# Jumpy – Reinforce Learning about Reinforcement Learning from an AI's Point of View

Annika Vielsack
Department of Informatics
Karlsruhe Institute of Technology
Karlsruhe, Germany
vielsack@kit.edu

Martina Landman*
Department of Computer Science and Technology
University of Cambridge
Cambridge, United Kingdom
ml2258@cam.ac.uk

Tobias Kohn
Department of Informatics
Karlsruhe Institute of Technology
Karlsruhe, Germany
tobias.kohn@kit.edu

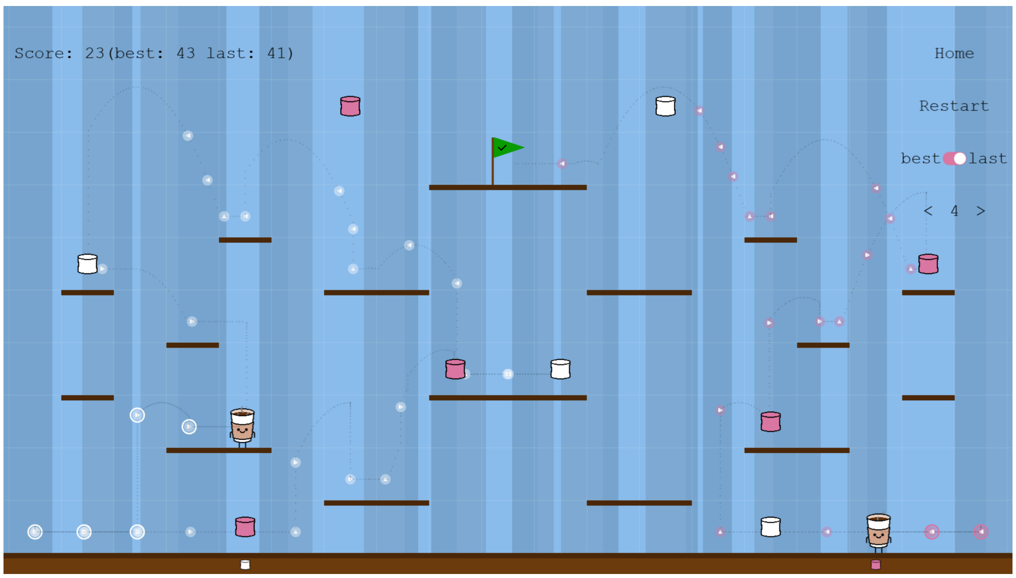


**Figure 1: Gameplay of *Jumpy* in the adaptation stage.**

## Abstract

The topic of Artificial Intelligence is gaining in importance and so is the need to learn about the core principles of AI. In this tool paper, we present *Jumpy*, an educational jump 'n' run game designed to provide hands-on experience with reinforcement learning. The students take the AI's point of view and act as an AI system themselves. While playing and thereby using reinforcement learning, they learn about key concepts like agents, reward functions, and policies. The tool is intended for use in classroom settings, with students working in pairs and participating in guided discussions. We evaluated *Jumpy* with more than 120 students aged 11–12. One third formed an experimental group while the remaining students served as a control group and used the tool after the formal evaluation. Both groups completed a questionnaire before and after the intervention. The results indicate a shift in students' attitudes towards AI and an increased familiarity with AI-related concepts.

## CCS Concepts

• **Theory of computation → Reinforcement learning**; • **Applied computing → Education**.

## Keywords

AI, AI Literacy, Reinforcement Learning, Computer Science Education, K-12 Education, Tool

**ACM Reference Format:**
Annika Vielsack, Martina Landman, and Tobias Kohn. 2026. Jumpy – Reinforce Learning about Reinforcement Learning from an AI's Point of View. In *28th Australasian Computing Education Conference (ACE 2026), February 09–13, 2026, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3786228.3786253

*Work done while at TU Wien, Austria.


*ACE 2026, Melbourne, VIC, Australia*

ACM ISBN 979-8-4007-2352-0/26/02
https://doi.org/10.1145/3786228.3786253

## 1 Introduction

Recent years have seen an unprecedented proliferation of artificial intelligence (AI), particularly in the form of generative AI. As a consequence, there seems to be widespread agreement that AI has to be integrated into (computer science) education to reflect its importance in modern digital systems [23, 24, 32], although there is no clear consensus which parts or aspects of AI should be prioritised or included, how and in what depth. In particular, two schools of thought seem to have emerged, focusing on either an understanding of the underlying technology behind AI or the ability to use and apply existing AI systems. However, both aspects have been promoted under the term 'AI literacy' [15, 20].

Achieving AI literacy is not a trivial task and research on effective teaching methods is relatively new and sparse [15, 20]. In particular, teaching the technology underlying modern AI systems faces several challenges such as the 'black box character' of AI systems [1], the use of advanced mathematical concepts and the scale and magnitude of training required to obtain effective AI systems. In the future, explainable AI (XAI) [1, 5] might offer a promising avenue to address the black box character, but this line of research is still in its infancy. The mathematical requirements, on the other hand, might pose an even larger hurdle as teaching the necessary fundamentals first is often not feasible. Particularly in K-12, there is already an overloaded curriculum, in which the cost of investment for building up the required foundations is prohibitively high. Indeed, in order to be effective, we must find ways to teach the fundamental ideas and essence of AI quickly and on an 'intuitive' level. One promising avenue is demonstrated by the *CS Unplugged* initiative [3, 4, 13, 16, 22], which has opened up core concepts of computer science (CS) through hands-on and practical experience to a wide variety of ages in a minimum of required time and investment.

Following one of the ideas of CS Unplugged approaches, we designed an AI initiative where students take on the role of the AI itself. However, in contrast to CS Unplugged, our initiative uses a simple video game, picking up on the idea of AI agents learning to play *Super Mario Bros* [31], using reinforcement learning. The use of a video game allows us to provide a much finer-grained reward function than is possible with CS Unplugged approaches.

