Teaching New Teammates

Doran Chakraborty
Department of Mathematics & Computer Science
The University of Tulsa
USA
doran@utulsa.edu

Sandip Sen
Department of Mathematics & Computer Science
The University of Tulsa
USA
sandip@utulsa.edu

ABSTRACT

Knowledge transfer between expert and novice agents is a challenging problem given that the knowledge representation and learning algorithms used by the novice learner can be fundamentally different from and inaccessible to the expert trainer. We are particularly interested in team tasks, robotic or otherwise, where new teammates need to replace currently indisposed team member(s). We are interested in a general knowledge transfer framework where existing team members or experts can train a new agent to follow its role in team coordination by using exemplars of desirable behavior. Each such exemplar presents a team situation and a preferred action. We envisage an iterative training process where the trainer selects more exemplars in the next iteration based on the errors made by the learner in action choices for test exemplars presented in the current iteration. Such an iterative, exemplar based generic knowledge transfer scheme can be used by agents using arbitrary knowledge representation and learning methods. We evaluate the success of training new teammates in the well-known pursuit problem, where some of the current set of expert predators is being replaced by new ones with no *a priori* hunting knowledge. Experimental results demonstrate the robustness of our knowledge transfer scheme with a graceful performance degradation.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*; I.2.6 [**Artificial Intelligence**]: Learning

General Terms

Experimentation, Performance

Keywords

Knowledge transfer, coordination, training

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.

1. INTRODUCTION

We have proposed an Agent Teaching Agent (ATA) framework to address the problem of transfer of concept description knowledge between a trainer and a trainee agent. A concept description is a boolean-valued function that classifies input examples into various discrete classifications of the target concept [5]. We assumed that the trainer agent does not have access to the internal knowledge representation of the trainee agent, but can evaluate the latter's concept recognition abilities by asking it to categorize selected exemplars and non-exemplars of the target concept. We focused on the incremental selection of training exemplars by the trainer to expedite the learning of the trainee agent. The ATA framework involves several iterations of training and test set presentations to the trainee, where the trainer chooses the training set of one iteration based on the mistakes made by the trainee on the test set in the previous iterations.

The goal of this paper is to show that the ATA framework can be used to transfer coordination knowledge to other agents, without information about the other agent's internal knowledge representation and learning algorithms. Such coordination knowledge transfer allows teams to recover from failure of some team members as well as deploy "clone" teams in the environment to reduce load on the current team of trainers. We assume that the trainer itself is a learner (a member of the current team), who learns its coordination knowledge from training instances provided by a domain expert, e.g., a coach. We further assume that this training set provided by the expert is no longer available, and the trainer must generate new exemplars from introspection to train new teammates.

To support our claims for the effectiveness of the iterative training process used in ATA, we also compare the performance of ATA-trained trainees with the performance of other trainee agents taught by the teacher agent in one iteration by presenting a single training set. Our results clearly show that the concept description generated by the trainee agent in the latter case is less effective in comparison to the concept description formed by the trainee agent trained using the ATA framework.

2. ATA FRAMEWORK

The trainer agent uses its learning module to acquire the target concept from its interaction with an environment, which may include an expert providing feedback. This learning process produces a target concept description in the knowledge representation used by the trainer agent. The

Train-Agent(Trainer,Trainee,Trainer-knowledge){
Select initial training set N_0 and initial testing set T_0 from Trainer-knowledge
$i \leftarrow 0$
repeat
  trainee agent trains on N_i
  $M_i \subset T_i$, examples misclassified by trainee after training on N_i
  $T_{i+1} \leftarrow T_i \cup newTestInstances(M_i)$
  $N_{i+1} \leftarrow N_i \cup newTrainingInstances(M_i)$
  $i \leftarrow i+1$
until $|M_i| < threshold$

Figure 1: Algorithm for incremental training.

trainer agent also has a training module which interacts with the trainee and provides successive training and testing sets to train and evaluate the progress in learning of the trainee agent. The trainee learns its own concept description from the set of classified training examples provided by the trainer. It also classifies each of the unclassified test examples provided by the trainer and returns these classified instances to the trainer for evaluation.

