The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Active Lifelong Learning with "Watchdog"

Gan Sun,[1, 2] **Yang Cong,**[1] **Xiaowei Xu**[3]
[1]State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, China. *
[2]University of Chinese Academy of Sciences, China.
[3]Department of Information Science, University of Arkansas at Little Rock, USA.
sungan@sia.cn, congyang81@gmail.com, xwxu@ualr.edu

Abstract

Lifelong learning intends to learn new consecutive tasks depending on previously accumulated experiences, i.e., knowledge library. However, the knowledge among different new coming tasks are imbalance. Therefore, in this paper, we try to mimic an effective "human cognition" strategy by actively sorting the importance of new tasks in the process of unknown-to-known and selecting to learn the important tasks with more information preferentially. To achieve this, we consider to assess the importance of the new coming task, i.e., unknown or not, as an outlier detection issue, and design a hierarchical dictionary learning model consisting of two-level task descriptors to sparse reconstruct each task with the ℓ_0 norm constraint. The new coming tasks are sorted depending on the sparse reconstruction score in descending order, and the task with high reconstruction score will be permitted to pass, where this mechanism is called as "watchdog". Next, the knowledge library of the lifelong learning framework encode the selected task by transferring previous knowledge, and then can also update itself with knowledge from both previously learned task and current task automatically. For model optimization, the alternating direction method is employed to solve our model and converges to a fixed point. Extensive experiments on both benchmark datasets and our own dataset demonstrate the effectiveness of our proposed model especially in task selection and dictionary learning.

Introduction

Multi-task learning (MTL) (Caruana 1997) is a learning paradigm designed to learn multiple tasks such as classification or regression task simultaneously. One of the basic assumptions for MTL is that taking into account the related / shared information among different tasks can lead to a better generalization performance than independently learning single task. In this setting, dramatic successes have been achieved in many areas by utilizing MTL, such as medical diagnosis (Bi et al. 2008), handwritten character recognition (Obozinski, Taskar, and Jordan 2007), relative attributes learning (Chen, Zhang, and Li 2014) and text classification

*This work is supported by NSFC (61722311, U1613214, 61533015), CAS-Youth Innovation Promotion Association Scholarship (2012163).

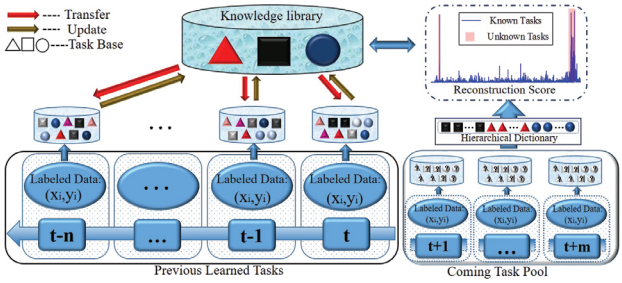

Figure 1: Demonstration of our active lifelong learning framework with a pool of m candidate tasks, where different shapes and colors denote different task base and weights, respectively. Whether each coming task is unknown or not is determined via the reconstruction score, and the one with higher reconstruction score will be marked as the next task.

(Zhang, Ghahramani, and Yang 2008). However, when encountering a new task, such MTL learning system must be capable of efficiently and continually acquiring knowledge over a serial tasks, i.e., *lifelong learning*.

In the context of *lifelong learning* framework, most state-of-the-arts (Saha et al. 2011; Ruvolo and Eaton 2013b; Ammar et al. 2014) intend to learn tasks sequentially while maximizing its performance across all learned tasks. The major procedure in these methods is to transfer knowledge from the previously learned tasks to the next coming task, where new task arrives in a stochastic manner. Further, nearly equivalent model performance has been demonstrated in comparison with batch multi-task learning (Kumar and Daumé 2012) while also exhibiting impressive speedups. However, *tasks in a candidate pool do not have equal knowledge*. Some new tasks that are similar to tasks learned before are well-known and further being redundant, while other irrelevant / strange tasks are unknown. To accumulate knowledge rapidly in "human learning", it is reasonable to filter the well-known tasks out automatically, and pay more exogenous attention (And and Yantis 1997) to the unknown / novel information. Take birds categorization task as an example: suppose that we have only learned different species of Gull, e.g., Ring billed Gull. When a pool of candidate tasks contains California Gull, Herring Gull and Cardinal, categorization task of whether a bird is California Gull or

not can be learned easily by transferring previously accumulated knowledge such as *does it have webbed feet like Ring billed Gull?* On the contrary, a Cardinal will be marked as unknown category since similar knowledge has not been collected over learned tasks. Therefore, selecting Cardinal from these candidate tasks as next task can acquire more available knowledge and information for the lifelong learning system.

Motivated by above analysis, as depicted in Figure 1, we propose a new Active Lifelong Learning (AcLL) framework in the context of Efficient Lifelong Learning Algorithm (ELLA) (Ruvolo and Eaton 2013b), where AcLL integrates a hierarchical dictionary in a perspective of outlier detection scenario, called "watchdog". Specifically, the hierarchical dictionary is learned over the previously learned tasks with the help of two-level task descriptor, i.e., high-dimensional subdictionary and low-dimensional one. As a pool of candidate tasks arrives at the "watchdog", the new task models can be best sparse reconstructed over this hierarchical dictionary via ℓ_0 norm. Whether a coming task is unknown (outlier) or not is detected by sorting the reconstruction cost in descending order (i.e, the tasks with higher reconstruction cost are considered as unknown, and vice versa), and the task with profound knowledge will be allowed to pass. Afterwards, the selected task can be encoded by the knowledge library, and knowledge in the new task will then refine the base of both knowledge library and hierarchical dictionary. For model optimization, the alternating direction strategy is presented to solve our task selection framework. Finally, we evaluate our proposed AcLL model against several active selection methods on several multi-task datasets, and several dictionary learning methods on extended Yale B dataset. The experimental results strongly support our proposed framework.

