Cañamero, L., Oudeyer, P.-Y., and Balkenius, C. (Eds) Proceedings of the Ninth International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems. Lund University Cognitive Studies, 146.

Formalization of different learning strategies in a continuous learning framework

Danijel Skočaj **Matej Kristan** **Aleš Leonardis**
University of Ljubljana
Faculty of Computer and Information Science
Tržaška 25, SI-1001 Ljubljana, Slovenia
{danijel.skocaj,matej.kristan,ales.leonardis}@fri.uni-lj.si

Abstract

While the ability to learn on its own is an important feature of a learning agent, another, equally important feature is ability to interact with its environment and to learn in an interaction with other cognitive agents and humans. In this paper we analyze such interactive learning and define several learning strategies requiring different levels of tutor involvement and robot autonomy. We propose a new formal model for describing the learning strategies. The formalism takes into account different levels and types of communication between the robot and the tutor and different actions that can be undertaken. We also propose appropriate performance measures and show the experimental results of the evaluation of the proposed learning strategies.

1. Introduction

An important characteristic of a robot that operates in a real-life environment is the ability to expand its current knowledge. The system has to create and extend concepts by observing the environment – and has to do so continuously, in a life-long manner.

As an example of such a learning framework, we need look no further than at the successful application of *continuous learning* in human beings. As humans, we can learn, for example, a new visual concept (e.g., an object category, an object property, an action pattern, an object affordance, etc.) by encountering a few examples of one. Later, as we come across more instances, different to the original examples, we not only recognise them, but also update our representation of learned visual concepts based on the salient properties of the new examples and without having visual access to the previous examples. In this way, we update or enlarge our ontology in an efficient and structured way by encapsulating new information extracted from the perceived data, which enables adaptation to new visual inputs and the handling of novel situations we may encounter.

Since humans are social beings this learning often takes place not in isolation, but rather in communication with other people. This communication can facilitate learning by exposing the knowledge that other possess also to the learner. It is very important for a robot, which is supposed to operate in a real world environment, to possess similar capabilities as well. The robot should be able to learn by interacting with the environment and with other knowledgable cognitive systems (e.g., a tutor), which may facilitate the learning process and make it robust and reliable.

In this paper we focus on such interactive continuous learning, where the robot is learning and continuously updating its knowledge autonomously or in a dialogue with a tutor. With respect to this, several learning strategies can be used; the robot can continuously learn while communicating with the tutor with different levels of tutor involvement and different levels of robot autonomy.

For performing a thorough analysis and evaluation of various learning strategies, it is necessary to formally describe the learning process and defined performance metrics. In this paper we propose such a formalism for specifying different learning strategies. In the proposed formal framework we also define four learning strategies ranging from tutor-driven to tutor-unassisted learning.

The paper is organised as follows. In the next section we first describe the related work. In Section 3. we then describe four learning strategies and in Section 4. the general formal model of learning strategies. This is followed by experimental evaluation of the presented learning strategies. The paper concludes with a final discussion and outlook.

2. Related work

A tutor's involvement by interaction plays an important role in the learning process in cognitive agents. Studies of human infants, for example (Pea, 1993), indicate that being able to exploit the expertise of others is a critical part of learning. Another point is the capability of the infants to take lead in the interaction, which is a foundation for many situated learning activities. Weng et al. (Weng et al., 2001) propose that similar measures should be undertaken in machine learning scenarios, in which the tutor should mentally *rise* the developmental robot through real-time interaction. This assumption is supported in the theory of cognitive development proposed by Vygotsky (Vygotsky, 1962), which states that social interactions are of essential importance for the development of individual intelligence. Building on a similar assumption, Thomaz (Thomaz, 2006) casts the machine learning problem as a strongly involved interaction between the human and the machine. As a feature of strong interaction (Thomaz, 2006) propose that the tutor has to have *a level of insight* into what the learner knows and which parts of the knowledge are ambiguous – the learner should be *transparent* to the tutor. In that respect, an involved interaction as a dialogue based learning scenario was also presented by Roy et. al (Roy and Pentland, 2002, Roy, 2002). Their system in (Roy and Pentland, 2002) was designed to learn word forms and visual attributes from speech and video recordings, and subsequently, Roy extended this work for generating spoken descriptions of scenes (Roy, 2002).

