Adaptive Feedback Selection for Learning to Avoid Negative Side Effects in Autonomous Agents

Abstract

Autonomous agents operating with underspecified objectives often produce negative side effects (NSEs) that are difficult to identify at design time. We examine how a deployed agent can learn a penalty function associated with NSEs, from diverse sources of information which are collected actively or passively from a user who interacts explicitly or implicitly with the agent. Unlike prior works that learn to avoid NSEs from a single form of feedback, our framework facilitates learning from *multiple* forms of feedback during the course of agent operation. Our framework for *adaptive feedback selection* enables the agent to query for feedback in formats that maximize information gain about NSE severities, given the human's feedback preference model that specifies the cost and probability of receiving feedback in a certain format. When query budget is limited, the agent prioritizes querying in states that provide important information to learn an NSE prediction model. We present an algorithm that clusters states and iteratively selects critical states for querying, by updating the weights of each cluster based on the number of new NSEs identified from feedback. Empirical evaluation on three domains show our framework's effectiveness in learning to avoid NSEs from explicit and implicit feedback.

1 Introduction

Autonomous agents in complex real-world settings often operate with underspecified or incomplete objectives and reward functions, which can lead to negative side effects (NSEs). NSEs are the undesired, unmodeled effects of agent actions on the environment (Amodei et al., 2016; Saisubramanian et al., 2021a). For example, an indoor agent optimizing distance to goal may unintentionally break a vase, as a negative side effect, if its model lacks details on the undesirability of its actions (Krakovna et al., 2020). Agents typically lack prior knowledge about the NSEs of their actions.

A popular approach to overcome this concern is to learn about NSEs from human feedback (Saisubramanian et al., 2021a; Srivastava et al., 2023; Zhang et al., 2020b), which can be explicit (e.g., approving agent's actions, providing demonstrations), or implicit (e.g., gaze and facial expressions) (Cui & Niekum, 2018; Cui et al., 2021; Lakkaraju et al., 2017; Saran et al., 2021). These approaches typically assume that the user will always provide immediate feedback in a *single* format throughout the course of agent operation, either in response to an agent query or based on observed agent trajectories (Ghosal et al., 2023; Ibarz et al., 2018; Saisubramanian et al., 2022). However, these assumptions often do not reflect real-world interactions. In practice, the human (1) may not be able to provide feedback when they are busy or away; and (2) may be able to provide feedback in more than one format during agent learning and operation (Loftin et al., 2014). In fact, a recent user study conducted specifically on the side effects problem indicates that users are generally willing to engage with the agent in more than one feedback format (Saisubramanian et al., 2021b).

For example, in the vase problem, the human may provide feedback through binary signals to approve actions, demonstrate safe ways of performing the task, or correct agent actions. Each format offers different level of information and requires varying human effort. However, the human may be uncertain about the most effective format for agent learning. The key question we aim to address in this paper is: *how can an agent leverage the human's ability to provide feedback in multiple formats, by querying for feedback in a format that maximizes information gain?*

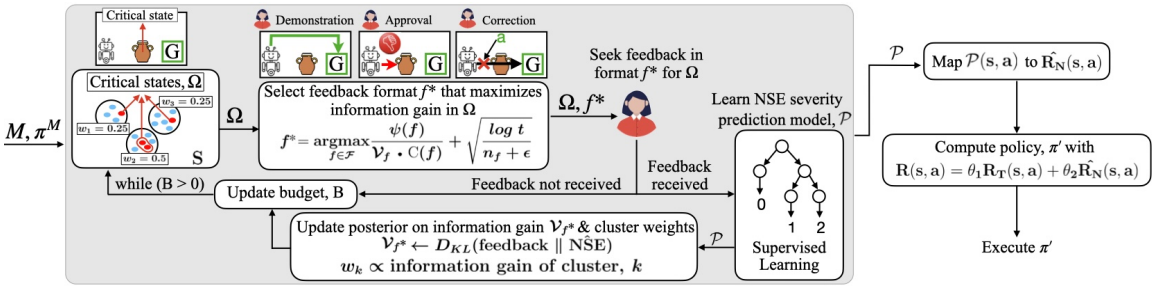


Figure 1: Framework for learning to avoid NSEs from diverse human feedback in critical states.

We present a framework for *adaptive feedback selection* (AFS) enabling the agent to request feedback in a format that maximizes information gain, given a model of human feedback preferences. This model specifies the cost and probability of receiving feedback in a certain format. Information gain of a feedback is measured as the KL divergence between the feedback, sampled from underlying true NSE distribution, and the agent's current knowledge of NSEs.

When collecting feedback in every state is infeasible, the agent must prioritize querying in *critical states*—states where human feedback is crucial for learning an association of state features and NSEs, i.e., a predictive model of NSE severities. Querying in critical states maximizes information gain about NSEs, compared to other states. In the vase example, states with a vase are critical states, as they provide valuable information about state features correlating with NSEs. Prior works, however, query for feedback in states randomly sampled or along the shortest path to the goal, which may not lead to a faithful NSE model (Saisubramanian et al., 2021a; Zhang et al., 2020b). Our algorithm iteratively selects critical states and the format to query in these states (Fig. 1).

We use a four-step solution approach to gather NSE information under a limited query budget: (1) states are partitioned into clusters, with a cluster weight proportional to the number of NSEs discovered in it; (2) a critical states set is formed by sampling from each cluster based on its weight; (3) a feedback format that maximizes the information gain in critical states is identified, while accounting for the cost and uncertainty in receiving a feedback, using the human feedback preference model; and (4) cluster weights are updated and new set of critical states are sampled to learn about NSEs, until the querying budget expires. The gathered NSE information is mapped to a penalty function and augmented to the agent's model to compute an NSE-minimizing policy to complete its task. Empirical evaluation on three domains in simulation demonstrate the effectiveness of our approach in learning to mitigate NSEs from explicit and implicit forms of feedback.

2 Problem Formulation

Consider an agent operating in an environment modeled as a Markov decision process (MDP), using its acquired model $M = \langle S, A, T, R_T \rangle$. The agent optimizes the completion of its assigned task, which is its primary objective described by reward R_T. A *primary policy* is an optimal policy for the agent's primary objective. Similar to Saisubramanian et al. (2021a), we assume that the agent's model M has all the necessary information for the agent to successfully complete its assigned task but lacks other superfluous details that are unrelated to the task. As the model is incomplete in ways unrelated to the primary objective, executing the primary policy produces NSEs that are difficult to identify at design time (Saisubramanian et al., 2021a; Zhang et al., 2020b). Similar to Saisubramanian et al. (2021a), we define NSEs as the immediate, undesired, unmodeled effects of an agent's actions on the environment.