The novelty of our proposed framework is threefold:

- Benefitting from outlier detection scenario, called "watchdog" in this paper, we design a lifelong machine learning framework, referred to as Active Lifelong Learning (AcLL), to autonomously judge whether the coming task is unknown or not to learn instead of human beings.
- A hierarchical dictionary framework consisting of high-dimensional and low-dimensional subdictionaries is proposed in the "watchdog" to select the tasks actively, which can sparsely reconstruct new coming tasks while preserving the previous task distribution well.
- Experimental results on multi-task datasets and extended Yale B dataset show that our proposed AcLL framework outperforms the state-of-the-art active task selection methods and dictionary learning methods, respectively.

The rest of the paper is organized as follows. The first section gives a brief review of some related works. The second one introduces our proposed active lifelong machine learning framework. Then, how to solve the proposed model efficiently via alternating direction method of multipliers is proposed. The last two sections report the experimental results and conclusion of this paper.

Related Works

In this section, we firstly provide a brief review of the most related works on learning tasks from "easy" to "hard": **Self-paced / Curriculum learning**, and then introduce some state-of-the-art methods related to **Active Learning**.

For the **Self-paced Learning** based MTL, the representative methods (Li et al. 2017; Murugesan and Carbonell 2017) aim to first learn the more "simple" or "easy" instances or tasks, and then add "complex" or "hard" ones gradually, inspired by the established human educational process. However, simple incorporation of self-paced learning into the multi-task learning may cause intractable increasing in the number of learning parameters and thus induce computation and storage requirements to grow gradually. For the **Curriculum Learning** based MTL, instead of learning all tasks jointly, the state-of-the-art approaches (Pentina, Sharmanska, and Lampert 2015) firstly sort multiple tasks in the best order, and then solve each task via transferring useful information between subsequent tasks. The objective function of searching next task which is not included in the task order π is given as:

$$\min_{w_k} : \left\| w_k - w_{\pi(k-1)} \right\|_2^2 + \mathcal{L}(w_k), \tag{1}$$

where w_k is the task parameter of most similar one. However, such method may be not flexible since it contains a fixed task order, and only transfer knowledge between local subsequent tasks instead of the global task relationship.

For the **Active Leaning**, the representative methods (Tong and Koller 2001; Cohn, Ghahramani, and Jordan 1996) typically focus on the most uncertain or informative samples. (Saha et al. 2010) has extended active learning into MTL via adopting an adaptive matrix to evaluate the informativeness of an incoming sample across all the tasks. It has also been shown to generally reduce the size of training data (Saha et al. 2011; Zhang 2010). In the context of lifelong learning, (Ruvolo and Eaton 2013a) proposes to employ two effective active task selection mechanisms for selecting the next best task: Information maximization and Diversity heuristic. Unlike most existing active MTL models which intend to enhance the model performance, (Ruvolo and Eaton 2013a) focuses on providing maximal benefit to learning future tasks. However, (Ruvolo and Eaton 2013a) aims to learn the knowledge library efficiently, which may cause the loss of original task parameter details. Therefore, in order to alleviate the problem of existing models, we extend outlier detection scenario (Cong, Yuan, and Liu 2011) to lifelong learning system by introducing an efficient hierarchical dictionary for reconstructing coming tasks, and further select the next unknown task to learn.

Active Lifelong Learning

Preliminaries

In this paper, we firstly begin by introducing batch multi-task learning (MTL). Suppose that we have m batch learning tasks $\{(x_i^t, y_i^t)\}_{i=1}^{n_t}$ $(t = 1, ..., m)$, where $x_i^t \in \mathbb{R}^d$, y_i^t and n_t denote the i-th sample of the t-th task, the corresponding output of the i-th sample and the total number of samples of

the t-th task, respectively. Let us define $f_t : X^t \rightarrow Y^t$ as the associated predictor for the t-th learning tasks, such as $Y^t \in \{-1, 1\}$ for binary classification problem and $Y^t \in \mathbb{R}$ for regression problem. Moreover, the objective function for batch MTL is summarized as:

$$\min_{W} : \quad \sum_{t=1}^{m} \sum_{i=1}^{n_t} \mathcal{L}(w_t^T x_i^t, y_i^t) + \Phi(W), \tag{2}$$

where the weight matrix W is defined as $[w_1, ..., w_m] \in \mathbb{R}^{d \times m}$ with each column $w_t \in \mathbb{R}^d$ corresponding to the t-th task. The first term in Eq. (2) is the loss function for regression or classification problem, and second term $\Phi(W) : \mathbb{R}^{d \times m} \rightarrow \mathbb{R}^+$ is the regularization term to shrink model complexity. However, batch MTL cannot adapt to a new task without accessing to previous training data, which may result in a computation and storage burden.

Background on Lifelong Learning

Different from the traditional batch MTL, the lifelong learning could incrementally learn new tasks. More specifically, the learner has not enough *prior* knowledge about the total number of tasks, task order or their distributions. When the learner receives data from the t-th task at each time step (either a new task or learned tasks), it needs to optimize the model performance across all the tasks encountered so far. The following section provides a common lifelong learning framework without active task selection: Efficient Lifelong Learning Algorithm (ELLA) (Ruvolo and Eaton 2013b).

The key idea of ELLA is that the model vector w_t of each coming task can be represented as a linear combination of k basis tasks, i.e., shared knowledge library $L \in \mathbb{R}^{d \times k}$. More specifically, the model parameters for task t are given by $w_t = Ls_t$, where s_t is the sparse coefficients over the knowledge library. Therefore, the tasks with the same basis can be considered as belonging to the same group, while the tasks whose basis are orthogonal between each other are concerned from different groups. Meanwhile, the partially overlapping of base to model the online tasks which has something in common but not in the same group. Under this assumption, the lifelong learning objective function for ELLA is:

$$e_T(L) = \frac{1}{T} \sum_{t=1}^{T} \min_{s_t} \left\{ \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}(f(x_i^t; Ls_t), y_i^t) + \mu \left\| s_t \right\|_1 \right\} + \lambda \left\| L \right\|_F^2, \tag{3}$$

where μ and λ provide a trade-off with the loss function, and loss function $\mathcal{L}$ can be squared loss, logistic loss, etc. Similarly, this optimization problem is extended from GO_MTL (Kumar and Daumé 2012), which focuses on the batch MTL in practical applications. Even through ELLA has achieved dramatic improvement across reinforcement learning (e.g., (Ammar et al. 2014)), the learner in ELLA has no control over the order in the coming tasks. In the next section, we consider an active lifelong leaning framework which can select the unknown tasks to learn.