Instead of using Super Mario Bros directly, we created a modified platformer game (a. k. a. jump 'n' run game). The challenge was to create a game whose basic controls would be familiar and easy enough for the students, but whose scoring system might not be. Slightly obscuring the scoring system picks up an idea that Russell and Norvig presented in their well-known textbook: "Imagine *playing a game whose rules you don't know*; after a hundred or so moves, the referee tells you 'you lose,' That is reinforcement learning in a nutshell" [26] (our highlights).

*Goals and Contribution.* The objective of our initiative was to expose students to reinforcement learning by making them take on the role of an AI 'learning' to play a game. To this end, we developed a novel platformer/jump 'n' run game that would be played by two students simultaneously, allowed for strategic decisions and with a scoring system that students would have to figure out first. This paper presents the open-source jump 'n' run game and its design rationale.

While conducting the initiative we asked students about their disposition towards AI as well as their familiarity with some central concepts of AI. We found that students' attitudes towards AI are very diverse but could be slightly influenced taking part in the presented activity. Furthermore, we found an improvement in self-reported familiarity with the AI-related concepts covered by the initiative.

## 2 Related Work

### 2.1 Reinforcement Learning

*Reinforcement learning* (RL) is based on the idea of maximising a 'reward' for decisions or actions taken in a Markov process. The goal is that an agent infers (learns) a 'policy' from the rewards, i.e., a probability distribution that guides actions for each state. Reinforcement learning is particularly powerful if a clear reward function is available, from which the agent can learn, eliminating the need for fixed (training) datasets. It has therefore seen widespread application in playing games such as 'Chess' or 'Go' [27, 29], since such games are governed by a small set of rules, yet the number of configurations and possible actions exceeds any feasible size of a dataset. However, many of these games provide only 'sparse reward', i.e., there is no evaluation of individual steps, but only for the overall game play.

Advances in deep neural networks and image recognition allowed to apply reinforcement learning to video games [18, 28], which offer a much finer-grained reward function through its score system, say, than traditional board games. Indeed, Super Mario Bros has seen widespread adoption in the context of reinforcement learning [11, 12, 31, 34], up to AI benchmarks based on the game [10].

### 2.2 Teaching AI

Despite the long history of artificial intelligence, educational initiatives are rather recent with a steeply rising awareness of its importance [15, 20, 32]. In particular, the sudden rise of large language models (LLMs) has led to a huge number of publications on the topic [6, 23, 24]. A common theme is often the evaluation of LLMs' capabilities in answering student problems, say, or their use as tools to improve education itself. Both LLMs and the use of AI as a tool to improve teaching and learning itself, however, are beyond the scope of this paper.

In 2016, the term 'AI literacy' was first proposed, highlighting the increasing relevance of a basic understanding of AI systems [9, 15]. However, the exact extent of what 'AI literacy' entails is still under discussion [15]. The two main aspects revolve around either teaching the effective use of AI to solve particular problems, or teaching the underlying technology to foster a deeper understanding of what AI is. Within these, our work addresses the latter.

In a survey of AI teaching and learning, Ng et al. highlighted the importance and success to "reduce entry barriers through playful game experience for AI interventions" [20]. They also found evidence in the research literature for the positive effects of using game elements.

In terms of fostering understanding of AI, there have been a number of initiatives based on the CS Unplugged concept [3, 4]. Arguably one of the oldest and best known such initiatives is the 'Hexapawn computer' [17], which originated with Martin Gardner in the 1960s and has been used for studies of students' perceptions of AI [7, 19]. Other initiatives mostly targeted classification, using, e. g., decision trees [13, 14, 16, 21, 22]. While our own work is clearly not an unplugged activity, we nonetheless picked up the notion of the pupils taking on the role of the AI, which is very much in tradition with the unplugged initiatives mentioned above.

On the other hand, there are designated tools to teach about RL [8, 25, 33]. These tools focus on altering the learning process, mostly by changing the reward function to influence the learning outcome. In contrast to this approach, when using *Jumpy* the students face an environment with presets that cannot be changed. Instead, they have to comply with the given parameters – just like an AI-system would have to.

## 3 Designing Jumpy

### 3.1 Concept

*Jumpy* is a jump 'n' run/platformer game as seen in Figure 1. To make it easy to get started, it has a lot of familiar features like platforms, tokens to collect and a flag to mark the goal. But when playing, one will notice some differences in the game mechanics compared to commercial games. For example, the controls are different as the movement is grid based. Furthermore, there are two players who cannot interact with each other in-game but do not play against each other, either. These differences help reach the learning objectives and make sure that everyone is new to the game mechanics and therefore has potential to learn.

The main idea of Jumpy is to introduce the concept of reinforcement learning (RL), but not by using an AI, but instead by 'becoming' one – or at least by learning like one. Therefore, the game itself is divided into different stages, each focusing on a specific aspect of RL. The concepts covered follow the third idea, 'Learning' from the Five Big Ideas in AI [30], slightly adjusting it to fit the concept of RL.

### 3.2 Game Design

*Jumpy* is designed to be played by two players simultaneously on one computer, one player using the WASD keys, the other using the arrow keys of the keyboard. Each player controls one cup of hot chocolate that can walk to the left and to the right and jump upward. Each level consists of multiple platforms that are steady and impenetrable for players. Each level has one flag, which marks the goal. Once a player reaches the goal, the game is won and restarts after a few seconds. In addition, there are marshmallows spread across the level. The players can collect marshmallows by passing them. Once collected, they disappear.

Each stage comes with its own additional features, such as counters, history, or additional menu items as described in detail in subsection 3.3. The first three stages consist of only one level, whereas the later stages offer multiple levels.

