TOWARDS ROBUST ACTIVE FEATURE ACQUISITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Truly intelligent systems are expected to make critical decisions with incomplete and uncertain data. Active feature acquisition (AFA), where features are sequentially acquired to improve the prediction, is a step towards this goal. However, current AFA methods lack robustness in two key areas that limits their applicability in the real world. First, current AFA models only consider a small set of candidate features and have difficulty scaling to a large feature space. Second, they are ignorant about the valid domains where they can predict confidently, thus they can be vulnerable to out-of-distribution (OOD) inputs. In order to remedy these deficiencies and bring AFA models closer to practical use, we propose several techniques to advance the current AFA approaches. Our framework can easily handle a large number of features using a hierarchical acquisition policy and is more robust to OOD inputs with the help of an OOD detector for partially observed data. Extensive experiments demonstrate the efficacy of our framework over strong baselines.

1 INTRODUCTION

A typical machine learning system will first collect all the features and then predict the target variables based on the collected feature values. Unlike the two-step paradigm, active feature acquisition performs feature value acquisition and target prediction at the same time. Features are actively acquired to improve the prediction, and the prediction in turn informs the acquisition process. Ideally, only features that provide useful information and outweigh their cost will be acquired. The AFA model will stop acquiring more features when the prediction is sufficiently accurate or it exceeds the given acquisition budget (Li & Oliva, 2020; Shim et al., 2018; Ma et al., 2018). Since each instance could have different set of informative features, active feature acquisition is expected to acquire different features for different instances.

As a motivating example, consider a doctor making a diagnosis on a patient (an instance). The doctor usually has not observed all the possible measurements (such as blood samples, x-rays, etc.) from the patient. The doctor is also not forced to make a diagnosis based on the currently observed measurements; instead, he/she may dynamically decide to take more measurements to help determine the diagnosis. The next measurement to make (feature to observe), if any, will depend on the values of the already observed features; thus, the doctor may determine a different set of features to observe from patient to patient (instance to instance) depending on the values of the features that were observed. Hence, each patient will not have the same subset of features selected (as would be the case with typical feature selection).

In the current literature, there are mainly two types of approaches to acquire features actively: greedy acquisition approaches and reinforcement learning based approaches. Both approaches acquire features sequentially, that is, one candidate feature is acquired at each acquisition step based on the previously observed features. Greedy approaches directly optimize the utility of the next acquisition, while reinforcement learning based approaches optimize a discounted reward along the acquisition trajectories. As a result, the reinforcement learning (RL) based approaches tend to find better solutions to the AFA problem as shown in (Li & Oliva, 2020). Here, we base ourselves on the Markov decision process (MDP) formulation of the AFA problem proposed in (Li & Oliva, 2020; Shim et al., 2018) and focus on resolving the deficiencies of the current AFA models.

One of the obstacles to extending the current AFA models to practical use is the potentially large number of candidate features. Greedy approaches are computationally difficult to scale, because the utilities need to be recalculated for each candidate feature based on the updated set of observed

features at each acquisition step, which incurs an $O(d^2)$ complexity for a d dimensional feature space. Reinforcement learning algorithms are known to have difficulties with a high dimensional action space (Dulac-Arnold et al., 2015). In this work, we propose to cluster the candidate features into groups and use a hierarchical reinforcement learning agent to select the next feature to be acquired.

Another challenge in deploying AFA models is assessing confidence and identifying "in distribution" queries. In a practical application, it is very likely for an AFA model to encounter inputs that are different from its training distribution. For example, it may be asked to acquire features for patients with unknown disease. For those out-of-distribution instances, the AFA model may acquire an arbitrary subset of features and predict one of its known categories. Dealing with out-of-distribution inputs is difficult in general, and is even more challenging for AFA models, since the model only has access to a subset of features at any acquisition step. In this work, we propose a novel algorithm for detecting out-of-distribution inputs with partially observed features, and further utilize it to improve the robustness of the AFA model.

Our contributions are as follows: 1) We propose to reduce the action space for active feature acquisition by grouping similar actions and learn a hierarchical policy to select the next candidate feature to be acquired. 2) We develop a novel out-of-distribution detection algorithm that can distinguish OOD inputs using an arbitrary subset of features. 3) Armed with the partially observed OOD detection algorithm, we encourage the AFA agent to acquire features that are most informative for distinguishing OOD inputs. 4) Our approach achieves the state-of-the-art performance for active feature acquisition, while identifying out-of-distribution inputs.

2 Background and Related Works

2.1 Active Feature Acquisition (AFA)

Typical discriminative models predict a target y using all d-dimensional features $x \in \mathbb{R}^d$. AFA, instead, actively acquires feature values to improve the prediction. It typically starts from an empty set of features and sequentially acquires more features x_i until the prediction is sufficiently accurate or it exceeds the given acquisition budget. The goal of an AFA model is to minimize the following objective

$$\mathcal{L}(\hat{y}(x_o), y) + \alpha \mathcal{C}(o), \tag{1}$$

where $\mathcal{L}(\hat{y}(x_o), y)$ is the prediction error between the groundtruth target y and the prediction $\hat{y}(x_o)$ using the acquired features x_o, $\mathcal{C}(o)$ measures the total cost of acquiring a subset of features $o \subseteq \{1, \dots, d\}$, and α balances these two terms.

However, directly optimizing equation 1 is not trivial, since it involves optimizing over a combinatorial number of possible subsets. Many heuristic approaches have been developed to approximately solve this problem. For example, in (Ling et al., 2004), the authors propose to take into account the cost of features when selecting an attribute for building a decision tree so that the final tree will have a minimum total cost. (Chai et al., 2004) utilizes a naive Bayes classifier to handle the partially observed features, where the unobserved features are simply ignored in the likelihood objective. They then assess the utility of each unobserved feature by their expected reduction of the misclassification cost. At each acquisition step, the feature with highest utility is acquired. Nan et al. (2014) instead leverage a margin-based classifier. An instance is classified by retrieving the nearest neighbors from training set using the partially observed features, and the utility of each unobserved feature is calculated by the one-step ahead classification accuracy. Following the same greedy solution, EDDI (Ma et al., 2018) utilizes modern generative models to handle partially observed instances. Specifically, they propose a partial VAE to model the arbitrary marginal likelihoods $p(x_o)$ (target variable y is concatenated into x and modeled together). Inspired by the experimental design approaches (Bernardo, 1979), they assess the utility of each unobserved feature with their expected information gain to the target variable y, i.e.,

$$\mathcal{U}_i = \mathbb{E}_{p(x_i|x_o)} D_{\mathrm{KL}}[p(y \mid x_o, x_i) \| p(y \mid x_o)]. \tag{2}$$

The feature with highest utility is acquired at each step. Similar to EDDI, Icebreaker (Gong et al., 2019) proposes to use a Bayesian Deep Latent Gaussian model to capture the uncertainty of unobserved features and to assess their utilities. They further extend the problem to actively acquire additional information during training.