Researchers have dealt with various levels of tutor involvement in the process of learning in machines. At one extreme is an example in which the tutor is absent and the agent has to *learn on its own* starting from a very small or no prior knowledge, e.g., (Mugan and Kuipers, 2008, Oudeyer and Kaplan, 2004). However, allowing *learning from demonstration* (Argall et al., 2009) or *learning by imitating* (Schaal, 1999) the tutor can drastically reduce the search space for the agent's task and speed up learning. Examples of implicit or explicit learning from a *passive observation* can be found, for example, in the works of (Kuniyoshi et al., 1994, Billard and Dautenhahn, 1999, Lieberman, 2001). Another level of tutor's involvement is teaching by *directly influencing* the the actions of the machine. Such an example is when user biases the action selection in the machine (Maclin et al., 2005) or to allow direct control of robot's actions to supervise the process of reinforcement learning (Smart and Kaelbling, 2002). Kaplan et al. (Kaplan et al., 2001) explored animal training techniques to teach a robot to perform complex tasks. An example where the tutor plays an oracle was explored by Schohn and Cohn (Schohn and Cohn, 2000) – in that scenario, the agent provides some *level of transparency* by identifying the relevant examples and querying the tutor for the required labels. Allowing the robot to *actively express its uncertainty*, or a gap in the knowledge, was explored in the "Ask for Help" framework (Clouse, 1996) and, for example, (Nicolescu and Mataric, 2003). An approach to reinforcement learning which can *learn from tutor's feedback* was presented in (Knox and Stone, 2008).

Learning in cognitive robots can therefore be described in terms of different levels of tutor involvement as well as levels of learner's responsiveness and learner's transparency. As noted above, various researchers have dealt with scenarios with various levels of the tutor-learner interaction, leading to different learning strategies. With this respect, the closest related work is (Chernova and Veloso, 2009), where the authors propose and evaluate similar learning strategies to those discussed in this paper (although in a different learning domain). The main contribution of this paper, however, goes beyond the definition of the learning strategies; we also propose a formalism for modeling these strategies. In fact, also the learning strategies like those presented in (Chernova and Veloso, 2009) could be modeled with the formal model presented here. This is also the main goal of our work; to introduce a formalism that would enable simple and efficient definition, evaluation and comparison of different learning strategies.

3. Learning strategies

The interaction between the tutor and the robot plays an important role in a continuous learning framework. The goal of the learning mechanism is to continuously learn and update the acquired concepts, i.e., to find associations between the words spoken by the tutor (and related amodal concepts) and features, which are automatically extracted from the observations. Such a continuous learning framework should communicate with the tutor, perform recognition, and update the representations according to the current learning strategy. In this section we define several learning strategies which alter the behaviour of the system and require different levels of tutor involvement.

In the core of any learning strategy is a **learning algorithm** that actually builds and updates the representations. Before we proceed with the definition of the learning strategies, let us introduce several requirements for the learning algorithm.

Most importantly, the learning algorithm has to be **incremental**; the representation, which is used for modeling the observed world, has to allow for updates when presented with newly acquired information. This update step should be efficient and should not require access to previously observed data, while still preserving the previously acquired knowledge.

In addition, in continuous learning scenarios the noise in the input data has a detrimental effect on the learnt representations, especially when the robot learns autonomously. If, for example, the recognition algorithm fails at some point to correctly interpret the visual scene and erroneously updates the current knowledge, the models of the concepts tend to degrade and the performance of the system will typically decrease severely. However, in interactive settings the tutor can help the robot to recover from the errors through interaction, by, e.g., indicating to the robot that its belief about a certain concept is wrong. The system should be then able to **unlearn**, i.e. to update the representation by considering the wrongly classified sample as a negative training example. Unlearning step may lead to the correction of the current representation, which can improve the performance considerably.

