Active Feature Acquisition with Generative Surrogate Models

Yang Li [1] **Junier B. Oliva** [1]

Abstract

Many real-world situations allow for the acquisition of additional relevant information when making an assessment with limited or uncertain data. However, traditional ML approaches either require all features to be acquired beforehand or regard part of them as missing data that cannot be acquired. In this work, we consider models that perform active feature acquisition (AFA) and query the environment for unobserved features to improve the prediction assessments at evaluation time. Our work reformulates the Markov decision process (MDP) that underlies the AFA problem as a generative modeling task and optimizes a policy via a novel model-based approach. We propose learning a generative surrogate model (GSM) that captures the dependencies among input features to assess potential information gain from acquisitions. The GSM is leveraged to provide intermediate rewards and auxiliary information to aid the agent navigate a complicated high-dimensional action space and sparse rewards. Furthermore, we extend AFA in a task we coin *active instance recognition* (AIR) for the unsupervised case where the target variables are the unobserved features themselves and the goal is to collect information for a particular instance in a cost-efficient way. Empirical results demonstrate that our approach achieves considerably better performance than previous state of the art methods on both supervised and unsupervised tasks.

1. Introduction

A typical machine learning paradigm for discriminative tasks is to learn the distribution of an output, y given a complete set of features, $x \in \mathbb{R}^d$: $p(y \mid x)$. Although this paradigm is successful in a multitude of domains, it is incongruous with the expectations of many real-world intelligent systems in two key ways: first, it assumes that a complete set of features has been observed; second, as a consequence, it also assumes that no additional information (features) of an instance may be obtained at evaluation time. These assumptions often do not hold; human agents routinely reason over instances with incomplete data and decide when and what additional information to obtain. For example, consider a doctor diagnosing a patient. The doctor usually has not observed all possible measurements (such as blood samples, x-rays, etc.) for the patient. He/she is not forced to make a diagnosis based on the observed measurements; instead, he/she may dynamically decide to take more measurements to help determine the diagnosis. Of course, 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). Furthermore, acquiring features typically involves some cost (in time, money and risk), and intelligent systems are expected to automatically balance the cost and improvement on performance. In order to more closely match the needs of many real-world applications, we propose an active feature acquisition (AFA) model that not only makes predictions with incomplete/missing features, but also determines the next feature that would be the most valuable to obtain for a particular instance.

As noted in (Shim et al., 2018), the active feature acquisition problem may be formulated as a Markov decision process (MDP), where the state is the set of currently observed features and the action is the next feature to acquire. A special action indicates whether to stop the acquisition process and make a final prediction. After acquiring its value and paying the acquisition cost, the newly acquired feature is added to the observed subset and the agent proceeds to the next acquisition step. Once the agent decides to terminate the acquisition, it makes a final prediction based on the features acquired thus far. For example, in an image classification task (Fig. 1), the agent would dynamically acquire pixels until it is certain of the image class. The goal of the agent is

[1] Department of Computer Science, University of North Carolina at Chaple Hill, Chapel Hill, NC, USA. Correspondence to: Yang Li <yangli95@cs.unc.edu>, Junier B. Oliva <joliva@cs.unc.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

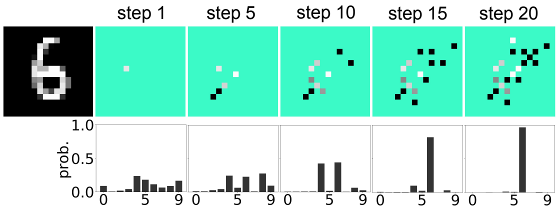

Figure 1. Active feature acquisition on MNIST. Example of our acquisition process (top) and the prediction probabilities (bottom). The green masks indicate the unobserved features.

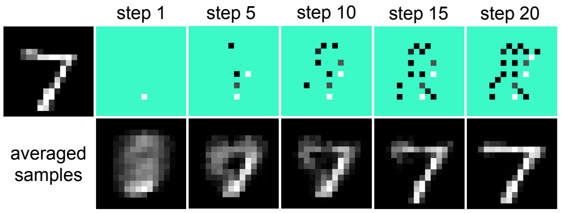

Figure 2. Active instance recognition on MNIST. Example of our acquisition process (top) and averaged inpaintings (bottom). The green masks indicate the unobserved features.

to maximize the prediction performance while minimizing the acquisition cost.

The key insight of this work is that the dynamics model for the AFA MDP is based on the conditionals of the features: $p(x_j \mid x_o)$, where x_j is an unobserved feature selected for acquisition and x_o are the previously acquired features. Thus, we develop a model-based approach through generative modeling of *all* conditional dependencies. Equipped with the surrogate model, our method, *Generative Surrogate Models for RL* (GSMRL), essentially combines model-free and model-based RL into a holistic framework.

GSMRL rectifies several short-comings of a model-free scheme such as JAFA (Shim et al., 2018). 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. To reduce the sparsity of the rewards and simplify the credit assignment problem for potentially long episodes (Minsky, 1961; Sutton, 1988), we leverage a surrogate model to provide intermediate rewards by assessing the information gain of a newly acquired feature, which quantifies how much our confidence about the prediction improves by acquiring this feature. In addition to sparse rewards, an AFA agent must also navigate a complicated high-dimensional action space (Dulac-Arnold et al., 2015), and must manage multiple roles as it has to: implicitly model dependencies, perform a cost/benefit analysis, and act as a classifier. To lessen the burden, we also propose using the surrogate model to provide side information that assists the agent. The side information shall explicitly inform the agent of: 1) uncertainty and imputations for unobserved features; 2) an estimate of the expected information gain of future acquisitions; 3) uncertainty of the target output. This allows the agent to easily assess its current uncertainty and helps the agent 'look ahead' to expected outcomes from future acquisitions.