Greedy approaches are easy to understand, but they are also inherently flawed, since they are myopic and unaware of the long-term goal of obtaining multiple features that are *jointly* informative. Instead of acquiring features greedily, the AFA problem has been formulated as a Markov Decision Process (MDP) (Zubek et al., 2004; Rückstieß et al., 2011). Therefore, reinforcement learning based approaches can be utilized, where a long-term discounted reward is optimized. In the MDP formulation of the AFA problem, the state is the current observed features, the action is the next feature to acquire, and the reward contains the final prediction reward and the cost of each acquired feature. In (Li & Oliva, 2020) and (Shim et al., 2018), a special action indicating the termination of the acquisition process is also introduced. The agent will stop acquiring more features when it selects the termination action. Specifically, we have

$$s = [o, x_o], \qquad a \in u \cup \phi, \qquad r(s, a) = -\mathcal{L}(\hat{y}(x_o), y)\mathbb{I}(a = \phi) - \alpha\mathcal{C}(a)\mathbb{I}(a \neq \phi), \tag{3}$$

where the state, s, consists of the current acquired feature subset, $o \subseteq \{1, \ldots, d\}$, and their values, x_o. The action, a, is either one of the remaining unobserved features, $u = \{1, \ldots, d\} \setminus o$, or the termination action, ϕ. When a new feature, i, is acquired, the current state transits to a new state following $o \xrightarrow{i} o \cup i$, $x_o \xrightarrow{i} x_o \cup x_i$, and the agent receives the negative acquisition cost of this feature as a reward. If the termination action is selected (i.e., $a = \phi$), the agent makes a prediction based on all acquired features, x_o, and receives a final reward as $-\mathcal{L}(\hat{y}(x_o), y)$.

Given the above MDP formulation, several RL approaches have been explored. (Zubek et al., 2004) fits a transition model using complete data, and then uses the AO* heuristic search algorithm to find an optimal policy. (Rückstieß et al., 2011) utilizes Fitted Q-Iteration to optimize the MDP. (He et al., 2012) and (He et al., 2016) instead employ a imitation learning approach coached by a greedy reference policy. JAFA (Shim et al., 2018) jointly learns an RL agent and a classifier, where the classifier is deemed as the environment to calculate the reward.

Although MDPs are broad enough to encapsulate the active acquisition of features, there are several challenges that limit the success of a naive reinforcement learning approach. In the aforementioned MDP, the agent pays the acquisition cost at each acquisition step but only receives a reward about the prediction after completing the acquisition process. This results in sparse rewards leading to credit assignment problems for potentially long episodes (Minsky, 1961; Sutton, 1988), which may make training difficult. In addition, an agent that is making feature acquisitions must also navigate a complicated high-dimensional action space, as the action space scales with the number of features, making for a challenging RL problem (Dulac-Arnold et al., 2015). To assuage these challenges, GSMRL (Li et al., 2020) proposes a model-based alternative. The key observation of GSMRL is that the dynamics of the above MDP can be modeled by the conditional dependencies among features. That is, the state transitions are based on the conditionals: $p(x_j \mid x_o)$, where x_o is current observed features and x_j is an unobserved feature. Base on this observation, a surrogate model that captures the arbitrary conditionals, $p(x_u, y \mid x_o)$, is employed to assist the agent. Specifically, GSMRL defines an intermediate reward using the information gain of the acquired feature, x_i, at the current acquisition step, i.e., $r_m(s, i) = H(y \mid x_o) - H(y \mid x_o, x_i)$. The entropy terms are estimated using the learned surrogate model. Furthermore, the expected information gain for each candidate acquisition is provided to the agent as side information, i.e.,

$$\mathcal{U}_j = H(y \mid x_o) - \mathbb{E}_{p(x_j|x_o)} H(y \mid x_o, x_j), \quad j \in u. \tag{4}$$

In addition to $\mathcal{U}_j$, the surrogate model can also provide the current prediction $\hat{y}$, the prediction probability, $p(y \mid x_o)$, the imputed values of unobserved features and their uncertainties, $p(x_u \mid x_o)$, as auxiliary information. Armed with the auxiliary information and the intermediate reward, GSMRL alleviates the challenge of a model-free approach and obtains state-of-the-art performance for several AFA problems. Given their established excellence, we use GSMRL as the base model for our robust AFA framework.

2.2 Active Instance Recognition (AIR)

AFA acquires features actively to improve the prediction of a target variable, while some application do not have an explicit target; instead, the features are acquired to improve our understanding of the instance. In GSMRL (Li & Oliva, 2020), the authors propose a task named AIR, where an agent acquires features actively to reconstruct the unobserved features. A similar model-based RL approach is used for AIR, where a dynamics model $p(x_u \mid x_o)$ captures the state transition. The intermediate reward and auxiliary information can be similarly derived by replacing y with $x_{u \setminus i}$ (the

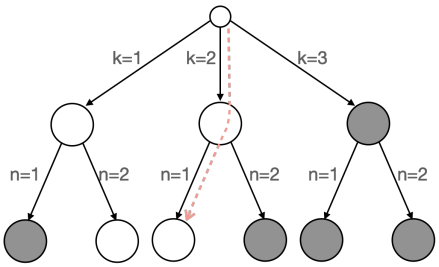


Figure 1: An illustrative example of the grouped action space, where 6 features are grouped into 3 clusters. The grayed circles represent the current observed features (or fully observed groups) and are not considered as candidates anymore. The dashed line shows one acquisition at the current step, which acquires the feature $g_2^{(1)}$. The corresponding circles will be grayed after this acquisition step.

Algorithm 1 Robust Active Feature Acquisition

Input: acquisition environment *env*; dynamics model *M*; partially observed OOD detector *D*; AFA agent *agent*; acquisition budget *B*
Output: reward, ood_likelihood, prediction

x_o, o, reward = *env*.reset()
while $|o| < B$ **do**
  aux = *M*.query(x_o, o)
  action = *agent*.act(x_o, o, aux)
  r_m = *M*.reward(x_o, o, action)
  x_o, o, r_e = *env*.step(action)
  reward += r_e + r_m
end while
aux = *M*.query(x_o, o)
prediction = *agent*.predict(x_o, o, aux)
r_p = *env*.reward(x_o, o, prediction)
r_d = *D*.reward(x_o, o)
reward += r_p + r_d
ood_likelihood = *D*.log_prob(x_o, o)

unobserved features excluding the current candidate i). A special case of AIR is to acquire features in k-space (Fourier frequency domain) for accelerated MRI reconstruction. (Pineda et al., 2020) and (Bakker et al., 2020) have explored this application using deep Q-learning and policy gradient respectively. Several non-RL approaches (Zhang et al., 2019; van Gorp et al., 2021) have also been proposed.