Our Active Lifelong Learning (AcLL)

This subsection introduces our Active Lifelong Learning framework (AcLL) in the perspective of outlier detection scenario, called "watchdog", i.e., we consider that the coming tasks in a pool are not equally in the lifelong learning.

We firstly formalize this problem following the basic setting of active task selection (Ruvolo and Eaton 2013a). Given a pool of candidate tasks $f_{t+1}, \ldots, f_{t_{\text{pool}}}$, where $t + 1 \leq t_{\text{pool}} \leq t_{\max}$ (the value of $t_{\max}$ could be a fixed value or be set dynamically during learning), the learner should actively choose the index t_{next} of the next task from $\{t + 1, \ldots, t_{\text{pool}}\}$. Once the learner decides the index of next task, the corresponding training dataset are transmitted to the learner, allowing knowledge library L to learn the new task. Obviously, lifelong learning system needs judge mechanism like "watchdog", which can detect whether the coming tasks are known or unknown, e.g., the coming persons are familiar or strange, and update the accumulated knowledge in the "watchdog" real-timely. **Since the basic assumption of MTL is that the learned tasks share the common relationship and information, there exists an optimal dictionary playing the same role as "watchdog".** Specifically, it can reconstruct the coming tasks, and the tasks which can not be well represented are regarded as unknown / outlier. Motivated by (Cong, Yuan, and Liu 2011; 2013), in this paper, we focus on "watchdog" based task selection framework by considering it as an outlier detection problem, i.e., AcLL. Generally, there are two steps for this challenge:

- **Online Dictionary Learning:** Given the previously learned tasks pool as $W = [w_1, w_2, \ldots, w_t] \in \mathbb{R}^{d \times t}$, where each column vector $w_i \in \mathbb{R}^d$ denotes a learned task model. Our goal is to establish a dictionary D such that it is well adapted to represent learned tasks W and reconstruct a set of candidate tasks. Formally, the objective function of learning D can be formulated as:

$$\min_{D} : \ \frac{1}{2} \left\| W - DR \right\|_F^2 + \Phi(D), \tag{4}$$

where $D = [d_1, d_2, \ldots, d_k] \in \mathbb{R}^{d \times k}$ has the same size as the library L, $R = [r_i, r_2, \ldots, r_t] \in \mathbb{R}^{k \times t}$ is the corresponding sparse representation matrix, and second term $\Phi(D) : \mathbb{R}^{d \times k} \rightarrow \mathbb{R}^+$ is the regularization term. Due to the fact that two-level task descriptor can improve knowledge transfer between multiple tasks in (Isele, Rostami, and Eaton 2016), we propose to learn a two-level dictionary called hierarchical dictionary in this paper. More specifically, we assume that there are two components contained in the hierarchical dictionary: one component is high-dimensional global dictionary, and the other component is a parameterized low-dimensional local dictionary for capturing shared subspace among multiple atoms. Therefore, the dictionary D over the previously learned tasks can be expressed as:

$$D = D' + \Theta A, \tag{5}$$

where the dictionary map Θ is cast as the form of an $d \times k$ matrix with orthonormal columns, i.e., $\Theta^T \Theta = I_k$, which

is imposed to make the problem tractable. $D \in \mathbb{R}^{d\times k}$, D' and ΘA correspond to the full dictionary space, the high-dimensional global dictionary, and the low-dimensional local dictionary, respectively. In order to model shared subspace over Θ, we impose a $\ell_{2,0}$-norm constraint on matrix A in the setting of dictionary selection (Cong et al. 2017). Since the knowledge in "watchdog" should be online updated as the new task comes, we describe an online hierarchical dictionary learning framework in this paper. Mathematically, given the sparse reconstruction coefficient r_i's, the dictionary learning problem can be formulated as:

$$\begin{aligned}\min_{D',A,\Theta^T\Theta=I_k} &: \frac{1}{2}\sum_{i=1}^{t}\|w_i-(D'+\Theta A)r_i\|_2^2+\lambda_1\|D'\|_F^2\\ s.t. &: \|A\|_{2,0}\le\tau,\end{aligned} \tag{6}$$

where $\lambda_1 \ge 0$ and $\tau \ge 0$ are tuning parameters to indicate the importance of the corresponding regularization component. As we can see, the proposed formulation in Eq. (6) subsumes several dictionary learning methods as special cases: by setting $\tau = 0$, the formulation in Eq. (6) falls back to the common online dictionary learning (Mairal et al. 2009) in some extent; by setting $\lambda_1 = \infty$, it reduces to coupled dictionary learning (Zhu et al. 2016).

- **Sparse Reconstruction Cost:** When a pool of candidate tasks comes, each task can be linearly constructed by only a few bases in the hierarchical dictionary D, i.e., $w_{t+1} = \sum_j d_j r_j$, where $r_j \in \mathbb{R}$ is the coefficient corresponding to d_j. Motivated by (Cong, Yuan, and Liu 2011; 2013), whether w_{t+1} is unknown or not is determined by the linear reconstruction cost, defined as:

$$S(w_{t+1}) = \frac{1}{2}\|w_{t+1}-Dr^*\|_2^2,\ \ s.t.: \|r^*\|_0 \le \mu_1, \tag{7}$$

where r^* is the optimal sparse coding coefficients, $\mu \ge 0$ controls the sparsity of r^*, and $S(w_{t+1})$ can estimate how easily the coming task w_{t+1} can be modeled by the knowledge library L.