In this work, we also propose the first (to the best of our knowledge) unsupervised AFA task, which we coin *active instance recognition* (AIR). Here we consider the case where there is not a single target variable, but instead the target of interest may be the remaining unobserved features themselves. That is, rather than reducing the uncertainty with respect to some desired output response (that cannot be directly queried and must be predicted), the task is to query as few features as possible that allows the agent to correctly uncover the remaining unobserved features. For example, in image data AIR, an agent queries new pixels until it can reliably uncover the remaining pixels (see Fig. 2). AIR is especially relevant in surveying tasks, which are broadly applicable across various domains and applications. Most surveys aim to discover a broad set of underlying characteristics of instances (e.g., citizens in a census) using a limited number of queries (questions in the census form), which is at the core of AIR. Policies for AIR would build a personalized subset of survey questions (for individual instances) that quickly uncovered the likely answers to all remaining questions.

Our contributions are as follows:

- We reformulate the AFA problem as a generative modeling task and build surrogate models that capture the state transitions with arbitrary conditional distributions.
- We develop methodology to leverage the surrogate model to provide intermediate rewards as training signals and to provide auxiliary information that assists the agent. Our framework represents a novel combination of model-free and model-based RL.
- We propose the first unsupervised active feature acquisition task where the target variables are the unobserved features themselves.
- We achieve state-of-the-art performance on both supervised and unsupervised tasks in the largest scale AFA study to date.
- We open-source a standardized environment inheriting the OpenAI gym interfaces (Brockman et al., 2016) to assist future research on active feature acquisition. Code is publicly available at `https://github.com/lupalab/GSMRL`.

2. Methods

In this section, we first describe our GSMRL framework for both active feature acquisition (AFA) and active instance recognition (AIR) problems. We then develop our RL algorithm and the corresponding surrogate models for different settings. We also introduce a special application that acquires features for time series data.

2.1. AFA and AIR with GSMRL

Consider a discriminative task with features $x \in \mathbb{R}^d$ and target y. Instead of predicting the target by first collecting all the features, we perform a sequential feature acquisition process in which we start from an empty set of features and actively acquire more features. There is typically a cost associated with features and the goal is to maximize the task performance while minimizing the acquisition cost, i.e.,

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

where $\mathcal{L}(\hat{y}(x_o), y)$ represents the loss function between the prediction $\hat{y}(x_o)$ and the target y. Note that the prediction is made with the acquired feature subset $x_o, o \subseteq \{1, \dots, d\}$. Therefore the agent should be able to predict with arbitrary subsets. $\mathcal{C}(o)$ represents the cost of the acquired features o. The hyperparameter α controls the trade-off between prediction loss and acquisition cost. For unsupervised tasks, the target variable y equals to x; that is, we acquire features actively to represent the instance with a selected subset.

In order to solve the optimization problem in (1), we formulate it as a Markov decision process as in (Zubek et al., 2004; Shim et al., 2018):

$$\begin{aligned} &s = [o, x_o], \quad a \in u \cup \phi, \\ &r(s,a) = -\mathcal{L}(\hat{y}, y)\mathbb{I}(a = \phi) - \alpha\mathcal{C}(a)\mathbb{I}(a \neq \phi). \end{aligned} \tag{2}$$

The state s is the current acquired feature subset $o \subseteq \{1, \dots, d\}$ and their values x_o. The action space contains the remaining candidate features $u = \{1, \dots, d\} \setminus o$ and a special action ϕ that indicates the termination of the acquisition process. To optimize the MDP, a reinforcement learning agent acts based on the observed state and receives rewards from the environment. When the agent acquires a new feature i, 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 reward is the negative acquisition cost of this feature. x_i is obtained from the environment (i.e. we observe the true i^{th} feature value for the instance).

Feature Dependencies as Dynamics Model A surprisingly unexplored property of the AFA MDP, and the driving observation to our work, is that the dynamics of the problem are dictated by *conditional dependencies among the data's features*. That is, the state transitions are based on the conditionals: $p(x_j \mid x_o)$, where x_j is an unobserved feature. Therefore we frame our approach according to the estimation of conditionals among features with generative models. We build a surrogate model to learn the distribution $p(y, x_j \mid x_o)$, where x_j and x_o contain arbitrary features from x. We find that the most efficacious use of our generative surrogate model (see section 5) is a hybrid model-based approach that utilizes intermediate rewards and side information stemming from dependencies.

Intermediate Rewards When the agent terminates the acquisition and makes a prediction, the reward equals to the negative prediction loss using current acquired features. Since the prediction is made at the end of acquisitions, the reward of the prediction is received only when the agent decides to terminate the acquisition process. This is a typical temporal credit assignment problem, which may affect the learning of the agent (Minsky, 1961; Sutton, 1988). Given the surrogate model, we propose to remedy the credit assignment problem by providing intermediate rewards for each acquisition. Inspired by the information gain, the surrogate model assesses the intermediate reward for a newly acquired feature i with

$$r_m(s, i) = H(y \mid x_o) - \gamma H(y \mid x_o, x_i), \tag{3}$$

where γ is a discount factor for the MDP. In appendix A, we show that our intermediate rewards will not change the optimal policy.

