university of groningen / faculty of science and engineering / bernoulli institute

Designing Game-Based Machine Learning Education for Dutch High School Students

Bachelor's Project Thesis

Tex McGinley, s4299035, t.x.m.mcginley@student.rug.nl
Supervisors: J.D. Cárdenas Cartagena Artificial Intelligence, University of Groningen

Abstract: This paper explores the use of game-based learning (GBL) to teach machine learning (ML) concepts, specifically recommender systems, to Dutch high school students in the VWO educational stream. The study addresses the gap in AI education within the Dutch curriculum by developing an engaging and accessible educational program. This program includes a comprehensive teaching guide, presentation slides, and the Movie Ranker game, which simulates a Netflix-style recommendation system to illustrate ML principles. The program was created using the preparation, design, and improvement methodology adapted from Park and Kwon's South Korean paper, with the goal of demystifying AI, educating about recommender systems, introducing reinforcement learning (RL) concepts through the Multi-Armed Bandit (MAB) problem, and highlighting the ethical considerations and implications of these systems.

The created Movie Ranker game incorporates a competitive mechanic through the use of an ϵ-Greedy bandit, which the player must compete against. The results of the implemented ϵ-Greedy algorithm showed that it is effective in the created environment, performing better than a random agent in a statistically significant manner. This competitive element not only engages students but also helps them understand RL concepts like the exploration vs. exploitation dilemma.

While the program has yet to be tested and evaluated in a real-world classroom setting, preliminary findings suggest that the game's interactive and competitive nature effectively enhances student motivation and understanding of AI concepts. The program aims to prepare students for a future where AI plays an integral role in their personal and professional lives whilst encouraging them to pursue careers in technical fields.

1 Introduction

The artificial intelligence (AI) revolution is upon us (Moch, 2023), and with it comes the need to transform the current education system to reflect the changes this revolution will bring. Implementing AI education into the current curriculum will equip students with the skills they need to thrive in the rapidly changing technological landscape.

Incorporating AI and its tools in education enables students to receive personalized instructions matching their own learning style and pace, significantly enhancing their learning ability (Willige, 2024, Milberg, 2024). AI's capacity to allow for personalized education can make education far more accessible and inclusive for people with disabilities (Milberg, 2024). Furthermore, incorporating AI into current education practices will benefit not only the students but also the educators themselves, due to the fact that their administrative tasks can be made significantly easier and quicker through AI. This will allow them to focus on more crucial tasks, such as direct interaction with students and creating more stimulating academic material (Milberg, 2024, Chen, 2023).

Studying AI provides a comprehensive learning experience by bridging a wide range of disciplines, giving students an understanding of both technological and societal aspects. The field of AI builds upon multiple fields, including mathematics, computer science, ethics, and social sciences, utilizing algorithms, data structures, and statistical methods to create intelligent systems. In addition, the study of AI incorporates ethics and social sciences by making students think about the impacts of AI on society through teaching ethical AI development (Holmes, Hui, Miao, and Ronghuai, 2021). By integrating these diverse fields, those studying AI develop valuable technical, critical and ethical thinking skills, providing them with the rounded perspective needed to ethically develop AI systems.

This study sets out to explore an innovative way to approach AI education for Dutch high school students. To achieve this, the current state of AI education was investigated and an educational program was developed.

1.1 Motivations

1.1.1 Preparing the Younger Generation

AI's prevalence in modern society is growing rapidly. Whether it be in healthcare, economics, education, or even transportation, AI is having a significant influence on almost all fields (M. West and R. Allen, 2018). With new technologies such as generative AIs and large language models (LLMs) improving exponentially, the impact of AI on society will only continue to grow. For instance, technologies such as OpenAI's ChatGPT are already revolutionizing how people find information and approach work. It is clear that AI is here to stay and will only become more prevalent in daily life.

Additionally, with an increasing portion of people relying on digital sources for their information, the influence of the recommender systems behind these platforms, which decide what content you get shown, continues to grow (Liedke and Wang, 2023). These systems can have a significant negative impact on users, which can range from misinformation to negative mental health effects like depression and anxiety (Bojic, 2024). Younger users in particular are susceptible to these influences (Bojic, 2024). Despite the potential problems and prevalence of these systems, most people do not understand how they work or the issues that come with them.

As AI's hold on our society expands, so does the need for relevant education about the field. This education is vital to prepare the younger generation for a future where AI will be an integral part of their personal and professional lives. Understanding these AI systems empowers people to make informed decisions about how they want to incorporate them into their daily lives.

1.1.2 Addressing Dutch Educational Gaps

The education system in the Netherlands differs from that of most other countries. In figure 1.1, you can see the various educational streams in the Dutch system. Upon completion of elementary school, students undergo an assessment and are placed into one of three groups based on their academic performance. The group they are placed in determines the level of education they will receive and the types of colleges and universities they will be eligible to attend (Wikipedia, 2019). This study will focus on students in VWO (Preparatory Scientific Education), which represents the highest academic and research-based high school education level. Specifically, we will focus on VWO students who are around 16 years old who have yet to choose their course specialization.

A VWO student wanting to learn AI-related subjects only has a few options. In the upper years of HAVO and VWO, they must specialize in one of four subject combinations. HAVO (Higher

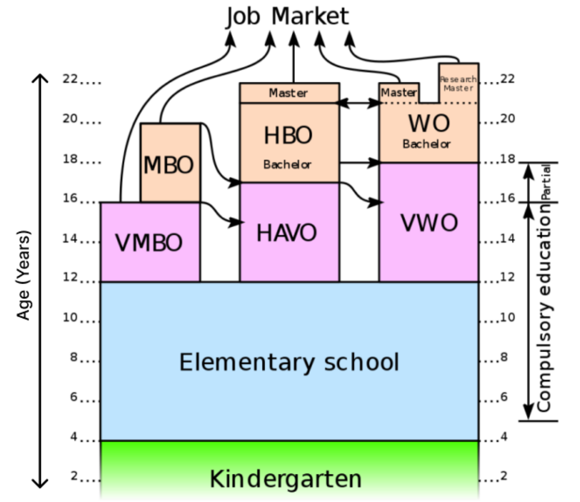

Figure 1.1: The Dutch Education Adapted from "Education in the Netherlands" (Wikipedia, 2019)

general continued education) is the second highest high school education level available to Dutch students (Ministerie van Onderwijs, 2014). Science and Technology is the combination that offers the most digital-related classes (Ministerie van Onderwijs, 2014). But even with this specialization, computer science is an optional choice. If a student decides to pursue computer science, the quality of the educational material they receive will depend on the school and its available resources. This is also just for computer science. There are no current options for students who want to learn solely about the technical side of AI. Therefore, making projects like this especially relevant in our current society.

There have been some pushes to add more digital education opportunities in Dutch education with the addition of programs like Technasium, where students have the opportunity to gain real-world skills and experiences by working on practical assignments commissioned by real companies (Wikipedia, 2020). However, this is an optional path to follow, which is only offered to HAVO and VWO students, and even if a student chooses to follow the program, they will have to find and choose a company that offers assignments in the field of AI or computer science to gain any AI-related education. This shows that digital-based skills, specifically AI-related ones, are not currently prioritized in Dutch high school education.

The *2024 Report of the State of Education* conducted by the Ministerie van Onderwijs (the Ministry of Education) found concerning trends, including teacher shortages, declining academic performances, and disparities in the quality of education. They reported that, "Het lijkt erop dat veel leerlingen niet het onderwijs krijgen dat ze nodig hebben" ("It seems that many students are not receiving the education they need") (Ministerie van Onderwijs, 2024).

These problems further highlight the need for AI education in the Dutch education system. Its ability to allow for personalized learning and streamline administrative tasks for educators has the potential to significantly aid in addressing the issues currently faced in the Dutch education system (Milberg, 2024).

1.1.3 Exploring Innovating Teaching Approaches

AI pedagogy education is a relatively new concept. Therefore, educators are still trying to figure out the best method. Given AI's multidisciplinary nature, there is a wide variety of different approaches to teaching it. Thus, there is no such thing as 'the best approach' as it is contextually driven depending on the teacher's aim and their target audience. For instance, for an educator trying to teach 17-year-old students the coding skills needed to create a simple hand-written digit recognition system, combining traditional teaching methods with project-based learning would probably be ideal. The first step would be implementing traditional teaching methods to explain the necessary concepts. Then, students would engage in project-based learning, where they must apply the concepts they have learned to create the actual system. Meanwhile, an educator trying to teach 10-year-old students about the basics of AI might opt for more interactive and visual teaching methods. Educational games or simple robotics projects could be particularly effective, engaging younger learners' curiosity and making abstract concepts more tangible.

Through this project, we hope to explore how game-based learning can be employed to educate Dutch high school students about machine learning (ML) concepts and recommender systems.

1.1.4 Dutch Economy

With the AI market expected to grow exponentially in the coming years (Research, 2023), countries must prioritize AI education in schools to avoid falling behind in the AI tech race. This is especially important for the Netherlands, as while the Dutch technological sector is by no means performing poorly, it does appear to be slowing down compared to countries such as France and England (van Oranje, Hooijman, Schutten, Parisi, van Rossum, and Windsor, 2024).

By focusing on VWO students who have yet to choose their course specialization, we want to encourage students to follow more technical paths. This will hopefully increase the number of digitally skilled workers in the Netherlands, giving the Netherlands the best shot at benefiting from the AI tech race.