We focus on settings where the agent has no prior knowledge about the NSEs of its actions or the underlying true NSE penalty function R_N. It learns to avoid NSEs by learning a penalty function $\hat{R_N}$ from human feedback that is consistent with R_N. The agent computes an NSE-minimizing policy to complete its task by optimizing the reward function, $R(s, a) = \theta_1 R_T(s, a) + \theta_2 \hat{R_N}(s, a)$, where θ_1 and θ_2 are fixed, tunable weights.

Learning $\hat{R_N}$ from multiple forms of feedback Unlike existing approaches that learn from a single feedback format (Ramakrishnan et al., 2020; Saisubramanian et al., 2021a; Saran et al., 2021), we target real-world settings where the human can provide feedback about NSEs in many forms, and the agent can query for feedback by specifying the format, such as requesting for action approval or demonstrations.

The human's *feedback preference model* is denoted by $D = \langle \mathcal{F}, \psi, C \rangle$ where,

- $\mathcal{F}$ is a predefined set of all feedback formats that the human can provide, for example, demonstrations and corrections;
- $\psi : \mathcal{F} \rightarrow [0, 1]$ is the probability of receiving feedback in a format f, denoted as $\psi(f)$;
- $C : \mathcal{F} \rightarrow \mathbb{R}$ is a cost function that assigns a cost to each feedback format f, representing the human's time or cognitive load involved in providing that feedback.

Abstracting user feedback preferences into probabilities and costs enables generalizing the preferences across similar tasks. We take the pragmatic stance that ψ is independent of time and state, denoting the user's preference about a format such as not preferring formats that require constant supervision of agent performance. While this can be relaxed and the approach can be extended to account for state-dependent preferences, getting an accurate state-dependent ψ could be challenging in practice.

We assume that the agent has access to the human's feedback preference model. This model may be provided by the user or learned by the agent via interactions with the user over time. In this paper, we assume it is user provided. Human feedback is immediate and accurate, when available. Given the human feedback preference model, *when* and in *what format* should the agent seek feedback to effectively learn about NSEs?

Adaptive Feedback Selection Our framework for *adaptive feedback selection* (AFS) enables the agent to query for feedback in specific formats in states that reveal important information about NSE severities. An instance of AFS is denoted by $L = \langle M, D \rangle$, where M is the agent's decision making model and D is the user's feedback preference model.

When the agent asks for a feedback in a particular state in a specific format f, the human may either provide it immediately or provide no feedback, corresponding to $\psi(f)$. We simulate the feedback for a state-action pair using a softmax action selection. The probability of choosing an action a' from a set of all safe actions A^* in state s is, $\Pr(a'|s) = \frac{e^{Q(s,a')}}{\sum_{a \in A^*} e^{Q(s,a)}}$ (Ghosal et al., 2023; Jeon et al., 2020).

2.1 Feedback Formats Studied

While our approach supports a wide range of implicit and explicit feedback formats, we present the following six formats, each providing different level of detail about NSEs. For simplicity, we assume an action in a state may cause mild NSEs, severe NSEs, or no NSEs (Saisubramanian et al., 2021a; Srivastava et al., 2023). In practice, our approach can be applied to settings with any number of NSE categories, provided the feedback formats align with it.

Approval (App): The agent randomly selects N state-action pairs from all possible actions in critical states and queries the human for approval or disapproval. Approved actions are labeled as acceptable ($\hat{R_N}(s, a) = l_a$), while disapproved actions are labelled as unacceptable ($\hat{R_N}(s, a) = l_u$). l_a and l_u denote problem-specific NSE penalty values.

Annotated Approval (Ann. App): An extension of the Approval approach where the human specifies the *NSE severity* (or category) for each disapproved action in the critical states. Actions causing mild NSEs are mapped to $\hat{R_N}(s, a) = l_m$ and severe NSEs are mapped to $\hat{R_N}(s, a) = l_h$.

Demo-Action Mismatch (DAM): The human demonstrates a safe action in each critical state, which the agent compares with its primary policy. Mismatched actions in its primary policy are labelled as unacceptable ($\hat{R_N}(s, a) = l_u$), and matched actions are labelled as acceptable ($\hat{R_N}(s, a) = l_a$). Thus, this format cannot represent varying NSE severities.

Corrections (Corr): The agent performs a trajectory of its primary policy, under human supervision. If the agent's action is unacceptable, then the human intervenes with an acceptable action

in that state. If all actions in a state lead to NSE, the human specifies an action with the least NSE. When interrupted, the agent assumes all actions except the correction are unacceptable in that state, $\hat{R_N}(s,a) = l_u$. Acceptable actions are mapped to $\hat{R_N}(s,a) = l_a$. While this format informs about acceptable and unacceptable actions, it does not indicate the NSE severity.

Annotated Corrections (Ann. Corr): An extension of Corrections approach where the human specifies the severity of NSEs caused by the agent's unacceptable action in the critical states. Actions leading to mild and severe NSEs are mapped to $\hat{R_N}(s,a) = l_m$ and $\hat{R_N}(s,a) = l_h$ respectively, and acceptable actions to $\hat{R_N}(s,a) = l_a$.

Gaze: In this implicit feedback format, the agent compares its action outcomes with the gaze positions of the user (Saran et al., 2021). Actions with outcomes opposite in direction to the human's mean gaze direction are labeled as unacceptable, $\hat{R_N}(s,a) = l_u$, and actions aligning with the mean gaze direction are labeled as acceptable, $\hat{R_N}(s,a) = l_a$.

3 Solution Approach

Prior works focus on learning from a single feedback format, which can be inefficient due to sampling biases (Saisubramanian et al., 2022) and the agent may benefit significantly by leveraging multiple forms of information. Our experiments show that learning about NSEs from two formats combined is more effective than learning from a single format, for example, DAM+Corrections outperforms DAM alone (Fig. 5). The results also suggest that the *order* of the feedback formats can influence the agent's overall performance, demonstrating the need for a principled approach to select formats and their order for efficient learning. Our adaptive feedback selection (AFS) framework enables the agent to query for feedback in critical states using formats that accelerate agent learning.