Side Information In addition to intermediate rewards, we propose using the surrogate model to also provide side information to assist the agent, which includes the current prediction and output likelihood, the possible values and corresponding uncertainties of the unobserved features, and the estimated utilities of the candidate acquisitions. The current prediction $\hat{y}$ and likelihood $p(y \mid x_o)$ inform the agent about its confidence, which can help the agent determine whether to stop the acquisition. The imputed values and uncertainties of the unobserved features give the agent the ability to look ahead into and future and guide its exploration. For example, if the surrogate model is very confident about the value of a currently unobserved feature, then acquiring it would be redundant. The utility of a feature i is estimated by its *expected* information gain to the target variable:

$$\begin{aligned} \mathcal{U}_i &= H(y \mid x_o) - \mathbb{E}_{p(x_i \mid x_o)} H(y \mid x_i, x_o) \\ &= H(x_i \mid x_o) - \mathbb{E}_{p(y \mid x_o)} H(x_i \mid y, x_o), \end{aligned} \tag{4}$$

where the surrogate model is used to estimate the expected entropies. The utility essentially quantifies the conditional mutual information $I(x_i; y \mid x_o)$ between each candidate feature and the target variable. A greedy policy can be easily built based on the utilities where the next feature to acquire is the one with maximum utility (Ma et al., 2018; Gong et al., 2019). Here, our agent takes the utilities as side information to help balance exploration and exploitation, and eventually learns a non-greedy policy.

Prediction Model When the agent deems that acquisition is complete, it makes a final prediction based on the acquired

Algorithm 1 Active Feature Acquisition with GSMRL

```
input pretrained surrogate model $M$; agent *agent*; prediction model $f_\theta(\cdot)$; test dataset $D$ to be acquired
  1. instantiate an environment: $env$ = Env($D$)
  2. $x_o$, o = $env$.reset()
  3. done = False; reward = 0
  while not done 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, done, r = $env$.step(action)
    reward = reward + r + $r_m$
  end while
  prediction = $agent$.predict($x_o$, o, aux)
```

features thus far. The final prediction may be made using the surrogate model, i.e., $p(y \mid x_o)$, but it might be beneficial to train predictions specifically based on the agent's own distribution of acquired features o, since the surrogate model is agnostic to the feature acquisition policy of the agent. Therefore, we build a prediction model $f_\theta(\cdot)$ that takes both the current state x_o and the side information as inputs (i.e. the same inputs as the policy). The prediction model can be trained simultaneously with the policy as an auxiliary task. Weight sharing between the policy and prediction function facilitates the learning of better representations. Given the two predictions from the surrogate model and the prediction model respectively, the final reward $-\mathcal{L}(\hat{y}, y)$ during training is the maximum one using either predictions. During test time, we choose one prediction based on validation performance. An illustration of our framework is presented in Fig. 3. Please refer to Algorithm 1 for pseudo-code of the acquisition process with our GSMRL framework. Please also see Algorithm 2 in the appendix for a detailed version. We will expound on the surrogate models for different settings below.

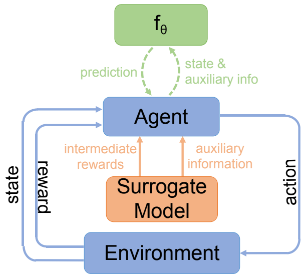

Figure 3. Illustration of our GSMRL framework with a prediction model f_θ.

2.1.1. Surrogate Model for AFA

As we mentioned above, the surrogate model learns the conditional distributions $p(y, x_j \mid x_o)$. Note that x_o is an arbitrary subset of the features and x_j is an arbitrary unobserved feature since the surrogate model must be able to assist arbitrary policies, and acquired features will vary from instance to instance. Thus, there are an exponential number of different conditionals that the surrogate model must estimate for a d-dimensional feature space. Therefore, learning a separate model for each different conditional is intractable. Fortunately, Ivanov et al. (2018) and Li et al. (2019) have proposed models to learn arbitrary conditional distributions $p(x_u \mid x_o)$. They regard different conditionals as different tasks and train VAE and normalizing flow based generative models, respectively, in a multi-task fashion to capture the arbitrary conditionals with a unified model. In this work, we leverage arbitrary conditionals and extend them to model the target variable y as well. For continuous target variables, we concatenate them with the features, thus $p(y, x_j \mid x_o)$ can be directly modeled. For discrete target variables, where we have a mix of continuous features and discrete labels, we use Bayes' rule:

$$p(y, x_j \mid x_o) = \frac{p(x_j \mid y, x_o)p(x_o \mid y)P(y)}{\sum_{y'} p(x_o \mid y')P(y')}. \tag{5}$$

We employ a variant arbitrary conditioning model that conditions on the target y to obtain the arbitrary conditional likelihoods $p(x_j \mid y, x_o)$ and $p(x_o \mid y)$ in (5).