When the "watchdog" detects the t_{next}-th task with higher reconstruction cost as the unknown task and allows it pass, the learner in lifelong system will be updated as:

$$\begin{aligned} s_{t_{\text{next}}} &\leftarrow \arg\min_{s_{t_{\text{next}}}} \ell(L_t, s_{t_{\text{next}}}, w_{t_{\text{next}}}, \Omega_{t_{\text{next}}}),\\ L_{t+1} &\leftarrow \arg\min_{L} g_t(L),\\ g_t(L) &= \frac{1}{T}\sum_{t=1}^{T}\ell(L, s_{t_{\text{next}}}, w_{t_{\text{next}}}, \Omega_{t_{\text{next}}}),\end{aligned} \tag{8}$$

where $\ell(L,s,w,\Omega_{t_{\text{next}}}) = \|w-Ls\|^2_{\Omega_{t_{\text{next}}}} + \mu_2\|s\|_1$, knowledge library L_t corresponds to the beginning of the t-th iteration, $\Omega_{t_{\text{next}}}$ denotes the Hessian matrix of the loss function $\mathcal{L}$ with respect to $w_{t_{\text{next}}}$, t_{next} is the index of current task. Additionally, $w_{t_{\text{next}}}$ and $\Omega_{t_{\text{next}}}$ are from the training data of the t_{next}-th task using a single task learner:

$$(w_{t_{\text{next}}}, \Omega_{t_{\text{next}}}) \leftarrow \text{singleTaskLearner}(X^{t_{\text{next}}}, Y^{t_{\text{next}}}). \tag{9}$$

Algorithm 1 Active Lifelong Learning (AcLL)

Input: $\lambda_1 > 0$, $\lambda_2 > 0$, $\mu_1 > 0$, $\mu_2 > 0$, $t = 0$.
Initialize: L_0, D_0, Θ_0, A_0 and Σ_0.
while isMoreTrainingDataAvailable() **do**
  $\{X^i, Y^i, w_i, i\}_{i=t+1}^{t_{\max}} \leftarrow$ getCandidateTaskData()
  $\{X^{\text{new}}, Y^{\text{new}}, w_{\text{new}}\} \leftarrow$ OutlierSelection() via Eq. (7)
  $(w_{\text{new}}, \Omega_{\text{new}}) \leftarrow \text{singleTaskLearner}(X^{\text{new}}, Y^{\text{new}})$
  $s_{\text{new}} \leftarrow \arg\min_{s_{\text{new}}} \ell(L_t, s_{\text{new}}, w_{\text{new}}, \Omega_{\text{new}})$
  $L_{t+1} \leftarrow \arg\min_L g_t(L)$
  Update D_{t+1} via **Algorithm 2**
  Update A_{t+1} via Eq. (16)
  Update Θ_{t+1} via Eq. (20)
end while
Return: L_{t+1}, S_{t+1}, D_{t+1}, Θ_{t+1} and A_{t+1}.

where $\text{singleTaskLearner}(\cdot,\cdot)$ can be defined as linear regression or logic regression. Generally, we summarize our Active Lifelong Learning framework (AcLL) in **Algorithm 1**.

Model Optimization

This section describes how to achieve online dictionary learning since it is the main optimization problem in **Algorithm 1**. Specifically, the optimization problem in Eq. (6) is non-convex due to its orthonormal constraints and penalty term with respect to Θ, and A. In order to envelop ℓ_0-norm, we replace it with ℓ_1-norm in our formulation, and convert our proposed formulation as:

$$\min_{\substack{D,A,\\ \Theta^T\Theta=I_k}} \frac{1}{2}\sum_{i=1}^{t}\|w_i-Dr_i\|_2^2+\lambda_1\|D-\Theta A\|_F^2+\lambda_2\|A\|_{2,1}, \tag{10}$$

where $D = D' + \Theta A$, and the regularization term $\|A\|_{2,1}$ guarantees that the optimal solution of A is row sparsity, which can select the discriminative features in the latent space Θ to encode coming tasks. In the following, we introduce the proposed update rules in brief.

Online Dictionary D Update: Assuming we have selected the $t+1$-th task as the most unknown one, and the decomposition r_{t+1} of w_{t+1} over the dictionary D_t obtained at the previous iteration. To store the previous knowledge, we then introduce two statistical records:

$$M_{t+1} = M_t + r_{t+1}r_{t+1}^T,\ C_{t+1} = C_t + w_{t+1}r_{t+1}^T, \tag{11}$$

where $M_t = \sum_{i=1}^{t} r_i r_i^T$, and $C_t = \sum_{i=1}^{t} w_i r_i^T$. Therefore, information of new task is stored as $r_{t+1}r_{t+1}^T$ and $w_{t+1}r_{t+1}^T$. By adopting D_t as a warm start, we then have:

$$\begin{aligned} D_{t+1} &= \arg\min: \frac{1}{2}\sum_{i=1}^{t+1}\|w_i-Dr_i\|_2^2+\lambda_1\|D-\Theta_t A_t\|_F^2,\\ &= \arg\min: \frac{1}{2}\big(\text{Tr}(D^T D(M_{t+1}+\lambda_1 I_k))\\ &\qquad -\text{Tr}(D^T(C_{t+1}+\Theta_t A_t))\big).\end{aligned} \tag{12}$$

Algorithm 2 Online Dictionary Update

1: **Input:** New task parameter w_{t+1}, corresponding coefficient r_{t+1}; $D = D_t$, M_t, C_t; $\lambda_1 > 0$.
2: **for** $i = 1 : d$ **do**
3: $M_{t+1}^i \leftarrow M_t^i + (r_{t+1} r_{t+1}^T)^i$
4: $C_{t+1}^i \leftarrow C_t^i + (w_{t+1} r_{t+1}^T)^i$
5: Solve linear system via Eq. (13).
6: **end for**
7: **Return:** D_{t+1}, M_{t+1} and C_{t+1}.