3.1 Feedback Format Selection

Let there exist a true underlying NSE distribution that the human knows. Human feedback, when available, is sampled from this true distribution. Let p denote the distribution of state-action pairs causing NSEs, consistent with the human *feedback received so far*. That is, p denotes the aggregate NSE information provided by the human so far. As the agent has no prior knowledge about NSEs, it assumes that its actions do not cause NSEs, unless a feedback indicates otherwise. Let q denote the agent's *current* NSE distribution that is learned from p. We define the information gain of a format f as the inverse of the Kullback-Leibler (KL) divergence between p and q, over a set of N critical states Ω. Hence, a *lower* value indicates that feedback f helps the agent learn about NSEs. We refer to this inverse KL divergence value as *information divergence* and is calculated as follows,

$$\mathcal{V}_f = \frac{1}{N} \sum_{s \in \Omega} D_{KL}(p \parallel q) \tag{1}$$

$$= \frac{1}{N} \sum_{s \in \Omega} \sum_{a \in A} p(a|s) \cdot \log \left(\frac{p(a|s)}{q(a|s)} \right) \tag{2}$$

The agent selects the most informative feedback format f^*, given its knowledge of each format's information gain, cost and probability of receiving it, using the following equation,

$$f^* = \underset{f \in \mathcal{F}}{\operatorname{argmax}} \frac{\psi(f)}{\mathcal{V}_f \cdot \mathrm{C}(f)} + \sqrt{\frac{log\ t}{n_f + \epsilon}} \tag{3}$$

where $\psi(f)$ is the probability of receiving a feedback in format f and $C(f)$ is the feedback cost, determined using the human preference model D, $\mathcal{V}_f$ is the information divergence of f calculated using Eqn. 2. t denotes the current learning iteration, n_f is the number of times f was received, and ϵ is a small value added for numeric stability. This approach effectively balances the trade-off between selecting previously used feedback formats and examining unexplored formats.

Alg. 1 outlines our NSE learning approach. The agent initializes a safe action distribution across all states in S, based on its knowledge of NSEs (Line 1). All actions are considered safe initially

Algorithm 1 NSE learning approach

Require: B, Budget; D, Human preference model; N, No. of critical states to sample
1: $t \leftarrow 1$; $\mathcal{V} \leftarrow \mathbf{0}$; $p \leftarrow$ Agent's distribution of safe actions
2: **while** $B > 0$ **do**
3: Sample N critical states using Alg. 2, $\Omega = \{s_1, ..., s_N\}$
4: Select feedback format f^* for querying, using sampled Ω and Eqn. 3
5: **if** feedback received in format f^* **then**
6: $p \leftarrow$ Update distribution based on the received feedback f^*
7: $\mathcal{P} \leftarrow$ TrainClassifier(p)
8: $q \leftarrow \{\mathcal{P}(s,a), \forall s \in \Omega, \forall a \in A\}$
9: $\mathcal{V}_{f^*} \leftarrow \frac{1}{N} \sum_{s \in \Omega} D_{KL}(p \parallel q)$
10: $n_{f^*} \leftarrow n_{f^*} + 1$
11: $B \leftarrow B - C(f^*)$; $t \leftarrow t + 1$
12: **return** NSE classifier model, $\mathcal{P}$

when the agent has no prior knowledge of NSEs. A set of N critical states are sampled using Alg. 2 (Line 3). Feedback format, f^*, that maximizes the information gain in these states is identified using Eqn. 3. The agent queries the human for feedback in format f^* (Line 4). The human provides it with probability $\psi(f^*)$. If feedback is received, the agent updates the distribution, p, based on the new NSE information (Line 5-6). The agent trains an NSE prediction model, $\mathcal{P}$ using p (Line 7). An NSE distribution q is derived for all the actions in the critical states, from $\mathcal{P}$ (Line 8). Information divergence $\mathcal{V}_{f^*}$ is updated using Eqn. 2 and n_{f^*} is incremented (Lines 9-10). The algorithm terminates when the query budget is exhausted, and outputs a model of NSEs.

3.2 Critical States Selection

To effectively learn from human feedback under a limited querying budget, the agent must identify critical states for learning—states where human feedback is pivotal to learning a good predictive model of NSEs. We define critical states as states in which receiving a feedback maximizes the agent's information gain about NSEs.

Alg. 2 outlines our approach to select critical states in each learning iteration of Alg. 1. The algorithm begins by clustering the state space S into $\mathcal{K}$ clusters, based on state features (Line 2). As NSEs do not occur at random and are correlated with state features, clustering allows the agent to group states that potentially lead to similar NSE severity. In our experiments, we use KMeans clustering algorithm with Jaccard distance over state features to measure the distance between states. In practice, any clustering algorithm can be used, including manual clustering by users.

In every iteration t of Alg. 1, the agent assigns a weight w_k to each cluster, proportional to the new information about NSEs that the current informative format f^* reveals, quantified by information divergence. Clusters are assigned equal weights when there is no prior feedback (Line 4). The cluster weight determines the number of states n_k to be sampled from it. At least one state is sampled from each cluster so that there is sufficient information to calculate the information gain for every cluster (Line 5). The agent randomly samples n_k states from the corresponding cluster and adds them to the critical state set Ω (Lines 6, 7). If the total number of critical states sampled is less than the required number due to rounding of values, then the remaining number of states N_r are sampled from the cluster with the highest weight and added to Ω (Lines 9-11). The information gained from sampled states in cluster k at iteration t is calculated using,

$$IG(k)^t = \frac{1}{|\Omega_k^{t-1}|} \sum_{s \in \Omega_k^{t-1}} D_{KL}(p \parallel q^{t-1}) \tag{4}$$

$$= \frac{1}{|\Omega_k^{t-1}|} \sum_{s \in \Omega_k^{t-1}} \sum_{a \in A} p(a|s) \cdot \log \left(\frac{p(a|s)}{q^{t-1}(a|s)} \right) \tag{5}$$

Algorithm 2 Critical States Selection Approach

Require: N, Number of critical states; $\mathcal{K}$, Number of clusters

$\Omega \leftarrow \emptyset$

Cluster states into $\mathcal{K}$ clusters, $K = \{k_1, \ldots, k_{\mathcal{K}}\}$

for each cluster $k \in K$ **do**

$$W_k \leftarrow \begin{cases} \frac{1}{\mathcal{K}}, & \text{if no feedback received in any iteration} \\ \frac{IG(k)}{\sum_{k \in K} IG(k)}, & \text{if feedback received (using Eqn. 5 for } IG(k)) \end{cases}$$

$n_k \leftarrow \max(1, \lfloor W_k \cdot N \rfloor)$

Sample n_k states at random, $\Omega_k \leftarrow \text{Sample}(k, n_k)$

$\Omega \leftarrow \Omega \cup \Omega_k$

$N_r \leftarrow N - |\Omega|$

if $N_r > 0$ **then**

$k' \leftarrow \arg\max_{k \in K} W_k$

$\Omega \leftarrow \Omega \cup \text{Sample}(k', N_r)$

return Set of selected critical states Ω

where, the information divergence between p and q^{t-1} is calculated over a set of critical states sampled at previous iteration $t-1$, Ω_k^{t-1}. The NSE distribution based on the feedback received until t is denoted by p, and q^{t-1} is the NSE distribution across all actions in the critical states, derived from the prediction model learned at iteration $t-1$. A high information divergence value implies that the feedback received reveals new information about NSEs, reflected via p. The cluster weights are updated to reflect the information divergence value of that cluster.