We envisage an iterative training procedure in which alternatively the trainer selects a set of training and testing exemplars, the trainee trains using the training set and then classifies the testing set, the trainer observes errors made by the trainee in classifying the instances in the last testing set and accordingly generates the next training and testing sets. This iterative process converges when the error of the trainee falls below a given threshold. We present these iterative training steps in an algorithmic form in Figure 1.

3. DOMAIN AND TRAINING DATA

The domain used for our experimentation is the predator-prey pursuit problem [3] in a continuous world. This problem requires effective coordination by a group of agents to achieve a goal. The goal of the predator in this paper is to capture a prey which is moving in a straight line with half the speed of the predator. The predators who move simultaneously can see each other and the prey, but cannot communicate explicitly to coordinate their moves. To capture the prey, predators must surround it from four sides and be within a threshold distance of the prey. However if they come in close proximity of each other, a collision occurs and each of the colliding parties move a significant number of steps away from the collision point. To avoid such collisions, an expert predator moves a few steps orthogonal to the direction of the prey when it senses another predator in close proximity. The predators must capture the prey within a maximum number of steps, after which the game stops.

For our experimental setup, we have considered C4.5, a decision tree based algorithm [7] as the learning mechanism for the trainer and an instance based learning algorithm, IB2, as the learning mechanism for the trainee. IB2 uses an incremental learning algorithm that stores a subset of the training examples as its classification knowledge [1].

The experts provide the trainers with 17500 instances from various configurations to learn from. The large size of training is required to capture the complexities of coordinating four teammates in continuous space. To derive a rich set of training instances, the experts calculated all possible orientations of the four predators in the four quadrants centered at the prey. A configuration is a vector $< n1, n2, n3, n4 >$ where $n_i \in \{0, 4\}$ and $\sum_{i=1}^{4} n_i = 4$ and n_i is the number of predators in quadrant i with respect to the prey. Several initial predator-prey situations were generated for each such configuration, and the move of the experts from each such situation was noted. A training instance is created for each situation-move pairs by noting polar coordinates of the other predators and the prey, assuming this expert's positions as origin, were the classification of the instance is the move(right, left or ahead) selected by this expert at this situation. A typical training set data is a tuple $(r1, \theta 1, r2, \theta 2, r3, \theta 3, rp, \theta\text{p}, M)$, where the different r and θ values signify the polar quadrants of the other three experts and the prey with respect to the expert who is teaching the trainer. M signifies the move that would be taken by the teacher expert under such an orientation. Once the trainers learn their coordination knowledge, they use the ATA framework to teach a new team member, the trainee.

4. RESULTS

Now we present the results from our experiments. The game is played on a 20×20 grid. We denote the various participants in our experiments by the following notations.

- Expert, E ,
- Trainer agent C: a decision tree based [7] learner,
- Trainee agent I: an instance based learner taught by a C agent in one iteration,
- Trainee agent A: an instance based learner taught by a C agent iteratively using the ATA framework.

Table 1 shows the performance of the different mixes of predators in the predator-prey pursuit game averaged over 1750 games. The first notable observation from Table 1 is the relative performances of the homogeneous groups of trainees who are taught either through the ATA framework or in one iteration. The relevant rows from Table 1 in this case are those with configurations AAAA and IIII. Clearly the AAAA configuration shows a higher capture rate and a faster capture to signify that the trainees who have undergone the learning process through the ATA framework have developed better coordination knowledge than those who have been taught in one iteration. Also the percentage capture rate for the AAAA configuration is fairly close to that of the CCCC configuration, consisting of all trainers, which shows that ATA framework has done a good job in teaching coordination knowledge to the new agents.

Now we shift our attention to heterogeneous predator team compositions. Interestingly, heterogeneous groups have smaller standard deviation compared to homogeneous groups, except for the group of experts. One reason for heterogeneous groups performing better than homogeneous groups can be that the differences in the knowledge of C and A agents can present useful complementarity that actually facilitate coordination. We also ran the Wilcoxon Matched-Pair Signed-Ranks test for pairs of groups where the team compositions differ by one trainee. The results show that the difference in step sizes to capture the prey is statistically significant for all such head-to-head comparisons.