1.2 State of the Art

Research into AI education is increasing, with more studies coming out evaluating different programs and teaching methods. A key study that was used as a foundation for this paper is Tedre, Toivonen, Kahila, Vartiainen, Valtonen, Jormanainen, and Pears's article evaluating the emerging trends, challenges, and necessary paradigm shifts for effectively integrating machine learning into K–12 education (Tedre et al., 2021). Their study looked at a wide range of existing programs, from 3-year-olds learning to identify and describe their own emotions through teaching a computer to recognize them to 9th-grade students training ML models to recognize different types of instruments (Tedre et al., 2021). Their research showed that an essential step to developing "next-generation computational thinking" is to move away from the belief that traditional rule-based programming is a key aspect of educating computer science and ML concepts (Tedre et al., 2021). It also highlighted the importance of creating material that is "low-floor and high-ceiling", meaning that the material is easily approachable (low-floor) for students but also allows for opportunities for advanced exploration and deeper understanding (high-ceiling)(Tedre et al., 2021).

Another key foundation study used for this paper is Park and Kwon's article "*Implementing Artificial Intelligence Education for Middle School Technology Education in the Republic of Korea*" (Park and Kwon, 2023). Their study developed and evaluated an AI educational program for Korean middle school students in their free semester. They developed their education program with the goal of analyzing the effect of AI activities on students' attitudes towards technological and AI competencies and to help with career exploration (Park and Kwon, 2023). Their results showed that their program was effective in increasing "interest in technology," "career aspirations in technology," and AI competency in the tested students (Park and Kwon, 2023). Additionally, their paper highlighted the need to create programs that focus on fostering learning from working on real-world context-based tasks (Park and Kwon, 2023).

Furthermore, Swart's study "*Experiencing Algorithms: How Young People Understand, Feel About, and Engage With Algorithmic News Selection on Social Media*" (Swart, 2021) offers valuable insights into how young people interact with and understand algorithms, specifically in the context of recommender systems. Swart's research revealed that young people's understanding of algorithms is often shaped by their everyday interactions and the transparency of the algorithms they encounter (Swart, 2021). This study emphasizes the importance of algorithmic literacy, which is a critical aspect when teaching about recommender systems in AI education.

Additionally, Zhan, Tong, Lan, and Zhong's systematic literature review of game-based learning (GBL) in AI education found that GBL is an effective approach to increase student engagement, satisfaction, motivation, and collaboration (Zhan et al., 2022). Their analysis categorized games into puzzle games, reasoning strategy games, robot games, role-playing games, and simulation games, with puzzle and reasoning strategy games being the most frequently used (Zhan et al., 2022). The review highlighted GBL's role as a teaching tool, enhancing the explanation of AI concepts through interactive methods (Zhan et al., 2022).

Zhan et al. emphasized the importance of designing educational yet engaging games to maintain student interest. They also noted the benefits of using GBL as a competitive mechanism, further enhancing involvement and understanding (Zhan et al., 2022). The review supports integrating GBL in AI education to improve learning outcomes and foster a deeper understanding of AI through practical application and interactive experiences (Zhan et al., 2022).

1.3 Contributions

The contributions of this study are as follows: the design and development of an educational game-based program for Dutch high schools, including a comprehensive teaching guide, class presentation slides, and the educational game itself. Through the creation of this program, we hope to increase AI literacy, in addition to increasing awareness and understanding of recommender systems and their implications.

2 Theoretical Framework

2.1 Game-based Learning

Game-based learning (GBL) is an educational method that utilizes game principles and mechanics to improve learning experiences (Qian and Clark, 2016). It involves incorporating game elements, such as point scoring, competition, and rules of play, into educational activities to engage students, motivate them, and facilitate deeper learning of the subject matter.

GBL's effectiveness stems from its ability to combine educational content with engaging gameplay. Doing so makes learning not only engaging and fun but also meaningful. It has been shown to increase student engagement and motivation, resulting in higher retention rates and understanding of the material (Hamari, Shernoff, Rowe, Coller, AsbellClarke, and Edwards, 2016).

However, implementing GBL can present some challenges. First, it requires adequate technological infrastructure, which can be a problem in lower-income areas where there isn't necessarily access to digital devices. Second, GBL can have a higher potential for distractions, as students use devices such as computers or even their own smartphones to play the game. Even with these issues, when implemented effectively, GBL has the potential to transform traditional educational paradigms, making learning more engaging, effective, and enjoyable for students.

2.2 Recommender Systems

From NVIDIA, "a recommender system is a class of machine learning that uses data to help predict, narrow down, and find what people are looking for among an exponentially growing number of options" (NVIDIA, 2024). These systems are ubiquitous nowadays. They determine nearly everything that you are shown online, from the posts you see on social media platforms such as Instagram to the movies recommended to you on streaming platforms such as Netflix and even the ads and products you see when online shopping. Recommender systems play a role in all of these.

2.2.1 Types of Recommender Systems

There are multiple different algorithms and techniques that recommender systems can utilize. However, most of them fall into one of three categories (NVIDIA, 2024):

- **Collaborative filtering** is based on "the idea that if some people have made similar decisions and purchases in the past, like a movie choice, then there is a high probability they will agree on additional future selections." (NVIDIA, 2024). For example, if the system knows you and another user like the same music, it may recommend a song to you that it knows the other user already likes.

- **Content filtering** uses the attributes of items to make recommendations. It analyzes the features of items previously liked by the user to suggest similar items. For instance, if a user enjoys horror movies, the system will recommend other horror movies (NVIDIA, 2024).

- **Context filtering** "uses a sequence of contextual user actions, plus the current context, to predict the probability of the next action" (NVIDIA, 2024). The types of information context filtering incorporates when making recommendations could include the user's country, the device they are using, the date and time they interacted with an item, and more (NVIDIA, 2024).

Most modern-day recommender systems use hybrid systems, which combine elements of the above-mentioned methods to create a more thorough recommendation system (NVIDIA, 2024).

2.2.2 Issues with Recommender Systems

Recommender systems can be helpful in assisting users to find what they want and need in a constantly expanding realm of choices; however, they can also generate issues if not implemented carefully. Because these algorithms learn from the provided training data, they are vulnerable to biases (Bojic, 2024). These biases can have tangible repercussions. For instance, a study revealed that Google's advertising system displayed high-paying job ads more frequently to men than to women (Datta and Carl Tschantz, 2015). Such biases reinforce stereotypes and perpetuate inequality.

Furthermore, recommender systems of digital platforms like streaming services and social media networks are designed to get users to spend the most amount of time on these platforms. While this is commercially advantageous for these platforms, it can lead to addictive behaviours in users that can have a negative impact on mental health, resulting in anxiety, stress, and even depression (Bojic, 2024, Park, Jeong, and Rho, 2021).

These systems have the ability to greatly influence users' opinions and beliefs by selectively presenting information. As a result, they can shape how users perceive the world. For example, in social media, users may be exposed only to content that aligns with their existing views, leading to a positive response. This risks polarizing users by promoting and reinforcing biased and misinformed perspectives (Bojic, 2024).

Addressing these issues is a challenging task that requires ongoing efforts and moderation. Recognizing and mitigating biases is essential to ensure users are exposed to a diverse range of perspectives. It is crucial for companies to maintain transparency about how their systems operate and the types of data they collect. These companies should also provide users with the necessary tools to choose what data is collected and how it is used. This can be done through privacy settings that allow the user to opt out of data collection and delete their data.

2.3 Bandits

The Multi-Armed Bandit (MAB) problem (also known as the k-armed bandit problem) is a classical reinforcement learning (RL) challenge (Sutton and Barto, 2018). It consists of T trials, in which an agent can choose from a set of K-number of arms (actions) a to pull. Pulling an arm returns an unknown rewards R_a.

The goal of an agent in the MAB is to maximize long-term rewards. To achieve this goal, the agent must balance two opposing strategies: exploration and exploitation. Exploration refers to the action of choosing a random arm to pull to learn more about its environment and how good or bad the chosen arm was. On the other hand, exploitation is the action of selecting the arm that, based on current knowledge, should return the highest reward. The exploitative action is also referred to as the greedy action, as in choosing this action, the agent prioritizes current rewards over possibly finding actions that may result in higher future rewards.

In most scenarios, the best strategy for maximizing long-term rewards involves exploring heavily at the beginning to gain more knowledge of the environment. As the knowledge of the environment grows, it becomes increasingly favourable to exploit this knowledge to get higher rewards.

There are a few different variations of the MAB problem. One of the more common ones is the stochastic MAB problem (Slivkins, 2020). This version assigns a reward distribution D_a to each arm. Thus, whenever an arm a_t is pulled, the reward R_a $(a = a_t)$ is independently sampled from this distribution. Since the reward of each action is sampled from a set distribution, an agent in this MAB environment will most likely have to try an action multiple times to learn the underlying reward distribution.

Another variation of the MAB is the deterministic MAB problem. In this version, each arm a_t has a set reward R_a $(a = a_t)$. Therefore, an agent in this environment only has to choose an arm once to know how good or bad an action it is. If the number of trials T is not a constraint, then the optimal strategy for an agent in this environment would be to try all arms a once to find the one that returns the highest reward R_a. Once this action a_t is found, the agent can simply only exploit this action to get the highest rewards.

2.3.1 ϵ-Greedy Algorithm

Multiple different algorithms have been developed to solve the MAB (Sutton and Barto, 2018). This project focuses on the ϵ-Greedy algorithm. This algorithm addresses the exploration vs exploitation dilemma by setting a probability ϵ, to explore.