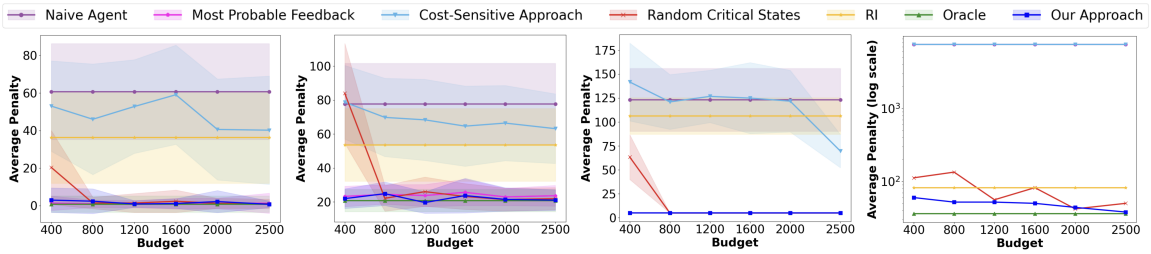
3.3 Model Learning

The gathered feedback is generalized to unseen situations by training a random forest classifier (RF) model to predict NSE severity of an action in a state. Any classifier can be used in practice. The model labels a state-action pair as no NSE, mild or severe NSE, associated with penalties l_a, l_m, and l_h respectively. In our experiments, the penalties are $l_a = +1$, $l_m = +5$, $l_h = +10$, and $l_u = +10$. Hyperparameters for training are determined by a randomized search in the RF parameter space, using three-fold cross validation and selecting parameters with the least mean squared error for training and subsequently to determine the penalty $\hat{R_N}$.

4 Experimental Setup

Baselines (i) *Naive Agent Policy*: The agent naively executes its primary policy without learning about NSEs, providing an upper bound on the NSE penalty incurred. (ii) *Oracle*: The agent has complete knowledge about R_T and R_N, providing a lower bound on the NSE penalty incurred. (iii) *Reward Inference with β Modeling (RI)* (Ghosal et al., 2023): The agent selects a feedback format that maximizes information gain according to the human's inferred rationality β. (iv) *Cost-Sensitive Approach*: The agent selects a feedback method with the least cost, according to the preference model D. (v) *Most-Probable Feedback*: The agent selects a feedback format that the human is most likely to provide, based on D. (vi) *Random Critical States*: The agent uses our AFS framework to learn about the NSEs, but the states are sampled randomly from the entire state space. We implement AFS with a learned $\hat{R_N}$, $\theta_1 = 1$ and $\theta_2 = 1$.

Metrics and Feedback Formats We evaluate the performance of various techniques on three domains in simulation: vase, boxpushing, and Atari freeway. We optimize costs (negations of rewards) and compare techniques using average NSE penalty and average cost to goal, averaged over 100 trials. The learned NSE models are evaluated using F1 score and prediction accuracy. For vase and boxpushing, we simulate explicit human feedback formats. For Atari we use both explicit (demonstration) and implicit (gaze) feedback from the Atari-HEAD dataset (Zhang et al., 2020a).



(a) Vase:avoidable NSE (b) Vase:unavoidable NSE (c) Boxpushing (d) ATARI Freeway

Figure 2: Average penalty incurred by the agent when feedback is selected using different techniques.

Vase This gridworld environment requires the agent to quickly reach a goal state (Krakovna et al., 2020). A state is represented as $\langle x, y, v, c \rangle$ where, x and y are the agent's coordinates, v indicates the presence of a vase and c indicates if the floor is carpeted. The agent can move in all four directions and each costs +1. Actions succeed with probability 0.8, and if they fail, the agent moves in one of the other directions. Penalty for breaking a vase on a carpet (mild NSE) is +5 and for breaking a vase not on a carpet (severe NSE) is +10. All other cases do not create NSEs. The state features used for training are $\langle v, c \rangle$. Both avoidable and unavoidable NSE settings are considered.

Boxpushing In this domain, the agent aims to push the box quickly to a goal state (Saisubramanian et al., 2021a). A state is represented as $\langle x, y, b, w, c \rangle$ where, x and y are the agent's coordinates, b indicates if the agent carries a box, w indicates if the box is wrapped and c indicates if the floor is carpeted. The agent can move in all four directions and wrap a box, and each costs +1. Actions succeed with probability 0.8, otherwise, the agent moves in one of the other directions. Penalty for pushing an unwrapped box over wooden surface (mild NSE) is +5 and over a carpeted surface (severe NSE) is +10. Other cases have no NSE. $\langle b, w, c \rangle$ are used for training. NSEs in this setting are always avoidable by wrapping the box before pushing.

Atari Freeway In this Atari game, the agent (a chicken) navigates ten cars moving at varying speeds to reach the destination quickly while avoiding being hit. Being hit moves the agent back to its previous position. A game state is defined by coordinates (x_1, y_1) and (x_2, y_2), i.e., the top left and bottom right corners of the agent and cars, extracted the Atari-HEAD dataset (Zhang et al., 2020a). Only car coordinates within a specific range of the agent are considered (Saran et al., 2021). The agent can move up, down or stay in place, with deterministic transitions. Move actions cost +1 and colliding with a car within 5-pixel radius of the agent incurs +10.

5 Results and Discussion

Effect of feedback cost and probability Two key factors in our framework for selecting feedback format are cost and probability. We compare the impact of optimizing these factors individually on mitigating NSEs (Figure 2). In our experiments on vase and boxpushing domains, Demo-Action Mismatch has the least cost and Correction has high probability of being received. The Cost-Sensitive approach will consistently choose Demo-Action Mismatch, while the Most Probable Feedback will always select Corrections. Our approach mitigates NSEs while balancing cost and probability, by selecting formats based on information gain in Eqn. 3. The appendix includes additional plots showing the frequency of selecting a format using our approach, under varying budget. We also tested a case with uniform feedback cost and uniform probability. In such cases, our framework consistently selects the most informative format (results in appendix).

