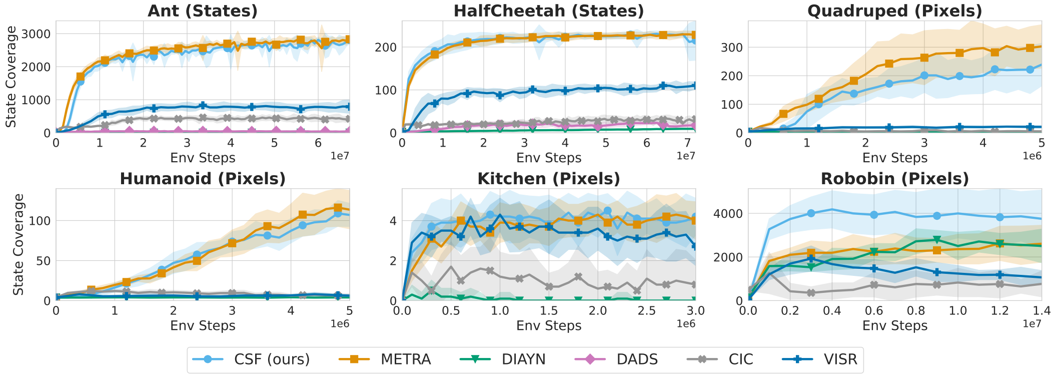


Figure 5: **State space coverage.** We plot the unique number of coordinates visited by the agent, except for Kitchen where we plot the task coverage. We find CSF matches the prior state-of-the-art MISL algorithms on $4/6$ tasks, and strongly outperforms METRA in `Robobin`. Shaded areas indicate one standard deviation.

Evaluation Unlike prior methods [49, 52, 63, 42], we choose the *staying time fraction* instead of the *success rate* as our evaluation metric. The staying time indicates the number of time steps that the agent stays at the goal divided by the horizon length, while the success rate simply indicates whether the agent reaches the goal at *any* time step. Importantly, a high success rate does not necessarily imply a high staying time fraction (e.g., the agent might overshoot the goal after success).

Skill inference. Prior work [49, 52, 50] has proposed a simple inference method by setting the skill to the difference in representations $z = \frac{\phi(g)-\phi(s)}{\|\phi(g)-\phi(s)\|_2}$, where g indicates the goal. We choose to use the same approach for CSF and METRA and provide some theoretical intuition for this strategy in Appendix G.7. For DIAYN, we follow prior work [52] and set $z = \text{one\_hot}[\arg\max_i q(z|g)_i]$.

H.5 Hierarchical control

Please see Fig. 7 for the full set of hierarchical control results. We find CSF to perform closely to METRA in most environments, though it outperforms METRA on `AntMultiGoal` and underperforms METRA on `QuadrupedGoal`. CSF outperforms all other baselines on all environments.

We use SAC [27] for `AntMultiGoal`, `HumanoidGoal`, and `QuadrupedGoal`. We use PPO [60] for `CheetahGoal` and `CheetahHurdle`. For all state-based environments, we initialize (and freeze) the child policy with a checkpoint trained with 64M environment steps. For image-based environments, we use checkpoints trained with 4.8M environments. A table with all relevant hyperparameters for training the hierarchical control policy can be found in Table 3.

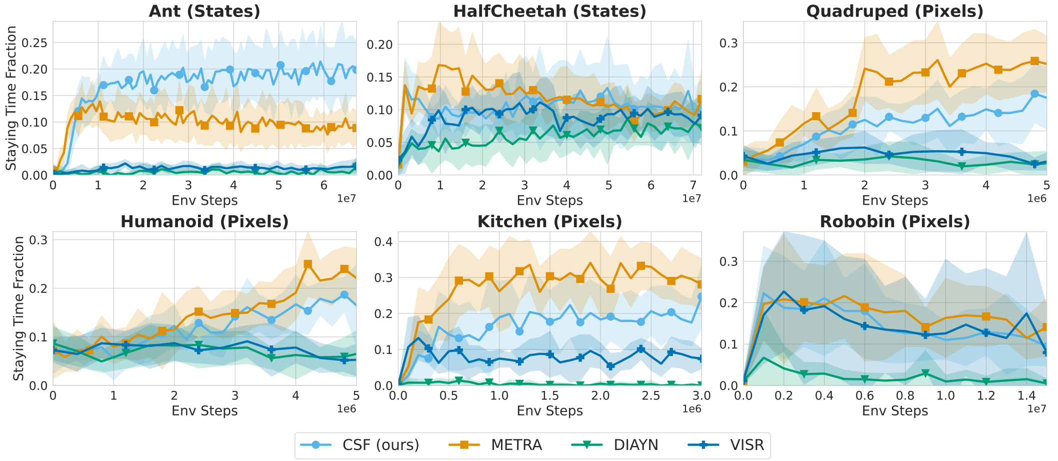


Figure 6: **Goal reaching.** We compare CSF with baselines on goal-reaching tasks. We find that CSF achieves strong performance on `Ant` and mostly outperforms DIAYN and VISR. However, CSF lags a bit behind METRA on `Quadruped`, `Kitchen`, and `Humanoid`. All means and standard deviations are computed across ten random seeds. Shaded areas indicate one standard deviation.