Algorithm 2.1 A simple ϵ-Greedy algorithm pseudo-code (adapted from Sutton and Barto, 2018)

Initialize, for $a = 1$ **to** k**:**
$Q(a) \leftarrow 0$
$N(a) \leftarrow 0$
for $t = 1$ to T **do**

$$A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1-\epsilon \\ \text{a random action} & \text{with probability } \epsilon \end{cases}$$

(breaking ties randomly)
$R \leftarrow \text{bandit}(A)$
$N(A) \leftarrow N(A) + 1$
$Q(A) \leftarrow Q(A) + \frac{1}{N(A)}[R - Q(A)]$
end for

Algorithm 2.1 above shows the pseudo-code for a simple ϵ-Greedy algorithm. The first stage of the algorithm involves initializing the expected rewards $Q(a)$, and the action counts $N(a)$ for each arm a to 0. The algorithm then loops from $t = 1$ to $t = T$. In each iteration of the loop, depending on the value of ϵ, the algorithm chooses a random or greedy action. After performing the action a, the environment returns a reward of $R(a)$. The algorithm then updates the action count $N(a)$ and the estimated reward $Q(a)$ for said action. There are a wide range of different methods to keep track of estimated rewards. In the case of the above algorithm 2.1, it uses what is essentially a running average with a learning rate $\frac{1}{N(A)}$. This learning rate decreases as an action is chosen more often, ensuring that the estimates stabilize over time. By reducing the impact of each new reward on the running average, the algorithm effectively balances between integrating new information and maintaining reliable estimates from accumulated data.

More complicated versions of the ϵ-Greedy algorithm employ a ϵ-decay value to achieve the desired behaviour of high exploration at the beginning and increasingly exploitative as knowledge of the environment grows. The value reduces ϵ by a specified factor after a defined amount of actions or time.

3 Methodology

The preparation, development, and improvement methodology used in Park and Kwon's South Korean paper was chosen for the development process of the educational material. Figure 3.1 shows the steps of this methodology.

In the **Preparation phase**, we set the program's goals and topics to be as follows:

- **Demystifying AI**: Use clear, relatable examples and avoid technical jargon to break down some of the complexities of AI to make it approachable for students.

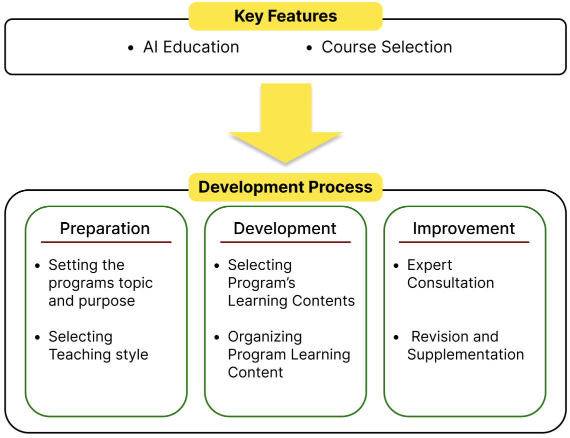

Figure 3.1: Development Methodology. Adapted from Park and Kwon (2023)

- **Educating about recommender systems**: Introduce some of the fundamental concepts behind recommender systems.
- **Introduce RL concepts through the MAB**: RL is a key part of modern-day AI systems. The MAB problem is well-suited to introduce learners to AI concepts because it provides many easily digestible examples and analogies.
- **Highlight the ethical considerations and implications of recommender systems**: Get students to start thinking about how these systems impact what they see and the implications that they can have.
- **Create engaging material**: Ensure students take away something from the lesson through designing the content to be engaging and relevant to them.
- **Encourage students' curiosity into technical career paths**: Provide students with opportunities to explore their areas of curiosity.

GBL was chosen for the teaching style due to its effectiveness in increasing students' motivation and engagement, allowing for enhanced learning experiences Zhan et al. (2022).

In the **Development phase** of the methodology, the program learning contents were selected and organized. The teaching guide, educational game, and presentation slides were all developed in this phase. The process of creating these materials will be discussed in more detail later on. Figure 3.2 shows the developed program's organization and learning contents.

Due to time constraints, the **Improvement phase** has yet to be conducted. This would involve consulting with AI and educational experts and revising the program based on their feedback.

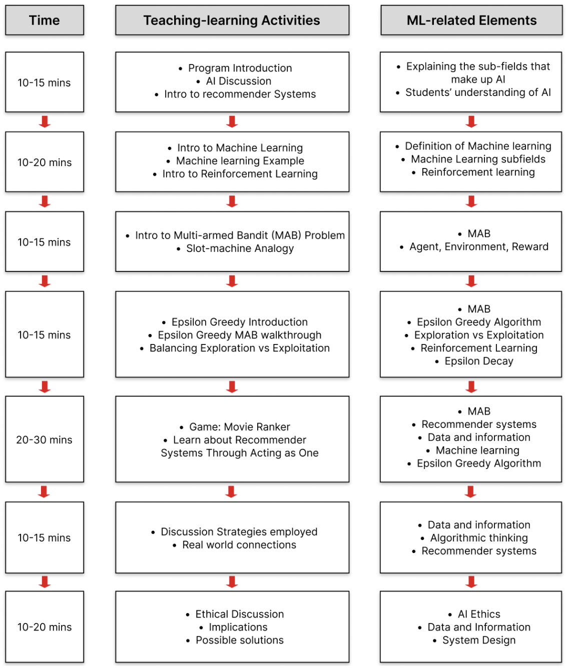

Figure 3.2: The Developed Program. Adapted from Park and Kwon, 2023

3.1 Teaching Material

The design decisions for the teaching guide were primarily influenced by the finding that educators identified "the biggest challenge in educating students on AI was the teachers' lack of confidence in content mastery" (Park, Teo, Teo, Chang, Huang, and Koo, 2023). The teaching guide was therefore developed to provide educators from any background with the knowledge and confidence needed to teach the class effectively. To achieve this, it focuses on using easily digestible examples, simplifying complicated terms, and, where possible, including real-world examples.

To meet the program's goals of creating engaging material that also encourages students' curiosity into technical career paths, the decision was made to emphasize in-class discussions. Where they can lead the discussion to their interests. The goal of doing this is to allow them to explore their areas of curiosity, hopefully leading to better engagement.

To ensure that educators feel confident in leading these discussions, the choice was made to include a section breaking down some of the ethical implications of recommender systems. Again, to make this part approachable for high school teachers regardless of background, it simplifies complicated terms and uses real-world examples where possible.

With the current teacher shortage in the Netherlands (Ministerie van Onderwijs, 2024), removing the requirement of prior knowledge increases the number of educators who can teach the lesson. Furthermore, reducing the workload put on teaching staff by providing all necessary educational material should enable the lesson to have the greatest impact possible using the limited resources available.

3.2 The Game

As previously mentioned, recommender systems are so prevalent nowadays that nearly everything one sees online has been curated and personalized by a recommendation system. People are so used to recommender systems controlling everything they see online that they don't even stop to consider how they work or what the possible problems are. This is especially true of younger generations, who are increasingly growing up online (Anderson, Faverio, and Gottfried, 2023). Younger people are also more vulnerable to being influenced by what they see as they haven't had the time to develop their own opinions through lived experiences (Bojic, 2024). Because of this, it is crucial to get them thinking about how the content they see affects them and how this content is chosen. With that in mind, it was chosen to make a game where students would learn about recommender systems by having to act as one.

3.2.1 Decision to Make a Movie Recommender Game

In the early stages of this project, there was deliberation on which type of recommender system would be best to demonstrate. In order to capture the students' interest, we examined recommender systems with which they are likely familiar, such as those behind social media platforms like Instagram and TikTok. While these are effective at showcasing some of the ethical implications of recommender systems, they are very complicated and would require more time than available to explain the underlying concepts. In addition, the AI and ML techniques that these systems employ are beyond the scope of a 16-year-old student's skill set. After further deliberation, it was decided to create a simplified Netflix-style movie recommendation system. This approach was chosen for the following reasons:

- Movie recommendation environment engages students as they all are familiar with Netflix or a different movie streaming service.

- Most likely, students have already had to act as a movie recommender system at some point. For instance, if they have ever been asked for a movie recommendation from a friend.

- The movie recommendation environment can be simplified to an appropriate level, where basic RL concepts can be demonstrated and taught.

In the game, the player is tasked with finding the ten best movies to recommend to a given user profile. They need to use the information provided to develop a strategy that yields the highest-scoring list of recommendations. In doing so, they will have to apply much of the same thinking as someone who is actually developing a recommender system. Such as what features are important to focus on. Figure 3.3 shows a screenshot of the main game window. The game's design takes a lot of inspiration from Netflix's design to further emphasize real-world connections.

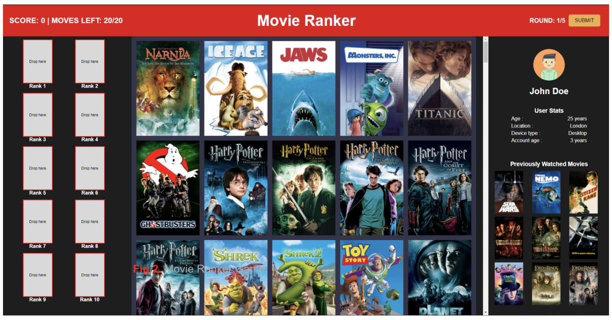

Figure 3.3: Movie Ranker Game

3.2.2 Connection to the MAB

The game, which is designed to educate students about recommender systems, can be represented as a MAB problem. By drawing parallels between the game mechanics and the components of the MAB framework, students can better understand some of the core concepts of RL and decision-making under uncertainty. Here are the key elements of the game mapped to the MAB problem:

- **Environment**: In this context, the environment is the game itself. It consists of a collection of movies, a user profile, and a list of the user's previously watched movies. Each with its own specific attributes.

