The Application of Learning Algorithms in the Development of Natural Interaction

Emilia Barakova, Ruud Mestrom, Willem Willemsen
Department Industrial Design
Technical University of Eindhoven
Den Dolech 2, 5612AZ, Eindhoven
+31 (0) 40 247 9111

ABSTRACT
This paper argues that natural interaction with a machine can be realized and improved by using learning algorithms. Through the use of supervised and reinforcement learning algorithms, a robot was created that can be trained to perform actions using only verbal commands. The user has complete freedom in choosing preferred commands and what actions should be linked to these commands. The combination of supervised and reinforcement learning resulted in a fundamentally different way of interaction with a robot. The way this system was set up can be used as a framework for new projects, giving designers a new tool to improve human-machine interaction.

Keywords
Natural interaction, learning algorithms, design principles

INTRODUCTION
Nowadays people interact with many digital devices, but the interaction is not developing at the same rate as the technology that enables them. The interaction between people and machines has been determined mainly by technological constraints; people had to adapt to the machines. Modern technologies make it possible to design machines that adapt to human interaction [1].

This paper suggests that we should move towards more natural interaction. Natural interaction is inspired by human-human communication and uses other ways of human expressions, like voice, gesture and face/expression recognition/reading. [2] New, smarter and more powerful technologies allow these new ways of interaction.

The industrial design discipline is moving more and more in the direction of interactive digital products. 30 years ago, in 1982, Xerox PARC realized the first form of visual computing using the WIMP (Windows, Icons, Menus and Pointers) model. Today’s computer looks much like those developed in 1982. They may be smaller, faster, sleeker and more powerful; their basic interaction models have changed very little. [3] The interaction with other devices (e.g. mobile phones, kitchen appliances, alarm clocks) is still largely based on the WIMP model or interaction through buttons and switches. This form of interaction is highly unnatural for people and should be redesigned.

Enabling an embodied system to take part in human interaction/communication, requires designing robot behaviours that bring the human-machine interaction to a new level, as the robot takes intelligent decisions on its interactive response that comes as a result of a learning and interpretation of human behaviour. (New kind of mapping that robot can do itself). There are many examples of human interaction with robots, such as robots that have autobiographical memory (10,11), emotions (8) and interpret mental states (9).

This creates new opportunities for designers, because they are not limited to predefined possibilities but give the user the possibility to interact the way they prefer. Humans are used to by interacting with the real world in everyday life, as evolution (and their cultural past) and education taught them to do [2]. One could think of recognising a word. People can easily recognise the same word said by different people, because they have heard the word before and can recognise it based on memory.. Machines can also learn how to analyse and give meaning to the sensor data by using learning algorithms and by providing it with a memory, a range of examples, as reference for new input.

This paper will discuss how learning algorithms can be applied to implement natural interaction with machines. One example has been worked out in a functioning prototype that shows the possibilities that learning algorithms give designers to create new interactions with digital devices.

MATERIALS AND METHODS
The goal of our work was to create a robot that can be trained and controlled using only voice commands. Just like training a dog, the user can give feedback to reinforce desired behaviour. Speech is a form of interaction that has become a natural way of communication for people.
Algorithms
Training the robot consists of two parts. Training to recognize different voice commands using a supervised learning algorithm and a reinforced learning algorithm that connects commands to actions, using reward and punishment. Together they enable the robot to recognize commands and map them to corresponding actions.

Supervised learning requires sample input-output pairs from the function to be learned [4]. In other words, supervised learning requires a set of questions with the right answers. In our case the input was a set of recordings of the commands for the robot. Each command was recorded 10 times by different people to provide the robot with a wide range of data for analysis. The answers were given by giving the robot the input data in meaningful sets instead of random input data. This will result in the robot being able to distinguish the different commands. Supervised learning was used, because the action possibilities of the robot (and voice commands) were limited and known beforehand. A multi-layer perceptron model was used to create an algorithm based on this training set.

In reinforcement learning, a robot is given a high level goal to achieve. The robot then learns how to achieve that goal by trial-and-error interactions with its environment. [5] For this robot a Q-learning algorithm was used to create a fast and flexible mapping of input and output. The software for the robot has variables for rewards for each command and it will try to get the highest rewards possible. In theory the robot can be trained entirely with commands, using positive and negative feedback loops. When a response to a command receives positive feedback (“well done”), the robot will try that combination more often. However to speed up the initial training we implemented keyboard feedback to improve training speed. Q-learning is quick to learn, but errors can slow down the learning process.

Experiment approach
The AdMoVeo robot was used as platform for our experiments. This robot is wirelessly connected to a computer which contains the software. It has several actuators (wheels, LEDs and a buzzer) and sensors (IR-distance, sound, line reader) to interact with the world.

Implementing the learning algorithms
The goal of this experiment was to find out how learning algorithms could enable a robot to be trained with voice commands. Learning algorithms were used and combined to create a one-way verbal communication between the robot and the user. We wanted to create robot that is able to recognize a specific set of commands (e.g. go, stop, left or right) and attempts to perform the action requested. The robot received positive or negative vocal feedback and updates the q-value for that specific action. After several iterations it is able to successfully perform actions.

The supervised learning algorithm was applied in a program called ‘Neuroph’. This program allows training a neural network. Voice commands were recorded several times to improve accuracy of voice recognition. The initial setup analysed 50 sections of maximum volume of the total sample of 100 ms. This approach allowed us to train two different words, but when they are more alike it did not recognize the word anymore. Another 10 sections using FFT (Fast Fourier Transfer) with 8 frequencies for each section was added to improve accuracy.