Table 3: **CSF hyperparameters for hierarchical control.**

Hyperparameter	**Value**
Learning rate	0.0001
Option timesteps length	25
Total horizon length	200
Parallel workers	8
Trajectories per data collection round	8, except for `Cheetah` where we use 64
Algorithm	SAC, except for `Cheetah` where we use PPO
State normalizer	used in state-based environments only
Replay buffer batch size	256
Gradient updates per trajectory collection round	50, except for `Cheetah` where we use 10
Frame stack	3 for image-based, n/a for state-based
π (parent, child) networks hidden dimension	1024
π (parent, child) networks number of layers	1 input, 1 hidden, 1 output
π (parent, child) networks nonlinearity	tanh
Child policy frozen?	yes

I Additional Experiments

I.1 Quadratic approximation of $\mathrm{LB}_2^\beta(\phi)$

We conduct experiments to study the accuracy of the quadratic approximation in Prop. 2 in practice. To answer this question, we reuse the METRA algorithm trained on the didactic `Ant` environment and compare $\log \mathbb{E}_{p(z)}[e^{(\phi(s')-\phi(s))^\top z}]$ against $\|\phi(s') - \phi(s)\|_2^2$. We can compute $\log \mathbb{E}_{p(z)}[e^{(\phi(s')-\phi(s))^\top z}]$ analytically because $d = 2$ in our experiments. Results in Fig. 8 shows a clear linear relationship between $\log \mathbb{E}_{p(z)}[e^{(\phi(s')-\phi(s))^\top z}]$ and $\|\phi(s') - \phi(s)\|_2^2$, suggesting that the slope of the least squares linear regression is near the theoretical prediction, i.e., $\lambda_0(d) = \frac{1}{2d} = 0.25 \approx 0.2309$. We conjecture that this linear relationship still exists for higher dimensional d and, therefore, the second-order Taylor approximation proposed by Prop. 2 is practical.

I.2 METRA and CSF are Sensitive to the Skill Dimension

METRA leverages different skill dimensions for different environments. This caused us to investigate what the impact of the skill dimension on exploration performance is. In Fig. 9, we find that both

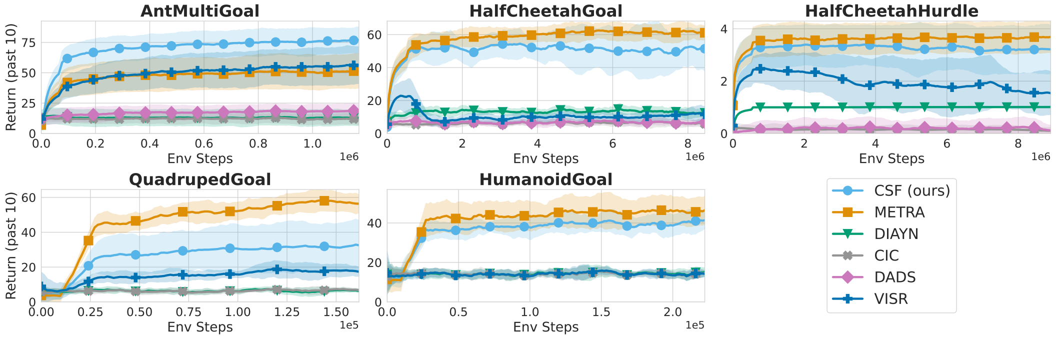
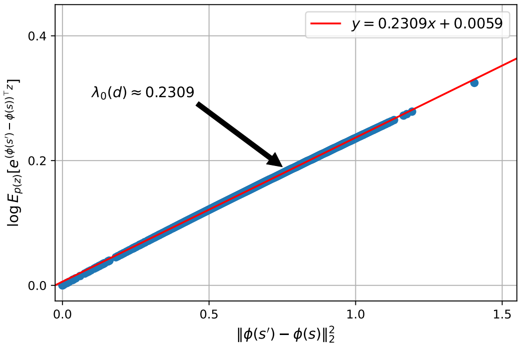


Figure 7: **Hierarchical control.** We compare CSF with baselines on hierarchical control tasks using returns averaged over the 10 past evaluations. We find CSF to perform mostly competitively compared to METRA, outperforming METRA in `AntMultiGoal`, but underperforming in `QuadrupedGoal` (and to a small extent in `HalfCheetahGoal` and `HumanoidGoal`). All means and standard deviations are computed across ten random seeds. Shaded areas indicate one standard deviation.



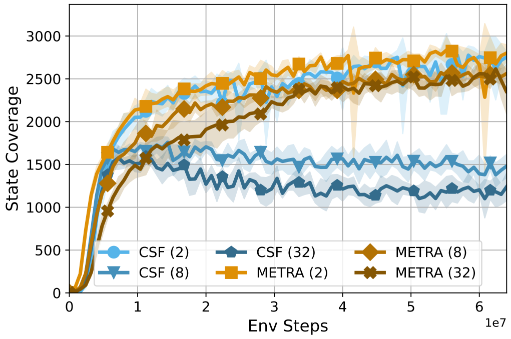


Figure 8: $\lambda_0(d)\|\phi(s') - \phi(s)\|_2^2$ is a second order Taylor approximation of $\log \mathbb{E}_{p(z)}[e^{(\phi(s')-\phi(s))^\top z}]$, where $\lambda_0(d) = \frac{1}{2d}$.

Figure 9: **Skill Dimension.** CSF and METRA (to a lesser extent) are sensitive to the skill dimension (indicated in parentheses).

METRA (to a lesser extent) and CSF are quite sensitive to the skill dimension. We conclude that skill dimension is a key parameter to tune for practitioners when training their MISL algorithm.