2.3 Out-of-distribution Detection

ML models are typically trained with a specific data distribution, however, when deployed, the model may encounter data that is outside the training distribution. For those out-of-distribution inputs, the prediction could be arbitrarily bad. Therefore, detecting OOD inputs has been an active research direction. A possible approach is to use the uncertainty of prediction, since the prediction for OOD inputs are expected to have higher uncertainty. Bayesian neural networks (BNNs) (Blundell et al., 2015), model ensemble (Lakshminarayanan et al., 2016) and MC dropout based ensemble (Gal & Ghahramani, 2016) have been leveraged to obtain the prediction uncertainties. Another approach is to quantify the distance to the "in distribution" manifold. Representative methods include DUQ (Van Amersfoort et al., 2020), SNGP (Liu et al., 2020) and DUE (van Amersfoort et al., 2021). Generative models have also been used for OOD detection, where the OOD inputs are expected to have lower likelihood (Bishop, 1994). However, recent works (Nalisnick et al., 2019a; Hendrycks et al., 2019) show that it is not the case for high dimensional distributions, and since then many methods have been proposed to rectify this pathology (Choi et al., 2018; Ren et al., 2019; Nalisnick et al., 2019b; Morningstar et al., 2021; Mahmood et al., 2021).

3 Method

In this section, we introduce each component of our framework. We use the model-based AFA approach, GSMRL (Li & Oliva, 2020), as the base model, which utilizes an arbitrary conditional model $p(x_u, y \mid x_o)$ to assist the agent by providing the intermediate rewards and the auxiliary information. We further leverage the arbitrary conditionals to cluster features into groups and develop a hierarchical acquisition policy to deal with the large action space. After, we introduce the OOD detection algorithm for partially observed instances along the acquisition trajectories. We then compose all those components together and propose the robust active feature acquisition framework. For convenience of evaluating OOD detection performance, we do not use the termination action for AFA but specify a budget of the acquisition (i.e., the number of features being acquired).

3.1 Action Space Grouping

As described in Sec. 2.1, the AFA problem can be interpreted as a MDP, where the action space at each acquisition step contains the current unobserved features. For certain problems, the action

space could be enormous. For example, in the aforementioned health care example, the action space could contain an exhaustive list of possible inspections a hospital can offer. Dealing with large action space for RL is generally challenging, since the agent may not be able to effectively explore the entire action space. Several approaches have been proposed to train RL agent with a large discrete action space. For instance, Dulac-Arnold et al. (2015) propose a Wolpertinger policy that maps a state to a continuous proto-action embedding. The proto-action embedding is then used to look up k-nearest valid actions using the given action embeddings. Finally, the action with the highest Q value is selected and executed in the environment. Wolpertinger policy assumes the apriori availability of action representations (embeddings) for k-nearest neighbor searching. However, there is no such representation for general AFA problems. That is, in general datasets, features are enumerated and proximity in indices (i.e., $|i-j|, i, j \in \{1, \dots, d\}$) is typically *not* informative of feature similarity. Majeed & Hutter (2020) propose a sequentialization scheme, where the action space is transformed into a sequence of $\mathcal{B}$-ary decision code words. A pair of bijective encoder-decoder is defined to perform this transformation. Running the agent will produce a sequence of decisions, which are subsequently decoded as a valid action that can be executed in the environment.

Similar to (Majeed & Hutter, 2020), we also formulate the action space as a sequence of decisions. Here, we propose to utilize the inherited clustering properties of the candidate features. Given a set of features, $\{x_1, \dots, x_d\}$, we assume features can be clustered based on their informativeness to the target variable y. That is, there might be a subset of features that are decisive about y and another subset of features that are not relevant to y. Based on this intuition, we propose to assess the informativeness of the candidate features using their mutual information to the target variable, y, i.e., $I(x_i; y)$, where $i \in \{1, \dots, d\}$. The mutual information can be estimated using the learned arbitrary conditionals of the surrogate model

$$I(x_i; y) = \mathbb{E}_{x_i, y} \log \frac{p(x_i, y)}{p(x_i)p(y)} = \mathbb{E}_{x_i, y} \log \frac{p(y \mid x_i)}{p(y \mid \emptyset)}, \tag{5}$$

where the expectation is estimated using a held-out validation set. Given the estimated mutual information, we can simply sort and divide the candidate features into different groups. For the sake of implementation simplicity, we use clusters with the same number of features. We can further group features inside each cluster into smaller clusters and develop a tree-structured action space as in (Majeed & Hutter, 2020), which we leave for future works. Note that the clustering is not performed actively for each instance; instead, we cluster once for each dataset and keep the cluster structure fixed throughout the acquisition process. Our grouping scheme partitions features based on how informative individual features are (marginally, i.e. in isolation) to the target. This acts as an additional form of auxiliary information, which guides the agent in earlier acquisitions (as marginally informative features will be more useful). The grouping shall also help guide the agent in later acquisitions, where it may seek less marginally informative features (that may be *jointly* informative with the current observations) to obtain more nuanced discriminations.

It is worth noting that the mutual information $I(x_i; y)$ is not the only choice for clustering features. Alternative quantities, such as the pairwise mutual information $I(x_i; x_j)$ or a metric $d(x_i, x_j) = H(x_i, x_j) - I(x_i; x_j)$, can be used together with a hierarchical clustering procedure to group candidate features. However, these alternatives need to be estimated for each pair of candidate features, which incurs a $O(d^2)$ computational complexity, while the mutual information, $I(x_i; y)$, only has $O(d)$ complexity.

Given the grouped action space, $\mathcal{A} = \{g_k\}_{k=1}^K$, with K distinct clusters, we develop a hierarchical policy to select one candidate feature at each acquisition step. $g_k = \{g_k^{(1)}, \dots, g_k^{(N)}\} \subseteq \{1, \dots, d\}$ represents the k_{th} group of features of size N, where $\forall k \neq k',\ g_k \cap g_{k'} = \emptyset$ and $\cup_{k=1}^K g_k = \{1, \dots, d\}$. The policy factorizes autoregressively by first selecting the group index, k, and then selecting the feature index, n, inside the selected group, i.e.,

$$p(a \mid s) = p(k \mid s)p(n \mid k, s), \quad k \in \{1, \dots, K\}, \quad n \in \{1, \dots, N\}. \tag{6}$$

The actual feature index being acquired is then decoded as $g_k^{(n)}$. As the agent acquires features, the already acquired features are removed from the candidate set. We simply set the probabilities of those features to zeros and renormalize the distribution. Similarly, if all features of a group have been acquired, the probability of this group is set to zero. With the proposed action space grouping, the original d-dimensional action space is reduced to $K + N$ decisions. Please refer to Fig. 1 for an illustration.