Configurations	$\%Capture$	Steps	
		$Mean$	STD
EEEE	100	14.77	6.63
CCCC	86.74	45.49	32.53
CCCA	100	17.85	8.19
CCAA	95.77	24.12	11.39
CAAA	89.54	32.69	18.59
AAAA	81.54	43.34	29.90
IIII	67.42	62.18	50.32

Table 1: Summary performance on an average of 1750 games

5. RELATED WORK

Multiagent learning has been an active area of research in the late 90s [8, 9]. The most relevant work involves one learner telling another agent what portions of the search space to ignore [6], a learner sharing experience [2], problem-solving traces or even learned policies [10] with another concurrent learner. A recent paper discusses a method for two agent to mutually define a concept [11]. Though emphasis is placed on instance selection, in contrast to the current work, there is no pre-existing concept, and hence the learners are peers rather than a trainer-trainee pair. Tan's work [10] of an expert sharing effective problem solving traces with a novice agent and Clouse's work of a trainer suggesting actions to take [4] are perhaps the closest in motivation to the current work, but the iterative nature of teaching, at the heart of the ATA framework, is not addressed by them.

6. CONCLUSIONS

We developed a generic approach for knowledge transfer between experts and novice agents where the knowledge representation and learning algorithms of the experts and the trainees are fundamentally different. We present arguments for the generality of our approach for learning coordination knowledge and evaluate its effectiveness in the pursuit problem with a decision tree learner used as the learning module of the trainer and an instance based learner used as the learning module of the trainee. Initial results are encouraging and demonstrates effective transfer of coordination. The learners trained through the ATA framework exhibit consistently higher performance compared to the learners trained with a single presentation of training data. A team composed entirely of new trainees perform close to the team of trainers. More importantly, heterogeneous teams of trainers and trainees outperform homogeneous teams of either the trainers or the trainees. The results show effective transfer of coordination knowledge to new teammates by existing players using the ATA framework.

We plan to run experiments on a wider set of problem instances with variants of the pursuit problem. We plan to use a support-vector machine based agent as trainer and trainee agents in conjunction with the current instance-based agent and decision-tree based agents. We also plan to explore possible combinations of active learning and ATA approaches to train new teammates. In the future, we would like to run our ATA framework on predators that collude between themselves to evolve a strategy for capturing the prey in optimal number of steps as well as for transferring coordination knowledge in other challenging multiagent problems.

Acknowledgments: This work has been supported in part by an NSF award IIS-0209208 and an NASA EPSCoR RIG.

7. REFERENCES

[1] D. W. Aha, D. Kibler, and M. K. Albert. Instance-based learning algorithms. *Machine Learning*, 6(1), 1991.

[2] B. Banerjee, S. Debnath, and S. Sen. Combining multiple perspectives. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 33–40, San Francisco: CA, 2000. Morgan Kaufmann.

[3] M. Benda, V. Jagannathan, and R. Dodhiawalla. On optimal cooperation of knowledge sources. Technical Report BCS-G2010-28, Boeing AI Center, Boeing Computer Services, Bellevue, WA, Aug. 1985.

[4] J. A. Clouse. Learning from an automated training agent. ML-95 Workshop on Agents that Learn From Other Agents, 1995.

[5] T. Mitchell. *Machine Learning*. WCB McGraw-Hill, Boston, MA, 1997.

[6] F. J. Provost and D. N. Hennessy. Scaling up: Distributed machine learning with cooperation. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 74–79, Menlo Park, CA, 1996. AAAI Press.

[7] R. J. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, California, 1993.

[8] S. Sen and G. Weisß. Learning in multiagent systems. In G. Weiß, editor, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, chapter 6, pages 259–298. MIT Press, 1999.

[9] P. Stone and M. Veloso. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3), July 2000.

[10] M. Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 330–337, June 1993.

[11] J. Wang and L. Gasser. Mutual online concept learning for multiple agents. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 362–369, New York: NY, 2002. ACM Press.