The horizontal movement is grid based, meaning that the player can only move in predefined intervals. As the corresponding key can be pressed multiple times before the player reaches its destination, a little marshmallow at the bottom of the screen indicates the current destination column the player is moving to. While on the ground (either the floor or a platform), the player can jump two tiles high and if moving while jumping, up to three tiles wide. This means that the player can jump on platforms two tiles above the current height and easily jump over two tile wide gaps. With the right timing, also three tile wide gaps can be crossed. The grid based movement makes it easy to replicate previous moves, as the only skill based part of the movement is the timing of the jumps, which is rarely necessary. This makes the control of the game slightly different from classical jump 'n' run games while maintaining the main idea of the genre. This ensures that all students have comparable starting conditions. Even experienced players of jump 'n' run games need some time to get used to the game mechanics but inexperienced players can master the game quite easily as well. This ensures that experienced players of video games do not have an unfair advantage over inexperienced students.

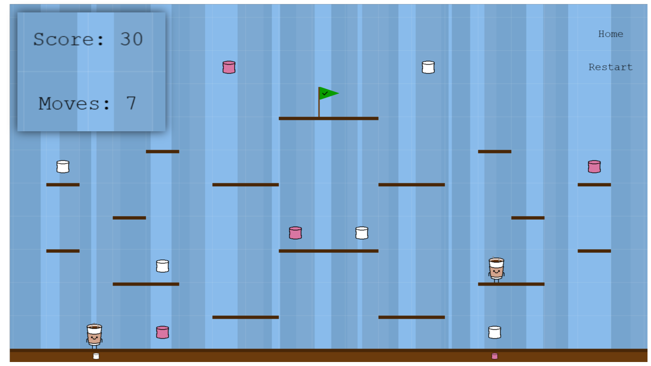


**Figure 2: The Exploration stage with individual counters for 'Score' and 'Moves'.**

### 3.3 Stage Design

For the greatest benefit, the tool should be used in combination with a classroom discussion following each completion of a stage. In these guided discussions, the teacher builds upon the students' experiences and connects these experiences to the learning process of an actual AI by highlighting similarities and differences in the learning process.

In the following section, every stage is described in detail highlighting the changes to the previous stages, explaining the students' tasks and describing their envisaged experiences. Furthermore, the students' experiences are connected to the actual process of reinforcement learning.

*3.3.1 Exploration.* In the first stage *Exploration* the students face their first level. It consists of multiple platforms with marshmallows all over the stage and the mandatory finish flag. They are presented with two counters ('Score' and 'Moves') as seen in Figure 2, and can restart the level or return to the 'Home'-Screen.

The students' task is to explore the game and figure out how everything works. They are expected to learn how to control their character and how to finish the level. Furthermore, they need to figure out how the counters work. Here, the 'Moves' counter increases by one for every move they make and the 'Score' counter increases by five or ten for every collected marshmallow, depending on the colour of the marshmallow and on the indicator below the collecting cup. Accordingly, the left player adds 10 points to the score when they collect a white marshmallow and 5 points when they collect a pink marshmallow, and the right player adds 10 points for every pink marshmallow and 5 points for every white marshmallow.

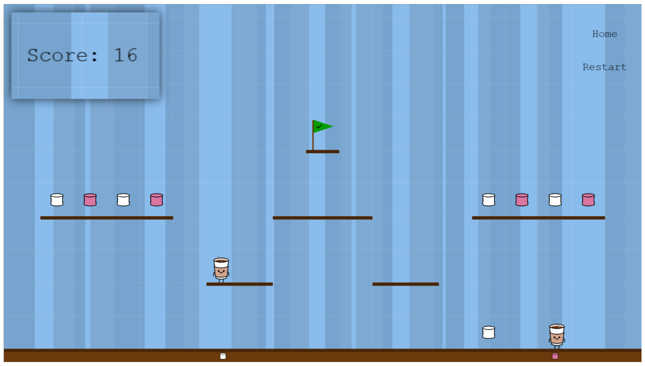


**Figure 3: The Reward stage with a single counter representing the reward function.**

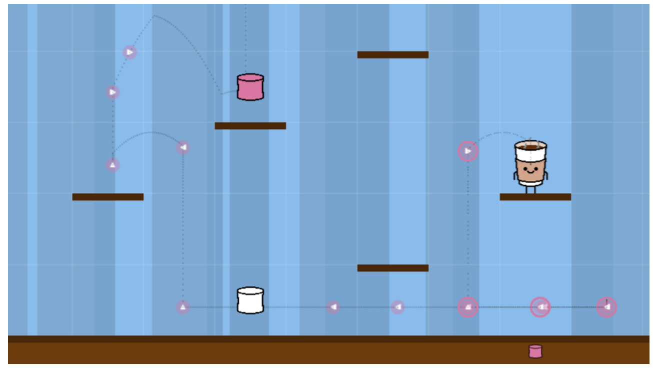

**Figure 4: The bottom right corner from the Strategy stage with the right players history from the previous run.**

While exploring the first stage the students will make random inputs before figuring out how the mechanics of the game work. They do not get any instructions, neither from the teacher nor from the game itself, apart from which keys to use. But still they will get better. Furthermore, they need to try different paths in order to learn how the counters work and what the flag will do.

These experiences connect to an AI's point of view, as an untrained AI has no model of the game. By making random choices, the untrained AI collects data from the game and thus creates a basis for learning. The same applies to the students, at least at the very beginning.

*3.3.2 Reward.* In the second stage *Reward* the two counters are replaced by a single 'Score' counter as seen in Figure 3. Now only the score is calculated while the 'Move' counter is omitted. The score increases as before when the marshmallows are collected and decreases by one for every move made. The platform design of this stage is simpler than the one before so that the runs can be completed faster. Therefore, the focus is shifted from playing and randomly trying to targeted testing.

The students' task is to figure out the aim of the game by deciding which rounds played were good. They are expected to figure out how the score is calculated. From that, they derive the aim of the game. This includes making as few moves as possible while reaching the flag and thereby collecting as many marshmallows as possible, preferably in their respective colours. Also, they don't play against each other, but instead share a score and end the game the moment the first player reaches the flag.

As before, the students are not given any explanation and have to derive every information from the game. By figuring out the score they are expected to deduce the aforementioned aim of the game.