3.2 PARTIALLY OBSERVED OUT-OF-DISTRIBUTION DETECTION

In Sec. 2.3, we introduce several advanced techniques to detect out-of-distribution inputs. However, those approaches require fully observed data. In an AFA framework, data are partially observed at any acquisition step, which renders those approaches inappropriate. In this section, we develop a novel OOD detection algorithm specifically tailored for partially observed data. Inspired by MSMA (Mahmood et al., 2021), we propose to use the norm of scores from an arbitrary marginal distribution $p(x_o)$ as summary statistics and further detect partially observed OOD inputs with a DoSE (Morningstar et al., 2021) approach. MSMA for fully observed data is built by the following steps:

(i) Train a noise conditioned score matching network s_θ (Song & Ermon, 2019) with L noise levels by optimizing

$$\frac{1}{L}\sum_{i=1}^{L}\frac{\sigma_i^2}{2}\mathbb{E}_{p_{data}(x)}\mathbb{E}_{\tilde{x}\sim\mathcal{N}(x,\sigma_i^2 I)}\left[\left\|s_\theta(\tilde{x},\sigma_i)+\frac{\tilde{x}-x}{\sigma_i^2}\right\|_2^2\right]. \tag{7}$$

The score network essentially approximates the score of a series of smoothed data distributions $\nabla_{\tilde{x}}\log q_{\sigma_i}(\tilde{x})$, where $q_{\sigma_i}(\tilde{x})=\int p_{data}(x)q_{\sigma_i}(\tilde{x}\mid x)dx$, and $q_{\sigma_i}(\tilde{x}\mid x)$ transforms x by adding some Gaussian noise form $\mathcal{N}(0,\sigma_i^2 I)$.

(ii) For a given input x, compute the L2 norm of scores at each noise level, i.e., $s_i = \|s_\theta(x,\sigma_i)\|$.

(iii) Fit a low dimensional likelihood model for the norm of scores using in-distribution data, i.e., $p(s_1,\ldots,s_L)$, which is called density of states in (Morningstar et al., 2021) following the concept in statistical mechanics.

(iv) Threshold the likelihood to determine whether the input x is OOD or not.

In order to deal with partially observed data, we modify the score network to output scores of arbitrary marginal distributions, i.e., $\nabla_{\tilde{x}_m}\log q_{\sigma_i}(\tilde{x}_m)$, where $m\subseteq\{1,\ldots,d\}$ represents an arbitrary subset of features. We propose to do so by extending equation 7 to

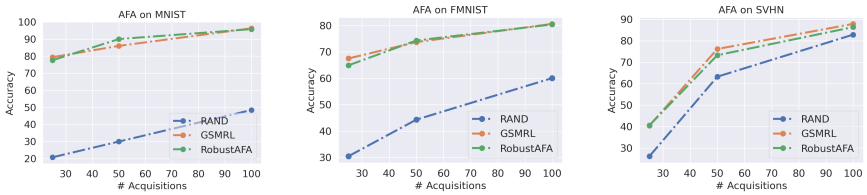
$$\frac{1}{L}\sum_{i=1}^{L}\frac{\sigma_i^2}{2}\mathbb{E}_{p_{data}(x)}\mathbb{E}_{\tilde{x}\sim\mathcal{N}(x,\sigma_i^2 I)}\mathbb{E}_{m\sim p(m)}\left[\left\|s_\theta(\tilde{x}\odot\mathbb{I}_m,\mathbb{I}_m,\sigma_i)\odot\mathbb{I}_m+\frac{\tilde{x}\odot\mathbb{I}_m-x\odot\mathbb{I}_m}{\sigma_i^2}\right\|_2^2\right], \tag{8}$$

where $\mathbb{I}_m$ represents a d-dimensional binary mask indicating the partially observed features, $\odot$ represents the element-wise product operation, and $p(m)$ is the distribution for generating observed dimensions. Similar to the fully observed case, we compute the L2 norm of scores at each noise level, i.e., $s_i = \|s_\theta(x\odot\mathbb{I}_m,\mathbb{I}_m,\sigma_i)\odot\mathbb{I}_m\|$, and fit a likelihood model in this transformed low-dimensional space. The likelihood model is also conditioned on the binary mask $\mathbb{I}_m$ to indicate the observed dimensions, i.e., $p(s_1,\ldots,s_L\mid\mathbb{I}_m)$. Given an input x with observed dimensions m, we threshold the likelihood $p(s_i,\ldots,s_L\mid\mathbb{I}_m)$ to determine whether the partially observed data x_m is OOD or not. To train the partially observed MSMA (PO-MSMA), we generate a mask for each input data x at random. The conditional likelihood over norm of scores is estimated by a conditional autoregressive model, for which we utilize the efficient masked autoregressive implementation (Papamakarios et al., 2017).

One benefit of our proposed PO-MSMA approach is that a single model can be used to detect OOD inputs with arbitrary observed features, which is convenient for detecting OOD inputs along the acquisition trajectories. Furthermore, sharing weights across different tasks (i.e., different marginal distributions) could act as a regularization (as discussed in (Li et al., 2020)), thus the unified score matching network can potentially perform better than separately trained ones for each different conditional, which we will investigate in future works.

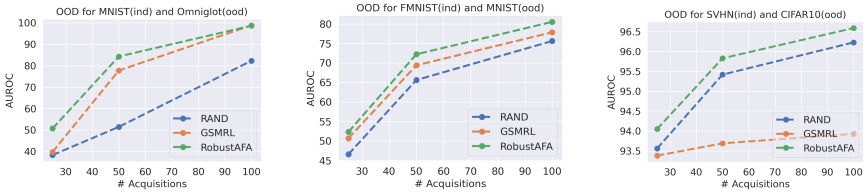
3.3 ROBUST ACTIVE FEATURE ACQUISITION

Above, we introduce our proposed action space grouping technique and a partially observed OOD detection algorithm. Combining those components, we can now actively acquire features for a problem with a large action space and simultaneously detect OOD inputs using the acquired subset of features. In order to guide the agent to acquire features that are informative for OOD detection, we propose an auxiliary reward that utilizes the likelihood of score norms of a partially observed



(a) MNIST (b) FMNIST (c) SVHN

Figure 3: Classification accuracy for acquiring different number of features.



(a) MNIST - Omniglot (b) FMNIST - MNIST (c) SVHN - CIFAR10

Figure 4: AUROC for OOD detection with acquired features.

input (with mask $\mathbb{I}_m$), $p(s_1, \ldots, s_L \mid \mathbb{I}_m)$. This encourages the agent to acquire features that more closely resemble the in-distribution ones, and thus reduces the false positive detection.

In summary, our robust AFA framework contains a dynamics model, a grouping of actions (features), an OOD detector and an RL agent. The dynamics model captures the arbitrary conditionals, $p(x_u, y \mid x_o)$, and is utilized to provide auxiliary information and intermediate rewards. It also enables a simple and efficient action space grouping technique and thus scales AFA up to applications with large action spaces. The partially observed OOD detector is used to distinguish OOD inputs alongside the acquisition procedure and also used to provide an auxiliary reward so that the agent is encouraged to acquire informative features for OOD detection. The RL agent takes in the current acquired features and auxiliary information from the dynamics model and predicts what next feature to acquire. When the feature is actually acquired, the agent pays the acquisition cost of the feature and receives an intermediate reward from the dynamics model. When the acquisition process is terminated, the agent makes a final prediction about the target, y, using all its acquired features and receives an reward about its prediction. It also receives an reward from the OOD detector about the likelihood of the acquired feature subset in the transformed space (i.e., the norm of the scores). Please refer to Algorithm 1 for additional details and to Fig. 2 for an illustration.