Finally, it is obvious that the system is supposed to have a certain level of **self-understanding**; it should be able to estimate whether its current knowledge suffices to interpret the current scene, or it should ask the tutor for help. Therefore, it should have a recognition capability, i.e., the ability to interpret the current observation to some extent. And even more importantly, the system should be able to evaluate the reliability of this recognition process.

We therefore assume that the learning algorithm, which is used in the continuous learning framework, fulfills the criteria mentioned above.

We define a **learning strategy** as a common strategy of the tutor and the robot that specifies the behaviour of the robot and the tutor in the continuous learning process. It specifies when the robot updates its knowledge autonomously and how and when the tutor and the robot communicate in order to extend the robot's knowledge. According to this definition and considering different levels of interaction between the tutor and the robot, various learning strategies are possible. Here we identify four such strategies:

- **Tutor-driven.** The tutor drives the learning by describing the observation and giving all available information to the robot. The communication is one-directional, the learning process is completely controlled by the tutor.

- **Tutor-supervised.** The robot establishes transparency; the tutor assesses the robot's knowledge and detects its ignorance. When the robot fails to correctly interpret the current observation, the tutor provides the correct information, which helps the robot to update or unlearn the current representations accordingly.

- **Tutor-assisted.** The robot tries to interpret the current observation. If it succeeds to do this reliably, it updates the current model, otherwise asks the tutor for the correct interpretation. The tutor therefore gives the information to the robot only when asked for assistance.

- **Tutor-unassisted.** The system updates the model with the automatically obtained interpretation of the visual input. No assistance from the tutor is required. There is no communication between the tutor and the robot.

The dialogue in the first two learning strategies is initiated by the tutor, while in the second two cases the robot takes the initiative. These four learning strategies range across the entire spectrum of different levels of the tutor involvement and the robot's autonomy. In Tutor-driven mode the tutor completely drives the learning process, in Tutor-supervised mode he intervenes only when necessary, in Tutor-assisted mode only when he is asked for, and in Tutor-unassisted mode even never. On the other hand, the autonomy of the robot increases from Tutor-driven mode, where the robot does not influence the learning process, to Tutor-unassisted mode, where it completely autonomously controls the learning. This is also depicted in Fig. 1.

The spectrum of different learning modes is of course not discrete as presented here; it is continuous and one could define additional learning strategies with similar properties. It is also possible to combine different learning strategies, to execute them in a sequence and to switch between them when necessary. In practice, the learning strategy should change over time, adapting to the current level of knowledge and complexity and novelty of the environment the robot is currently situated in. We believe, however, that the presented four learning strategies span across the entire space of possible learning strategies and cover a major part of its variability.

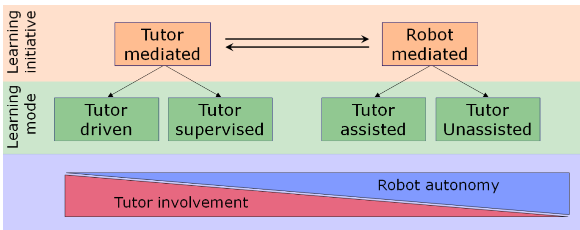


Figure 1: Learning strategies.

4. Formal model

In the previous section we have conceptually described a few possible learning strategies. Here we present a general formalism, which can be used to formally define these or many other learning strategies.

We will limit our analysis on the continuous learning scenarios, in which a robot observes a scene and learns new concepts through interaction with a tutor. This interaction can be quite simple or very complex; different learning strategies employ different levels of communication. We assume that the robot and the tutor can establish the common ground; they have all necessary communication capabilities, they observe the same scene, and in the dialogue they refer to the same object.