The score with which the students work is a representation of a reward function. An AI learns what is considered 'good behaviour' solely through the reward function, so do the students. By changing the reward function, the learnt behaviour of an AI can be altered. In this case, the score, and thus also the reward function, is a combination from rewarding the collection of marshmallows and punishing additional moves. Furthermore, whoever defines the reward function decides what is considered good. The same applies to the students' experience. Even though they might like to play the game differently (e.g., reaching the goal without considering the marshmallows or playing competitively instead of cooperatively), this would not lead to a good score and would therefore leave their human AI system behind in terms of performance.

*3.3.3 Strategy.* In the third stage *Strategy*, history is added. Every time a movement is triggered, a symbol is added at the current position of the character indicating which movement was triggered. Furthermore, a line shows the path that a character has taken. An example can be seen in Figures 1 and 4. The history view can be toggled between the best and the last run and both corresponding scores are shown next to the current score.

So far, the students have learnt about the goal of the game, but not how to achieve it. From this stage onward, before starting a game, the students have to talk to their partner and agree on a strategy for playing in order to achieve the highest possible score. While playing, they must not talk to each other any more. After they finish, they can talk again and improve their strategy.

The students are expected to talk about their strategy for playing, thus making it explicit. In this stage the strategies are expected to be very concrete (e. g. "I walk here, then you jump there."). The history will help the teams to improve their strategy and will encourage them to try out new paths and correct mistakes in the next run.

An AI learns by trying, being rewarded and thereby improving its strategy (or policy) over multiple generations. This sequential learning process cannot be replicated in students, as they tend to adjust their strategy more holistically rather than gradually and purely based on rewards. But the history still encourages the students to iterate over their strategies, mutate a strategy and compare the new generation to the previous one, thus gradually improving their used strategy. The students will still use their logical thinking to accelerate the improvement of the strategy but this can be discussed in the next classroom discussion.

*3.3.4 Adaptation.* The fourth stage *Adaptation* adds six different levels to the stage as seen in Figure 5. Changing the level resets the score and the history.

The same rules as before apply. But this time the students have to decide on their strategy before entering a new level. After playing the level once, they can adapt their strategy.

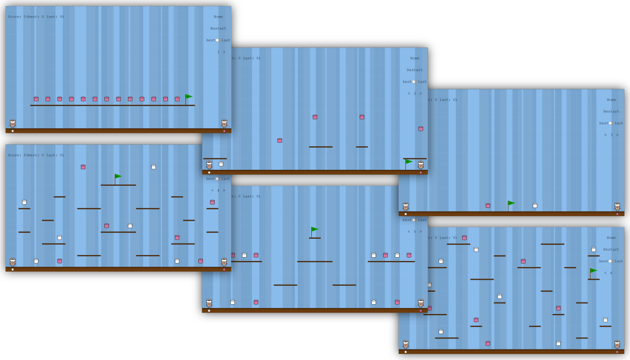

**Figure 5: Overview of the six different levels from the Adaptation stage. The same levels are used in the Testing stage, but in different order.**

As the levels change, the students have to adapt their strategy over and over again, coping with new levels, and therefore are encouraged to use more general strategies. The strategies are expected to be more general than before (e. g. "Collect a marshmallow of the other colour only, if you can reach it in less than five moves").

For real world AI systems, the learning environment needs to fit the operating environment, otherwise the learned strategy will not match the actual problem. This connects to the students' experience, as the strategies from the previous stage do not match the levels of this stage.

*3.3.5 Testing.* The final stage *Testing* is a collection of all previous levels in random order.

The students' task is to play through all the levels without talking to each other and putting their strategies to the test.

They are expected to either use their strategy or recap how they played these levels before. Either way they will use their learnt behaviour in a testing environment, simulating a real-world operating environment.

Speaking in terms of AI, the students now change from the learning phase to the application phase.

## 3.4 Didactic Design

The stages of *Jumpy* are based on the third of the *Five Big Ideas in AI* [30]: Learning. While this idea is formulated for machine learning in general, we transfer it to reinforcement learning in the context of using Jumpy. Touretzky et al. list four essential insights that students need to acquire to demystify the process of machine learning [30]:

(1) *Definition of machine learning*: "Machine learning allows a computer to acquire behaviors without people explicitly programming those behaviors." [30, p. 251]
The students experience that they learn a specific behaviour, in this case playing Jumpy successfully, without explicitly being told how to do so. This is the case for all stages starting from the second one where the students are introduced to the reward function.

(2) *How machine learning algorithms work*: "Learning new behaviors results from changes the learning algorithm makes to the internal representations of a reasoning model, such as a decision tree or a neural network. [...] What we want students to understand is that this kind of learning is a simple, mechanical process; there is no self-awareness or any kind of magic involved." [30, p. 251]
Even though we skip the step of introducing decision trees or neural networks, Jumpy is designed to demystify the process of machine learning. As the students enter the fourth stage *Adaptation*, they are presented with the history of their last and best run. Instead of remembering everything, they can experiment and iteratively improve their strategy – even though logical thinking helps to speed up this process.

(3) *The role of training data*: "When the reasoning model is capable of a great variety of behaviors, large amounts of training data are required to narrow down the learning algorithm's choices." [30, p. 252]
Jumpy is designed to model reinforcement learning and although reinforcement learning does not rely on training data in the traditional sense, the game still provides feedback signals that guide learning. This contrast can serve as a valuable discussion point in class. The first stage gives a hint in this direction by providing two different counters for moving and collecting marshmallows. This information is still present in the following stages but it is integrated into the score that represents the reward function. This way, the students experience that they are able to learn how to play the game just from the immediate feedback from the game and the reward function.

(4) *Learning phase vs. application phase*: "The reasoner constructed by the machine learning algorithm can be applied to new data to solve problems or make decisions." [30, p. 252]
The transition from the third stage *Strategy* to the fourth stage *Adaptation* encourages the students to apply their strategy to new levels, thus discovering the need to develop a more general strategy. The final stage *Testing* represents the application phase.