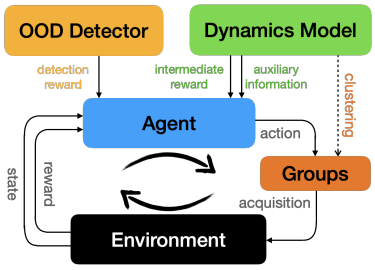
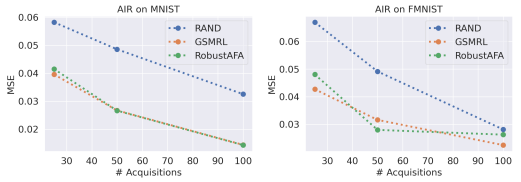


Figure 2: Schematic illustration of our robust AFA framework.

In GSMRL (Li & Oliva, 2020), the acquisition procedure is terminated when the agent selects a special termination action, which means each instance could have different number of features acquired. Although intriguing for practical use, it introduces additional complexity to assess OOD detection performance. To simplify the evaluation, we instead specify a fixed acquisition budget (i.e., the number of acquired features). The agent will terminate the acquisition process when it exceeds the specified acquisition budget. However, it is possible to incorporate a termination action into our framework.

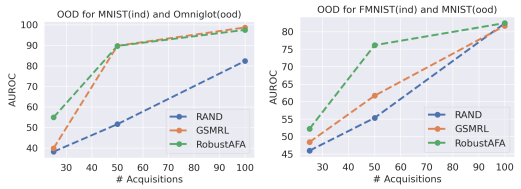
4 Experiments

In this section, we evaluate our framework on several commonly used OOD detection benchmarks. Our model actively acquires features to predict the target and meanwhile determines whether the input is OOD using only the acquired features. Given that these benchmarks typically have a large number of candidate features, current AFA approaches cannot be applied directly. We instead compare to a modified GSMRL algorithm, where candidate features are clustered with our proposed



(a) MNIST (b) FMNIST

Figure 6: Reconstruction MSE for robust AIR.



(a) MNIST - Omniglot (b) FMNIST - MNIST

Figure 7: OOD detection for robust AIR.

Figure 8: Examples of the acquisition process for robust AIR.

action space grouping technique. We also compare to a simple random acquisition baseline, where a random unobserved feature is acquired at each acquisition step. The random policy is repeated for 5 times and the metrics are averaged from different runs. Please refer to Appendix A for experimental details. For each dataset, we assess the performance under several prespecified acquisition budgets. For classification task, the performance is evaluated by the classification accuracy; for reconstruction task, the performance is evaluated by the reconstruction MSE. We also detect OOD inputs using the acquired features and report the AUROC scores.

Robust Active Feature Acquisition We first evaluate the AFA tasks using several classification datasets. The agent is trained to acquire the pixel values. For color images, the agent acquires all three channels at once. For MNIST (LeCun et al., 2010) and FMNIST (Xiao et al., 2017), we follow GSMRL to train the surrogate model using a class conditioned ACFlow (Li et al., 2020); for SVHN (Netzer et al., 2011), we simply use a partially observed classifier to learn $p(y \mid x_o)$ since we found ACFlow difficult to train for this dataset. The auxiliary information is accordingly modified to contain only the prediction probability. Figure 3 and 4 report the classification accuracy and OOD detection AUROC respectively. The accuracy is significantly higher for RL approaches than the random acquisition policy. Although we expect a trade-off between accuracy and OOD detection performance for our robust AFA framework, the accuracy is actually comparable to GSMRL and sometimes even better across the datasets. Meanwhile, the OOD detection performance for our robust AFA framework is significantly improved by enforcing the agent to acquire informative features for OOD identification. For SVHN and CIFAR10 detection, the AUROC for GSMRL is even lower than the random policy, which we believe is because of the discrepancy of informative features for two different goals. Augmented with the detector reward solves the problem and improves the detection performance even further. Figure 5 presents several examples of the acquisition process from our robust AFA framework. We can see the prediction becomes certain after only a few acquisition steps. See appendix A for additional examples.

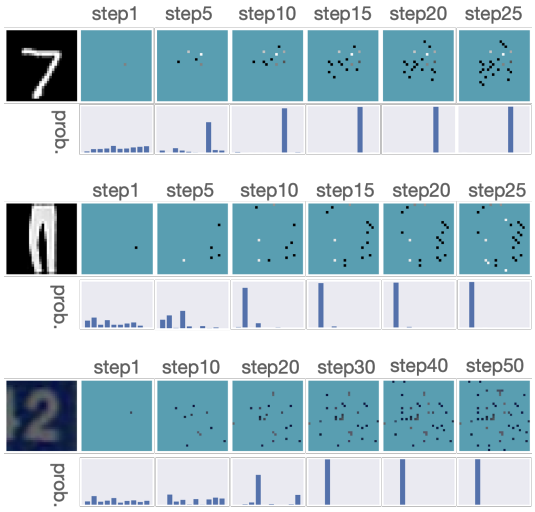


Figure 5: Examples of the acquisition process from our robust AFA framework. The bar charts demonstrate the class prediction probability at the corresponding acquisition step.

Robust Active Instance Recognition In this section, we evaluate the AIR task using MNIST and FashionMNSIT datasets. Following GSMRL (Li & Oliva, 2020), we use ACFlow as the surrogate

model. Figure 6 and 7 report the reconstruction MSE and OOD detection performance respectively using the acquired features. We can see our robust AIR framework improves the OOD detection performance significantly, especially when the acquisition budget is low, while the reconstruction MSEs are comparable to GSMRL. Figure 8 presents several examples of the acquisition process for robust AIR.

Ablations Our proposed action grouping technique enables the agent to acquire features from a potentially large pool of candidates. However, it also introduces some complexity due to the autoregressive factorization in equation 6. In Fig. 9, we compare two agents with and without the action grouping using a downsampled MNIST. We can see the action grouping does not degrade the performance on smaller dimensionalities whilst allowing one to work over larger dimensionalities that previous methods cannot scale to.

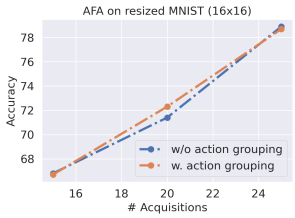


Figure 9: Compare AFA performance with or without action grouping.

The mutual information based clustering scheme could help the agent navigate the action space, thus simplifying exploration. Figure A.3 in appendix presents the acquisition process for MNIST AFA. We can see the acquired features concentrate on the informative groups especially at the early stage, which verifies the effectiveness of our action space grouping scheme.

We also compare with several alternative clustering schemes. Random clustering groups the candidate features into several equal-sized clusters. Row clustering groups pixels by their rows. Graph clustering first builds an undirected graph over candidate features and groups the nodes using spectral clustering methods. Figure 10 presents the AFA performance using different clustering schemes. We can see our information based scheme outperforms other alternatives significantly, especially when the acquisition budget is low. Figure A.4 in the appendix shows several clusters obtained from each grouping scheme.