The robot and the tutor are involved in a continuous and interactive learning process; the robot continuously observes objects, it tries to recognize them and learn something new about them. Every learning step therefore starts with the robot trying to interpret the current scene. It tries to recognize all the concepts it currently knows. Based on the classification confidence (see Fig. 2), the robot can assign **soft labels** when trying to determine whether the current observation is indicative of a given concept or not:

- '**Yes**' (YES): The recognition confidence is very high, the robot reliably classifies the current observation as being an instance of a particular concept.
- '**Probably yes**' (PY): The recognition confidence is relatively high, however the robot is not certain about its current interpretation.
- '**Probably no**' (PN): The recognition confidence is relatively low; the current observation probably does not indicate the particular concept.
- '**No**' (NO): The recognition confidence is very low, therefore the robot reliably classifies the current observation as not being an instance of a particular concept.
- '**Don't know**' (DK): The recognition was not sufficiently reliable to determine the answer.
- '**Unknown**' (UK): The robot has not yet encountered the certain concept it was asked about.

Based on the output of the classifier and as instructed by the chosen learning strategy, one of the following four **actions** follows:

- **Do nothing.** The robot does not update its current knowledge nor does request an interaction with the tutor.
- **Autonomously update.** The robot updates the current knowledge with the information autonomously inferred from the current observation without involving the tutor.
- **Tell.** The tutor gives the correct information about the current observation to the robot.
- **Ask.** The robot asks the tutor for clarification about the current observation and the tutor replies with the correct answer.

In the latter three cases an update of the current knowledge follows (either based on the automatically extracted information or on information obtained by the tutor). Two different kinds of **update** are possible:

- **Update with a positive example.** The robot updates its current knowledge by integrating the positive training sample into its current representation of the particular concept.
- **Unlearn with a negative example.** The robot unlearns its current knowledge; based on the given negative example, it corrects the current representations not to model this negative example.

To fully describe the learning strategy we also need to define the intensity of communication between the robot and the tutor. We define three such **communication levels**:

- **Ignoring.** The tutor ignores the robot's output; the state and performance of the robot do not influence the tutor's behavior.
- **Listening.** The tutor listens to the robot and correctly answers with 'yes' or 'no' when being asked a polar question.
- **Transparency facilitated assessment.** The robot establishes transparency and the tutor is able to assess the robot's current interpretation of the observation.

Now, let us denote the above mentioned four actions with the following signs: '/' for *'do nothing'*, '**U**' for *'auto-update'*, '**T**' for '*tell*', and '**A**' for *'ask'*. In addition, with a subfix next to these signs we will denote an *update with positive example* with the plus sign ('+') and an *unlearning* request with the minus sign ('**-**'). For instance, 'U_+' means that the system will automatically update the current knowledge with the information inferred from the current observation, while 'A_-' means that the robot will ask the tutor for clarification, the tutor will reply with a negative answer and the robot will unlearn its current knowledge accordingly. Similarly, let us denote the communication levels with '**ign**' (*ignoring*), '**lst**' (*listening*), and '**tfa**' (*transparency facilitated assessment*).

To fully describe a learning strategy, we need to define what will happen if the robot correctly or incorrectly interprets the current observation with respect to all known concepts. Therefore, we need to define the action that will be undertaken depending on the robot's autonomous interpretation of the scene (**soft label** *sl* that is autonomously assigned for a particular concept). We assume that the tutor is omniscient and always gives the correct information to the robot; therefore the tutor's actions will also depend on the **ground truth data** (gt), which tells if the observation is an instance of the particular concept or not.

Now, a learning strategy can be defined as a 13-tuple LS:

$$\begin{aligned} LS &= [act_{sl,gt}, cl]\,, \text{where} \qquad (1)\\ sl &\in \{YES, PY, PN, NO, DK, UK\}\\ gt &\in \{yes, no\}\\ act_{\cdot,\cdot} &\in \{/, U_+, U_-, T_+, T_-, A_+, A_-\}\\ cl &\in \{ign, lst, tfa\} \end{aligned}$$

Note that $act_{sl,gt}$ denotes 12 elements (2×6 combinations of sl and gt, i.e., $act_{YES,yes}$, $act_{YES,no}$, $act_{PY,yes}$, etc.[1]). This vector exactly specifies what will happen in certain situations. When the robot observes a new observation it tries to determine whether it belongs to a certain concept or not, and assigns a soft label (sl) as described above. This label is then together with the known ground truth (gt) used to index in the vector LS; the obtained action $act_{sl,gt}$ exactly specifies which action (or sequence of actions) will be undertaken.

