Research Statement

Keerthiram Murugesan
kmuruges@cs.cmu.edu

My current research is on developing efficient algorithms for large-scale multitask and lifelong learning under realistic assumptions. I am also interested in related areas such as multi-armed bandit, reinforcement learning, hyper-parameter optimization, active learning, etc. with applications to recommendation, conversation systems, computational advertising, spam detection, sentiment analysis and other natural language understanding tasks.

Multitask learning solves multiple related tasks simultaneously and achieves a better performance using the relevant information shared by multiple tasks ([Car98]). The power of joint learning in multiple tasks arises from the transfer of relevant knowledge across said tasks, especially from information-rich tasks to information-poor ones. Unlike traditional multitask learning, in lifelong learning the tasks arrive sequentially ([Thr96]). It provides an efficient way to learn new tasks faster by utilizing the knowledge learned from the previous tasks and prevent catastrophic forgetting or significantly degrading performance on the old tasks.

Scalable Multitask and Lifelong Learning

In the recent decades, multitask learning have received significant attention due to its empirical success in many important fields of AI, from natural language understanding, computational biology [KMCKS17] to bandit learning, reinforcement learning, etc [MC15]. Despite several advantages of learning from related tasks, it poses considerable challenges in terms of effectiveness by minimizing prediction errors for all tasks and overall computational tractability for real-time performance. In contrast, human beings seem natural in accumulating and retaining the knowledge from the past and leverage this knowledge to acquire new skills and solve new problems efficiently. For my research, I consider two key challenges in multitask and lifelong learning:

1. **Sequential data**: In many real-world applications such as optimizing financial trading, email prioritization, personalized news, and spam filtering, etc., data arrive sequentially. In such cases, we need to make predictions and update the per-task models in an efficient real-time manner when a new observation or task is available.
2. **Scalability**: Most existing algorithms for multitask learning cannot scale to very large dataset especially when the number of tasks is large. The computational complexity arises due to the difficulty in learning the shared knowledge from all the tasks.

Together, these two challenges may hinder the practical application of multitask learning to several real-world problems. My coauthors and I tackle these challenges with simple and efficient online learning algorithms for jointly learning both the per-task model parameters and the task relatedness. In my work, I chose pairwise inter-task relationship to keep track of the knowledge transfer. This allows us to consider task relatedness as a graph of relations among multiple tasks where each node corresponds to a task and an edge between the nodes represents similarity/correlation between the tasks. We observe this notion of pairwise inter-task relationship graph in many applications such as user preference learning (social graph), recommendation systems, computational advertising, web search, conversation systems, etc. One important consequence of this approach in online multitask learning is that without manageable *efficient updates* at each round, learning the relationship between the tasks automatically from data may impose a severe computational burden.

Online Smooth Multitask Learning: Based on the motivation to address the aforementioned challenges, we proposed an algorithm that features probabilistic interpretation, efficient updating rules for learning the asymmetric relationship between the tasks and flexible modulations on whether learners focus on their specific task or on jointly address all tasks [MLCY16]. The key idea is based on smoothing the loss function of each task w.r.t. a probabilistic distribution over all tasks, and adaptively refining such distribution over time. We proved a sub-linear regret bound as compared to the best linear predictor in hindsight.

We extended our learning setting to multiple kernel learning where each task is associated with multiple kernels and proposed a two-stage learning approach to learn the model parameters, optimal kernel weights, and the task relationship efficiently [MC17b]. In addition to the graph relations between the task, I plan to explore other structural assumptions such as clusters [MCY17], overlapping groups, hierarchical cascades, etc. under online learning setting. Following our previous work, I am also interested in learning with partial feedback where we run an instance of (contextual) bandit learning algorithm for each task and leverage the inter-task relationship to speed-up the learning process.

Learning with a Self-paced Curriculum: When it comes to lifelong learning, it is natural to ask, Is it possible to use an existing multitask algorithm to solve a lifelong learning problem? We proposed a novel framework based on self-paced learning which utilizes a curriculum defined dynamically by the learner (”self-paced”) instead of a fixed curriculum set *a-priori* by a teacher [MC17c]. It allowed us to use a certain class of existing multitask algorithms for solving lifelong learning setting. Our proposed method starts with an easier set of tasks, and gradually introduces more difficult ones to build the shared knowledge base. It provides a natural way to specify the trade-off between choosing the easier tasks to update the shared knowledge and learning new tasks using the knowledge acquired from previously learned tasks.

Continuous Lifelong Learning with Output Kernels: Along this line of research, my coauthors and I considered a difficult problem setting in lifelong learning where both the observation and task arrives sequentially [MC18]. We define the task relatedness as *output* kernels in *Reproducing Kernel Hilbert Space* (RKHS) on multitask instances. We proposed a novel algorithm called *Online Output Kernel Learning Algorithm* (OOKLA) for lifelong learning setting with simple and quite intuitive update rules for learning the output kernel sequentially. When we receive a new example, the algorithm updates the output kernel when the learner made a mistake by computing the similarity between the new example and the set of representative examples (stored in the memory) that belongs to a specific task. If the two examples have similar (different) labels and high similarity, then the relationship between the tasks is increased (decreased) to reflect the positive (negative) correlation and vice versa.