- **Agent**: The agent in the game is the entity making the actions. Whether it be an actual student or an algorithm. The agent's goal in the environment is to maximize total rewards, which represents how good a recommendation is.

- **Action**: An action in the game is recommending a movie. Each movie not currently recommended is considered an arm in the game. Pulling an arm means recommending a movie.

- **Reward**: The reward in the game is the score that the recommendation receives. It quantifies how good a given recommendation is.

3.2.3 Reward Function

The reward function in the game calculates the score for each movie recommendation based on the user's preferences. Each user profile has an assigned genre preference list. This list contains a score from 1 to 10 for all the different movie genres, indicating their level of interest, with higher scores reflecting stronger preferences. To determine a movie's score, the game matches the movie's genres with the user's preferences. For instance, if a user has a preference score of 9 for action movies, and the recommended film is an action movie, then this will positively affect the recommendation score. The final score for a film is calculated by averaging the user's preference scores for all the genres the movie belongs to. For example, suppose John Doe's preferences for action, adventure, and science fiction are 7, 4, and 8, respectively, and the recommended movie "Spider-Man" falls into these genres. In that case, the score is computed as $(7 + 4 + 8)/3$, resulting in a final score of 6.333. The total recommendation score is all the recommended movies' scores added together. It reflects how well the created recommendation list matches the user's tastes.

3.2.4 Selection of Movies

One method employed to meet the program's goal of creating engaging GBL material was selecting movies that match student interests. In order to ensure age appropriateness, only movies with a rating of PG-13 or below were chosen. The TMDB API was used to gather all the movies and their relevant details. Movies were pulled from a comprehensive list of popular movies to ensure that the selections were appropriate and likely to be familiar and appealing to the students.

3.2.5 Incorporating ϵ-Greedy Agent

In the game, the students compete against an ϵ-Greedy agent. An ϵ-Greedy agent was chosen for its ability to allow for easily absorbable explanations of reinforcement learning and MAB concepts such as the exploration vs exploitation dilemma.

Initially, the game was going to be set up so that you would only see the score of your recommendation after submitting it. However, this would not allow for immediate feedback on how good an individual recommendation is. Therefore, it meant that it would be a poor environment to showcase the MAB and the ϵ-Greedy algorithm. Thus, to fix this, a real-time scoring mechanic was added to the game where a player (real or simulated) could see the score of their recommendation at all times. In addition, incorporating the ϵ-Greedy agent helps meet the program's goal of introducing reinforcement learning concepts through the MAB.

3.2.6 Competing Against the ϵ-Greedy Agent

Another method employed to create engaging GBL material was the decision to have the player compete against the ϵ-Greedy agent. To add this competitive mechanic, the decision was made to split the game up into 5 rounds. In each round, the player and the agent could make 20 moves. An action is only considered a move if a movie that is not already recommended gets added to the recommendation list. Furthermore, restricting the number of movies a player can recommend in a given round forces them to be more strategic in their actions.

The ϵ-Greedy agent is designed to improve each round, adding the extra challenge to the player to see how many rounds they are able to beat it for. Moreover, adding this competitive element will likely increase student engagement by creating a video-game-like atmosphere. This improvement is achieved by decaying the ϵ-value to a predefined value each round. These ϵ-values are as follows: [1, 0.9, 0.75, 0.5, 0.001].

In the first round, ϵ is set to 1, meaning it will only explore. By the last round, ϵ is set to 0.001, which should result in the agent only exploiting. Thus, setting up the ϵ-Greedy agent in this manner gives us the desired behaviour of heavily favouring exploration at the start to gain more knowledge of how good each action (recommendation) is. As the agent's knowledge grows, the agent increasingly exploits the learned information to achieve the goal of maximizing rewards. Furthermore, in order to keep track of how good each action is, the agent makes use of a running average.

3.2.7 Stand-alone Web Application

To minimize students' distraction, the decision was made to go with a stand-alone web application. This means that it operates independently of other applications and browser tabs, ensuring that students remain focused on the task at hand without the temptation to switch to unrelated websites or apps.

4 Results

4.1 ϵ-Greedy Algorithm

Figure 4.1 shows the performance comparison of the ϵ-greedy algorithm vs a random agent averaged over 100 runs. That being shown, a key thing to note is the sharp increase of the ϵ-greedy agent and the random agent from 0-10 moves. This happens because both agents start with an empty recommendation list, and they both randomly add the first ten movies. Since the ϵ-greedy agent starts with an ϵ-value of 1 for the first round (20 moves), it means that it is practically acting as a random agent for this duration. However, after ten moves, the ϵ-greedy agent starts performing better than the random agent, even though it only randomly explores movies until move 20. This occurs due to the implementation of the ϵ-greedy agent. The ϵ-greedy agent sorts the recommendation list by the movies that return the best score. This means that the movie in the first position in the list will be the one that returns the highest score. Once the recommendation list is full, the ϵ-greedy agent will only replace the last movie in the list if the selected movie returns a higher score than it. This is why it starts outperforming the random agent after move 10.

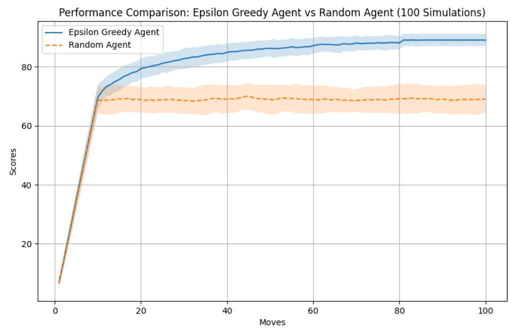

Figure 4.1: Performance comparison of the ϵ-greedy agent vs random agent

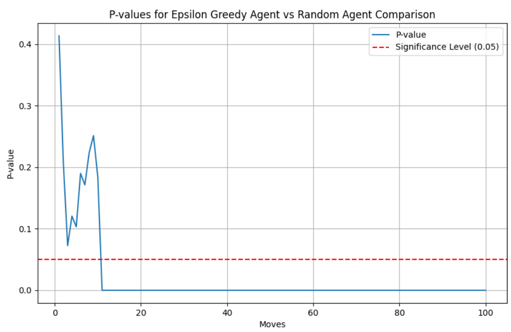

Figure 4.2: P-values for ϵ-Greedy Agent vs Random Agent Comparison

We can see this in Figure 4.2, which plots the p-values for the ϵ-greedy agents vs the random agent. This plot shows when the ϵ-greedy agent performs better than the random agent in a statistically significant manner. The p-values were calculated using a paired t-test, chosen for its ability to compare the means of two related samples.

A p-value below 0.05 indicates a statistically significant difference between the two agents.

4.2 Teaching Guide

The resulting teaching guide is designed to provide educators from all backgrounds with the necessary knowledge to confidently teach the lesson program. The entire teaching guide and presentation slides are included in appendix A at the end. The guide is split up into several key sections:

4.2.1 Motivations

The motivation section explains why this lesson matters. It breaks down why this topic was chosen, how it connects to the students' lives, and the educational values it adds. Explaining these things, in addition to the key takeaways, puts the educators in the right mindset when leading the class.

4.2.2 Theoretical Framework

This section explains all the necessary concepts that this lesson covers. It provides definitions of the key terms and concepts. Additionally, it contains clear examples that illustrate these concepts. Examples are made specifically in the context of the Movie Ranker Game and the MAB problem. This shows the educators how these concepts directly relate the game and the MAB problem.

The MAB problem is explained through the use of the slot machine analogy. This analogy is a concise and engaging way to explain the concepts that make up the MAB problem, as it has many direct real world comparisons.

The MAB problem's environment is also useful to explain related concepts such as the exploration vs exploitation dilemma, ϵ-Greedy algorithm, and more. Complicated terms such as ϵ are simplified through giving them a more descriptive name like "exploration rate".

4.2.3 Movie Ranker Game

The Movie Ranker game section, explains the main concept of the game and what the desired learning outcomes are for playing the game. Furthermore, it explains the rules and the game mechanics, such as how the ϵ-Greedy agent improves each round through increasingly exploiting.

The Movie Ranker Game is available on github here: `https://github.com/TexMcGinley/Bachelor_Thesis`

4.2.4 Ethical Discussion

This section breaks down some ethical discussion surrounding recommender systems and their issues. It explains how these problems arise and why they need to be addressed. The section includes multiple real world examples to further emphasize the impact of these systems. Possible solutions to these problems are also included. This section is designed to provide educators with the knowledge to lead discussion to important areas.

4.2.5 Lesson Breakdown

The goal of this section is to lay out the structure of the lesson. It includes the key points that should be covered for each slide and gives questions that can be used to lead the classroom discussions.

4.2.6 Additional Resources

To ensure that educators feel confident in teaching the lesson material, the additional resources section lists extra sources that provide more information and can help clarify concepts educators may feel uncertain about.

5 Conclusions and Discussions

This project set out to develop an innovative education program to teach Dutch high school students about ML concepts, specifically recommender systems, through GBL. The developed educational program includes a comprehensive teaching guide, an educational Movie Ranker game, and a set of lesson slides. The motivations for this project were as follows:

- **Preparing the Younger Generation:** To equip students with the knowledge needed to navigate a future where AI plays an integral part in their personal and professional lives.
- **Addressing Educational Gaps in the Dutch System:** To provide an engaging and accessible way to introduce AI and ML concepts in schools, addressing the current lack of priority placed on AI-focused education.
- **Exploring an Innovative Approach to teach ML Concepts:** To employ GBL to increase student motivation and engagement to create an enhanced learning experience.
- **Supporting the Dutch Economy:** To foster interest in technical career paths, ultimately contributing to a more skilled workforce in the AI and technology sectors.