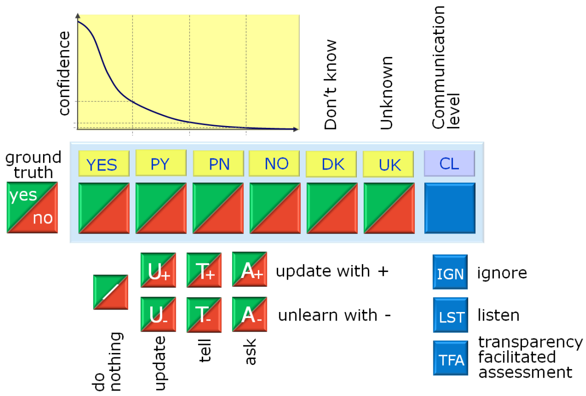


Figure 2: Parametrisation of learning strategies.

To demonstrate this formalism, let us formally define the four learning strategies presented in the previous section (see also Fig. 3):
$LS_{TD} = [T_+, /, T_+, /, T_+, /, T_+, /, T_+, /, T_+, /, ign]$
$LS_{TS} = [U_+, T_-, U_+, T_-, T_+, /, T_+, /, T_+, /, T_+, /, tfa]$
$LS_{TA} = [U_+, U_+, A_+, A_-, A_+, A_-, /, /, A_+, A_-, T_+, /, lst]$
$LS_{TU} = [U_+, U_+, U_+, U_+, /, /, /, /, /, /, T_+, /, ign]$

In *Tutor-driven* learning mode, the tutor ignores the output of the robot (ign); it always gives the robot the correct (positive) information about the current observation (T_+). In *Tutor-supervised* mode, the tutor observes the robot and assesses its current knowledge (tfa). The tutor lets the robot automatically update the current knowledge (U_+), when its interpretation is correct, or he corrects the robot, when its interpretation is incorrect by telling it the correct information (T_- or T_+). In *Tutor-assisted* mode the tutor listens to the robot (lst), which autonomously decides either to update the knowledge automatically (U_+), when it trusts to its recognition result, or to ask the tutor for help, when the recognition was not reliable. In the latter case, the tutor responds with 'yes' (A_+) or 'no' (A_-) according to the ground truth label, which in turn enables the robot to update or unlearn its current knowledge. Finally, in the *Tutor-unassisted* learning, the robot only relies on its current recognition abilities and does not ask the tutor for help. The robot is therefore ignored by the tutor (ign) and updates its current knowledge autonomously (U_+).

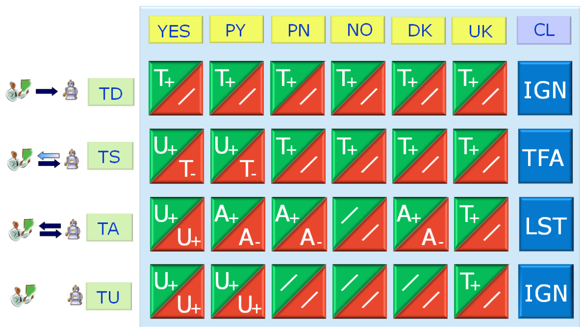


Figure 3: Formal definition of four learning strategies.

Such learning formalism allows us to formally define evaluation measures. Instead of standard recognition rate we propose to use a **recognition score**, which rewards successful recognition (true positives and true negatives) and penalizes incorrectly recognised concepts (false positives and false negatives) by taking into account soft labels. The scoring rules are presented in Table 1; it shows how many points (-1 to 1) the system is rewarded with for each of the answers given in the first row, depending on the correct answer as given in the first column.

Table 1: Scoring table.