After evaluating the derivative of Eq. (12) and setting it to zeros, the global optimum of D_{t+1} can be obtained and further lead to the following linear problem:

$$(C_{t+1} + \Theta_t A_t)^i = D(i,:)(M_{t+1} + \lambda_1 I_k)^i. \quad (13)$$

With this configuration, D_{t+1} can be updated via **Algorithm 1** in an online manner.

Coefficient A Update: With fixed matrix D_{t+1} and Θ_t, A is the single variable in this subproblem, where the optimization function can be rewritten as:

$$\min_A : \ \lambda_1 \left\| D_{t+1} - \Theta_t A \right\|_F^2 + \lambda_2 \left\| A \right\|_{2,1}. \quad (14)$$

A simple idea is to set the derivative of A as 0, and adopting the property that $\Theta_t^T \Theta_t = I_k$, we have:

$$\lambda_1(\Theta_t^T \Theta_t A - \Theta_t^T D_{t+1}) + \lambda_2 \Sigma A = 0, \quad (15)$$

where Σ is a diagonal matrix (Nie et al. 2010) of A_t with $\Sigma_{ii} = \frac{1}{2\|a_i\|_2}, i = 1, \ldots, k$. Therefore, the solution of A can be given as:

$$A = \lambda_1(\lambda_1 I_k + \lambda_2 \Sigma)^{-1} \Theta_t^T D_{t+1}. \quad (16)$$

Dictionary Map Θ Update: With fixed matrix D_{t+1} and A_{t+1}, the optimization function with respect to Θ can be rewritten as:

$$\min_{\Theta^T\Theta = I_k} : \ \lambda_1 \left\| D_{t+1} - \Theta A_{t+1} \right\|_F^2 + \lambda_2 \left\| A_{t+1} \right\|_{2,1}. \quad (17)$$

After substituting A_{t+1} with Eq. (16), the objective function is written as follows:

$$\begin{aligned} \min_{\Theta^T\Theta = I_k} : \lambda_1 \left\| D_{t+1} - \lambda_1 \Theta V^{-1} \Theta^T D_{t+1} \right\|_F^2 \\ + \lambda_2 \mathrm{Tr}(\lambda_1^2 D_{t+1}^T \Theta V^{-1} D_{t+1} V^{-1} \Theta^T D_{t+1}), \end{aligned} \quad (18)$$

where $V = \lambda_1 I_k + \lambda_2 \Sigma$. After using simple linear algebra, we then can rewrite Eq. (18) as:

$$\min_{\Theta^T\Theta = I_k} : \ \lambda_1 \mathrm{Tr}\Big(D_{t+1}^T (I_d - \lambda_1 \Theta V^{-1} \Theta^T) D_{t+1} \Big). \quad (19)$$

By eliminating $\mathrm{Tr}(D_{t+1}^T I_d D_{t+1})$, Eq. (18) is equivalent to:

$$\begin{aligned} &\max_{\Theta^T\Theta = I_k} : \ \lambda_1^2 \mathrm{Tr}(D_{t+1}^T \Theta V^{-1} \Theta^T D_{t+1}), \\ \Leftrightarrow &\max_{\Theta^T\Theta = I_k} : \lambda_1^2 \mathrm{Tr}(V^{-1} \Theta^T D_{t+1} D_{t+1}^T \Theta), \\ \Leftrightarrow &\max_{\Theta^T\Theta = I_k} : \lambda_1^2 \mathrm{Tr}\Big((\lambda_1 I_k + \lambda_2 \Sigma)^{-1} \Theta^T D_{t+1} D_{t+1}^T \Theta \Big), \\ \Leftrightarrow &\max_{\Theta^T\Theta = I_k} : \lambda_1^2 \mathrm{Tr}\Big(\Theta^T (\lambda_1 I_d + \lambda_2 \Theta \Sigma \Theta^T)^{-1} D_{t+1} D_{t+1}^T \Theta \Big). \end{aligned} \quad (20)$$

It is well-known that the solution of Θ can be relaxedly achieved by the eigen-decomposition of $(\frac{1}{\lambda_1} I_d + \frac{\lambda_2}{\lambda_1^2} \Theta_{t-1} \Sigma \Theta_{t-1}^T)^{-1} D_t D_t^T$. Note that although the input parameters in Eq. (20) contain Θ, the above solution is also effective since the proposed algorithm converges very quickly in the online style.

Computational Complexity Analysis The main computational cost in our AcLL model involves the online dictionary learning except for Eq. (8), i.e., the updating of D, Θ and A in each iteration. Specifically, the updating of D involves a $k \times k$ matrix, and the computational cost is $O(d^2k + k^3)$. A has a closed-form solution and the computational complexity is $O(k^3 + dk^2 + d^2k)$. The cost of computing Θ is $O(d^3 + d^2k + dk^2)$, in which the computational cost of matrix inversion and eigen-decomposition are $O(d^3)$. Therefore, the total computational cost of dictionary learning is $O(T(d^3 + d^2k + dk^2 + k^3))$, where T denote the total number of tasks in the lifelong learning system. Recall that, when the size of dictionary D is smaller than d (i.e., $k \ll d$), the computational cost approximates to $O(T(d^3))$.

Experiments

In this section, we carry out empirical comparisons and several experiments to validate the proposed model.

Competing Algorithms and Measurements

In our experiments, we evaluate our proposed AcLL framework on six well-known task selection strategies:

1. (Ruvolo and Eaton 2013a) concludes: Random (**ELLA-Rand**): the next task arrives randomly; Information Maximization (**ELLA-Info**): the next task should be maximize the expected information gain over the knowledge library L; Diversity (**ELLA-Diver**): the next task is chosen as the one that current library L obtains the worst performance; Diversity++ (**ELLA-Diver++**): a stochastic version of **ELLA-Diver**.
2. Curriculum learning (**CL**): this method for multiple tasks proposes to learn subsequent tasks based on the established best order.
3. Self-paced Multitask Feature Learning (***sp*MTFL**) (Murugesan and Carbonell 2017): *sp*MTFL presents to firstly select the easy tasks via parameter τ, and the objective function is given as:

$$\min_{W,\Omega \in \boldsymbol{S}_+^d} : \sum_{t=1}^m \tau_t \Big(\mathcal{L}(w_t^T X_t, y_t) + \gamma \langle w_t, \Omega^{-1} w_t \rangle \Big) + \lambda r(\tau), \quad (21)$$

where the value of τ_t controls the importance of a task in feature relathionship matrix Ω via assigning different weights.

