The Intelligent Management of Machine Learning

Christopher H. Lin
University of Washington
chrislin@cs.washington.edu

1 Overview

Research in artificial intelligence (AI) and machine learning (ML) has exploded in the last decade, bringing humanity to the cusp of self-driving cars, digital personal assistants, and unbeatable game-playing robots. My research, which spans the areas of AI, Crowdsourcing, and Natural Language Processing (NLP), focuses on an area where machines are still significantly inferior to humans, despite their super-human intelligence in so many other facets of life: the intelligent *management* of machine learning (iML), or the ability to reason about what they don't know so that they may independently and efficiently close gaps in knowledge. Imagine an agent that exhibits the following behavior: while reading a newspaper, when it sees a verb it doesn't understand, it utilizes various resources like the crowd to train a deep neural net to learn what that verb means. Then, it tests itself, by asking those resources, or others, whether it is correctly using that verb, perhaps asking a greater number of sources if the initial ones disagree, and then fine-tunes its parameters to more accurately understand that verb. This agent is what I would consider to have excellent iML ability. Unfortunately, no such agents exist today.

The idea to improve an agent's ability to manage its own learning is not novel. In particular, the AI subfield of Active Learning (AL) [11] has been a significant attempt at making progress on this problem. Research in AL tries to answer variations of the broad question: What is the next best training example to label in order to maximize the long-term performance of the learner? However, because of the recent popularity of using crowdsourcing for machine learning, many of traditional AL's core assumptions are outdated. For example, today, examples are not labeled only once, nor is the quality of every label uniform. While much modern AL research has made significant headway in relaxing those assumptions (*e.g.*, [12][3][14]), work still remains.

Traditional AL also, by definition, has failed to address the bigger vision of iML, which encompasses many other important questions, including, but not limited to: 1) What is the best way to collect labels for test examples (training and test examples have different requirements - see Section 2 for more.) 2) How can an agent that is trying to learn a new concept sift through all the unlabeled examples that exist in the world to identify exemplary subsets that would make good training and test sets? An agent must be able to identify examples that are positive for that concept. Learning is extremely expensive, if not impossible, if one cannot find representative examples. 3) Is the cost of learning too high to justify continued learning? Perhaps that effort might be better spent elsewhere [5]. I believe answering these questions is crucial to autonomous thinking and advancing AI. Although some research outside of AL has tried to answer some of these questions (*e.g.*, [2]), we are still far from a complete understanding.

My research endeavors to contribute to the community's ongoing modernization of AL and efforts to develop iML. Inside of this broad theme, my work can be organized into three parts. In the first part, I address the first iML question above by using decision-theoretic techniques like partially-observable Markov Decision Processes (POMDPs) to model and optimally control the process of constructing test sets via crowdsourcing. In the second part, I contribute a generalization of AL and devise algorithms for that generalization that achieve more cost-effective learning by trading off between gathering more diverse training examples and de-noising existing ones (Figure 1). And in the third part, I address the second iML question above by developing methods for using the crowd to generate training examples when sampling examples from a highly skewed dataset is not cost effective, for example when positive examples are difficult to find. In most of my work, I have used NLP domains like information extraction and entity linking as practical testbeds, in order to show that my contributions in iML help to boost performance on tasks of real-world importance.

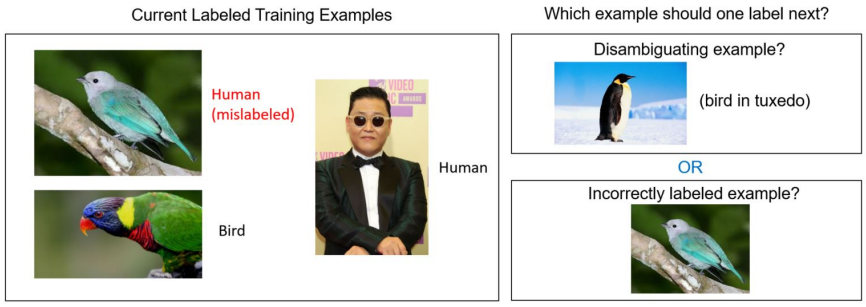

Figure 1: Suppose an agent is learning about birds and humans and has been trained on a correctly labeled bird, a correctly labeled human wearing a tuxedo, and an incorrectly labeled bird (perhaps because of a spammer). Should this agent relabel the incorrectly labeled example or should it label a new disambiguating example, like a bird wearing a tuxedo?