	YES	PY	PN	NO	DK	UK
yes	1	0.5	-0.5	-1	0	0
no	-1	-0.5	0.5	1	0	0

The recognition score thus measures how successfully the robot recognizes the learned concepts (therefore, how successful the learning was). However, in interactive learning scenarios another criterion is also important; the **tutoring costs**. Obviously, one would prefer that the robot learns autonomously as much as possible, without involving the tutor too frequently. During the learning process different types of tutoring costs may occur (in

[1] With capital letters (e.g., YES), we denote the label autonomously inferred by the robot, while with small letters (e.g., yes) we denote the actual (ground truth) label for a particular concept.

different learning strategies):

- C_{inf}: costs of providing some information to the robot.
- C_{ans}: costs of answering a polar question to the robot.
- C_{ign}: costs of ignoring the robot's output.
- C_{lst}: costs of listening to the robot.
- C_{tfa}: costs of assessing the current robot's knowledge.

Let us suppose that at a particular learning step the tutor gave N_{inf} concept labels about the correct observation to the tutor and answered N_{ans} polar questions. Now we can define the overall tutoring costs at that particular learning step as

$$TC = N_{inf}C_{inf} + N_{ans}C_{ans} + C_{cl} \qquad (2)$$

where cl is one of three communication levels as defined above.

The values of the parameters C_* depend on the actual costs that occur during the interactive learning. In this paper we use the values presented in Table 2. We set the cost of assessing the robots knowledge high, since this is not a trivial task for the tutor. If, for instance, the robot would establish the transparency by verbalizing its current beliefs, the tutor would just have to listen to it and the cost of assessing the knowledge would be lower, i.e., $C_{tfa} = C_{lst}$.

Table 2: Tutoring costs.

C_{inf}	C_{ans}	C_{ign}	C_{lst}	C_{tfa}
1	.25	0	.25	2

5. Experimental results

For performing large scale experiments and evaluating different learning strategies we have developed *Interactive Continuous Learning Simulator*, which implements the formal model of learning strategies presented in the previous section. This simulation environment uses as observations the features that were automatically extracted from the previously captured, automatically processed and manually labeled real data; the tutor is replaced by an omniscient oracle, which has the ground truth data available. The simulator enables large scale experiments and a thorough evaluation and comparison of different learning methods and strategies.

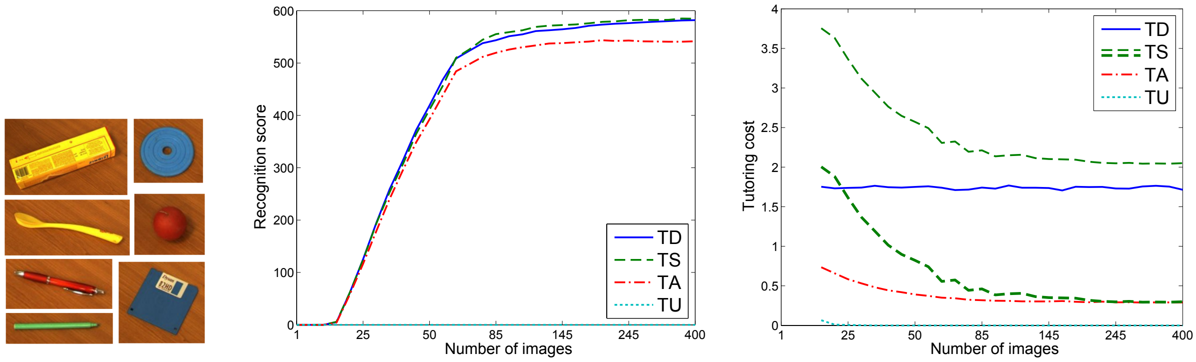
We performed a number of experiments to evaluate different learning strategies on different learning domains. Here we present the results of the experiment where the goal was to learn basic visual attributes like colour and shape by observing a set of everyday objects (some of them are depicted in Fig. 4(a)). Six visual attributes were considered; four colours (red, green, blue, yellow) and two shapes (elongated, compact). The database that we used for learning contains 500 images. 400 images were used to incrementally learn the representations of six visual properties, while the rest 100 of them were used as test images. We repeated the experiment for 100 runs by randomly splitting the set of images into the training and test set and averaged the results across all runs. In all the experiments we used the extended algorithm for incremental learning that we have previously proposed (Skočaj et al., 2008, Kristan et al., 2009).