Given a trained surrogate model, the prediction $p(y \mid x_o)$, the information gain in (3), and the utilities in (4) can all be estimated using the arbitrary conditionals. For continuous target variables, the prediction can be estimated by drawing samples from $p(y \mid x_o)$, and we express their uncertainties using sample variances. We calculate the entropy terms in (3) with Monte Carlo estimations. The utility in (4) can be further simplified as

$$\begin{aligned} \mathcal{U}_i &= \mathbb{E}_{p(y, x_i \mid x_o)} \log \frac{p(x_i, y \mid x_o)}{p(y \mid x_o)p(x_i \mid x_o)} \\ &= \mathbb{E}_{p(y, x_i \mid x_o)} \log \frac{p(y \mid x_i, x_o)}{p(y \mid x_o)}. \end{aligned} \tag{6}$$

Note that $p(y \mid x_i, x_o)$ is evaluated on sampled x_i rather than the true value, since we have not acquired its value yet.

For discrete target variables, we employ Bayes' rule to make a prediction

$$\begin{aligned} P(y \mid x_o) &= \frac{p(x_o \mid y)P(y)}{\sum_{y'} p(x_o \mid y')P(y')} \\ &= \operatorname{softmax}_y(\log p(x_o \mid y') + \log P(y')), \end{aligned} \tag{7}$$

and the uncertainty is expressed as the prediction probability. The information gain in (3) can be estimated analytically, since the entropy for a categorical distribution is analytically available. To estimate the utility, we further simplify (6) to

$$\begin{aligned} \mathcal{U}_i &= \mathbb{E}_{p(x_i \mid x_o)P(y \mid x_i, x_o)} \log \frac{P(y \mid x_i, x_o)}{P(y \mid x_o)} \\ &= \mathbb{E}_{p(x_i \mid x_o)} D_{\mathrm{KL}}[P(y \mid x_i, x_o) \| P(y \mid x_o)], \end{aligned} \tag{8}$$

where the KL divergence between two discrete distributions can be analytically computed. x_i is sampled from $p(x_i \mid x_o)$ as before. The expectation can be estimated using Monte Carlo samples.

Although the utility can be estimated accurately by (6) and (8), it involves some overhead especially for long episodes, since we need to calculate them for each candidate feature at each acquisition step. Moreover, each Monte Carlo estimation may require multiple samples. To reduce the computation overhead, we utilize (4) and estimate the entropy terms with Gaussian approximations. That is, we approximate $p(x_i \mid x_o)$ and $p(x_i \mid y, x_o)$ as Gaussian distributions and entropies reduce to a function of the variance. We use sample variance as an approximation. We found that this Gaussian entropy approximation performs comparably while being much faster.

2.1.2. Surrogate Model for AIR

For unsupervised tasks, our goal is to represent the full set of features with an actively selected subset. Since the target is also x, we modify our surrogate model to capture arbitrary conditional distributions $p(x_u \mid x_o)$, and modify the utility as the entropy of the unobserved features, i.e.,

$$\mathcal{U}_i = H(x_i \mid x_o). \tag{9}$$

We again use a Gaussian approximation to estimate the entropy. Therefore, the side information for AIR only contains imputed values and their variances of the unobserved features. Similar to the supervised case, we leverage the surrogate model to provide the intermediate rewards. Instead of using the information gain in (3), we use the reduction of negative log likelihood per dimension, i.e.,

$$r_m(s,i) = \frac{-\log p(x_u|x_o)}{|u|} - \gamma \frac{-\log p(x_{u\setminus i}|x_o,x_i)}{|u|-1}, \tag{10}$$

since (3) involves estimating the entropy for potentially high dimensional distributions, which itself is an open problem (Kybic, 2007). We show in appendix A that the optimal policy is invariant under this form of intermediate rewards. The final reward $-\mathcal{L}(\hat{x}, x)$ is calculated as the negative MSE of unobserved features $-\mathcal{L}(\hat{x}, x) = -\|\hat{x}_u - x_u\|_2^2$.

2.2. AFA for Time Series

We also apply our GSMRL framework on time series data. For example, consider a scenario where sensors are deployed in the field with limited resources. We would like the sensors to decide when to put themselves online to collect data. The goal is to make as few acquisitions as possible while still making an accurate prediction. In contrast to ordinary vector data, the acquired features must follow a chronological order, i.e., the newly acquired feature i must occur after all elements of o (since we may not go back in time to turn on sensors). In this case, it is detrimental to acquire a feature that occurs very late in an early acquisition step, since we will lose the opportunity to observe features ahead of it. The chronological constraint in action space removes all the features behind the acquired features from the candidate set. For example, after acquiring feature t, features $\{1, \dots, t\}$ are no longer considered as candidates for the next acquisition.

2.3. Implementation

We implement our GSMRL framework using the Proximal Policy Optimization (PPO) algorithm (Schulman et al., 2017). The policy network takes in a set of observed features and a set of auxiliary information from the surrogate model, extracts a set embedding from them using the set transformer (Lee et al., 2019), and outputs the actions. The critic network that estimates the value function shares the same set embedding as the policy network. To help learn useful representations, we also use the same set embedding as inputs for the prediction model f_θ. Arbitrary conditionals are estimated based on (Li et al., 2019).

To reflect the fact that acquiring the same feature repeatedly is redundant, we manually remove those acquired features from the candidate set. For time-series data, the acquired features must follow the chronological order since we cannot go back in time to acquire another feature, therefore we need to remove all the features behind the acquired features from the candidate set. Similar spatial constraints can also be applied for spatial data. To satisfy those constraints, we manually set the probabilities of the invalid actions to zeros.