Our tool is explicitly designed to make this abstract idea of machine learning tangible for learners respecting the multiple facets of this idea. Each of the five stages builds on core aspects of the learning process in reinforcement learning, gradually building up a conceptual understanding of how agents learn from experience. In this way, the tool operationalises the abstract concept of learning by enabling students to experience the iterative nature of behaviour acquisition through feedback and adaptation – core principles of reinforcement learning.

# 4 Methodology

The evaluation of the tool was part of a bigger and more diverse research project at an Austrian secondary school. Throughout the day the students participated in three different 80 minute workshops on different topics of computer science. One of them was the Jumpy-workshop, the others were focused on using block based programming and educational robotics, and therefore were not related to the topics of AI and reinforcement learning. Ethical approval for the research project lead by TU Wien was granted and written consent from the students' parents was collected. Students

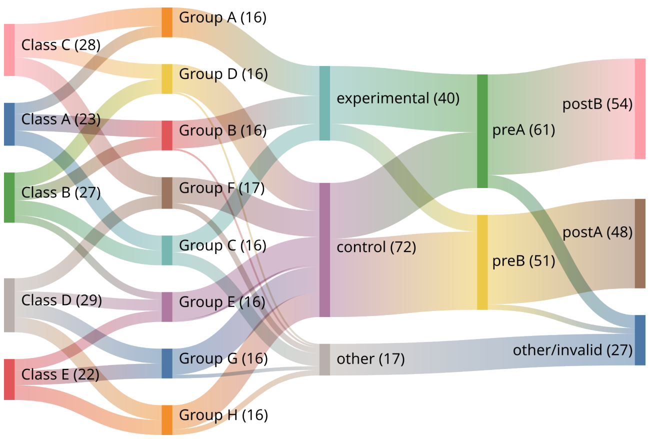


**Figure 6: Sampling – from classes to groups and testing**

who did not consent to taking part in the research still took part in the activity but are not represented in the numbers or the results.

The aim of the study was to answer the following research questions:

RQ1 How do students' attitudes towards AI change after undergoing the Jumpy-workshops?
RQ2 To what extent does the use of Jumpy improve students' self-reported familiarity with AI-related concepts?

Both research questions were addressed using a questionnaire with Likert scale items in a pre- and post-test scenario with control and experimental groups. The questionnaires were completed at the beginning and end of the session. Versions A and B of the same questionnaire with some reversely phrased questions and a different order of the questions were used to mitigate memory effect.

## 4.1 Sampling and Setting

We conducted this study with all students from year six in an Austrian school. In total there were 129 students in five classes aged 11–12. The students were divided into 8 smaller groups with about 16 students each and rotated through the different workshops. For this research project, only the first workshop of that day was evaluated. This means that three of those groups comprised the experimental group and took part in the presented activity while the other five groups comprised the control group and took part in other activities. Due to the rotation all students eventually underwent all workshops, although in different order, i. e. students in the control groups also attended the Jumpy-workshop, but not as part of our study. As some students did not consent to taking part in the research or were absent on the day of the study, 112 students took part in the pre-test and 110 students took part in the post-test. There were no dropouts during the activity as it was conducted in fixed groups in classrooms. The difference in the number of pre- and post-tests is due to students not submitting the questionnaire. The students chose a key in the pre-test and reused it in the post-test. Some students failed to provide the same unique key twice. Therefore, keys that could not be matched unmistakably were dropped. In the end 102 pre- and post-tests could be analysed with 64 tests in the control group and 38 tests in the experimental group. The sampling process is illustrated in Figure 6.

The students were all from the same school. This means that there might be some bias because of the location. But since there was no selection process, we believe they are still a good representation of students in their sixth school year.

The sampling process mentioned before ensures that the classes are distributed among the different groups while keeping some students of the classes together so that there are still peer groups within the groups. This approach helped maintain a natural classroom dynamic and supported collaborative learning, which was especially relevant for the intended use of the tool in pair-based activities and the classroom discussions. At the same time the distribution of the classes mitigated bias from prior experience within individual classes. Still there might be some bias from prior experience as the students from the different classes were not distributed equally over the experimental and control groups.

## 4.2 Questionnaire

The questionnaire was designed to assess students' attitudes towards AI as well as their self-reported understanding of AI-related concepts. It consisted of two main sections: (1) attitudinal statements rated on a 5-point Likert scale (from "strongly disagree" to "strongly agree"), and (2) familiarity ratings of technical terms on a 5-point Likert scale (from "never heard" to "know very well").

The questionnaire was administered as both a pre-test and a post-test to capture changes in students' perceptions and knowledge before and after using Jumpy. The items were developed in collaboration with CS education experts but have not been formally validated. The average completion time was approximately 5 minutes.

The questions are provided below as translation from the original German and used in this form in version A of the questionnaire. In version B the order of the items was changed and items 2, 3, 4 and 5 in Question I were formulated reversely and then recoded for evaluation.

I. Please indicate how much each of the following statements applies to you by selecting one of these options (Strongly agree / Agree / Neutral / Disagree / Strongly disagree)
(1) Computers and robots can learn to do anything.
(2) Machines learn the same way as humans do.
(3) There are things that computers should not learn.
(4) I know how an AI system learns.
(5) In an AI system the intelligence stems from the system itself, not from the programmer.
(6) If an AI can solve a task, it means it understands the task very well.