We also compare our results with independent task learning (**ITL**), where multiple tasks are trained in an independent way. For the evaluation, we adopt the RMSE (root mean squared error) and AUC (area under curve) for the regression and classification problems, respectively.

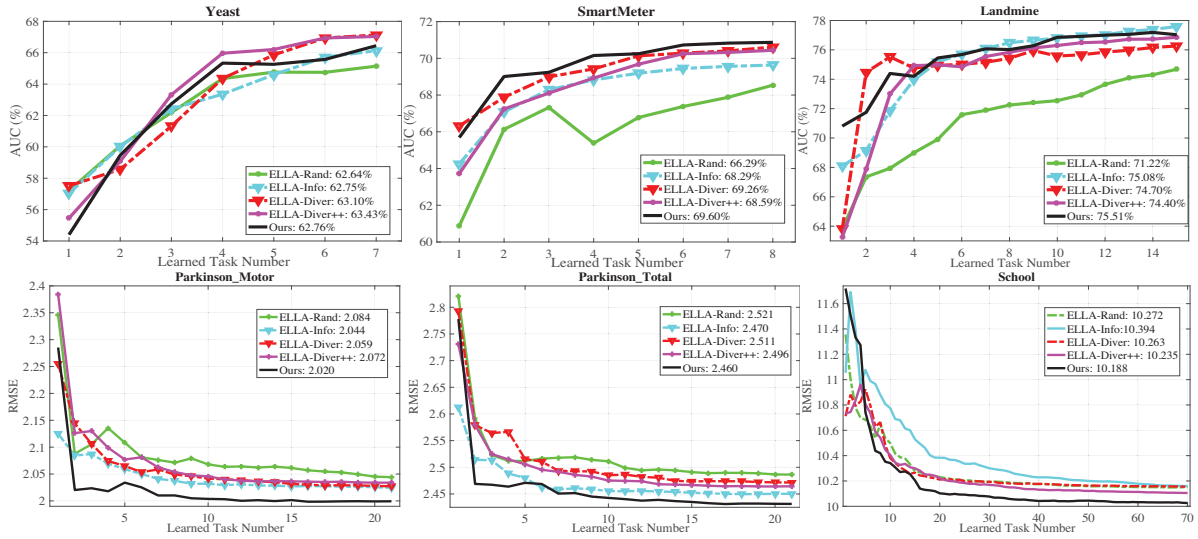

Figure 2: Results on six datasets with ELLA-Rand, ELLA-Info, ELLA-Diver, ELLA-Diver++ and Ours in terms of first tasks (around $\frac{1}{2}t_{\max}$). Each subfigure corresponds to a dataset, the corresponding legend gives the mean performance of each model, and AUC or RMSE of different models are shown by difference color lines.

Table 1: Comparisons between our method and state-of-the-arts in terms of AUC or RMSE on six datasets: mean and standard errors averaged over ten random runs. Methods with the best performance are bolded.

	Evaluation	#TaskNumber	ITL	ELLA-Rand	ELLA-Info	ELLA-Diver	ELLA-Diver++	*sp*MTFL	CL	Ours
Yeast	AUC	14	65.716±0.49	67.734±1.11	68.230±1.14	68.134±0.59	**68.246±0.62**	58.720±0.26	64.070±0.71	68.238±0.35
SmartMeter	AUC	16	51.193±0.31	70.471±0.26	70.725±0.60	70.860±0.24	70.820±0.21	56.330±0.50	50.481±0.61	**71.112±0.14**
Landmine	AUC	29	73.857±0.60	76.003±1.58	77.366±0.51	76.427±1.82	77.342±0.59	76.667±0.86	74.100±1.34	**78.190 ±0.68**
Parkinson-Total	RMSE	42	2.833±0.03	2.477±0.02	2.448±0.03	2.463±0.03	2.463±0.03	2.513±0.03	NaN	**2.421±0.01**
Parkinson-Motor	RMSE	42	2.253±0.01	2.035±0.03	2.021±0.02	2.025±0.02	2.032±0.02	2.058±0.02	NaN	**1.995±0.01**
School	RMSE	138	10.294±0.07	10.129±0.03	10.126±0.08	10.138±0.12	10.088±0.06	10.290±0.08	NaN	**9.982±0.03**

Real-World Datasets

We use six benchmark datasets for our experiments:

London School Data (**School**[1]) contains examination scores from 15,362 students in 139 schools, where each student has 27 binary features (e.g., student-specific features, school-specific features, etc), plus 1 bias feature. The corresponding response is the examination score. Therefore, we have 139 tasks in total by treating each school as a task.

Parkinson Data[2] consists of Parkinsons disease symptom score of 5,875 observations for 42 patients. Each task is a symptom score prediction problem and each sample consists of 16 biomedical features. We thus have 42 tasks in total with the number of samples for each patient varying from 101 to 168. Since the output of this dataset is a score consisting of Total and Motor, we establish two regression datasets in this experiment: **Parkinson-Total**, and **Parkinson-Motor**.

Landmine Data which can be modeled as a binary classification problem consists of 14,820 samples for 29 different geographical regions. Specifically, each task intends to detect whether or not a landmine is presented in a region based on 9-dimensional feature vector, and the corresponding binary label are: landmine (1) and clutter (-1). We thus have 29 classification tasks in total.

Yeast Data[3] consists of phylogenetic profiles of 2,417 genes for 14 functional categories. The input for each gene is micro-array expression data with 103-dimensional feature vector. In this experiment, we have 14 tasks in total by treating each functional category as a task.