3. Related Works

Active Learning Active learning (Fu et al., 2013; Konyushkova et al., 2017; Yoo & Kweon, 2019) is a related approach in ML to gather more information when a learner can query an oracle for the true label, y, of a complete feature vector $x \in \mathbb{R}^d$ to build a better estimator. However, our methods consider queries to the environment for the feature value corresponding to an unobserved feature dimension, i, in order to provide a better prediction on the current instance. Thus, while the active learning paradigm queries an oracle *during training* to build a classifier with complete features, our paradigm queries the environment *at evaluation* to obtain missing features of a current instance to help its current assessment.

Feature Selection Feature selection (Miao & Niu, 2016; Li et al., 2017; Cai et al., 2018), ascertains a static subset of important features to eliminate redundancies, which can help reduce computation and improve generalization. Feature selection methods choose a *fixed* subset of features $s \subseteq \{1, \dots, d\}$, and always predict y using this same subset

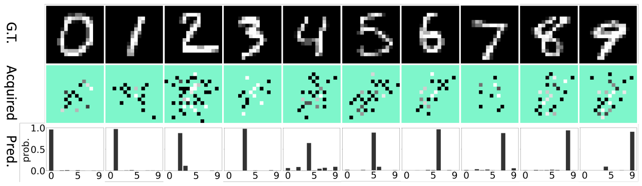

Figure 4. Example of acquired features and prediction. The green masks indicate the unobserved features.

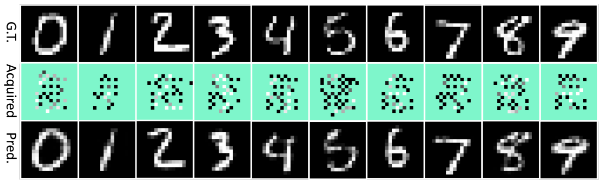

Figure 5. Example of acquired features and inpaintings. The green masks indicate the unobserved features.

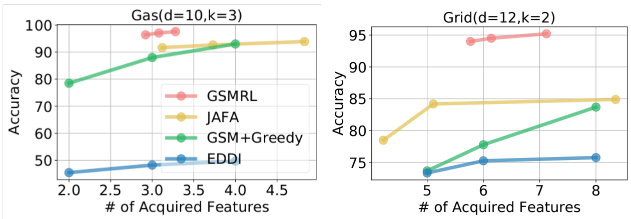

Figure 6. Test accuracy on UCI datasets.

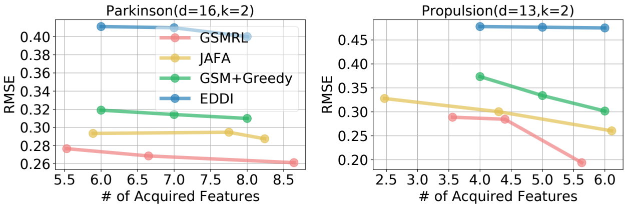

Figure 7. Test RMSE on UCI datasets.

of feature values, x_s. In contrast, our model considers a *dynamic* subset of features that is sequentially chosen and personalized on an instance-by-instance basis to increase useful information.

Active Feature Acquisition Instead of predicting the target passively using collected features, previous works have explored actively acquiring features in the cost-sensitive setting. Active perception is a relevant sub-field where a robot with a mounted camera is planning by selecting the best next view (Bajcsy, 1988; Aloimonos et al., 1988; Cheng et al., 2018; Jayaraman & Grauman, 2018). In this work we consider general features, and take images as one of many data sources. For general data, Ling et al. (2004), Chai et al. (2004) and Nan et al. (2014) propose decision tree, naive Bayes and maximum margin based classifiers, respectively, to jointly minimize the misclassification cost and feature acquisition cost. Ma et al. (2018) and Gong et al. (2019) acquire features greedily using mutual information as the estimated utility. Zubek et al. (2004) formulate the AFA problem as a MDP and fit a transition model using complete data, then they use the AO* heuristic search algorithm to find an optimal policy. Rückstieß et al. (2011) formulate the problem as a partially observable MDP and solve it using Fitted Q-Iteration. He et al. (2012) and He et al. (2016) instead employ a imitation learning approach guided by a greedy reference policy. Shim et al. (2018) utilize Deep Q-Learning and jointly learn a policy and a classifier. The classifier is treated as an environment that calculates the classification loss as the reward. ODIN (Zannone et al., 2019) presents an approach to learn a policy and a prediction model using augmented data with a Partial VAE (Ma et al., 2018). In contrast, GSMRL uses a surrogate model, which estimates both the state transitions and the prediction in a unified model, to directly provide intermediate rewards and auxiliary information to an agent.

Model-based and Model-free RL Reinforcement learning can be roughly grouped into model-based methods and model-free methods depending on whether they use a transition model (Li, 2017). Model-based methods are more data efficient but could suffer from significant bias if the dynamics are misspecified. On the contrary, model-free methods can handle arbitrary dynamic system but typically requires substantially more data samples. There have been works that combine model-free and model-based methods to compensate with each other. The usage of the model includes generating synthetic samples to learn a policy (Gu et al., 2016), back-propagating the reward to the policy along a trajectory (Heess et al., 2015), and planning (Chebotar et al., 2017; Pong et al., 2018). In this work, we rely on the model to provide intermediate rewards and side information. We compare this strategy to other model-based approaches in section 5.