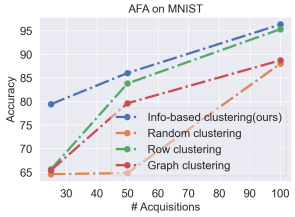


Figure 10: Ablation study about clustering methods.

Several attempts have been made to deal with large action space in general RL. Here, we compare with the Wolpertinger policy. Although there is no universal action embeddings available for general AFA problems (as described in Sec. 3.1), we can use the Cartesian coordinates of pixels as the embeddings for image datasets. We perform extensive hyperparameter tuning for training the policy. Results are presented in Fig. A.5, which demonstrates that our action space grouping is more effective in the AFA setting. Please see the appendix for additional details.

Although our PO-MSMA is designed for partially observed instances, it can handle fully observed ones as special cases. In Table 1, we report the AUROC scores for both methods. We can see our PO-MSMA is competitive even though it is not trained to detect fully observed instances.

Table 1: Comparison with MSMA for fully observed OOD detection. AUROC scores are reported.

	MNIST - Omniglot	FMNIST - MNIST	SVHN - CIFAR10	CIFAR10 - SVHN
MSMA	-	82.56	97.60	95.50
PO-MSMA	99.55	96.62	97.77	74.74

5 Discussion and Conclusion

In this work, we investigate an understudied problem in AFA, increasing robustness. Previous AFA methodology fails to produce meaningful acquisitions in high dimensional settings and do not flag when they are applied to out of distribution instances. Both short-commings limit the applicability of AFA in real-world scenarios. We propose a robust AFA framework to acquire feature actively and determine whether the input is OOD using only the acquired subset of features. In order to scale up the AFA models to practical use, we develop a hierarchical acquisition policy, where the candidate features are grouped together based on their relevance to the target. Our framework represents the first AFA model that can deal with a potentially large pool of candidate features. Extensive experiments are conducted to showcase the effectiveness of our framework.

REFERENCES

Tim Bakker, Herke van Hoof, and Max Welling. Experimental design for mri by greedy policy search. *Advances in Neural Information Processing Systems*, 33, 2020.

José M Bernardo. Expected information as expected utility. *the Annals of Statistics*, pp. 686–690, 1979.

Christopher M Bishop. Novelty detection and neural network validation. *IEE Proceedings-Vision, Image and Signal processing*, 141(4):217–222, 1994.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pp. 1613–1622. PMLR, 2015.

Xiaoyong Chai, Lin Deng, Qiang Yang, and Charles X Ling. Test-cost sensitive naive bayes classification. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pp. 51–58. IEEE, 2004.

Hyunsun Choi, Eric Jang, and Alexander A Alemi. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.

Gabriel Dulac-Arnold, Richard Evans, Hado van Hasselt, Peter Sunehag, Timothy Lillicrap, Jonathan Hunt, Timothy Mann, Theophane Weber, Thomas Degris, and Ben Coppin. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Wenbo Gong, Sebastian Tschiatschek, Sebastian Nowozin, Richard E Turner, José Miguel Hernández-Lobato, and Cheng Zhang. Icebreaker: Element-wise efficient information acquisition with a bayesian deep latent gaussian model. 2019.

He He, Jason Eisner, and Hal Daume. Imitation learning by coaching. *Advances in Neural Information Processing Systems*, 25:3149–3157, 2012.

He He, Paul Mineiro, and Nikos Karampatziakis. Active information acquisition. *arXiv preprint arXiv:1602.02181*, 2016.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=HyxCxhRcY7`.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474*, 2016.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Yang Li and Junier B Oliva. Active feature acquisition with generative surrogate models. *arXiv preprint arXiv:2010.02433*, 2020.

Yang Li, Shoaib Akbar, and Junier Oliva. ACFlow: Flow models for arbitrary conditional likelihoods. In *International Conference on Machine Learning*, pp. 5831–5841. PMLR, 2020.

Charles X Ling, Qiang Yang, Jianning Wang, and Shichao Zhang. Decision trees with minimal costs. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 69, 2004.

Jeremiah Zhe Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax-Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *arXiv preprint arXiv:2006.10108*, 2020.

Chao Ma, Sebastian Tschiatschek, Konstantina Palla, José Miguel Hernández-Lobato, Sebastian Nowozin, and Cheng Zhang. Eddi: Efficient dynamic discovery of high-value information with partial vae. *arXiv preprint arXiv:1809.11142*, 2018.

Ahsan Mahmood, Junier Oliva, and Martin Andreas Styner. Multiscale score matching for out-of-distribution detection. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=xoHdgbQJohv.

Sultan Javed Majeed and Marcus Hutter. Exact reduction of huge action spaces in general reinforcement learning. *arXiv preprint arXiv:2012.10200*, 2020.

Marvin Minsky. Steps toward artificial intelligence. *Proceedings of the IRE*, 49(1):8–30, 1961.

Warren Morningstar, Cusuh Ham, Andrew Gallagher, Balaji Lakshminarayanan, Alex Alemi, and Joshua Dillon. Density of states estimation for out of distribution detection. In *International Conference on Artificial Intelligence and Statistics*, pp. 3232–3240. PMLR, 2021.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019a. URL https://openreview.net/forum?id=H1xwNhCcYm.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, and Balaji Lakshminarayanan. Detecting out-of-distribution inputs to deep generative models using typicality. *arXiv preprint arXiv:1906.02994*, 2019b.

Feng Nan, Joseph Wang, Kirill Trapeznikov, and Venkatesh Saligrama. Fast margin-based cost-sensitive classification. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2952–2956. IEEE, 2014.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. *arXiv preprint arXiv:1705.07057*, 2017.

Luis Pineda, Sumana Basu, Adriana Romero, Roberto Calandra, and Michal Drozdzal. Active mr k-space sampling with reinforcement learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 23–33. Springer, 2020.

Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark A DePristo, Joshua V Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *arXiv preprint arXiv:1906.02845*, 2019.

Thomas Rückstieß, Christian Osendorfer, and Patrick van der Smagt. Sequential feature selection for classification. In *Australasian joint conference on artificial intelligence*, pp. 132–141. Springer, 2011.

Hajin Shim, Sung Ju Hwang, and Eunho Yang. Joint active feature acquisition and classification with variable-size set encoding. *Advances in neural information processing systems*, 31:1368–1378, 2018.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*, 2019.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3 (1):9–44, 1988.

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning*, pp. 9690–9700. PMLR, 2020.

Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. Improving deterministic uncertainty estimation in deep learning for classification and regression. *arXiv preprint arXiv:2102.11409*, 2021.