NSE mitigation and task completion There is a trade-off between optimizing task completion and mitigating NSEs, especially when NSEs are unavoidable. While some techniques are better at mitigating NSEs, they significantly impact task performance. Fig. 2 shows the average NSE penalty of different techniques, and Table 1 shows the average total cost incurred upon task completion. The Naive Agent policy, with lower average cumulative cost compared to other strategies, incurs the highest NSE penalty as it operates based on R_T and has no knowledge of R_N. The RI policy has better task performance but causes more NSEs when they are unavoidable. In the avoidable NSEs

Method	Vase: avoidable NSE	Vase: unavoidable NSE	Boxpushing: avoidable NSE	Freeway: avoidable NSE
Oracle	37.42 ± 4.07	53.72 ± 5.95	44.62 ± 9.97	3759.8 ± 0.0
Naive	35.84 ± 3.04	36.0 ± 2.89	39.82 ± 5.44	61661.0 ± 0.0
RI	39.28 ± 4.05	37.44 ± 3.31	59.91 ± 19.63	71716.57 ± 0.0
Ours	38.79 ± 6.35	52.81 ± 5.62	43.75 ± 3.32	1726.5 ± 0.0

Table 1: Average cost for task completion, along with standard errors.

setting, RI mitigates a smaller number of NSEs and incurs high costs, as its reward function does not fully model the penalties for mild and severe NSEs. Additional results on the number of mild and severe NSEs encountered are included in the appendix.

We discuss in detail our method's effectiveness in learning an NSE prediction model and compare the effects of learning from single and multiple feedback types in the appendix.

6 Related Works

Negative Side Effects The problem of avoiding negative side effects is gaining increasing attention (Krakovna et al., 2018; Saisubramanian et al., 2021a;b; Zhang et al., 2020b; Klassen et al., 2022; Srivastava et al., 2023). Different notions of side effects have been addressed, such as undesired changes to the environment during operation (Krakovna et al., 2018; Saisubramanian et al., 2021a; Saisubramanian & Zilberstein, 2021), affecting the ability to perform future tasks (Krakovna et al., 2020), and negatively impacting the behavior of other agents in the environment (Alizadeh Alamdari et al., 2022; Klassen et al., 2022). Our focus is on side effects due to model incompleteness that affect the environment but not the agent's ability to complete its task. Our approach is akin to Saisubramanian et al. (2021a) in learning a penalty function for NSEs from human feedback, but we allow for learning from multiple feedback types.

Learning from Human Feedback Human feedback is a popular approach for training agents when reward functions are unavailable (Abbeel & Ng, 2004; Ng et al., 2000; Pomerleau, 1988; Ross et al., 2011). It has been widely used to improve the safety and reliability of agent operation (Hadfield-Menell et al., 2017; Bajcsy et al., 2017; Brown et al., 2018; Ramakrishnan et al., 2020; Zhang et al., 2020b; Brown et al., 2020b;c; Saisubramanian et al., 2021a). Recent works explore various forms feedback for reward learning, including demonstrations (Ramachandran & Amir, 2007; Brown & Niekum, 2018), corrections (Losey & O'Malley, 2018; Bobu et al., 2021; Cui et al., 2023), critiques (Cui & Niekum, 2018; Saisubramanian et al., 2021a), ranking (Brown et al., 2019; 2020a), and implicit feedback like facial expressions and gestures (Cui et al., 2021; Xu et al., 2020).

Existing literature focuses on a single form of feedback for agent learning, limiting the efficiency of learning. Ibarz et al. (2018) and Bıyık et al. (2022) employ two feedback formats, demonstrations and preferences, to learn a reward function. Their approach shows that it is more efficient than using a single format. However, they assume that the order of these formats are predetermined, and the approach does not scale well if the human is willing to provide more than two feedback formats. Recently Ghosal et al. (2023) proposed a method to estimate the human's ability in providing feedback using the Boltzmann rationality model. Their method focuses on a single feedback setting, where a feedback format is selected based on the rationality of the user providing feedback. Unlike their approach, we dynamically select the most informative feedback and do not require pre-processing.

7 Summary and Future Work

We propose an adaptive feedback selection (AFS) framework to learn about negative side effects (NSEs) from diverse forms of human feedback. Our algorithm identifies critical states for learning about NSEs and selects the most informative feedback format, considering the cost and probability of each format. Experimental results on three domains show our approach's effectiveness in learning to mitigate avoidable and unavoidable NSEs, from explicit and implicit feedback. In the future, we aim to learn a human preference model, such as in a calibration phase and validate using user studies.

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 1, 2004.

Parand Alizadeh Alamdari, Toryn Q Klassen, Rodrigo Toro Icarte, and Sheila A McIlraith. Be considerate: Avoiding negative side effects in reinforcement learning. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pp. 18–26, 2022.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety, 2016.

Andrea Bajcsy, Dylan P. Losey, Marcia K. O'Malley, and Anca D. Dragan. Learning robot objectives from physical human interaction. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg (eds.), *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 217–226. PMLR, 13–15 Nov 2017. URL https://proceedings.mlr.press/v78/bajcsy17a.html.

Erdem Bıyık, Dylan P Losey, Malayandi Palan, Nicholas C Landolfi, Gleb Shevchuk, and Dorsa Sadigh. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research*, 41(1):45–67, 2022.

Andreea Bobu, Marius Wiggert, Claire Tomlin, and Anca D Dragan. Feature expansive reward learning: Rethinking human input. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 216–224, 2021.

Daniel Brown and Scott Niekum. Efficient probabilistic performance bounds for inverse reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Daniel Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Niekum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International conference on machine learning*, pp. 783–792. PMLR, 2019.

Daniel Brown, Russell Coleman, Ravi Srinivasan, and Scott Niekum. Safe imitation learning via fast bayesian reward inference from preferences. In *International Conference on Machine Learning*, pp. 1165–1177. PMLR, 2020a.

Daniel Brown, Russell Coleman, Ravi Srinivasan, and Scott Niekum. Safe imitation learning via fast Bayesian reward inference from preferences. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1165–1177. PMLR, 13–18 Jul 2020b. URL https://proceedings.mlr.press/v119/brown20a.html.

Daniel Brown, Scott Niekum, and Marek Petrik. Bayesian robust optimization for imitation learning. *Advances in Neural Information Processing Systems*, 33:2479–2491, 2020c.