The project achieved its goals in these ways:

1. **The Created Educational Material:** The created material explains fundamental ML and RL concepts in an approachable and engaging manner for high school students.

2. **Engaging Learning Experience:** The developed Movie Ranker game demonstrates some of the key concepts behind recommender systems and the MAB problem, in an engaging way through the use of a competitive and interactive platform.
3. **Teacher Accessibility:** The teaching guide was designed to be as accessible and comprehensive as possible by providing educators with all the necessary understanding to teach the program confidently.

The development process followed in creating the education program was adapted from Park and Kwon South Korean paper. It is split into 3 phases: preparation, development, and improvement (Park and Kwon, 2023). This methodology involved setting clear educational goals, organizing the learning contents, and utilizing expert feedback to improve the program.

GBL was chosen for the learning style as research such as Zhan et al.'s systematic literature review showed the benefits of GBL in AI education. GBL effectively increases student motivation and engagement, allowing for an enhanced learning experience (Zhan et al., 2022).

The ϵ-Greedy algorithm was a key part of the Movie Ranker game. The implemented algorithm performed better than a random agent in a statistically significant manner, as seen in figures 4.2 and 4.1. This demonstrates that the algorithm is implemented correctly and highlights fundamental RL concepts such as the exploration vs. exploitation dilemma.

The developed program was created to have the greatest impact possible within the current Dutch educational system. Research into the current state of education in the Netherlands showed some concerning trends, with the most concerning being teacher shortages (Ministerie van Onderwijs, 2024). To address this, the teaching guide was created to be accessible to as many educators as possible by removing the need for prior knowledge. Unlike most AI programs that require a scientific background, this guide was developed to be easily understood by educators from any background, enabling them to confidently teach the class.

5.1 Limitations

The main limitation of this study is that it is currently untested in a real-world environment, meaning that the program's effectiveness is yet unknown. Additionally, due to time constraints, the last step of the development methodology, the improvement phase, has not been conducted. This means that there most likely are still many areas that need to be improved and revised to increase the program's effectiveness.

Another key limitation of the program is that all the material is currently in English. This means that, in its current state, the program would only be effective for schools that teach in English.

5.2 Future Work

Future work would first include finishing the improvement phase. This would comprise consulting educational and AI experts to get their feedback on the current program, then utilizing their feedback to revise and improve the program to increase its effectiveness. After completing this phase, the lesson would need to be evaluated to see if it meets the desired outcomes. To do this, a study should be conducted using pre- and post-tests to assess if the desired learning outcomes are met. This study would have to be done on participants that match the program's target demographic.

References

Monica Anderson, Michelle Faverio, and Jeffery Gottfried. Teens, social media and technology 2023, Dec 2023. URL `https://www.pewresearch.org/internet/2023/12/11/teens-social-media-and-technology-2023/`.

Ljubisa Bojic. Ai alignment: Assessing the global impact of recommender systems. *Futures*, 160:103383–103383, 04 2024. doi:10.1016/j.futures.2024.103383.

Claire Chen. Ai will transform teaching and learning. let's get it right., 03 2023.

Amit Datta and Michael Carl Tschantz. *Automated Experiments on Ad Privacy Settings a Tale of Opacity, Choice, and Discrimination **. 2015. URL `https://arxiv.org/pdf/1408.6491`.

Juho Hamari, David J Shernoff, Elizabeth Rowe, Brianno Coller, Jodi AsbellClarke, and Teon Edwards. Challenging games help students learn: An empirical study on engagement, flow and immersion in gamebased learning. *Computers in Human Behavior*, 54:170–179, 2016. doi:https://doi.org/10.1016/j.chb.2015.07.045. URL `https://www.sciencedirect.com/science/article/pii/S074756321530056X`.

Wayne Holmes, Zhang Hui, Fengchun Miao, and Huang Ronghuai. *AI and Education*. UNESCO Publishing, 04 2021.

Jacob Liedke and Luxuan Wang. News platform fact sheet, 11 2023. URL `https://www.pewresearch.org/journalism/fact-sheet/news-platform-fact-sheet/`.

Darrell M. West and John R. Allen. How artificial intelligence is transforming the world, 04 2018.

Tanya Milberg. The future of learning: Ai is revolutionizing education 4.0, 04 2024.

Cultuur en Wetenschap Ministerie van Onderwijs. Senior general secondary education (havo) and pre university education (vwo) - secondary education - government.nl, 12 2014.

Cultuur en Wetenschap Ministerie van Onderwijs. De staat van het onderwijs, 2024.

Enrico Moch. The industrial revolution of artificial intelligence: A new era of automation and innovation. *International Journal of Public Administration, Management and Economic Development*, 8(2), Dec 2023. doi:https://doi.org/10.60026/ijpamed.v8i2.104. URL `https://www.ijpamed.eu/index.php/journal/article/view/104/98`.

NVIDIA. What is a recommendation system?, 2024. URL `https://www.nvidia.com/en-us/glossary/recommendation-system/`.

Jihwan Park, Jo-Eun Jeong, and Mi Jung Rho. Predictors of habitual and addictive smartphone behavior in problematic smartphone use. *Psychiatry Investigation*, 18(2), Feb 2021. doi:https://doi.org/10.30773/pi.2020.0288.

Joonhyeong Park, Tat Joo Teo, Albert C.K. Teo, Jina Chang, Jun Huang, and Seungbum Koo. Integrating artificial intelligence into science lessons: teachers' experiences and views. *International Journal of STEM Education*, 10, 10 2023. doi:10.1186/s40594-023-00454-3.

Woongbin Park and Hyuksoo Kwon. Implementing artificial intelligence education for middle school technology education in republic of korea. *International Journal of Technology and Design Education*, 34, 02 2023. doi:10.1007/s10798-023-09812-2.

Meihua Qian and Karen R. Clark. Game-based learning and 21st century skills: A review of recent research. *Computers in Human Behavior*, 63:50–58, 2016. ISSN 0747-5632. doi:https://doi.org/10.1016/j.chb.2016.05.023. URL `https://www.sciencedirect.com/science/article/pii/S0747563216303491`.

Grand View Research. Artificial intelligence market size, share — ai industry report, 2025, 2023. URL `https://www.grandviewresearch.com/industry-analysis/artificial-intelligence-ai-market`.

Aleksandrs Slivkins. Book announcement: Introduction to multi-armed bandits. *ACM SIGecom Exchanges*, 18(1):28–30, Dec 2020. doi:https://doi.org/10.1145/3440959.3440965.

Richard S Sutton and Andrew Barto. *Reinforcement Learning : an Introduction.* The Mit Press, Cambridge, Ma ; Lodon, 2018. ISBN 9780262039246.

Joëlle Swart. Experiencing algorithms: How young people understand, feel about, and engage with algorithmic news selection on social media. *Social Media + Society*, 7(2):1–11, Apr 2021. doi:https://doi.org/10.1177/20563051211008828.

Matti Tedre, Tapani Toivonen, Juho Kahila, Henriikka Vartiainen, Teemu Valtonen, Ilkka Jormanainen, and Arnold Pears. Teaching machine learning in k–12 classroom: Pedagogical and technological trajectories for artificial intelligence education. *IEEE Access*, 9:110558–110572, 2021. doi:10.1109/access.2021.3097962.

Constantijn van Oranje, Myrthe Hooijman, Nika Schutten, Chiara Parisi, Sam van Rossum, and George Windsor. State of dutch tech 2024 2 state of dutch tech 2024, 2024.

Wikipedia. Education in the netherlands, 03 2019. URL `https://en.wikipedia.org/wiki/Education_in_the_Netherlands`.

Wikipedia. Technasium, 10 2020. URL `https://en.wikipedia.org/wiki/Technasium`.

Andrea Willige. 5 ways ai can benefit education, 05 2024. URL `https://www.weforum.org/agenda/2024/05/ways-ai-can-benefit-education/`.

Zehui Zhan, Yao Tong, Xixin Lan, and Baichang Zhong. A systematic literature review of game-based learning in artificial intelligence education. *Interactive Learning Environments*, 32(3):1–22, Sep 2022. doi:https://doi.org/10.1080/10494820.2022.2115077.

A Appendix

Below is the created teaching guide, followed by the lesson slides.

university of groningen

faculty of science and engineering

Teaching Guide

Recommender Systems and Movies

Author: T.X. Mcginley , 2024

Motivations

In today's digital age, recommender systems are everywhere, controlling almost everything we see online. With people increasingly relying on digital platforms for information, the influence of these systems on our society continues to grow (Liedke and Wang, 2023). Despite their prevalence, the average person has little to no understanding of how these systems work and their potential impacts. This lecture aims to fill that gap, particularly for young people who are growing up in an increasingly online world.

Why This Topic?

Recommender systems are highly relevant to high school students as they are constantly interacting with them through social media, online shopping, and streaming services. These systems determine what content they see, from Instagram posts to Netflix recommendations and YouTube videos. Educating students about how recommender systems work can make them more aware of their impacts and help them become informed digital citizens who can make better decisions about the content they consume.

Key Takeaways

1. **Demystifying AI**: Breaking down the complexities of AI, making it approachable and understandable to students through clear, relatable examples.
2. **Insight into Recommender Systems**: Help students better understand recommender systems, their impact on daily life, and the ethical implications of their widespread use.
3. **Introduction to Reinforcement Learning**: Teach students some fundamental reinforcement learning concepts, specifically the Epsilon Greedy Algorithm, and how these algorithms work.
4. **Critical Thinking about AI's Impact**: Get the students to reflect on the influence of recommender systems and make informed decisions about their use in personal lives.
5. **Increase students' curiosity about AI:** Foster students' curiosity about AI by teaching them interesting AI concepts and making direct connections to their interests.