Hans van Gorp, Iris A.M. Huijben, Bastiaan S. Veeling, Nicola Pezzotti, and Ruud Van Sloun. Active deep probabilistic subsampling, 2021. URL https://openreview.net/forum?id=0NQdxInFWT_.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

Zizhao Zhang, Adriana Romero, Matthew J Muckley, Pascal Vincent, Lin Yang, and Michal Drozdzal. Reducing uncertainty in undersampled mri reconstruction with active acquisition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2049–2058, 2019.

Valentina Bayer Zubek, Thomas Glen Dietterich, et al. Pruning improves heuristic search for cost-sensitive learning. 2004.

A EXPERIMENTAL DETAILS

A.1 ROBUST ACTIVE FEATURE ACQUISITION

Datasets We evaluate the performance for robust AFA using several classification datasets. MNIST and FashionMNIST are two gray-scale image datasets of size 28×28, and SVHN is a color image dataset of size 32×32. Our framework acquires one pixel value at each acquisition step. For color images, it acquires all three channels at once.

Dynamics Model For MNIST and FashionMNIST, we follow GSMRL (Li & Oliva, 2020) to use a class conditioned ACFlow for dynamics modeling. ACFlow captures the arbitrary conditional distribution $p(x_u \mid x_o, y)$, and the prediction using an arbitrary subset can be derived using the Bayes rule, i.e.,

$$p(y \mid x_o) = \frac{p(x_o \mid y)p(y)}{\sum_{y'} p(x_o \mid y')p(y')}. \tag{A.1}$$

The architecture of ACFlow closely follows GSMRL, which contains a stack of affine coupling layers and a Gaussian base likelihood module. The dynamics model is used to assess the intermediate reward of an acquisition and provide auxiliary information to assist the agent. please refer to Sec. 2 for details.

For SVHN, it is hard for ACFlow to balance the likelihood objective and the classification loss. Instead, we use a simple classifier with partially observed inputs to learn $p(y \mid x_o)$. We use the ResNet50 architecture and modify it to take in masked inputs and a binary mask. During training, we sample the mask at random. Since the partially observed classifier does not explicitly capture the dependencies among features, it cannot provide any prediction about the unobserved features. Although, it can still assess the intermediate reward using the information gain. We also use the current prediction probability as the auxiliary information.

PO-MSMA Our PO-MSMA model consists of a score matching network and a likelihood model over the norm of scores. We modify the original NCSN model to produce the scores for arbitrary marginal distributions $\nabla_{\tilde{x}_m} \log p(\tilde{x}_m)$. Specifically, the inputs contain the masked images x_o and a binary mask indicating the observed pixels. The output is a tensor with the same size as the input image. We then mask out the unobserved dimensions for the output and compute the norm only for those observed pixels. Throughout the experiment, we use 10 noise scales, thus obtaining 10 summary statistics, $s_1, \ldots, s_{10}$, for each input image. Then, we train a conditional autoregressive model for the norms conditioned on the binary mask $\mathbb{I}_m$, i.e.,

$$p(s_1, \ldots, s_{10} \mid \mathbb{I}_m). \tag{A.2}$$

Given a test image x with observed dimensions m, the OOD detection starts by calculating the norm of scores on different noise levels. Then, the conditional likelihood $p(s_1, \ldots, s_{10} \mid \mathbb{I}_m)$ is thresholded to determine whether the given partially observed input is OOD or not. Here, we report the AUROC scores to evaluate the OOD detection performance.

AFA Agent We use PPO algorithm to train our AFA agent. Given the observed dimensions as the state, we first use a two-layer convolutional network with max pooling to extract an embedding, from which the actor and critic are derived using two fully connected layers. The actor network predicts the probability of the next action, where the probabilities of observed features are manually set to zero. The critic network is used to estimate the state values. The AFA agent observes the current acquired features and determines which next feature to acquire. It stops acquiring more features when the acquired features exceed the acquisition budget. Throughout this work, we assume each feature has the same cost, thus the acquisition budget is equivalent to the number of features to be acquired. In GSMRL (Li & Oliva, 2020), the authors also learn a predictor along with the agent. However, we did not find it beneficial at the early stage of experiment. Therefore, we directly use the dynamics model to make a final prediction.

Baselines Both greedy and RL based approaches have been proposed for the AFA task. However, they all deal with a small number of candidate features and have difficulty scaling to large ones. For example, the greedy approach, EDDI (Ma et al., 2018), has a $O(Nd)$ computation complexity

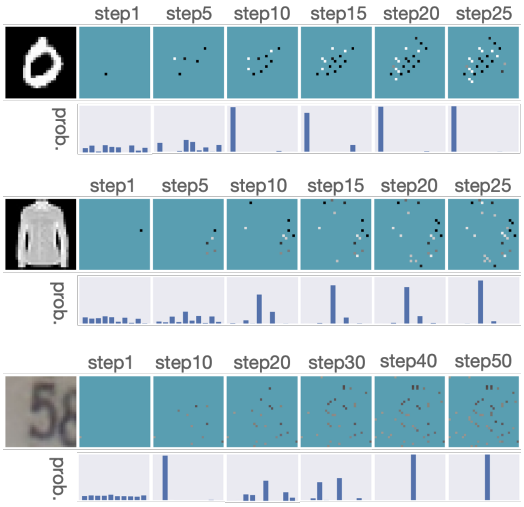


Figure A.1: Additional results from our robust AFA framework.

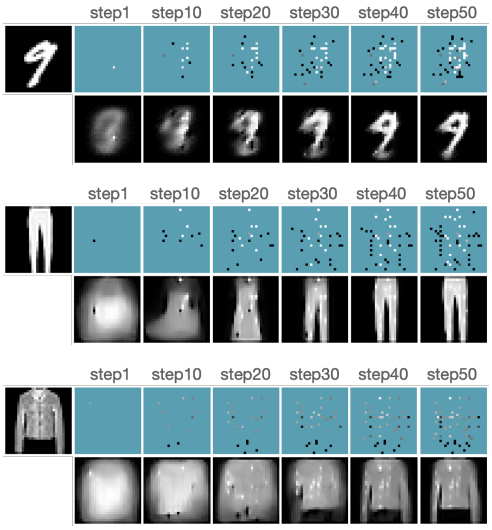


Figure A.2: Additional results from our robust AIR framework.

for acquiring N features from a d dimensional feature space. Model-free and model-based RL approaches are known difficult dealing with large action spaces (Li & Oliva, 2020). In order to evaluate the OOD detection performance using commonly used benchmarks, we modify the state-of-the-art AFA approach, GSMRL (Li & Oliva, 2020), with our proposed action space grouping technique. We also compare to a random policy where a random unobserved feature is acquired at each acquisition step.

Additional Results Figure A.1 presents additional results for acquiring features using our robust AFA framework. Our model successfully recognizes the underlying classes using only a small subset of features.

A.2 Robust Active Instance Recognition

Datasets We evaluate the AIR performance using MNIST and FashionMNIST. The model acquires one pixel at each acquisition step to reconstruct those unobserved pixels.