During the experiment, we kept incrementally updating the representations with the training images using different learning strategies as defined in the previous section. At each step, we evaluated the current knowledge by recognising the visual properties of all test images. The learning performance was evaluated using two above defined performance measures: recognition score and tutoring costs.

Figs. 4(b,c) show the evolution of the learning performance over time for all four learning strategies. First thing to note is that the overall results improve through time. The growth of the recognition score is very rapid at the beginning when new models of newly introduced concepts are being added, and still remains positive even after all models are formed due to refinement of the corresponding representations.

Tutor-driven and Tutor-supervised learning yield similar recognition score; they almost achieve the perfect score (600 in this case). Tutor-supervised learning performs slightly better, since it sooner achieves better results. This is somehow expected, since in this case the tutor corrects the robot when necessary and the robot unlearns the erroneous representations. The inherent problem of any continuous learning framework, which involves autonomous updating of the knowledge, is propagation of errors. The tutor supervision efficiently helps the robot to recover from this errors, if the robot transparency has been achieved. The error recovery is in this experiment less effective in the Tutor-assisted case. The errors are in this case detected by the robot (and not by the tutor). Obviously, this error detection is not so efficient, therefore the recognition score is lower. In this experiment, Tutor-unassisted learning did not perform well; without sufficiently good initial knowledge it was not able to improve without any assistance from the tutor.

We also have to take into account the tutoring costs that occur during the learning. In Tutor-driven learning mode they are almost constant; the tutor always gives all the information about the current object, which is available. The costs of Tutor-assisted learning are significantly lower. The robot keeps ask-



(a) (b) (c)

Figure 4: Experimental results: (a) Seven everyday objects from the database. (b) Evolution of Recognition Score, (c) Tutoring Costs. Note the logarithmic scale along abscissa.

ing the tutor only at the beginning of the learning process; after its knowledge gets improved the number of questions drops and most of the costs relate to the fact that the tutor has to listen to the robot and await for its questions. The costs of Tutor-supervised learning are relatively high, since in this experiment we use the settings presented in Table 2, which assume that it is relatively expensive to asses the robot's knowledge. In addition to that, at the beginning there is a lot of communication between the tutor and the robot, which again drops when the models of the concepts get stabilized. If the robot establishes its transparency by verbalizing its beliefs about current observations, the costs of assessing the knowledge are significantly lower, and the overall tutoring costs significantly decrease (the strong dashed line in Fig. 4(c)), making Tutor-supervised learning more efficient than the Tutor-driven. This holds true also in practice; it is more convenient (and effective) for the tutor just to listen and correct the learner occasionally than to continuously giving it new information.

6. Conclusion

In this paper we have introduced a new formal model for formalizing learning strategies. We define a learning strategy as a common strategy of the tutor and the robot that specifies the behaviour of the robot and the tutor in the continuous learning process. The formalism takes into account different levels and types of communication between the robot and the tutor and different actions that can be undertaken. By specifying these actions an communication levels, the learning strategy can be uniquely defined.

In general, it is very difficult to objectively compare different (incompatible) learning processes; the presented formalism makes this comparisons straightforward. This will allow us to analyse different learning strategies, to efficiently combine them and to find a way how to exploit the properties of the individual strategy best.

In addition, we introduced four learning strategies that span across the entire space of possible learning strategies and cover a major part of its variability. They range across the entire spectrum of different levels of the tutor involvement and the robot's autonomy. We also evaluated these four learning strategies using the proposed performance metrics.

While the currently presented formalism may appear to simplistic to apply to richer scenarios with shifting the focus of attention and more complex dialogues, we believe that it forms a solid base of building blocks for basic tutor-learner interaction. In our future work we will build upon this base and establish means of combining these blocks into more complex framework which will account for more complex situations.

Our primary goal is to develop a robot that would be able to efficiently acquire new concepts and to update the existing ones in collaboration with a human teacher. We have implemented the learning strategies introduced in this paper on a real robot (for details the reader is referred to (Vrečko et al., 2009)). When conducting research on interactive learning it is crucial to have a real implementation of the learning framework on real robots and to test its functionality in real-world settings. However, it is equally important also to have formalisms and tools to perform large scale experiments, which enable thorough evaluation and analysis of the proposed methods. We believe that the proposed formal model can facilitate such research and enable further development of related approaches.