2 Current Research

2.1 Decision-Theoretic Construction of Test Data

Machine learning, whether it is supervised, semi-supervised, or unsupervised, is heavily reliant on a test set with clean, gold-standard labels. (In contrast, the accuracy of training data may not be critical, because training data only needs to be as accurate as suffices to train an accurate classifier. See Section 2.2 for more details.) While in the past, providing these labels has been the task of the researcher or system designer, the advent of crowdsourcing is now enabling cheap and fast collection. However, the crowd is noisy, so one must pay particular attention to the quality of the labels that are returned. Typically, task requesters mitigate the risk of poor data by asking multiple workers to label each example. However, oftentimes their strategies for doing so are suboptimal. For instance, asking 10 workers to weigh in on a particular example may not be wise if the first 5 have already agreed. The first part of my thesis looks at how one can apply decision-theory to two different problems one encounters when relabeling testing data.

The first problem I tackle is the optimal control of workflows. Task designers often construct complex workflows to acquire labels from workers. Frequently, they A/B test several alternative workflows that can accomplish the task, and eventually deploy the one that achieves the best performance during early trials [4]. However, surprisingly, this seemingly natural design paradigm does not achieve the full potential of crowdsourcing. In particular, using a single workflow (even the best) to accomplish a task is suboptimal. I have shown that alternative workflows can compose synergistically to yield much higher quality output by modeling the problem with a novel probabilistic graphical model and designing and implementing a POMDP-based controller that dynamically switches between these workflows [7]. I have shown that switching between workflows can achieve up to 50% error reduction compared to using only one for the same cost.

Unfortunately, this model (and all other models for redundant labeling) assumes prior knowledge of all possible outcomes of the task. While not an unreasonable assumption for tasks that can be posited as multiple-choice questions (e.g. n-ary classification), many tasks do not naturally fit this paradigm, but instead demand a free-response formulation where the outcome space is of infinite size (e.g. audio transcription). The second problem I tackle is the removal of this assumption [6]. I show that one can model such tasks with a novel probabilistic graphical model using a Chinese Restaurant Process, and design and implement a decision-theoretic controller that dynamically requests responses as necessary in order to infer answers to these tasks. The controller is able to eliminate up to 83.2% of the error and achieve greater net utility compared to majority-voting.

2.2 Re-active Learning: Active Learning with Relabeling

Crowdsourcing is also a popular method for labeling *training* data for supervised machine learning algorithms. The second part of my research moves away from the realm of labeling testing data and into the realm of labeling training data.

As mentioned above, unlike for test data, the accuracy of training data does not necessarily need to be perfect. In many cases, noisy training data can work very well if you have a lot of it. Because of this key difference, relabeling training examples is not necessarily the most cost-effective strategy for creating the best classifier. Instead, one must make a tradeoff between decreasing the noise of the training set via relabeling and increasing the size and diversity of the (noisier) training set by labeling new examples [12]. Given a fixed budget, in some cases classifiers can often achieve higher accuracies when trained with large sets of noisy singly-labeled data, whereas in other cases, classifiers do better when trained with smaller sets of cleaner training data. I have found that at least three factors affect whether relabeling examples is an effective strategy: classifier expressiveness (VC dimension), worker accuracy, and budget [8].

Also as important as understanding the characteristics of learning problems that make them amenable to relabeling is devising strategies for dynamically controlling the relabeling. Previous work [12] has looked at strategies for picking which examples to relabel when relabeling is the *only* option. In contrast, I have formalized the problem of *re-active learning* as a generalization of AL that allows for picking examples to relabel or label for the first time [9]. Along with this formalization, I have shown how traditional AL methods can perform extremely poorly at re-active learning and presented a new class of algorithms specifically designed for the task. I have also shown that this new class of algorithm can be considered a generalization of uncertainty sampling, a popular algorithm for AL.

2.3 Discovering Relevant Training Data

In all the research I have discussed so far, I, like the many others who work in the intersection of machine learning and crowdsourcing, have assumed that relevant unlabeled examples, whether for training or testing purposes, can be easily found. The final part of my thesis tackles the unfortunate reality that in practice, this assumption is simply not true. Consider for example, event extraction. Imagine the following problem: the NSA has an extremely large corpus, and wants to find all sentences that talk about bombings. If they currently don't have an extractor for the event "bombing," how could they solve this problem? If we would like to teach a learner a new concept, one that has never been taught before, finding the positive examples of that concept in a very large or non-existent corpus can be extremely difficult, especially if that corpus has heavy skew. The probability that any given sentence in the corpus is positive for "bombing" is vanishingly small.

I have implemented an end-to-end, push-button system that can automatically learn convolutional neural net extractors for arbitrary events via non-expert crowdsourcing [10]. The system works via two kinds of actions. One type of action asks crowd workers to *generate* examples, as opposed to label them. Then, it can use these examples to train an extractor. The second type of action is to use the current extractor to make predictions about sentences in the large corpus and use this information to selectively label sentences using the crowd. By using the crowd to bootstrap an extractor and intelligently selecting actions to maximize performance, the system can train extractors for novel events.