Daniel S. Brown, Yuchen Cui, and Scott Niekum. Risk-aware active inverse reinforcement learning. In Aude Billard, Anca Dragan, Jan Peters, and Jun Morimoto (eds.), *Proceedings of The 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pp. 362–372. PMLR, 29–31 Oct 2018. URL https://proceedings.mlr.press/v87/brown18a.html.

Yuchen Cui and Scott Niekum. Active reward learning from critiques. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 6907–6914. IEEE, 2018.

Yuchen Cui, Qiping Zhang, Brad Knox, Alessandro Allievi, Peter Stone, and Scott Niekum. The empathic framework for task learning from implicit human feedback. In *Conference on Robot Learning*, pp. 604–626. PMLR, 2021.

Yuchen Cui, Siddharth Karamcheti, Raj Palleti, Nidhya Shivakumar, Percy Liang, and Dorsa Sadigh. No, to the right – online language corrections for robotic manipulation via shared autonomy. In *Proceedings of the 2023 ACM/IEEE Conference on Human-Robot Interaction (HRI)*, 2023.

Gaurav R Ghosal, Matthew Zurek, Daniel S Brown, and Anca D Dragan. The effect of modeling human rationality level on learning rewards from multiple feedback types. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 5983–5992, 2023.

Dylan Hadfield-Menell, Smitha Milli, Pieter Abbeel, Stuart J Russell, and Anca Dragan. Inverse reward design. *Advances in neural information processing systems*, 30, 2017.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. *Advances in neural information processing systems*, 31, 2018.

Hong Jun Jeon, Smitha Milli, and Anca Dragan. Reward-rational (implicit) choice: A unifying formalism for reward learning. *Advances in Neural Information Processing Systems*, 33:4415–4426, 2020.

Toryn Q. Klassen, Sheila A. McIlraith, Christian Muise, and Jarvis Xu. Planning to avoid side effects. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36:9830–9839, Jun. 2022. doi: 10.1609/aaai.v36i9.21219. URL https://ojs.aaai.org/index.php/AAAI/article/view/21219.

Victoria Krakovna, Laurent Orseau, Miljan Martic, and Shane Legg. Measuring and avoiding side effects using relative reachability. *CoRR*, abs/1806.01186, 2018. URL http://arxiv.org/abs/1806.01186.

Victoria Krakovna, Laurent Orseau, Richard Ngo, Miljan Martic, and Shane Legg. Avoiding side effects by considering future tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 19064–19074, 2020.

Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Eric Horvitz. Identifying unknown unknowns in the open world: Representations and policies for guided exploration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Robert Loftin, Bei Peng, James MacGlashan, Michael L. Littman, Matthew E. Taylor, Jeff Huang, and David L. Roberts. Learning something from nothing: Leveraging implicit human feedback strategies. In *The 23rd IEEE international symposium on robot and human interactive communication*, pp. 607–612. IEEE, 2014.

Dylan P Losey and Marcia K O'Malley. Including uncertainty when learning from human corrections. In *Conference on Robot Learning*, pp. 123–132. PMLR, 2018.

Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, pp. 2, 2000.

Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.

Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pp. 2586–2591, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

Ramya Ramakrishnan, Ece Kamar, Debadeepta Dey, Eric Horvitz, and Julie Shah. Blind spot detection for safe sim-to-real transfer. *Journal of Artificial Intelligence Research*, 67:191–234, 2020.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Sandhya Saisubramanian and Shlomo Zilberstein. Mitigating negative side effects via environment shaping. In Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé (eds.), *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, pp. 1640–1642. ACM, 2021. doi: 10.5555/3463952.3464186. URL https://www.ifaamas.org/Proceedings/aamas2021/pdfs/p1640.pdf.

Sandhya Saisubramanian, Ece Kamar, and Shlomo Zilberstein. A multi-objective approach to mitigate negative side effects. In *Proceedings of the 29th international conference on international joint conferences on artificial intelligence*, pp. 354–361, 2021a.

Sandhya Saisubramanian, Shannon C. Roberts, and Shlomo Zilberstein. Understanding user attitudes towards negative side effects of AI systems. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–6, 2021b.

Sandhya Saisubramanian, Ece Kamar, and Shlomo Zilberstein. Avoiding negative side effects of autonomous systems in the open world. *Journal of Artificial Intelligence Research*, 74:143–177, 2022.

Akanksha Saran, Ruohan Zhang, Elaine Schaertl Short, and Scott Niekum. Efficiently guiding imitation learning agents with human gaze. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2021.

Aishwarya Srivastava, Sandhya Saisubramanian, Praveen Paruchuri, Akshat Kumar, and Shlomo Zilberstein. Planning and learning for non-markovian negative side effects using finite state controllers. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence (AAAI)*, 2023.

Duo Xu, Mohit Agarwal, Faramarz Fekri, and Raghupathy Sivakumar. Playing games with implicit human feedback. In *Workshop on Reinforcement Learning in Games, AAAI*, volume 6, 2020.

Ruohan Zhang, Calen Walshe, Zhuode Liu, Lin Guan, Karl Muller, Jake Whritner, Luxin Zhang, Mary Hayhoe, and Dana Ballard. Atari-head: Atari human eye-tracking and demonstration dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 6811–6820, 2020a.

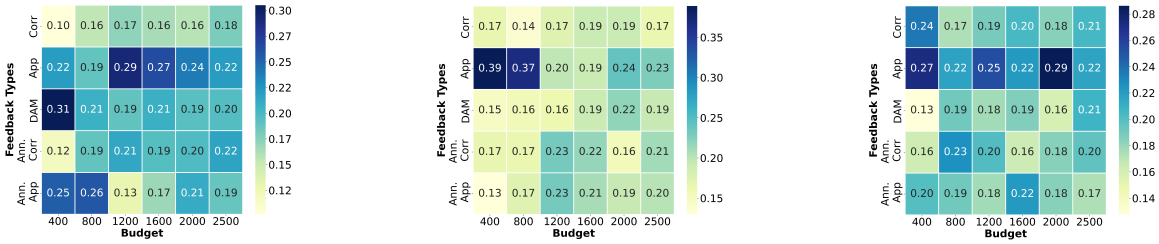
Shun Zhang, Edmund Durfee, and Satinder Singh. Querying to find a safe policy under uncertain safety constraints in markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 2552–2559, 2020b.

A Appendix

This appendix includes additional results evaluating the effect of feedback cost and probability, comparing the performance of single and multiple feedback approaches, and assessing the effectiveness of our approach in learning an NSE prediction model.

A.1 Effect of feedback cost and probability

We examine the influence of cost and probability associated with different feedback formats on the frequency at which the agent selects each format, with our AFS learning framework.