SmartMeter Data[4] which is collected during a smart metering trial conducted in Ireland by the Irish CER, and the goal is to research the influence of consumption on household characteristics. In this experiment, we establish a new dataset by extract 81 features (such as daily consumption figures, statistical aspects, etc) from consumption data and 16 characteristics (such as household income, cooking style, etc) from questionnaires, i.e, the total number of task is 16.

For each task, we randomly split 50%-50% train-test set for our experiments, and the $t_{\max}$ is set as the task number of each dataset. Specifically, the lifelong learner used in our AcLL is same as the ELLA (Ruvolo and Eaton 2013b), i.e., Eq. (8). The experimental results averaged over ten random repetitions are presented in Table 1, and we can conclude:

- Compared with other competing methods, our proposed

[1]http://cvn.ecp.fr/personnel/andreas/code/mtl/index.html

[2]https://archive.ics.uci.edu/ml/datasets/parkinsons+telemonitoring

[3]http://mulan.sourceforge.net/datasets-mlc.html

[4]http://www.ucd.ie/issda/data/commissionforenergyregulationcer/

Table 2: Runtime (seconds) on a standard CPU of all compared methods.

	ITL	ELLA-Rand	ELLA-Info	ELLA-Diver	ELLA-Diver++	*sp*MTFL	CL	Ours
Parkinson(s)	0.32±0.06	0.74±0.03	0.79±0.05	0.69±0.05	0.68±0.02	1.49±0.02	NaN	1.29±0.06
Landmine(s)	0.39±0.26	0.97±0.40	1.15±0.08	1.12±0.02	1.15±0.04	8.54±0.19	773.01±3.22	1.50±0.08

AcLL framework does well on almost all datasets, which verifies the effectiveness of the proposed "watchdog" idea to mimic "human cognition". Furthermore, the original algorithm ELLA outperforms MTFL may leads to the fact that the performance of self-paced / curriculum learning frameworks is worse than task selection based on ELLA.

- For the task selection method based on ELLA, as shown in Figure 2, our AcLL framework outperforms other strategies (such as ELLA-Rand, ELLA-Info, etc) in most datasets, due to the fact that we adopt the original task parameter to select the next task and further provide the maximal benefit to future tasks.
- For the six real datasets, Table 1 also show that when the task number in the lifelong system increases gradually (e.g., task number of Yeast, SmartMeter, Landmine, Parkinson and School is 14, 16, 29, 42 and 139, respectively), the performance of our proposed AcLL framework is improved gradually since the hierarchical dictionary can accumulate more useful knowledge as the task number becomes large.

Comparison in term of Runtime: We adopt the Parkinson and Landmine datasets to test the time consumption of our proposed AcLL with the state-of-the-arts as shown in Table 2. Specifically, we adopt 50%-50% train-test split on both two datasets. Generally, the task selection methods based on ELLA are more efficient than self-paced / curriculum learning methods, i.e., *sp*MTFL and CL. This is because ELLA does not need to process the original features with knowledge library L. For all the strategies based on ELLA, because we adopt the eigen-decomposition for model optimization, our method is a little slower than others, but has lower error than others. All the experiments are performed using Matlab on the computer with 12G RAM, i7 CPU.

Convergence Analysis: In order to investigate the convergence of the alternating direction method to solve our proposed AcLL framework, we plot the value of the reconstruction cost on the Parkinson-Motor dataset and School dataset, respectively. Specifically, we randomly select 70% of the total number of tasks as training set and the rest as the test set. The reconstruction cost is calculated on the test set with the learned task number increasing. As depicted in Figure 3, the cost values decrease with respect to number of learned tasks. The performance lends further evidence that our framework to select unknown task is effective.

Effect of Dictionary Learning Method

We further investigate our dictionary learning method on extended Yale B dataset in this subsection. Specifically, the extended Yale B dataset consists of 2,414 frontal face images for 38 persons under 64 illumination conditions and expressions, i.e., each person has 64 images. The original images

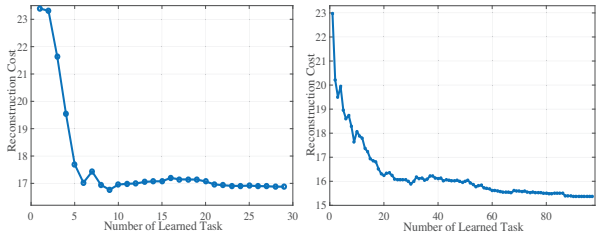

Figure 3: Convergence analysis of our proposed framework on the Parkinson (Left) and School (Right) dataset.

Table 3: Average classification accuracies on extended Yale B dataset.

Method	Dictionary Size	Accuracy (%)
TraDL	570	95.70
KSVD	570	95.54
LC-KSVD1	570	93.61
LC-KSVD2	570	94.48
D-KSVD2	570	93.58
ORDL	570	89.17
Ours	494	**95.89**

are cropped to the resolution 192×168 and then extracted into a 504-dimensional feature vector using a random matrix. For evaluation, we randomly split the dataset into 50%-50% train-test set, i.e., 32 images for each person are randomly selected for training the dictionary and the rest are for test.

The state-of-the-art methods used in this paper include: traditional dictionary learning (TraDL) (Mairal et al. 2009), K-SVD (Aharon, Elad, and Bruckstein 2006), LC-KSVD1 and LC-KSVD2 (Jiang, Lin, and Davis 2013), D-KSVD (Zhang and Li 2010) and ORDL (Lu, Shi, and Jia 2013). The parameters for these methods are selected via cross validation. The experimental results averaged over ten random repetitions and dictionary size among used methods are presented in Table 3, and we can conclude that: our proposed dictionary learning framework can achieve the best performance. Note that the dictionary size of ours is 494 (i.e., 13 items for person on average), which is smaller than other state-of-the-arts (15 items for person on average). The reason why ours can perform well with small dictionary size is that we adopt two-level dictionary structure, and it can capture more useful information among the Yale B face dataset. Additionally, we also evaluate the effect of local dictionary ΘA by fixing $\lambda_1 = 1$ and adjusting λ_2 in $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$. As illustrated in Figure. 4, accuracy changes with different values of λ_2, i.e., appropriate local dictionary can make the performance better.