To avoid the memory explosion, we introduce a robust budget-limited version of the proposed algorithm, which efficiently utilizes the relationships between the tasks to bound the total number of representative examples in the support set. We are currently working on a two-stage budgeted scheme for efficiently tackling the task-specific budget constraints in lifelong learning. During my recent internship at IBM research, we explored the problem of lifelong learning builds on this work for the conversation systems.

For more details on these topics, please read my thesis proposal document [Mur17].

Multi-agent (Reinforcement) Learning

I am more interested these days in the continuous adaptation of learning system to the changing environment especially when there are many learners (or agents) interact with each other under realistic assumptions (along the lines of bandit learning and multi-agent reinforcement learning). I consider this problem setting under two different learning environments for my work: 1) Collaborative where the learning agents work together to solve a problem 2) Competitive where the learning agents learn from their opponents to solve a problem in a competitive environment.

Learning from Peers: My first contribution towards this direction is my recent work on learning from peers in a collaborative environment [MC17a]. The primary intuition we leveraged in this work is that task performance can be improved both by querying external oracles and by querying peer tasks, where the former incurs a cost or at least a query-budget bound, but the latter requires no human attention. Hence, our hypothesis was that with bounded queries to the human expert, additionally querying peers should improve task performance. Instead of always requesting a label from a human oracle, our proposed method first determines if the learner for each task can acquire that label with sufficient confidence from its peers either as a task-similarity weighted sum or from the single most similar task. If so, it saves the oracle query for later use in more difficult cases, and if not it queries the human oracle.

Lifelong Multi-agent Reinforcement Learning: Following our work on learning from peers and lifelong learning, I am interested in pursuing a similar setting in lifelong multi-agent reinforcement learning. In this setting, the learning agents share the knowledge (policy) accumulated in the past with the other agents as options/skills (extended action set). In addition to the latent structural assumption among the tasks, we will consider the (fully/partially known) structural constraints (as successor representation) on the environment.

Distributed Multitask Learning

Besides the above interests, I am eager to explore the topic of distributed multitask learning. Unlike in the traditional multitask learning where we build models for data (from all the tasks) collected at a centralized location, distributed multitask learning focuses on communication and computationally-efficient algorithms for multitask learning where the data for each task resides in different geographical locations. This is common in many real-world applications such as clinical trails [KMCKS17], safety-alert system [MC15], etc.

Unlike in the standard approach, where we select the features that affect all the tasks using shared sparsity (such as ℓ_1/ℓ_2), I plan to leverage the pairwise interrelationship graph of the tasks for efficient communication between the models. The motivation behind this approach is that since task relationship is asymmetric and knowledge transfer is typically from information-rich tasks to information-poor ones, we can reduce the communication cost significantly by learning and utilizing the inter-task relationship graph. Motivated by this line of work, I plan to contribute towards developing privacy-preserving multitask learning algorithm in a distributed setting.

References

[Car98] Rich Caruana. Multitask learning. In *Learning to learn*, pages 95–133. Springer, 1998.

[KMCKS17] Meghana Kshirsagar, Keerthiram Murugesan, Jaime G Carbonell, and Judith Klein-Seetharaman. Multitask matrix completion for learning protein interactions across diseases. *Journal of Computational Biology*, 24(6):501–514, 2017.

[MC15] Keerthiram Murugesan and Jaime Carbonell. Predicting workplace incidents with temporal graph-guided fused lasso. 2015.

[MC17a] Keerthiram Murugesan and Jaime Carbonell. Active learning from peers. In *Advances in Neural Information Processing Systems*, pages 0–0, 2017.

[MC17b] Keerthiram Murugesan and Jaime Carbonell. Multi-task multiple kernel relationship learning. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 687–695. SIAM, 2017.

[MC17c] Keerthiram Murugesan and Jaime Carbonell. Self-paced multitask learning with shared knowledge. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2522–2528, 2017.

[MC18] Keerthiram Murugesan and Jaime Carbonell. Lifelong learning with output kernels. 2018.

[MCY17] Keerthiram Murugesan, Jaime Carbonell, and Yiming Yang. Co-clustering for multitask learning. *arXiv preprint arXiv:1703.00994*, 2017.

[MLCY16] Keerthiram Murugesan, Hanxiao Liu, Jaime Carbonell, and Yiming Yang. Adaptive smoothed online multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4296–4304, 2016.

[Mur17] Keerthiram Murugesan. Online and adaptive methods for multitask learning. *Thesis Proposal*, 2017.

[Thr96] Sebastian Thrun. Is learning the n-th thing any easier than learning the first? In *Advances in Neural Information Processing Systems*, pages 640–646, 1996.