Acknowledgements

This research has been supported in part by the EU FP7 project CogX (ICT-215181) and the Research program Computer Vision (RS).

References

Argall, B., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483.

Billard, A. and Dautenhahn, K. (1999). Experiments in learning by imitation - grounding and use of communication in robotic agents. *Adaptive Behavior*, 7(3/4):415–438.

Chernova, S. and Veloso, M. (2009). Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research*, 34:1–25.

Clouse, J. (1996). *On integrating apprentice learning and reinforcement learning*. PhD thesis, University of Massachusetts Amherst.

Kaplan, F., Oudeyer, P.-Y., Kubinyi, E., and Miklosi, A. (2001). Taming robots with clicker training : a solution for teaching complex behaviors. In *European workshop on learning robots, LNAI, Springer*.

Knox, W. and Stone, P. (2008). TAMER: Training an agent manually via evaluative reinforcement. In *IEEE 7th International Conference on Development and Learning*, pages 292–297.

Kristan, M., Skočaj, D., and Leonardis, A. (2009). Online kernel density estimation for interactive learning. *Image and Vision Computing*.

Kuniyoshi, Y., Inaba, M., and Inoue, H. (1994). Learning by watching: Extracting reusable task knowledge fromvisual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10:799–822.

Lieberman, H., (Ed.) (2001). *Your Wish is My Command: Programming by Example*. Morgan Kaufmann, San Francisco.

Maclin, R., Shavlik, J., Torrey, L., Walker, T., and Wild, E. (2005). Giving advice about preferred actions to reinforcement learners via knowledge-based kernel regression. In *National Conference on Artificial Intelligence*.

Mugan, J. and Kuipers, B. (2008). Towards the application of reinforcement learning to undirected developmental learning. In *8th International Conference on Epigenetic Robotics*.

Nicolescu, M. and Mataric, M. (2003). Natural methods for robot task learning: Instructive demonstrations, generalization and practice. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 241–248. ACM New York, NY, USA.

Oudeyer, P.-Y. and Kaplan, F. (2004). Intelligent adaptive curiosity: a source of self-development. In *Proceedings of the 4th International Workshop on Epigenetic Robotics*, volume 117, pages 127–130. Lund University Cognitive Studies.

Pea, R. D. (1993). *Distributed cognitions: Psychological and educational considerations*, chapter Practices of distributed intelligence and designs for education, pages 47–87. Cambridge University Press, New York.

Roy, D. K. (2002). Learning visually-grounded words and syntax for a scene description task. *Computer Speech and Language*, 16(3):353–385.

Roy, D. K. and Pentland, A. P. (2002). Learning words from sights and sounds: a computational model. *Cognitive Science*, 26(1):113–146.

Schaal, S. (1999). Is imitation learning the route to humanoid robots? *Trends in Cognitive Science*, 3(6):233–242.

Schohn, G. and Cohn, D. (2000). Less is more: Active learning with support vector machines. In *17th International Conference on Machine Learning*, pages 839–846.

Skočaj, D., Kristan, M., and Leonardis, A. (2008). Continuous learning of simple visual concepts using Incremental Kernel Density Estimation. In *International Conference on Computer Vision Theory and Applications*, pages 598–604.

Smart, W. D. and Kaelbling, L. P. (2002). Effective reinforcement learning for mobile robots. In *IEEE International Conference on Robotics and Automation*, pages 3404–3410.

Thomaz, A. L. (2006). *Socially Guided Machine Learning*. PhD thesis, Massachusetts Institute of Technology.

Vrečko, A., Skočaj, D., Hawes, N., and Leonardis, A. (2009). A computer vision integration model for a multi-modal cognitive system. In *2009 IEEE/RSJ International Conference on Intelligent RObots and Systems*.

Vygotsky, L. (1962). *Thought and Language*. Cambridge, MA: MIT Press.

Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2001). Autonomous mental development by robots and animals. *Science*, 291(5504):599 – 600.