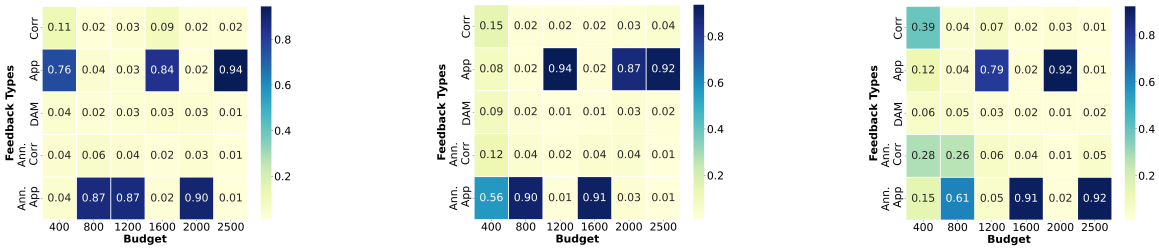
(a) Vase: avoidable NSE (b) Vase: unavoidable NSE (c) Boxpushing

Figure 3: Frequency of selecting a feedback format, with varying budget for querying.

Feedback Type	Probability	Cost
Corrections	0.70	8
Approval	0.50	9
Demo AM	0.60	5
Annotated Corr.	0.65	6
Annotated App.	0.60	7

Table 2: Cost and probability values for each feedback type in vase and boxpushing domains.

Figure 3 shows how frequently the agent selects each feedback format, normalized and presented across different budget values. Table 2 shows the cost and probability values corresponding to each feedback format used during the feedback selection process. In this case, the agent consistently selects more informative feedback formats, such as Approval, Annotated Corrections and Annotated Approval, across most budget values.



(a) Vase: avoidable NSE (b) Vase: unavoidable NSE (c) Boxpushing

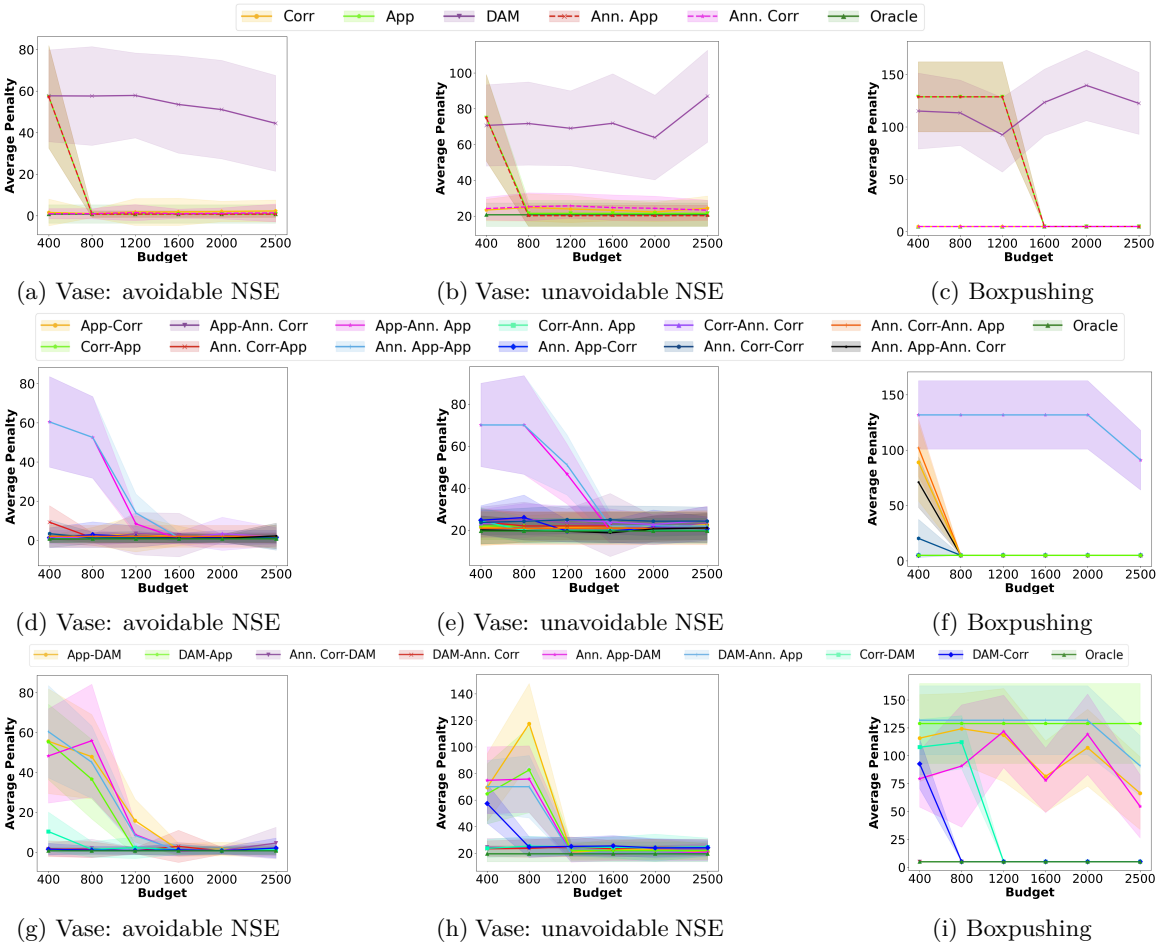
Figure 4: Frequency of our approach selecting a feedback format, under uniform feedback cost and probability for all formats.

When the feedback probability and costs are uniform for all the formats, the agent predominantly selects either Approval or Annotated Approval (Figure 4). In both cases, the agent initially explores and learns from the different feedback formats. With a higher querying budget, the agent learns

from the most informative format in later learning iterations. While Annotated Corrections provide information on both correct and incorrect actions in a state, this format is utilized primarily in the early learning iterations. Approval and Annotated Approval, on the other hand, are consistently selected in the later learning iterations, as they can provide information about every state-action pair, not just restricted to the agent's current policy as in corrections. This assists the agent in learning more fine-grained information about the association of NSEs with each state-action pair.

A.2 Effect of learning from multiple feedback types

We examine the benefit of learning from more than one feedback type, by comparing the average NSE penalties of learning from a single feedback and multiple feedback formats (Figure 5), with varying budget for querying. In the single feedback case (Figure 5 (a-c)), Corrections format successfully mitigates NSEs with fewer feedback across domains. However, its reliance on constant human guidance is a limitation. While Demo-Action Mismatch requires less human guidance, it is less effective in avoiding NSEs. The effectiveness of Demo-Action Mismatch improves significantly depending on its position within a sequence of feedback formats (Figure 5 (d-f)). For instance, using Demo-Action Mismatch before Corrections, in the avoidable NSE setting of vase and boxpushing domains, results in a lower average penalty with a smaller budget. However, in the vase domain with unavoidable NSEs, the agent performs better when Demo-Action Mismatch follows Corrections.