Connection to Students' Lives

AI and recommendation systems are becoming increasingly integral to daily life, especially for young people who spend a significant amount of time on social media. From the videos and accounts they get recommended on YouTube to the suggested products they see when shopping on Amazon, these recommender systems decide almost everything people see online today. By understanding how these systems work, students can become more conscious of their influence and make more informed choices.

Broader Implications

With AI improving rapidly and its adoption becoming increasingly widespread, Knowledge of AI and specific machine learning and recommender systems is increasingly valuable in today's job market. The field of machine learning is rapidly growing, and the number of career opportunities is rising steadily, making it an increasingly attractive field to be in (Shewale, 2023). By introducing students to these concepts early on, we hope to spark their interest and potentially guide them toward a lifelong passion for AI and machine learning.

Educational Value

This lecture effectively engages students through discussion and game-based learning. Discussions encourage students to share their ideas and relate the material to their own experiences, making the learning process more interactive and relevant. The Movie Ranker game fosters critical thinking as students must strategize and make decisions similar to those involved in creating a recommender system. This hands-on approach makes the learning process fun and deepens their understanding of the concepts taught.

Theoretical framework

Machine learning (ML)

AI is an umbrella term that refers to any program that can sense, reason, adapt, and act. Within AI, there are multiple separate sub-fields. The most famous one and the one most relevant for this lesson is machine learning. In the figure below, you can see the subfields that makeup AI.

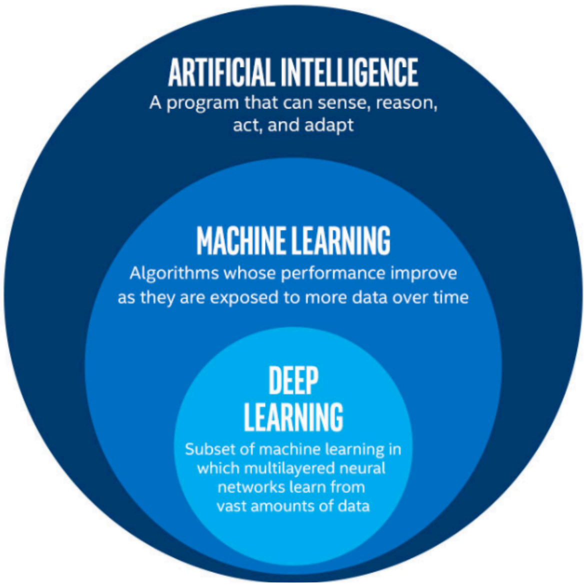

Fig.1 AI’s subfields (Singh, 2018)

Machine learning is the study of algorithms that can learn from data to generalize on new unseen data. In simple terms, it is teaching machines to make decisions by themselves on unseen data. For example, imagine you have a set of images. You want your computer to tell you which images have stop lights in them and which have crosswalks. To do this, you would start by giving the computer lots of different photos, each clearly labelled as "stoplights" or" crosswalks." The computer would repeatedly examine these images, analyzing the shapes, colours, and patterns. After seeing enough labelled examples, the algorithm begins to recognize certain features and patterns. The next time you show a new image, the algorithm uses what it has learned to guess whether it is a stoplight or a crosswalk. The more photos you show the algorithm, the better it gets at guessing. So, the computer learns from the actual data to get better at the task rather than the programmer programming the algorithm with specific rules. Within Machine learning, there are also 3 subfields.

- **Reinforcement learning** is where the algorithm makes an action and learns from the outcome of the action. It's similar to teaching a pet a new trick by rewarding good behaviour. The algorithm performs an action in an environment and receives rewards or penalties based on the action. Over time, the algorithm learns to maximize rewards by choosing the best action to take in different situations. Reinforcement learning is often used in games, robotics, and autonomous driving systems where the goal is to achieve the best outcome through trial and error.
- **Supervised learning** is just like the previous example we gave. It is when you train the algorithm with already labelled data. It is called supervised learning because an external source, usually a human, is needed to label the data.
- **Unsupervised learning** is when you give the algorithm a completely unlabeled data set. The algorithm's goal is then to find patterns and structures in the data on its own. For example, if you give the algorithm a bunch of pictures, it might group them in clusters based on similarities, like photos of cities, lakes, or sunsets, without being told what each group represents.

For the purpose of this class, we will focus only on **reinforcement learning.**

Agent and environment

In the context of reinforcement learning, "the agent" refers to the action-taker. It interacts with the environment, which is everything outside the agent that the agent can perceive and act upon. Based on what the agent observes in the environment, it decides on an action to take. Once the agent has performed an action on the environment, the environment responds by providing some rewards and new observations. Understanding the interaction between the agent and the environment is key to grasping how reinforcement learning works.

- **Agent**: The entity that makes decisions (e.g., the machine).
- **Environment**: The context in which the agent operates (e.g., the system or setting where the agent's decisions have consequences).

Example: In the Movie Ranker Game, the agent is the computer (or the students acting as one), and the environment is the set of available movies and the user's profile containing the relevant user details. The computer (the agent) interacts with the set of movies and the user profile (the environment) to perform the action of choosing the best movies to recommend.

Reward

The reward is the feedback that the agent receives after performing an action. The agent's goal is to maximize the total reward over time. A reward can be positive or negative. Rewards are a fundamental part of reinforcement learning as they guide the agents' learning by giving them a metric of how good or bad an action is.

- **Reward**: Feedback that the agent uses to learn how good or bad its action was.
- **Agent's objective**: Maximize total reward over time.

Example: In the Movie Ranker Game, the reward is the score of how well their recommendations match the user's preferences. The agent's objective is to get the highest possible score (reward) over time by finding the best 10 movies to recommend.

Movie Ranker Game Reward Function

Here's a simple breakdown of how the game calculates the score (reward) for each recommendation:

1. **User Preferences**: Each user profile has certain preferences for movie genres, like how much they like action, comedy, or drama movies. These preferences are given a score, such as 10 for a genre they love and 1 for a genre they dislike.
2. **Movie Details**: Each movie has information about its genres, rating, release year, and certification (like age ratings).
3. **Scoring the Genres**: The game looks at the movie's genres and matches them with the user's preferences. For example, if a user loves action movies (score 10) and the movie is an action movie, this will contribute positively to the movie's score.
4. **Final Score**: The final score for a recommended movie is calculated by averaging the user's genre preference scores for the genres that the movie belongs to. This means we add up the preference scores for all the genres of the movie and then divide this total by the number of genres the movie has. This gives us a score that reflects how well the movie matches the user's tastes.

Example calculation:

- Recommended movie: **Spider-Man**
    - Belongs to these genres: Action, adventure, science fiction
- User: **John Doe**
    - Shortened preference list: [Action: 7, Adventure: 4, Science fiction: 8]

- Score calculation:
    - Adding all the scores for the movie genres together:
        - 7 + 4 + 8 =19
    - Dividing the by the number of genres the movie belongs to:
        - 19 / 3= 6.333
    - Final score for Spider-Man = 6.333

Multi-armed bandit (MAB) problem

The multi-armed bandit problem is one of the most famous reinforcement learning problems. The problem is as follows:

- Imagine you are in a casino with a line of slot machines and want to win as much money as possible.
- Each slot machine gives out a different amount of money, and they all have a different chance to give a reward or not.
- You don't know the chance of hitting a reward for each slot machine, nor do you know how much money each machine will pay out.
- One machine may pay $5 once every two times you play it, and another may pay $1000 once every 50 times you play it. The only way you can find out how good a machine is is by playing it.
- How do you make the most possible money?

This type of trial-and-error problem is what reinforcement learning algorithms try to solve. The key dilemma of the Multi-Armed Bandit problem is the exploration vs exploitation dilemma. At each point, you must choose if it is better to explore a new slot machine and possibly find a new, better reward or exploit one you already know the reward of.

Exploration vs. Exploitation

The Exploration vs. Exploitation dilemma is the dilemma of how to balance two opposing strategies, exploration and exploitation, to maximize long-term rewards. Exploring new options results in gaining information about the environment, which may lead to better future outcomes at the cost of an exploitation opportunity. Exploiting involves choosing the option that will give the highest reward based on the system's current knowledge of its environment (Sutton and Barto, 2018). Since exploitation depends on the system's current knowledge of its environment, if this is incomplete, the action chosen by the exploitation strategy may not be the optimal choice. Because of this, it is important that the system explores enough to have a good understanding of its environment. This is why when the system starts out on a problem and knows very little about its environment, it is intelligent to favour exploration more, but as its understanding of the environment grows, it becomes increasingly better to favour exploitation.

- **Exploration:** Choosing an action that may not be optimal to learn more about the environment.

- **Exploitation:** Choosing the best action based on the system's current knowledge of its environment.

MAB example: In the MAB problem, the exploration choice would be choosing a slot machine you haven't played, and the exploitation choice would be choosing the slot machine that has been giving you the best rewards.

Movie Ranker Game example: In the Movie Ranker game, the exploration choice would be choosing a new movie to learn more about the user's preferences. The exploitation choice would be choosing a movie you already know the user likes.

Epsilon Greedy Algorithm

The Epsilon greedy algorithm is one of the most famous reinforcement learning algorithms. It addresses the exploration vs. exploitation dilemma by setting an exploration rate, Epsilon. Depending on the exploration rate, the algorithm will either randomly explore an action or exploit its knowledge to choose the action that it expects will return the highest reward at each opportunity to take an action. The exploit action is also referred to as the greedy action, as the system chooses the action that will give the best reward now over possible better future rewards. There are a few ways the algorithm can track how good an action is. One of the most commonly used methods is to keep a running average of the rewards received from each action. This simply involves adding all the rewards gained from an action and dividing it by the number of times the action was chosen.