4. Experiments

In this section, we evaluate our method on several benchmark environments built upon the UCI repository (Dua & Graff, 2017) and MNIST dataset (LeCun, 1998). We compare our method to another RL based approach, JAFA (Shim et al., 2018), which jointly trains an agent and a classifier. We also compare to a greedy policy EDDI (Ma et al., 2018) that estimates the utility for each candidate feature using a VAE based model and selects one feature with the highest utility at each acquisition step. As a baseline, we also acquire features greedily using our surrogate model that estimates the utility following (6), (8) and (9). We use a fixed cost for each feature and report multiple results with different α in (1) to control the trade-off between task performance and acquisition cost. We cross validate the best architecture and hyperparameters for baselines. Architectural details, hyperparameters and sensitivity analysis are provided in the appendix. In this work, we conduct the largest scale AFA study to date. Previous works have typically considered smaller datasets (both in terms of the number of features and the number of instances). We instead consider a broad range of datasets with more instances and higher dimensionality. In terms of comparisons, previous works often compare to naively simple baselines, such as a random acquisition order. In this work, we compare our GSMRL to the state-of-the-art models with both greedy policy and non-greedy RL policy.

Classification We first perform classification on the MNIST dataset. We downsample the original images to 16 × 16 to reduce the action space to accommodate baselines such as EDDI that have trouble scaling (see Sec. D in the appendix for details on full MNIST). Fig. 4 illustrates several examples of the acquired features and their prediction probability for different images. We can see that our model acquires a different subset of features for different images. Notice the checkerboard patterns of the acquired features, which indicates our model is able to exploit the spatial correlation of the data. Fig. 1 shows the acquisition process and the prediction probability along the acquisition. We can see the prediction become certain after acquiring only a small subset of features. The test accuracy in Fig. 10 demonstrates the superiority of our method over other baselines. It typically achieves higher accuracy with a lower acquisition cost. It is worth noting that our surrogate model with a greedy acquisition policy outperforms EDDI. We believe the improvement is due to the better distribution modeling ability of our surrogate model so that the utility and the prediction are more accurately estimated. We also perform classification using several UCI datasets. The test accuracy is presented in Fig. 6. Again, our method outperforms baselines under the same acquisition budget.

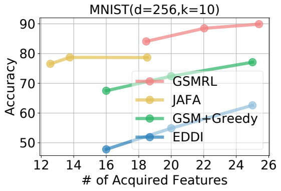

Figure 10. Test accuracy for AFA on MNIST.

Regression We also conduct experiments for regression tasks using several UCI datasets. We report the root mean squared error (RMSE) of the target variable in Fig. 7. Similar to the classification task, our model outperforms baselines with a lower acquisition cost.

Time Series To evaluate the performance with constraints in action space, we classify over time series data where the acquired features must follow chronological ordering. The datasets are from the UEA & UCR time series classification repository (Bagnall et al., 2017). For GSMRL and JAFA, we clip the probability of invalid actions to zero; for the greedy method, we use a prior to bias the selection towards earlier time points. Please refer to appendix B.4 for details. Fig. 8 shows the accuracy with different numbers of acquired features. Our method achieves high accuracy by collecting a small subset of the features.

Medical Diagnosis We evaluate the AFA performance for medical diagnosis. We use the Physionet challenge 2012 dataset (Goldberger et al., 2000) to predict the in-hospital mortality. Since the classes are heavily imbalanced, we use weighted cross entropy as training loss and the final rewards. For evaluation, we report the F1 scores in Fig. 11. Compared to baselines, our model achieves higher F1 with lower acquisition cost.

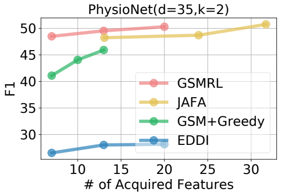

Figure 11. F1 for in-hospital mortality on Physionet.

Unsupervised Next, we evaluate our method on unsupervised tasks where features are actively acquired to impute the unobserved features. We use negative MSE as the reward for GSMRL and JAFA. The greedy policy calculates the utility following (9). For low dimensional UCI datasets, our method is comparable to baselines as shown in Fig. 9; but for the high dimensional case, as shown in Fig. 12, our method is doing better. Note JAFA is worse than the greedy policy for MNIST. We found it hard to train the policy and the reconstruction model jointly without the help of the surrogate model in this case. See Fig. 2 for an example of the acquisition process.

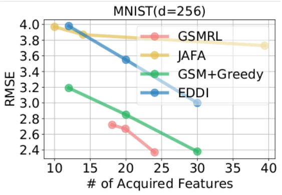

Figure 12. RMSE of x_u for AIR on MNIST.

5. Ablations

We now conduct a series of ablation studies to explore the capabilities of our GSMRL model.

Model-based Alternatives Our GSMRL model combines model-based and model-free approach into a holistic framework by providing the agent with auxiliary information and intermediate rewards. Here, we study different ways of utilizing the dynamics model. As in ODIN (Zannone et al., 2019), we utilize class conditioned generative models to generate synthetic trajectories. The agent is then trained with both real and synthetic data (PPO+Syn). Another way of using the model is to extract a semantic embedding from the observations (Kumar et al., 2018). We use a pretrained EDDI to embed the current observed features into a 100-dimensional feature vector.

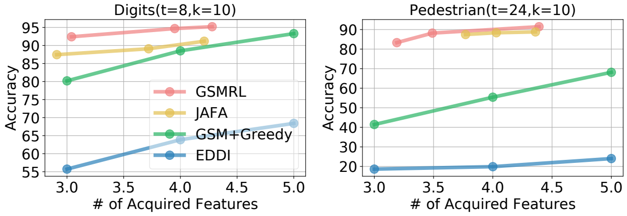

Figure 8. Classification on time series.