3 Future Research

In the future, I am interested in continuing to push forward on the design of agents that can intelligently manage their learning through rich interactions with the crowd. Additionally, I am broadly interested in exploring topics in AI Safety, Deep Learning, and NLP; the connections between these areas and my work; and in particular, problems in these spaces that have real-world applications. Some questions that I could work on include:

- When is the cost of learning too high to justify continued learning? For example, a medical assistant bot that cannot substantially increase its performance on identifying malignant tumors may be better off learning about warning signs of septic shock. While I have looked at this problem in the context of planning [5], understanding this question in the context of learning is important as well. One way to approach this problem is to use heuristics to cut off learning when an agent's learning curve has converged. A more principled way is to use decision-theoretic machinery like POMDPs to model the tradeoff between continuing to learn a concept versus switching to a new one, but this would require utility elicitation, a difficult problem. Yet another way is to train a system that determines when

learning should stop. This system could be trained by asking the crowd to label scenarios in which learning should stop or continue. More concretely, one might be able to show the crowd learning curves and ask them questions like "Which curve looks like it has the most potential to increase?" The crowd could also be asked on-demand by the agent whether it should move on.

- How does an agent identify actions that are potentially problematic, and when it does, how should it proceed? An agent that is trying to get better at reading different kinds of fonts around the house should not ask the crowd to label an image of a credit card. This problem is similar to the problem of "safe exploration" in Safe AI/Reinforcement Learning [1][13], but one can make progress by considering it in the context of iML. One way to approach the first part of this problem (identifying problematic actions) is to train a problem predictor, by asking the crowd to generate scenarios in which actions may have high variance or high negative rewards. For example, for an agent that learns primarily through vision, we could ask the crowd to provide example images from the web that should not be shared. The second part of the problem (how the agent should proceed) requires answering several more questions: Should the agent ask the owner for a label? If the agent queries the crowd, how can it reduce risk? The right answers probably depend on the preferences of the owner as well as the domain, which can be modeled and learned. As a starting point, the crowd can provide data to bootstrap generic algorithms, for example by answering hypothetical questions about how often they, as an owner, might wish to be queried, or providing ways of obscuring images so that they may be safely shown to others. Then, these baseline agents can become more personalized as they run.

In addition to being guided by the above research themes, I hope to pursue the idea that we should be able to use skilled crowds, which can both train and collaborate with agents, to solve our most difficult problems. I believe the domain of medicine is especially promising and ripe for innovation along this line. For example, instead of asking random crowd workers about learning curves, a medical assistant bot should instead be able to ask doctors about what their weaknesses are. Such interaction would pay off dividends when the bots become capable of working alongside doctors to make treatment and diagnosis recommendations, and has the potential to save countless lives through the reduction of medical errors as well as improve the quality of life for overworked doctors. I envision and hope to build agents that can work side-by-side with humans, learning from them, teaching them, and cooperating to advance society beyond what we can now imagine.

References

[1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *CoRR*, abs/1606.06565, 2016.

[2] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *AAMAS*, 2012.

[3] Ece Kamar, Ashish Kapoor, and Eric Horvitz. Lifelong learning for acquiring the wisdom of the crowd. In *IJCAI*, 2013.

[4] Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M. Henne. Controlled experiments on the web: Survey and practical guide. *Data Mining and Knowledge Discovery*, 18:140–181, 2009.

[5] Christopher H. Lin, Andrey Kolobov, Ece Kamar, and Eric Horvitz. Metareasoning for planning under uncertainty. In *IJCAI*, 2015.

[6] Christopher H. Lin, Mausam, and Daniel S. Weld. Crowdsourcing control: Moving beyond multiple choice. In *UAI*, 2012.

[7] Christopher H. Lin, Mausam, and Daniel S. Weld. Dynamically switching between synergistic workflows for crowdsourcing. In *AAAI*, 2012.

[8] Christopher H. Lin, Mausam, and Daniel S. Weld. To re(label), or not to re(label). In *HCOMP*, 2014.

[9] Christopher H. Lin, Mausam, and Daniel S. Weld. Re-active learning: Active learning with relabeling. In *AAAI*, 2016.

[10] Christopher H. Lin, Mausam, and Daniel S. Weld. Extremest extraction: Push-button learning of novel events. In Preparation.

[11] Burr Settles. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2012.

[12] Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.

[13] Matteo Turchetta, Felix Berkenkamp, and Andreas Krause. Safe exploration in finite markov decision processes. 2016.

[14] Chicheng Zhang and Kamalika Chaudhuri. Active learning from weak and strong labelers. In *NIPS*, 2015.