Dynamics Model Following GSMRL (Li & Oliva, 2020), we use ACFlow to model the dynamics. Specifically, ACFlow learns the arbitrary conditional distribution $p(x_u \mid x_o)$, and the prediction about the unobserved pixels are simply sampled from this distribution. The intermediate reward is defined as the improvement of the log likelihood per dimension, i.e.,

$$r_m(s, i) = \frac{\log p(x_{u\setminus i} \mid x_o)}{|u| - 1} - \frac{\log p(x_u \mid x_o)}{|u|}. \tag{A.3}$$

The dynamics model also provides auxiliary information to the agent, which contains the predicted mean and variance of the unobserved pixels.

PO-MSMA The OOD detector is the same as used in the AFA task.

AIR Agent The agent is also the same as the AFA task, except the final reward is given as the MSE between the prediction and the groundtruth.

Baselines Similar to the AFA task described above, we compare to a modified GSMRL algorithm and a random acquisition policy.

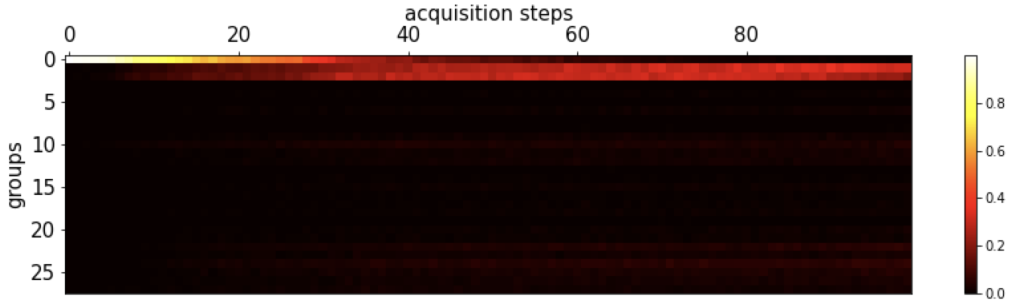


Figure A.3: Acquired groups along the acquisition process for MNIST AFA. Each column represents the frequency of each group being acquired at the corresponding acquisition step. Groups with smaller index have higher mutual information to the target variable.

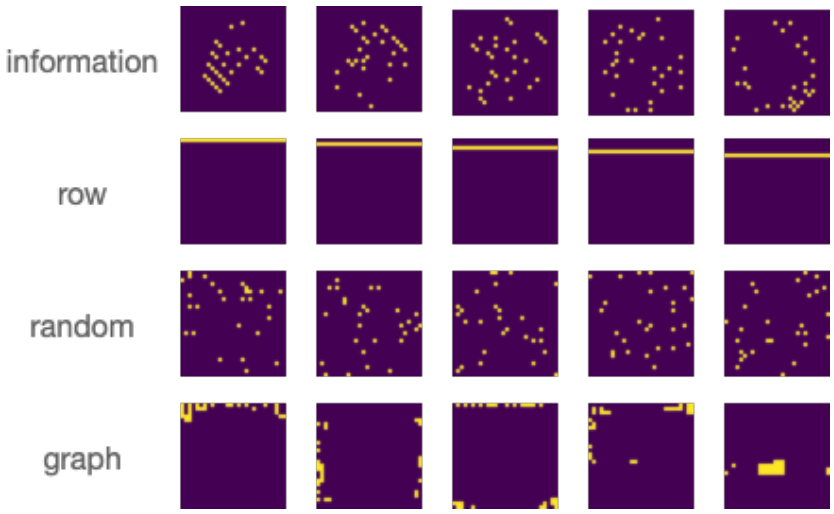


Figure A.4: Examples of the clusters from different grouping scheme.

Additional Results Figure A.2 presents several acquisition processes for the AIR task using our proposed framework. The prediction quickly becomes certain after only several acquisitions.

A.3 Ablations

Information based clustering Figure A.3 presents the frequencies of each group being acquired along the acquisition process for MNSIT AFA. We can see the acquired features concentrate on the informative groups especially at the early acquisition steps. Therefore, our grouping scheme acts like a curriculum that guides the agent towards informative acquisitions.

Grouping schemes In the main text, we propose using mutual information between each candidate feature and target variable to group actions. The mutual information based clustering scheme could help the agent eliminate non-informative actions, thus simplifying exploration. Here, we compare with several alternative clustering schemes. Random clustering groups the candidate features into several equal-sized clusters. Row clustering groups pixels by their rows. Graph clustering first builds an undirected graph over candidate features and groups the nodes using spectral clustering methods. We build the graph for MNIST pixels using the graphical lasso method. Since the spectral clustering cannot guarantee balanced cluster sizes, we further postprocess the clustering results by splitting large clusters and combining small clusters so that each cluster has equal size. Figure A.4 shows several clusters obtained from each grouping scheme.

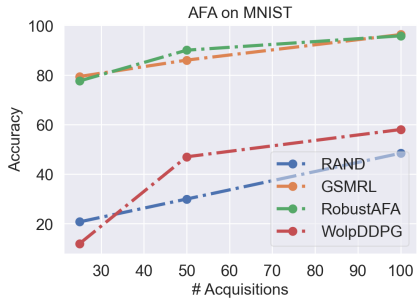


Figure A.5: Comparison with Wolpertinger DDPG policy for MNIST AFA.

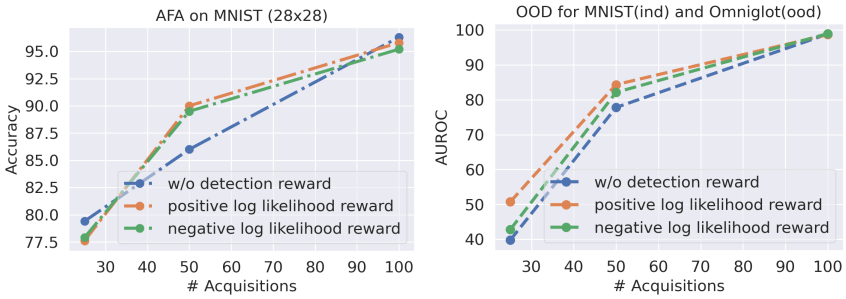


Figure A.6: Comparison of different detection reward.

Wolpertinger DDPG Policy Wolpertinger policy is specifically designed to deal with large discrete action space. Although there is no universal action embeddings available for general AFA problems, we can use the Cartesian coordinates of pixels as the embeddings for image datasets. We perform extensive hyperparameter tuning for training the policy, such as the network architectures, epsilon greedy exploration, and Ornstein Uhlenbeck Process parameters. Results are presented in Fig. A.5, which demonstrates that our action space grouping is more effective in the AFA setting.

Detection Reward In the main text, we use $\log p(s_1, \ldots, s_{10} \mid \mathbb{I}_m)$ as an auxiliary reward from the OOD detector. As we discussed in Sec. 3.3, the positive log likelihood reward helps to reduce the false positive, while the negative log likelihood reward helps to reduce the false negative. Figure A.6 compares different types of detection reward. We can see both positive and negative likelihood reward can improve the detection performance, and the classification accuracy does not degrade a lot from the baseline.