II. Are you familiar with these terms, and if so, do you know what they mean? (Never heard / Heard of it / Know a little / Know well / Know very well)
(1) Computer Science
(2) Data
(3) Algorithm
(4) Neural Network
(5) Agent (in computer science, not a spy)
(6) Artificial Intelligence

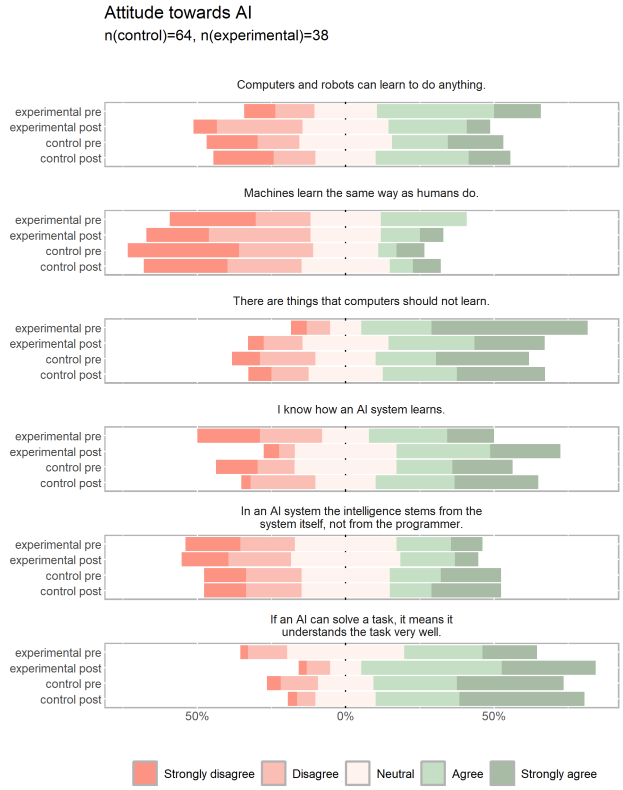


**Figure 7: Results from the questionnaire about students' attitude towards AI.**

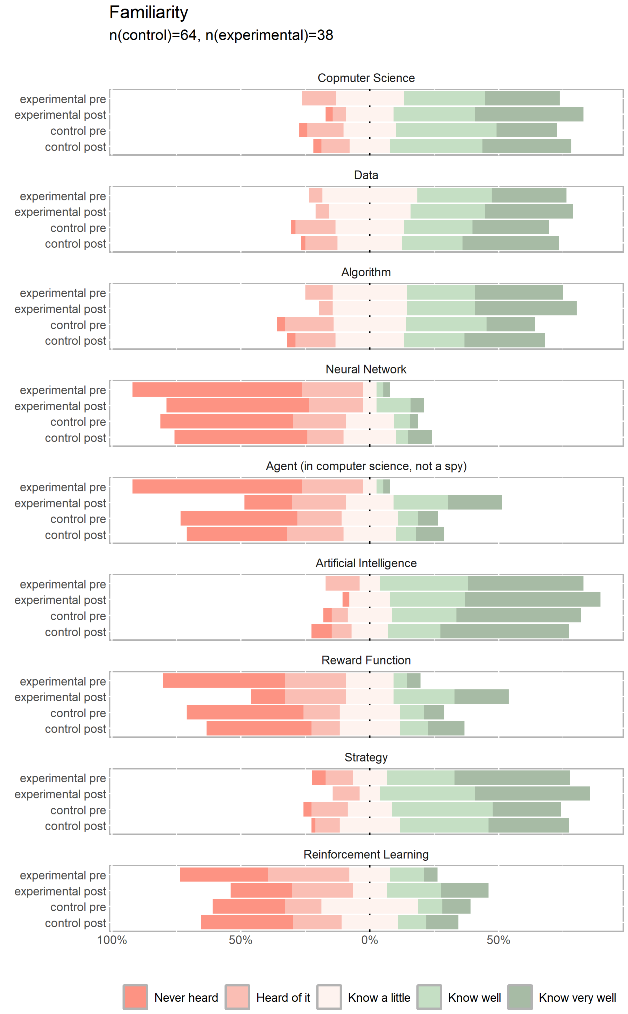


**Figure 8: Results from the questionnaire about students' familiarity of AI-related concepts.**

(7) Reward Function
(8) Strategy
(9) Reinforcement Learning

## 5 Results

As the research questions strongly correspond with the two parts of the questionnaire, the results presented in subsection 5.2 help address RQ1 and the results presented in subsection 5.3 help address RQ2.

### 5.1 Observations

Informal observations during sessions showed a high level of engagement. The students quickly immersed themselves in the game-like environment and demonstrated exploratory behaviour, especially in the early stages. The guided discussions showed that the students' experiences line up with the expectations and the transfer from the students' experience to the underlying reinforcement learning concepts occurred naturally.

### 5.2 Attitude Towards AI

The students' attitude towards AI showed an exceptionally high variance in the responses. In particular, the experimental and control groups clearly did not start from the same baseline as seen in Figure 7. Hence, a statistical analysis of pre- vs post-test between the two groups to compare shifts in attitudes makes no sense. However, there are still some anomalies worth mentioning:

(1) For items 1, 2 and 3 the experimental groups' attitude shifts towards disagreeing while the control groups' attitude shifts towards agreeing. This indicates a possible change in attitude caused by the intervention.
(2) For items 4 and 6 the change in the experimental group is remarkably greater than in the control group, which may suggest an influence from the intervention.
(3) Item 5 shows no notable change in the students' attitude towards AI at all.

### 5.3 Familiarity of AI-related Concepts

The analysis of the second part of the questionnaire focused on students' self-reported familiarity with AI-related concepts as seen in Figure 8. As the data is not as diverse as in the first part of the evaluation, statistical analysis was conducted. Linear mixed-effects models were used to assess the effects of time (pre vs. post), treatment (experimental vs. control group), and their interaction [2]. The results show that for most terms, there were no statistically significant changes in familiarity ratings. However, three terms showed significant interaction effects between time and treatment:

- Reward function ($p < .01$)
- Agent ($p < .01$)
- Reinforcement Learning ($p < 0.01$)

These results indicate that *Jumpy* was particularly effective in introducing and reinforcing key terminology directly related to the game mechanics and learning objectives.

# 6 Discussion

The results of the classroom implementation of *Jumpy* indicate that the tool is effective in fostering both engagement and conceptual understanding of reinforcement learning. In particular, the staged design appears to support a gradual and intuitive introduction to core RL concepts, aligning well with the third of the *Five Big Ideas in AI*: *Learning*.