- **Epsilon:** The probability to explore (value between 0-1).
- **Greedy action:** The action based on current knowledge will give the best reward, also known as exploitation.
- **Running average:** One of the methods used to keep track of how good each action is.

Exploring to Exploiting

As stated above, exploring a lot at the start of a problem is a good idea to gain more information about the system's environment. As the system's understanding of its environment grows, the exploitation option becomes an increasingly better option to get the highest possible rewards in the long term. To achieve this behaviour, we need a high exploration rate (Epsilon) at the start, and as time goes on, we need a lower and lower exploration rate (Epsilon). This is where the term "exploration rate adjustment" comes in. Exploration rate adjustment is the process of gradually reducing the exploration rate after each action.

There is a whole set of different methods for adjusting the exploration rate for a given problem. These are a little out of the scope of this class, but here is a simple explanation of one of the methods: The initial exploration rate and the rate at which it decreases are decided by trying different values, applying statistical methods, and analyzing the results. These results help us estimate the best exploration rate and adjustment method for the given problem.

- **Exploration rate adjustment:** The rate the exploration probability decreases after each action.

Movie Ranker Game

The game's objective is to get the students to learn about recommender systems by having to act as one. Specifically, the students will be tasked with acting as a movie recommender system. They will be shown a user's profile containing relevant user details and a list of movies they previously watched. Using this information, the students must create a list of 10 movies that best match the user's preferences. The students will have to develop strategies to get the highest possible rating. By doing so, they must apply some of the same thinking that someone creating an actual recommender system must consider. Such as which features are important to focus on.

Rules

The game is divided into five rounds. In each round, the player can make 20 moves. The player will compete against an Epsilon greedy algorithm that is performing the same task. At each round, the Epsilon greedy algorithm will have a lower exploration rate (Epsilon), meaning that it will favour exploitation more, resulting in a higher score each round.

Rules and instructions:

- Using the information available about the user, recommend 10 movies that should best match their preferences.
- Think about the strategies and decisions that led to each recommendation.
- To recommend a movie, drag and drop it from the movie grid into the recommendation rankings.
- Clicking on a movie poster will flip it over to show its details.
- Drag and drop a movie from the recommendation ranking to the movie grid to remove it from the recommendation rankings.
- Dragging and dropping a movie to an occupied ranked movie will swap the movies.
- 5 rounds
- 20-move limit per round
- An action is only considered a move when a movie that is not in the recommendation ranking is added to the recommendation rankings.
- Once satisfied with the recommendations, click the submit button.
- In the submit window, the player will see their score for the round, their high score for all the rounds they have played, the score the Epsilon greedy algorithm got for the round, and the exploration rate (Epsilon).
- The player should see how many rounds they are able to beat the algorithm's score.

Ethical discussion

As more people get online each day, the impact of recommender systems continues to grow (International Telecommunication Union, 2023). Most likely, a recommender system has decided the vast majority of the content you see online. Because these systems decide what you see online, they can significantly influence your beliefs and opinions. With this in mind, it is important to evaluate the possible issues with these systems and the impacts that they can have.

This section is meant to provide you with the necessary knowledge to confidently guide the classroom discussions to the right areas if needed.

Bias

A big issue with recommender systems and AI as a whole is bias. Since these algorithms learn from the data they train on, if there is any bias in the data, the algorithm will most likely learn this bias, too. For a movie recommender system, this could look like a bias stemming from a training dataset where romance movies are only favoured by females. The algorithm would then heavily recommend romance movies to females, not men. This bias would not reflect the reality that liking romantic movies doesn't depend on gender but rather taste.

While this example might seem minor, biases in AI can have serious real-world implications. For example, a study found that Google's ad system showed high-paying job ads more frequently to men than to women (Datta and Carl Tschantz, 2015). Such biases reinforce stereotypes and perpetuate inequality.

The more control we give AI, the bigger the impact these types of biases have. Therefore, it is crucial that the developers of recommender systems and AI systems actively work to identify and combat the biases of both the training data and the algorithms themselves. This involves:

- **Diverse Data**: Using diverse training datasets that represent multiple perspectives and groups.
- **Bias Detection**: Implement methods and checks to detect and measure bias in AI systems.
- **Algorithmic Fairness**: Developing algorithms that are designed to minimize bias and promote fairness.
- **Continuous Monitoring**: Regularly reviewing and updating AI systems to address any potential biases.

By taking these steps, we can help ensure that AI and recommender systems make fair and unbiased recommendations, promoting equality and reducing the risk of perpetuating harmful stereotypes.

Privacy concerns

The performance of these recommender systems is often directly tied to the quantity and quality of data available to them. Because of this, recommender systems often collect vast amounts of data about users, their behaviours, interactions, and preferences. Many users aren't aware of the extent of this data collection. There are multiple potential risks that come with the way this data is collected and used; the two biggest are:

- **Data Misuse:** The risk that the data collected by these recommender systems gets used in a manner beyond what the user agreed to, either accidentally or intentionally. The most famous recent case of data misuse is the Cambridge Analytica scandal. Where users who completed a personality test unknowingly gave permission to access their personal information and that of their friends. This resulted in data of up to 87 million people being harvessed, which was subsequently used to create detailed voter profiles. These profiles were then used to run target ads for susceptible users (Romano, 2018) (Chang, 2018).
- **Data Breaches:** Since these recommender systems collect and store huge amounts of user data, they are vulnerable to security breaches that result in sensitive user data being leaked. A recent example of this is LinkedIn's data breach, in which data from 92% of all its users was hacked (Hodson, 2021).

To ensure that user data is protected, there are a few methods that can be employed:

- **Data anonymization:** Companies can employ data anonymization techniques to remove personally identifiable information from the data they store and collect, protecting user privacy.
- **User control:** This allows users to control what data is collected and how this data is used. This can be done through privacy settings that allow the user to opt out of data collection and delete their data.
- **Transparency:** Companies need to clearly communicate what data they will collect, how they will use it, and who will have access to it.

Data misuse is a huge issue, and there are multiple methods that can be used to ensure ethical data use:

- **Consent:** All the data collected should be done with the user's consent. This also needs to be done in a way that is clear to the user what they are agreeing to.
- **Minimizing data collection:** Collecting only the data that is absolutely necessary for the intended purpose to minimize privacy risks.
- **Purpose limitation:** All data collected should only be used for the purposes for which it was collected. Using data beyond its intended purpose without user consent is unethical.

Implementing all these methods will make users' data safer and allow for ethical data use.

Impact on human behaviour

As discussed above, the prevalence of these systems and their influence on controlling what you see on the internet are immense. These systems can shape one's preferences and behaviours by influencing what content one is exposed to. Some of the most relevant impacts they have are:

- **Influencing opinions and beliefs:** Recommender systems can significantly shape users' opinions and beliefs. By selectively presenting information, these systems can influence how users perceive the world. In social media, users may only see content they agree with and, therefore, have a positive reaction too. Because of this, they risk users having potentially skewed and misinformed viewpoints.
- **Behavioural Addiction:** Recommender systems of digital platforms like streaming services and social media networks are designed to get users to spend the most amount of time on these platforms. While this is commercially advantageous for these platforms, it can lead to addictive behaviours in users that can have a negative impact on mental health, resulting in anxiety, stress, and even depression (Bojic, 2024)(Park et al., 2021).

The impact of recommender systems on human behaviour raises several ethical considerations. Developers need to be mindful of the potential consequences of their algorithms, ensuring they promote diverse and balanced content. Transparency in how recommendations are made and allowing users more control over their preferences are essential steps toward mitigating negative impacts. Additionally, there is a responsibility to address biases within these systems to prevent perpetuating stereotypes and unfair treatment.

Engagement vs user satisfaction

Another area of concern is the metric a recommender system uses to evaluate its performance. There are two key metrics to consider:

- **User engagement:** the interactions and time a user spends on the platform. This can be watch time, comments left, clickthrough rate, subscribing/following, and shares. There is no difference between negative and positive engagement.
- **User satisfaction:** this is how users feel when using a recommender system. Good user satisfaction happens when users get what they want and expect from the system. Other factors that play into good user satisfaction is if the user gets positive emotions from using the systems.

If a recommender system is designed to maximize user engagement, then an issue that can occur is that it starts to try to purposely evoke negative emotions/reactions from users. This is because negative reactions tend to be stronger and easier to elicit (Baumeister et al., 2001). In addition to this, a user is more likely to engage with a post they disagree with due to the stronger reactions they evoke. The result of this is that users of these recommender systems end up in a cycle of negative emotions, where they are constantly being shown posts that make them feel anxious, sad, angry, and/or depressed (Bojic, 2024). This is especially the case with younger people. In particular, younger females can end up with self-image problems due to comparing themselves with the unrealistic standards of beauty that they get recommended on social media (Papageorgiou et al., 2022).

To combat these issues, there are some methods the companies behind these recommender systems can use:

- **Use multiple metrics:** Using only user satisfaction as the evaluation metric may result in a far better user experience but may not return the desired results and performance. As discussed above, there are multiple issues with using only user engagement. However, it is arguably a better metric to evaluate the performance of the recommender system. This is because user satisfaction can be hard to measure accurately. They usually require some additional form of input from the user, whether it be a separate survey or a simple click of a like button. Even if users are very satisfied with their experience, they may still decide not to add that extra input. With user engagement, the user just has to use the platform, and the information gained from their actions can be immediately used as an evaluation metric. In order to get a system that performs well and prioritizes how users feel when using the recommender system, both metrics need to be combined. Finding the right balance between both metrics is difficult and depends on what the recommender system is designed for.
- **Distinguishing between good and bad user engagement:** Not all engagement is good, and it is up to the designer of these recommender systems to separate between good and bad engagement. This is a complex task, and there is no one-fits-all solution. Reddit, for instance, has a system that allows most subreddits to have a team of moderators who filter the content to ensure that the content and discussion happen positively.