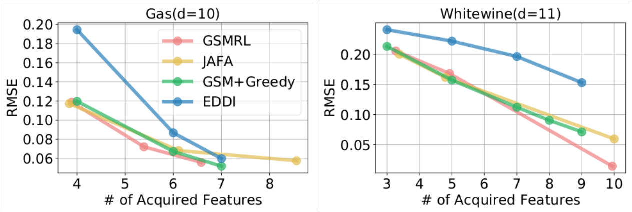

Figure 9. RMSE for unsupervised tasks.

An agent then takes the embedding as input and predicts the next acquisition (PPO+Embed). Figure 13 compares our method with these alternatives. We also present the results from a model-free approach as a baseline. We see our GSMRL outperforms other model-based approaches by a large margin.

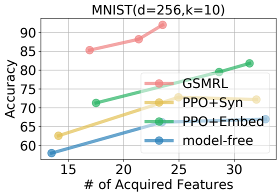

Figure 13. Other model-based approaches.

Surrogate Models Our method relies on the surrogate model to provide intermediate rewards and auxiliary information. To better understand the contributions each component does to the overall framework, we conduct ablation studies using the MNIST dataset. We gradually drop one component from the full model and report the results in Fig. 14. The 'Full Model' uses both intermediate rewards and auxiliary information. We then drop the intermediate rewards and denote it as 'w/o rm'. The model without auxiliary information is denoted as 'w/o aux'. We further drop both components and denote it as 'w/o rm & aux'. From Fig. 14, we see these two components contribute significantly to the final results. We also compare models with and without the surrogate model. For models without a surrogate model, we train a classifier jointly with the agent as in JAFA. We plot the smoothed rewards using moving window average during training in Fig. 15. The agent with a surrogate model not only produces higher and smoother rewards but also converges faster.

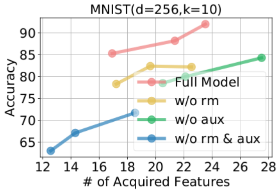

Figure 14. Ablations

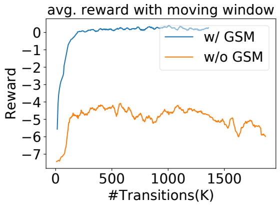

Figure 15. Rewards

Dynamic vs. Static Acquisition Our GSMRL acquires features following a dynamic order where it eventually acquires different features for different instances. A dynamic acquisition policy should perform better than a static one (i.e., the same set of features are acquired for each instance), since the dynamic policy allows the acquisition to be specifically adapted to the corresponding instance. To verify this is actually the case, we compare the dynamic and static acquisition under a greedy policy for MNIST classification. Similar to the dynamic greedy policy, the static acquisition policy acquires the feature with maximum utility at each step, but the utility is averaged over the whole testing set, therefore the same acquisition order is adopted for the whole testing set. Figure 16 shows the classification accuracy for both EDDI and GSM under a greedy acquisition policy. We can see the dynamic policy is always better than the corresponding static one. Furthermore, our GSM with a static acquisition can already outperform dynamic EDDI.

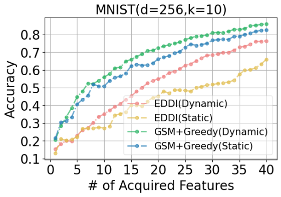

Figure 16. Compare dynamic and static acquisition strategy using greedy policies.

Greedy vs. Non-greedy Acquisition Our GSMRL will terminate the acquisition process if the agent deems the current acquisition achieves the optimal trade-off between the prediction performance and the acquisition cost. To evaluate how much the termination action affects the performance and to directly compare with the greedy policies under the same acquisition budget, we conduct an ablation study that removes the termination action and gives the agent a hard acquisition budget (i.e., forcing the agent to predict after some number of acquisitions). We can see (Fig. 17) GSMRL outperforms the greedy policy under all budgets. Moreover, we see that the agent is able to correctly assess whether or not more acquisitions are useful, since it obtains better performance when it dictates when to predict with the termination action.

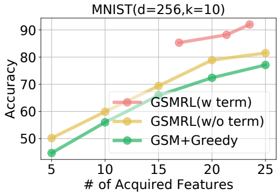

Figure 17. Acquisition with and without termination.

6. Conclusion

In this work, we reformulate the dynamics of the AFA MDP as a generative modeling task among features. We leverage a generative surrogate model to capture the state transitions across arbitrary feature subsets. The surrogate model also provides auxiliary information and intermediate rewards to assist the agent. Our GSMRL model essentially combines model-based and model-free approaches. We conduct a large scale study to evaluate our model on both supervised and unsupervised AFA problems. Our model achieves state-of-the-art performance on both problems. In future work, we will explore AFA in spatial-temporal setting with continuously indexed features.

Acknowledgements

This work was supported in part by NIH 1R01AA026879-01A1.

References

Aloimonos, J., Weiss, I., and Bandyopadhyay, A. Active vision. *International journal of computer vision*, 1(4): 333–356, 1988.

Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31(3):606–660, 2017.

Bajcsy, R. Active perception. *Proceedings of the IEEE*, 76 (8):966–1005, 1988.