## 6.1 Interpretation of Findings

The high level of engagement that we observed during the sessions confirms the motivational potential of game-based learning environments. Students actively explored the game mechanics and reflected on their strategies, which supports the idea that learning can emerge through interaction and feedback rather than direct instruction.

Although the students' attitudes towards AI were diverse and did not show consistent statistical trends, some shifts in the experimental group suggest that the intervention may have influenced their perception. The observed changes in items related to autonomy and control, namely items 1,2 and 3, could reflect a more nuanced understanding of how AI systems operate, potentially reducing misconceptions. The change in item 4 (*I know how an AI system learns*) indicates that the students perceived the initiative as helpful in understanding machine learning. On the other hand, the change in item 6 (*If an AI can solve a task, it means it understands the task very well.*) highlights the need to talk more about the differences between human behaviour and the inner workings of AI although the changes in item 2 (*Machines learn the same way as humans.*) show that students learned about the differences in learning between humans and machines.

The results of the second part of the questionnaire suggest that *Jumpy* can support students in developing a more concrete understanding of reinforcement learning concepts. While general terms such as *Computer Science* or *Algorithms* did not show significant changes, more specific terms that were directly embedded in the gameplay or addressed during classroom discussions – such as *Reward function*, *Strategy*, and *Agents* – showed significant gains in the experimental group.

This supports the idea that learning through doing, especially when combined with reflection and discussion, can foster conceptual understanding even in abstract domains like AI. The design of *Jumpy* aligns well with experiential learning principles, allowing students to engage with RL concepts in a hands-on and meaningful way.

Our findings support the idea that abstract AI concepts can be made accessible through carefully designed, interactive learning environments. By allowing students to take on the role of the agent, *Jumpy* aligns with constructivist and experiential learning theories. The tool also provides opportunities for reflection and discussion, which are essential for deeper conceptual understanding.

## 6.2 Limitations and Future Work

This study was exploratory in nature and limited in scope. The sample size was relatively small, and the intervention was short-term. Future studies on this topic should include larger and more diverse student populations, as well as longitudinal designs to assess long-term learning outcomes. Future work could also focus on evaluating the students' knowledge of the working of an AI as the presented study only assessed the self-reported familiarity with some AI-related concepts. Furthermore, the tool offers potential to be expanded and integrate other aspects of RL or to lead up to using a specific learning algorithm.

Despite these limitations, our study provides valuable insights into how young students engage with reinforcement learning concepts through playful interaction and how such tools can influence their understanding and attitudes towards AI.

# 7 Conclusion

This tool paper presented *Jumpy*, a game-based learning tool designed to introduce secondary school students to the core principles of reinforcement learning. Grounded in the third of the *Five Big Ideas in AI: Learning* the tool enables students to experience the learning process from the perspective of an AI, gradually building an understanding of key RL concepts such as reward functions, strategies, and policy adaptation.

The staged design of the tool supports a scaffolded learning experience, allowing students to explore, reflect, and refine their behaviour through interaction and feedback. Classroom observations and student feedback indicate high levels of engagement and motivation, while the evaluation results show significant gains in familiarity with central AI-related concepts.

*Jumpy* demonstrates that abstract AI concepts can be made accessible and meaningful through playful, interactive learning environments and reflective discussions. It offers a promising approach to fostering AI literacy at an early stage and contributes to the growing field of AI education in schools. The material is freely available at research.lehr-lern-labor.info/jumpy.

# References

[1] Plamen P Angelov, Eduardo A Soares, Richard Jiang, Nicholas I Arnold, and Peter M Atkinson. 2021. Explainable artificial intelligence: an analytical review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 11, 5 (2021), e1424.

[2] R. H. Baayen, D. J. Davidson, and D. M. Bates. 2008. Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language* 59, 4 (Nov. 2008), 390–412. doi:10.1016/j.jml.2007.12.005

[3] Tim Bell, Jason Alexander, Isaac Freeman, and Mick Grimley. 2009. Computer science unplugged: School students doing real computing without computers. *New Zealand Journal of applied computing and information technology* 13, 1 (2009), 20–29.

[4] Tim Bell, Paul Curzon, Quintin Cutts, Valentina Dagiene, and Bruria Haberman. 2011. Overcoming obstacles to CS education by using non-programming outreach

programmes. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*. Springer, 71–81.

[5] Roberto Confalonieri, Ludovik Coba, Benedikt Wagner, and Tarek R Besold. 2021. A historical perspective of explainable artificial intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 11, 1 (2021), e1391.

[6] James Finnie-Ansley, Paul Denny, Brett A Becker, Andrew Luxton-Reilly, and James Prather. 2022. The robots are coming: Exploring the implications of openai codex on introductory programming. In *Proceedings of the 24th Australasian computing education conference*. 10–19.

[7] Gregor Große-Bölting and Andreas Mühling. 2020. Students Perception of the Inner Workings of Learning Machines. In *2020 International Conference on Learning and Teaching in Computing and Engineering (LaTICE). IEEE Computer Society, Ho Chi Minh City*, Vol. 5.

[8] Cesar Hinojosa, Priyanka Kumar, Pragathi Durga Rajarajan, and Fred Martin. 2025. TrainYourSnakeAI: a novel tool to teach reinforcement learning to middle school students. In *Proceedings of the 56th ACM technical symposium on computer science education V. 1 (Sigcsets 2025)*. Association for Computing Machinery, Pittsburgh, PA, USA and New York, NY, USA, 506–512. doi:10.1145/3641554.3701907 Number of pages: 7.

[9] Martin Kandlhofer, Gerald Steinbauer, Julia P Laßnig, Wilfried Baumann, Sandra Plomer, Aron Ballagi, and Istvan Alfoldi. 2019. Enabling the creation of intelligent things: Bringing artificial intelligence and robotics to schools. In *2019 IEEE frontiers in education conference (FIE)*. IEEE, 1–5.