Voice command recordings: Too much similarity between the “go” and “right” command using only volume analysis.

The Q-learning algorithm was used to link commands to actions. The robot responds to a voice command with a random action. After performing a (random) action, it is rewarded or punished by the user and the robot remembers this. Next time it hears a command it will try to perform a command to receive a reward. By trying all actions with all commands several times, it is able to learn which action will result in the highest reward, thus giving that action a higher probability of being performed each time it is followed by a reward from the user. The more often this process is repeated, the better the robot will be trained.
RESULTS
The trained neural network was able to give a 90% accurate outcome in recognizing voice commands. Sometimes it was not able to recognize a command, or the wrong command was recognized. With more training and better/larger data sets, the accuracy could be further improved. Also, we used FFT to analyse the voice recordings because it was most feasible in this course. The Hidden Markov Model has been shown to be a more suitable algorithm for speech recognition, so this could further improve the accuracy [6].

The Q-learning algorithm is a fast learning algorithm if a single action is rewarded. When the actions became more complex (e.g. “zig-zag movement”), the robot must perform multiple actions and receive only one reward, it will take much more time. To speed up the training, the feedback loop was included in the programming instead of waiting for verbal feedback from the user to radically increase learning time. Training a robot using verbal rewards would seriously slow down the learning of the robot, because it would have to wait for feedback after each action before it can continue. Training a complex (sequence of) action(s) could take days for an active robot with this learning algorithm.

CONCLUSION
The outcomes of the experiments suggest that learning algorithms can make a product suitable for more natural interaction with/for the user. It enables machines to understand human interaction and this will result in natural human-machine interaction. As natural interaction develops, machines will no longer dictate the interaction, but humans will be empowered to define and design the interaction with their environment. Machine learning is crucial in the development of new ideas and future interaction with everyday devices.

There is a trade-off between learning rate (speed) and accuracy. A neural network offers “flexibility” and with a bigger training set it will become more accurate, but there will always be a 2% margin of error [7]. A limited set of voice commands was used, but the robot could use a different learning algorithm, like a neural gas algorithm to learn new commands more easily. This will however also greatly increase the list of possible commands an increase learning time for the Q-learning algorithm. In theory it would be possible to train the commands with an unsupervised training model, and let the system map commands on its own. However it could happen that commands are not mapped correctly and in that case there is no feedback to the user to indicate this problem. A supervised learning algorithm might not be as flexible as an unsupervised one, but in our case it was more suitable and it gave the user more feedback on the mapping of actions.

DISCUSSION
This exploration resulted in an example of a product that has a more natural form of interaction than the common control for robots that mainly use buttons. Learning algorithms were crucial for the robot to recognise and process the “human” input. The algorithm is able to recognize patterns that will create a product that can be used by any person speaking the English language. The programming allows people who have no previous knowledge or experience with (programming) digital interfaces, to interact with this product and enjoy it.

The beauty of learning algorithms is that a system is also able to recognize incomplete patterns. Due to a programming error in one of the early experiments, volume wasn’t recorded in the field test, but it still recognized over 70% of the given commands with an incomplete pattern. An absolute, digital system would not have been able to deal with this human error.

Accuracy and reliability of the work in this paper could be improved in several ways. Only the basic algorithms of different learning paradigms were implemented in the robot. Accuracy can be improved with a larger training set of multiple people. The learning algorithm itself can also
be fine-tuned for speech recognition, in this example we used the more general formulation.

Learning algorithms will always keep a certain error margin (of about 2%), even in optimal conditions. That means that there’s always the possibility that commands won’t be recognized. This could lead to frustration with the user when a system performs an action that doesn’t match the expectations of the user. However this fits the human nature because people also suffer from miscommunication. People often assign human characteristics to computers. “It did that on purpose, just to annoy me.” The fact that computers and other devices can become more “human” might have very interesting effects on our relationship with our environment (of artefacts). This is certainly an interesting topic for future research as well.

FUTURE WORK
Several design cases were explored to provide a range of application possibilities of learning algorithms in everyday life. Four different scenarios are proposed that rely on learning algorithms to improve natural interaction with products.

Alarm clock
The first is an alarm clock that tries different strategies to effectively wake a person. It has different means in its arsenal: snooze, radio, light, sound and movement. It can try different combinations, which will lead to a different wake up strategy. It will measure how the user responds and finally will determine the ideal strategy for a specific user to prevent them from oversleeping.

Adaptive chair
This chair changes its position depending on the body characteristics and the specific activity (e.g. relaxing, working, dining, etc.) The user is able to correct the position of the chair, which will serve as input for the system and the system will learn to match user activities to desired chair settings.

Phone
Mobile phones are limited in the way of input. Old phones use buttons; new developments incorporate a touch screen, GPS and motion sensors that offer more freedom of input. The phone could map its sensors data to following actions of the user. If the phone would then recognise the same input data (e.g being taken out of a pocket), it could perform the desired action automatically.

ACKNOWLEDGMENTS
We thank our other team members, Joey van Dun and Marco van Beers for their hard work and input.

REFERENCES
11. E. I. Barakova, T. Lourens. Spatial navigation based on novelty mediated autobiographical memory. LNCS 3561, pages 1 – 10, Las Palmas, Spain, June 2005