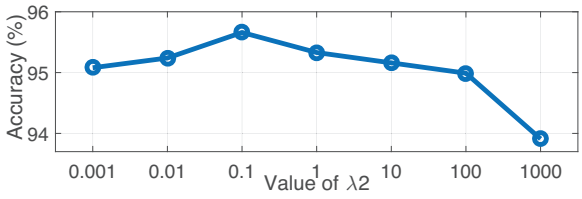

Figure 4: Illustration of accuracy by varying the value of λ_2.

Conclusion

In this paper, we propose an Active Lifelong Learning framework (AcLL) based on outlier detection scenario, where it can mimic an effective "human cognition" behavior, i.e., unknown-to-known. Specifically, whether the knowledge of coming tasks is unknown or not can be detected via a sparse reconstruction cost over a hierarchical dictionary, which consists of two-level task descriptors. The task with higher cost will be permitted to pass and further be analytically encoded by the knowledge library in the lifelong system. For dictionary optimization, we adopt the alternating direction method to solve our optimization problem. We have conducted experiments on several real-world datasets and extended Yale B dataset; the experimental results demonstrate the effectiveness of our proposed AcLL framework. In the future, we plan to apply the hierarchical dictionary to construct more useful knowledge library.

References

Aharon, M.; Elad, M.; and Bruckstein, A. 2006. rmk-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing* 54(11):4311–4322.

Ammar, H. B.; Eaton, E.; Ruvolo, P.; and Taylor, M. 2014. Online multi-task learning for policy gradient methods. In *ICML-14*, 1206–1214.

And, H. E. E., and Yantis, S. 1997. Visual attention: Control, representation, and time course. *Annual Review of Psychology* 48(1):269.

Bi, J.; Xiong, T.; Yu, S.; Dundar, M.; and Rao, R. B. 2008. An improved multi-task learning approach with applications in medical diagnosis. In *ECML/PKDD*, 117–132.

Caruana, R. 1997. Multitask learning. *Machine Learning* 28(1):41C–75.

Chen, L.; Zhang, Q.; and Li, B. 2014. Predicting multiple attributes via relative multi-task learning. In *CVPR*, 1027–1034.

Cohn, D. A.; Ghahramani, Z.; and Jordan, M. I. 1996. Active learning with statistical models. *Journal of artificial intelligence research* 129–145.

Cong, Y.; Liu, J.; Sun, G.; You, Q.; Li, Y.; and Luo, J. 2017. Adaptive greedy dictionary selection for web media summarization. *IEEE Transactions on Image Processing* 26(1):185–195.

Cong, Y.; Yuan, J.; and Liu, J. 2011. Sparse reconstruction cost for abnormal event detection. In *CVPR*, 3449–3456.

Cong, Y.; Yuan, J.; and Liu, J. 2013. Abnormal event detection in crowded scenes using sparse representation. *Pattern Recognition* 46(7):1851–1864.

Isele, D.; Rostami, M.; and Eaton, E. 2016. Using task features for zero-shot knowledge transfer in lifelong learning. In *IJCAI*, 1620–1626.

Jiang, Z.; Lin, Z.; and Davis, L. S. 2013. Label consistent k-svd: Learning a discriminative dictionary for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(11):2651–2664.

Kumar, A., and Daumé, III, H. 2012. Learning task grouping and overlap in multi-task learning. In *ICML*, 1723–1730.

Li, C.; Yan, J.; Wei, F.; Dong, W.; Liu, Q.; and Zha, H. 2017. Self-paced multi-task learning. In *AAAI*, 2175–2181.

Lu, C.; Shi, J.; and Jia, J. 2013. Online robust dictionary learning. In *CVPR*, 415–422.

Mairal, J.; Bach, F.; Ponce, J.; and Sapiro, G. 2009. Online dictionary learning for sparse coding. In *ICML*, 689–696.

Murugesan, K., and Carbonell, J. 2017. Self-paced multitask learning with shared knowledge. *arXiv preprint arXiv:1703.00977*.

Nie, F.; Huang, H.; Cai, X.; and Ding, C. H. 2010. Efficient and robust feature selection via joint l2, 1-norms minimization. In *NIPS*, 1813–1821.

Obozinski, G.; Taskar, B.; and Jordan, M. 2007. Multi-task feature selection. *Technical report*.

Pentina, A.; Sharmanska, V.; and Lampert, C. H. 2015. Curriculum learning of multiple tasks. In *CVPR*, 5492–5500.

Ruvolo, P., and Eaton, E. 2013a. Active task selection for lifelong machine learning. In *AAAI*, 862–868.

Ruvolo, P., and Eaton, E. 2013b. Ella: An efficient lifelong learning algorithm. In *ICML*, 507–515.

Saha, A.; Rai, P.; Daumé III, H.; and Venkatasubramanian, S. 2010. Active online multitask learning. In *ICML 2010 Workshop on Budget Learning*.

Saha, A.; Rai, P.; DaumÃ, H.; Venkatasubramanian, S.; et al. 2011. Online learning of multiple tasks and their relationships. In *AISTATS*, 643–651.

Tong, S., and Koller, D. 2001. Support vector machine active learning with applications to text classification. *Journal of machine learning research* 2(Nov):45–66.

Zhang, Q., and Li, B. 2010. Discriminative k-svd for dictionary learning in face recognition. In *CVPR*, 2691–2698.

Zhang, J.; Ghahramani, Z.; and Yang, Y. 2008. Flexible latent variable models for multi-task learning. *Machine Learning* (73):221C–242.

Zhang, Y. 2010. Multi-task active learning with output constraints. In *AAAI*, 667–672.

Zhu, P.; Hu, Q.; Zhang, C.; and Zuo, W. 2016. Coupled dictionary learning for unsupervised feature selection. In *AAAI*, 2422–2428.