A GRU-based Meta-learning Model Based on Active Learning

Honglan Huang
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: huanghonglan17@nudt.edu.cn

Yanghe Feng
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: fengyanghe@yeah.net

Shixuan Liu
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: liushixuan19@nudt.edu.cn

Jincai Huang
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: huangjincai@nudt.edu.cn

Zhong Liu
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: liuzhong@nudt.edu.cn

Abstract—In the realities of machine learning, labeling a data set may be expensive, tedious, or extremely difficult and it is often not easy to choose the common criteria for active learning to select samples for different data sets. In order to solve these difficulties, this paper introduces a Gated Recurrent Unit (GRU)-based meta-learner model, which combines active learning with reinforcement learning and uses it in a stream-based one-shot learning task. Based on the uncertainty of the instances, the model learns an action strategy that determines when to predict or request the label of each instance. Through the experiments on Omniglot dataset, the model shows its ability to achieve a good prediction accuracy with few label requests.

Keywords—active learning; meta learning; reinforcement learning; GRU.

I. INTRODUCTION

Active learning [1] uses unlabeled and labeled instances to train a highly accurate classifier to reduce the workload of human experts. The algorithm simulates the human learning process, selects part of instances to label and iteratively improves the generalization performance of the classifier. Therefore, it has been widely used in information retrieval and text, image and speech recognition in recent years.

Most of the traditional active learning methods are carefully formulating some criteria for selecting samples, such as uncertainty sampling [2], query-by-committee [3], margin [4] and representative and diversity-based sampling [5]. It's hard to pinpoint which approach is better, because each approach starts from a reasonable, meaningful and completely different motivation. However, for now, there is no universal criteria that performs well on all datasets. This paper introduces a learning-based approach, rather than a manually-designed sample-selection criterion, which integrates active learning algorithm with reinforcement learning. Our approach not only learns to use small supervisors to classify instances, but also learns about label-querying strategies. The model adopts a stream-based active learner that considers the online environment for active learning.

Our primary contribution in this work is using a GRU to improve the active one-shot learning model introduced by Woodward et al. [6]. We evaluate the model on Omniglot (“active” variants of existing one-shot learning tasks [7]), and our experiment results show that it can learn label-querying strategies efficiently with simpler structure.

The rest of this paper is structured as follows. Section II summarizes the existing approaches related to our work. Section III presents the task and the general framework of our proposed active learning model. Section IV presents the experiments and interprets the results. Finally, we conclude this paper in Section V.

II. RELATED WORK

Active learning has been well studied in the past few decades. The main idea of the active learning is that a learner should achieve higher accuracy with fewer labeled training instances, if it is able to choose the training instances from which it learns. Numerous algorithms have been proposed to design the criteria for the selection of which examples to label [2][5][8][9]. However, most of these traditional active learning methods are based on heuristics, which may be limited when the data distribution of the underlying learning problems vary (e.g. a new class appears). Instead, we used a meta-learning approach to train an active learner via reinforcement learning to solve a one-shot learning task. The idea of combing active learning and reinforcement learning was recently investigated by Woodward et al. [6]. In contrast to their work, we used a GRU instead of a Long Short-Term Memory (LSTM) network to approximate the action-value function in reinforcement learning. Compared with LSTM, GRU has fewer parameters, so it can effectively speed up the training process [10] and requires fewer samples, which is more suitable for our one-shot active learning task. Similar inspirations have also been studied by Bachman et al. [7] Pang et al. [11] and Puzanov et al. [12].
III. MODEL DESCRIPTION

A. Task Description

We mainly focus on the stream-based online active learning scenario [6], in which instances can be continually obtained from the data stream and presented in an exogenously-determined order. Thus, the input of the model is a stream of images, the label of which is queried or predicted by our model. The performance of our model was improved with a short training episode and a small number of examples per class to maximize the performance of test episodes, which consists of classes that are not encountered in training. The structure of our task can be seen in Figure 1. The classes and their labels and the specific samples are shuffled and randomly presented at each episode.

At each time step, the input of the model is an image along with a vector that depends on the output taken previous time step along with the next image. The true label \( y_t \) of image \( x_t \) is then provided at the next time step to 1, which means the output of timestep \( t \) is \([0, 1]\). Thus, the reward of this label request action is \( R_{req} \).