Brabec, J. and Machlica, L. Bad practices in evaluation methodology relevant to class-imbalanced problems. *arXiv preprint arXiv:1812.01388*, 2018.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym, 2016.

Cai, J., Luo, J., Wang, S., and Yang, S. Feature selection in machine learning: A new perspective. *Neurocomputing*, 300:70–79, 2018.

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

Chebotar, Y., Hausman, K., Zhang, M., Sukhatme, G., Schaal, S., and Levine, S. Combining model-based and model-free updates for trajectory-centric reinforcement learning. *arXiv preprint arXiv:1703.03078*, 2017.

Cheng, R., Agarwal, A., and Fragkiadaki, K. Reinforcement learning of active vision for manipulating objects under occlusions. *arXiv preprint arXiv:1811.08067*, 2018.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Dulac-Arnold, G., Evans, R., van Hasselt, H., Sunehag, P., Lillicrap, T., Hunt, J., Mann, T., Weber, T., Degris, T., and Coppin, B. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.

Fu, Y., Zhu, X., and Li, B. A survey on instance selection for active learning. *Knowledge and information systems*, 35(2):249–283, 2013.

Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23): e215–e220, 2000.

Gong, W., Tschiatschek, S., Nowozin, S., Turner, R. E., Hernández-Lobato, J. M., and Zhang, C. Icebreaker: Element-wise efficient information acquisition with a bayesian deep latent gaussian model. In *Advances in Neural Information Processing Systems*, pp. 14820–14831, 2019.

Gu, S., Lillicrap, T., Sutskever, I., and Levine, S. Continuous deep q-learning with model-based acceleration. In *International Conference on Machine Learning*, pp. 2829–2838, 2016.

He, H., Eisner, J., and Daume, H. Imitation learning by coaching. In *Advances in Neural Information Processing Systems*, pp. 3149–3157, 2012.

He, H., Mineiro, P., and Karampatziakis, N. Active information acquisition. *arXiv preprint arXiv:1602.02181*, 2016.

Heess, N., Wayne, G., Silver, D., Lillicrap, T., Erez, T., and Tassa, Y. Learning continuous control policies by stochastic value gradients. In *Advances in Neural Information Processing Systems*, pp. 2944–2952, 2015.

Ivanov, O., Figurnov, M., and Vetrov, D. Variational autoencoder with arbitrary conditioning. *arXiv preprint arXiv:1806.02382*, 2018.

Jayaraman, D. and Grauman, K. Learning to look around: Intelligently exploring unseen environments for unknown tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1238–1247, 2018.

Konyushkova, K., Sznitman, R., and Fua, P. Learning active learning from data. In *Advances in Neural Information Processing Systems*, pp. 4225–4235, 2017.

Kumar, A., Eslami, S., Rezende, D. J., Garnelo, M., Viola, F., Lockhart, E., and Shanahan, M. Consistent generative query networks. *arXiv preprint arXiv:1807.02033*, 2018.

Kybic, J. High-dimensional entropy estimation for finite accuracy data: R-nn entropy estimator. In *Biennial International Conference on Information Processing in Medical Imaging*, pp. 569–580. Springer, 2007.

LeCun, Y. The mnist database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*, 1998.

Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., and Teh, Y. W. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pp. 3744–3753. PMLR, 2019.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., and Liu, H. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, 50(6):1–45, 2017.

Li, Y. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*, 2017.

Li, Y., Akbar, S., and Oliva, J. B. Acflow: Flow models for arbitrary conditional likelihoods. *arXiv preprint arXiv:1909.06319*, 2019.

Ling, C. X., Yang, Q., Wang, J., and Zhang, S. Decision trees with minimal costs. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 69, 2004.

Ma, C., Tschiatschek, S., Palla, K., Hernández-Lobato, J. M., Nowozin, S., and Zhang, C. Eddi: Efficient dynamic discovery of high-value information with partial vae. *arXiv preprint arXiv:1809.11142*, 2018.

Miao, J. and Niu, L. A survey on feature selection. *Procedia Computer Science*, 91:919–926, 2016.

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

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

Ng, A. Y., Harada, D., and Russell, S. Policy invariance under reward transformations: Theory and application to reward shaping. 1999.

Pong, V., Gu, S., Dalal, M., and Levine, S. Temporal difference models: Model-free deep rl for model-based control. *arXiv preprint arXiv:1802.09081*, 2018.

Rückstieß, T., Osendorfer, C., and van der Smagt, P. Sequential feature selection for classification. In *Australasian Joint Conference on Artificial Intelligence*, pp. 132–141. Springer, 2011.

Russo, D., Van Roy, B., Kazerouni, A., Osband, I., and Wen, Z. A tutorial on thompson sampling. *arXiv preprint arXiv:1707.02038*, 2017.

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

Shim, H., Hwang, S. J., and Yang, E. Joint active feature acquisition and classification with variable-size set encoding. In *Advances in neural information processing systems*, pp. 1368–1378, 2018.

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

Thompson, W. R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

Yoo, D. and Kweon, I. S. Learning loss for active learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 93–102, 2019.

Zannone, S., Hernandez Lobato, J. M., Zhang, C., and Palla, K. Odin: Optimal discovery of high-value information using model-based deep reinforcement learning. In *Real-world Sequential Decision Making Workshop, ICML*, June 2019.

Zubek, V. B., Dietterich, T. G., et al. Pruning improves heuristic search for cost-sensitive learning. 2004.