(a) Vase: avoidable NSE (b) Vase: unavoidable NSE (c) Boxpushing

(d) Vase: avoidable NSE (e) Vase: unavoidable NSE (f) Boxpushing

(g) Vase: avoidable NSE (h) Vase: unavoidable NSE (i) Boxpushing

Figure 5: Average penalty incurred, along with standard error, when learning from a single feedback (a-c), and using combinations of two formats (d-i).

On the other hand, Approval and Annotated Approval have a similar performance across domains and require higher samples to learn the true distribution of NSEs. However, when combined with Corrections or Annotated Corrections, the performance improves considerably (Figure 5 (g-i)). Notably, in the boxpushing domain, the order of these formats affect the agent's performance — using Corrections before Approval has a better performance compared to using Approval before Corrections. Learning the underlying NSE severities demands a significantly higher number of samples when using a combination of Approval and Annotated Approval formats. These results show that learning from more than one feedback format is generally useful but the benefits depend on the formats considered together and the *order* in which they are combined. Identifying the right ordering of feedback formats manually is practically infeasible. Our AFS framework enables the agent to learn and mitigate most NSEs effectively, by automatically selecting effective feedback formats in every learning iteration (Figure 2).

A.3 Learning NSE model

We evaluate the effectiveness of our approach in learning to predict NSEs using F1 scores for each NSE category and overall prediction accuracy, compared to learning from a single feedback (Table 3). The results are averaged across three test instances in vase and boxpushing domains, and five instances for Atari freeway. Annotated Corrections format is most effective in predicting NSEs of different severity levels, in both the vase and boxpushing domains. Our AFS framework performs similar to Annotated Corrections in the boxpushing and vase domains. In the Freeway environment, Demo-Action Mismatch has a better performance in terms of the F1 score and accuracy. The F1 scores and accuracy of our approach are similar to that of Demo-Action Mismatch. However, Gaze, a low-cost implicit feedback, has a high overall accuracy but its F1 score for severe NSE is zero. In general, a low F1 score in this domain is a result of a highly imbalanced dataset—there are ∼300,000 states with no NSEs and ∼4000 states with severe NSE, which affects the learning process.

Domain	Method	Average F1 Score (↑)			Average Accuracy % (↑)
		No NSE	Mild	Severe	
Vase: avoidable NSE	Corrections	1.00 ± 0.00	0.72 ± 0.04	0.00 ± 0.00	88.15 ± 1.41
	Annotated Corr.	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	100 ± 0.00
	Approval	0.84 ± 0.05	0.00 ± 0.00	0.00 ± 0.00	70.96 ± 7.5
	Annotated App.	0.84 ± 0.05	0.00 ± 0.00	0.40 ± 0.00	73.93 ± 7.15
	Demo AM	0.82 ± 0.05	0.00 ± 0.00	0.00 ± 0.00	70.96 ± 7.5
	Our approach	1.00 ± 0.00	0.83 ± 0.02	0.67 ± 0.00	94.07 ± 0.70
Vase: unavoidable NSE	Corrections	0.97 ± 0.01	0.76 ± 0.03	0.00 ± 0.00	85.04 ± 0.00
	Annotated Corr.	0.97 ± 0.01	1.00 ± 0.00	0.86 ± 0.00	96.26 ± 0.00
	Approval	0.80 ± 0.04	0.00 ± 0.00	0.00 ± 0.00	65.33 ± 0.06
	Annotated App.	0.80 ± 0.04	0.00 ± 0.00	0.40 ± 0.00	69.07 ± 0.06
	Demo AM	0.79 ± 0.04	0.00 ± 0.00	0.00 ± 0.00	65.33 ± 0.06
	Our approach	0.97 ± 0.00	0.90 ± 0.01	0.67 ± 0.00	92.52 ± 0.00
Boxpushing: avoidable NSE	Corrections	0.94 ± 0.00	0.75 ± 0.04	0.00 ± 0.00	83.26 ± 0.02
	Annotated Corr.	0.94 ± 0.00	0.89 ± 0.00	0.89 ± 0.00	92.18 ± 0.00
	Approval	0.76 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	61.57 ± 0.00
	Annotated App.	0.76 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	61.57 ± 0.00
	Demo AM	0.76 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	61.57 ± 0.00
	Our approach	0.92 ± 0.02	0.99 ± 0.00	0.00 ± 0.00	88.71 ± 0.02
Atari Freeway: avoidable NSE	Gaze	0.99	-	0.00	98.68
	Demo AM	0.87	-	0.03	77.69
	Our approach	0.84	-	0.03	71.98

Table 3: Average F1 score and prediction accuracy of different formats, with standard errors, compared to our approach. All NSEs are severe in the Freeway domain.

A.4 NSE mitigation and task completion

To assess the agent's performance in mitigating NSEs, Table 4 presents the average number of mild and severe NSEs encountered during policy simulation, across various domains and approaches. The results indicate that our approach effectively mitigates both avoidable and unavoidable NSEs, as demonstrated by the significantly lower number of mild and severe NSEs encountered in comparison to the baseline methods.

Domain	NSE	Oracle	Naive	RI	Ours
Vase: avoidable NSE	Mild	0.09 ± 1.00	3.68 ± 3.17	1.44 ± 2.12	0.19 ± 0.65
	Severe	0.03 ± 0.20	4.22 ± 3.99	2.90 ± 3.49	0.2 ± 0.68
Vase: unavoidable NSE	Mild	1.97 ± 1.21	6.78 ± 4.20	1.03 ± 1.13	1.3 ± 0.78
	Severe	1.09 ± 0.95	4.83 ± 3.48	4.84 ± 3.46	1.54 ± 0.88
Boxpushing: avoidable NSE	Mild	1.00 ± 0.00	14.40 ± 5.32	20.21 ± 5.53	1.00 ± 0.00
	Severe	0.00 ± 0.00	5.11 ± 4.05	0.52 ± 1.51	0.00 ± 0.00
Freeway: avoidable NSE	Mild	-	-	-	-
	Severe	3.60 ± 2.54	751.80 ± 0.17	8.2 ± 2.06	6.0 ± 2.6

Table 4: Average #mild and severe NSEs, along with standard errors, with querying budget $B = 400$. All NSEs are severe in the Freeway domain.