If the model requests the label of the image \( x_t \), it sets the final bit of the output vector of this timestep to 1, which means the output of timestep \( t \) is \([0, 1]\). Thus, the reward of this label request action is \( R_{req} \).

The true label \( y_t \) of image \( x_t \) is then provided at the next time step along with the next image \( x_{t+1} \), so the input of \( t+1 \) is \((y_t, x_{t+1})\). Alternatively, if the model makes a prediction of \( x_t \), it sets one of the first \( k \) bits of the output vector to represent \( \hat{y} \), so the output of this step is \([\hat{y}, 0]\). The reward of this action is \( R_{cor} \) if the prediction is correct or \( R_{inc} \) if incorrect. If a prediction is made at time step \( t \), no information of its true label \( y_t \) is supplied at the next time step \( t+1 \), then the input is \((0, x_{t+1})\) instead.

B. Methodology

We use a model-free reinforcement learning method Q-learning to learn an optimal action strategy, which can maximize the rewards. The loss function we use is defined as,

\[
\mathcal{L}(\theta) := \sum_{t} \left[ Q(o_t, a_t) - \left( r_t + \gamma \max_{a_{t+1}} Q(\ast_{t+1}, a_{t+1}) \right) \right]^2
\]

where, \( \theta \) are the parameters of the function approximator, \( o_t \) are the observations such as images that the agent receives, \( a_t \) is the action the model chooses at timestep \( t \), \( Q^* \) is the optimal value of action-value function \( Q \).

We use a GRU [13] network connected to a linear output layer to adopt the methodology of using action-value function \( Q(o_t, a_t) \) in Q-learning. \( Q(o_t) \) outputs a vector, where each element corresponds to an action:

\[
Q(o_t, a_t) = Q(o_t) \cdot a_t
\]

\[
Q(o_t) = W^{ha} h_t + b^q
\]

where, \( b^q \) is the action-value bias, \( h_t \) is the output of the GRU, \( W^{ha} \) are the weights mapping from the GRU output to action-values.

IV. EXPERIMENTS

A. Setup

The experiments were carried out on the Omniglot dataset [14] that contains 32460 instances having 1623 classes of characters from 50 different alphabets, each handwritten by 20 different persons. The dataset was randomly divided into 1200 characters for training and the rest 423 characters are kept for testing. The images were downscapabled to 28x28 pixels and each pixel was normalized between 0.0 and 1.0. 30 Omniglot images from 3 random classes were chosen in each episode. Each class of images was randomly rotated in \{0°, 90°, 180°, 270°\}. A GRU with 200 hidden units was used here. We optimized the parameters of our model using Adam with the default parameters [15]. A grid search was performed over the following parameters, and the parameters of the results reported in this article are listed here. During the training process, epsilon-greedy (\( \varepsilon = 0.23 \)) exploration is set for actions selection. The learning rate of training was set to 0.001 and the discount factor \( \gamma \) was set to 0.8. The reward values were set as: \( R_{cor} = +1, R_{inc} = -1, R_{req} = -0.3 \).

B. Results

Here, we present the results of our experiments. The 1st, 2nd, 5th of all classes in each episode were identified. After 100,000 episodes, training is ceased and the model was given 10,000 more test episodes. No further updates occurred during these episodes. The results can be seen in Figure 3.
stable at 85%, while the label request rate was stable at 12%. Compared with supervised learning, our model greatly reduces the dependence on the number of labels and human workload, and achieves decent prediction accuracy. At the same time, our method speeds up the convergence of the algorithm compared to the LSTM-based method [6].

V. CONCLUSION

We introduced a GRU-based meta-learning model that learns active learning in an reinforcement learning way and experimented it on Omniglot one-shot learning tasks. Our results show that our model can learn an optimal query strategy and achieve a good classification accuracy with a small amount of labeled data.

As we used a GRU network to approximate the action-value function in reinforcement learning, a promising direction is that the GRU network can be replaced by a more sophisticated one-shot learning approach such as Matching Network [16] or Memory-Augmented Neural Networks [17]. We will leave this as our future work.

REFERENCES