[10] Sergey Karakovskiy and Julian Togelius. 2012. The Mario AI Benchmark and Competitions. *IEEE Transactions on Computational Intelligence and AI in Games* 4, 1 (2012), 55–67. doi:10.1109/TCIAIG.2012.2188528

[11] Manav Khambhayata, Shivansh Singh, Neetu Bala, and Arya Bhattacharyya. 2024. Mastering Super Mario Bros.: Overcoming Implementation Challenges in Reinforcement Learning with Stable-Baselines3. In *2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET)*. IEEE, 1–7.

[12] Geoffrey Lee, Min Luo, Fabio Zambetta, and Xiaodong Li. 2014. Learning a Super Mario controller from examples of human play. In *2014 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 1–8.

[13] Lukas Lehner and Martina Landman. 2024. Unplugged Decision Tree Learning–A Learning Activity for Machine Learning Education in K-12. In *International Conference on Creative Mathematical Sciences Communication*. Springer, 50–65.

[14] Annabel Lindner, Stefan Seegerer, and Ralf Romeike. 2019. Unplugged Activities in the Context of AI. In *International conference on informatics in schools: Situation, evolution, and perspectives*. Springer, 123–135.

[15] Duri Long and Brian Magerko. 2020. What is AI literacy? Competencies and design considerations. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–16.

[16] Ruizhe Ma, Ismaila Temitayo Sanusi, Vaishali Mahipal, Joseph E. Gonzales, and Fred G. Martin. 2023. Developing Machine Learning Algorithm Literacy with Novel Plugged and Unplugged Approaches *(SIGCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 298–304. doi:10.1145/3545945.3569772

[17] Erik Marx, Thiemo Leonhardt, David Baberowski, and Nadine Bergner. 2022. Using matchboxes to teach the basics of machine learning: An analysis of (possible) misconceptions. In *The Second Teaching Machine Learning and Artificial Intelligence Workshop*. PMLR, 25–29.

[18] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[19] Andreas Mühling and Gregor Große-Bölting. 2023. Novices' conceptions of machine learning. *Computers and Education: Artificial Intelligence* 4 (2023), 100142.

[20] Davy Tsz Kit Ng, Min Lee, Roy Jun Yi Tan, Xiao Hu, J Stephen Downie, and Samuel Kai Wah Chu. 2023. A review of AI teaching and learning from 2000 to 2020. *Education and Information Technologies* 28, 7 (2023), 8445–8501.

[21] Elisaweta Ossovski and Michael Brinkmeier. 2019. Machine learning unplugged-development and evaluation of a workshop about machine learning. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*. Springer, 136–146.

[22] Susanne Podworny, Yannik Fleischer, Sven Hüsing, Rolf Biehler, Daniel Frischemeier, Lukas Höper, and Carsten Schulte. 2021. Using data cards for teaching data based decision trees in middle school. In *Proceedings of the 21st Koli Calling International Conference on Computing Education Research* (Joensuu, Finland) *(Koli Calling '21)*. Association for Computing Machinery, New York, NY, USA, Article 39, 3 pages. doi:10.1145/3488042.3489966

[23] James Prather, Paul Denny, Juho Leinonen, Brett A Becker, Ibrahim Albluwi, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, Andrew Luxton-Reilly, et al. 2023. The robots are here: Navigating the generative ai revolution in computing education. In *Proceedings of the 2023 working group reports on innovation and technology in computer science education*. 108–159.

[24] James Prather, Juho Leinonen, Natalie Kiesler, Jamie Gorson Benario, Sam Lau, Stephen MacNeil, Narges Norouzi, Simone Opel, Vee Pettit, Leo Porter, et al. 2025. Beyond the hype: A comprehensive review of current trends in generative AI research, teaching practices, and tools. *2024 Working Group Reports on Innovation and Technology in Computer Science Education* (2025), 300–338.

[25] Matthew Presson, Anisha Gupta, Jessica Vandenberg, Alex Goslen, Wookhee Min, Veronica Cateté, and Bradford W. Mott. 2025. Introducing reinforcement learning concepts to middle school students with game-based learning. *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2* null (2025), null. doi:10.1145/3641555.3705206

[26] Stuart Jonathan Russell and Peter Norvig. 1995. *Artificial intelligence: A modern approach;[the intelligent agent book]*. Prentice hall.

[27] A. L. Samuel. 1959. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development* 3, 3 (July 1959), 210–229. doi:10.1147/rd.33.0210

[28] Kun Shao, Zhentao Tang, Yuanheng Zhu, Nannan Li, and Dongbin Zhao. 2019. A survey of deep reinforcement learning in video games. *arXiv preprint arXiv:1912.10944* (2019).

[29] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the game of go without human knowledge. *nature* 550, 7676 (2017), 354–359.

[30] David Touretzky, Christina Gardner-McCune, and Deborah Seehorn. 2023. Machine Learning and the Five Big Ideas in AI. *International Journal of Artificial Intelligence in Education* 33, 2 (June 2023), 233–266. doi:10.1007/s40593-022-00314-1 Publisher: Springer Science and Business Media LLC.

[31] Jyh-Jong Tsay, Chao-Cheng Chen, and Jyh-Jung Hsu. 2011. Evolving intelligent mario controller by reinforcement learning. In *2011 International Conference on Technologies and Applications of Artificial Intelligence*. IEEE, 266–272.

[32] UNESCO. 2022. K-12 AI curricula: a mapping of government-endorsed AI curricula. (2022). doi:10.54675/ELYF6010

[33] Iro Voulgari, Marvin Zammit, Elias Stouraitis, Antonios Liapis, and Georgios Yannakakis. 2021. Learn to Machine Learn: Designing a Game Based Approach for Teaching Machine Learning to Primary and Secondary Education Students. In *Interaction Design and Children*. ACM, Athens Greece, 593–598. doi:10.1145/3459990.3465176

[34] Nika Zahedi, Nils Kuhn, and Evelyn Yee. 2025. MARIO: Reinforcement Learning on Image Observations. https://cs224r.stanford.edu/projects/pdfs/CS224R_Project(1).pdf