Applying these methods, these recommender systems should result in high-performing systems and leave users more satisfied and well-off.

Lesson Breakdown

1. **Intro and go through the lesson overview (Slides 1-2)**
    - Briefly walk through what will happen in the lesson.
2. **Understanding the class's understanding of AI (Slide 3)**
    - Initial question: Ask the class what they know about AI and if they can give some examples of AIs they know of or use.
    - Guiding talking points:
        - Ask them where they see or use AI in their daily lives.
        - If they say yes, ask if they understand how that AI works.
        - Are they excited about AI or worried? And why?
3. **Recommender Systems (Slide 4)**
    - Explain to students what recommender systems using real-world examples such as YouTube, Netflix, Instagram, TikTok, etc.
    - After explaining what recommender systems are using one real-world example, ask students if they can think of any other examples of where recommender systems are used.
    - Key points:
        - Recommender systems are algorithms that determine what content to recommend to whom.
        - They control the majority of what you get shown online.
        - They decide what posts you see on Instagram, what movies are recommended on Netflix, what ads you see, and what products are shown when shopping online.
4. **Discussion Recommender Systems (Slide 5)**
    - Initial question: Can anyone think of any other areas where recommender systems are used?
        - Additional examples:
            1. Dating apps.
            2. Job websites such as LinkedIn.
            3. Travel websites such as Airbnb.
            4. Health and fitness apps that recommend workouts and recipes.
    - Additional question: Ask the students if they have any idea how these systems decide what to recommend to whom.
5. **Intro to Machine Learning (Slide 6)**
    - Briefly explain how AI is an umbrella term that encompasses multiple different sub-fields.
    - Introduce machine learning and provide them with an easily digestible example.
    - Example: Training an algorithm to distinguish between stoplights and crosswalks using labelled photos.
6. **Continuation (Slide 7)**
    - Discuss how completing CAPTCHAs helps train machine learning algorithms.
    - Example: Labeling data for autonomous driving algorithms to recognize stoplights.
7. **Reinforcement Learning (Slide 8)**

    - Briefly explain how machine learning consists of three subfields: reinforcement learning, supervised learning, and unsupervised learning.
    - Focus on reinforcement learning for this class.
8. **Intro to the Multi-Armed Bandit (MAB) Problem (Slide 9)**
    - Explain the MAB problem using the casino slot machine analogy.
    - Discuss the exploration vs. exploitation dilemma.
9. **Intro to the Epsilon Greedy Algorithm (Slide 10)**
    - Explain the Epsilon Greedy algorithm and its approach to the exploration vs. exploitation dilemma.
10. **Epsilon Greedy MAB Walkthrough (Slide 11)**
    - Walk through the Epsilon Greedy algorithm on a simplified MAB problem:
        - Initially, the algorithm has no information about any of the slot machines. So, the only option is to explore by randomly selecting one to play.
        - In this case, it selected the D2 slot machine.
        - It pulls the arm of the D2 slot machine and gets a reward of $5 **(First animation)**
        - Now, the Epsilon Greedy algorithm keeps track of the rewards it has received from each slot machine. It can do this in a few ways. For our example, we will keep a running average of the rewards received from each machine.
        - This means that it will add up all the rewards received from a single machine and divide that by the number of times the machine has been played.
        - The algorithm will now choose the next action to take.
        - Once again, it will have a chance to explore or exploit what it knows.
        - In this case, it chooses again to explore.
        - It randomly decided to select Slot Machine D4. **(Second animation)**
        - Slot machine D4 ends up paying out $20.
        - The algorithm learns and keeps track of the running average reward
        - The algorithm now takes another action. This time, it chooses to exploit.
        - Using the information it learned from previous rounds, the algorithm chooses the action that it believes will maximize the reward it receives, in this case, money.
        - Based on what it has learned, the best option is to pull the D4 Slot Machine since it is expecting another reward of $20.
        - This time, it returns a reward of $0
        - The algorithm learns from this reward and updates the expected awards for the D4 Slot Machine to $10 **(Third animation)**
        - As the algorithm plays the slot machines more and more times, it begins to get a more accurate estimate of each machine's expected reward. (**Fourth animation)**
        - Here, we can see the slot machines are their expected rewards after the algorithm played 100 rounds.

        - We can see that machines that seem really good, like Machine D4, can actually be a bad choice despite their initial high payout.
11. **Exploration Vs. Exploitation (Slide 12)**
    - Discuss the importance of balancing exploration and exploitation.
    - Explain epsilon decay and its role in controlling this balance.
12. **Game Rules Explanation (Slide 13)**
    - Explain the game's goal and rules :
        - Act as a movie recommender system.
        - Show a user's profile with previously watched movies.
        - Create a list of 10 movie recommendations based on this information.
        - Click on any movie poster to see its details.
        - Drag and drop a movie to add it or to remove it.
        - Dragging and dropping a movie on an occupied movie will swap the movies.
        - Compete against an Epsilon greedy agent in 5 rounds with 20 moves each.
        - The goal is to devise a strategy and see how many rounds you can beat the algorithm.
13. **Game - Main Session (Slide 14)**
    - 20 minutes of playtime
14. **Discussion (Slide 15)**
    - Initial question: How many rounds did you manage to beat the algorithm in?
    - Further questions:
        - What strategies did you use to come to your list of recommendations?
        - Are there any features that you decided to focus more on?
        - What other information do you think would be useful to have to make good movie recommendations?
        - How does this system differ from real-world recommender systems?
        - What other information do you think real-world recommender systems use to make their recommendations?
15. **Connection to Real-World Recommender Systems (Slide 16)**
    - Connect how real-world recommender systems, such as Netflix, YouTube, TikTok, etc.
    - Explain the types of data used in creating recommendations:
        - Watch time: How long you watch a video.
        - User interaction: Likes, comments, shares.
        - Video information: Titles, descriptions, tags.
        - Contextual information: Device type, location, time of day.
    - What type of reward do these systems use to train these recommender systems?
        - Briefly explain the difference between User engagement vs user satisfaction.
16. **Recommender systems issues discussion (Slide 17)**
    - Initial question: Do you think there are any possible issues with the influence of recommender systems on what we see online?

    - Areas to guide the discussion too:
        - Privacy concerns
        - Impact on human behaviour
        - Bias
        - Engagement vs user satisfaction
17. **Discussion of possible solutions (Slide 18)**
    - Initial question: How can we combat some of these issues?
    - Additional questions:
        - What are some things we can do to, to not fall into some of these traps?
        - Can we make better recommender systems?
        - What do you think would be good metrics to focus on when making recommendations?
18. **Final word + Questions (Slide 19)**
    - Make the final point of the need to be conscious of the impacts of the systems we incorporate into our daily lives and how we will need informed AI experts to create and ensure these systems don't result in negative ethical impacts.
    - Thank the students for listening and provide them with the opportunity to ask any other questions.

Additional Resources

Artificial Intelligence and Machine learning

- **AI vs Machine learning:** AI vs Machine Learning
- **Machine learning in more depth:** https://www.datacamp.com/blog/what-is-machine-learning

Reinforcement Learning

- **Video explaining the MAB problem:** Multi-Armed Bandit : Data Science Concepts
- **More in-depth to the Exploration vs Exploitation + More additional algorithm:** https://towardsdatascience.com/the-exploration-exploitation-dilemma-f5622fbe1e82

Recommender systems

- **How Recommender Systems Work:** How Recommender Systems Work (Netflix/Amazon)
- **Paper on the impacts of Recommender systems:** https://www.sciencedirect.com/science/article/pii/S0016328724000661#sec0155

References

Baumeister, Roy F., et al. “Bad Is Stronger than Good.” *Review of General Psychology*, vol. 5, no. 4, 2001, pp. 323–370, https://doi.org/10.1037//1089-2680.5.4.323.

Bojic, Ljubisa. “AI Alignment: Assessing the Global Impact of Recommender Systems.” *Futures*, vol. 160, 1 Apr. 2024, pp. 103383–103383, https://doi.org/10.1016/j.futures.2024.103383. Accessed 9 May 2024.

Chang, Alvin. “The Facebook and Cambridge Analytica Scandal, Explained with a Simple Diagram.” *Vox*, 2 May 2018, www.vox.com/policy-and-politics/2018/3/23/17151916/facebook-cambridge-analytica-trump-diagram.

Datta, Amit, and Michael Carl Tschantz. *Automated Experiments on Ad Privacy Settings a Tale of Opacity, Choice, and Discrimination* *. 2015.

Hodson, Madeleine. “Exclusive: 700 Million LinkedIn Records Leaked June 2021 | Safety First.” *PrivacySharks*, 27 June 2021, www.privacysharks.com/exclusive-700-million-linkedin-records-for-sale-on-hacker-forum-june-22nd-2021/.

International Telecommunication Union. “Statistics.” *Itu.int*, 2023, www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx.

Liedke, Jacob, and Luxuan Wang. “News Platform Fact Sheet.” *Pew Research Center’s Journalism Project*, 15 Nov. 2023, www.pewresearch.org/journalism/fact-sheet/news-platform-fact-sheet/.

Papageorgiou, Alana, et al. "“Why Don’t I Look like Her?” How Adolescent Girls View Social Media and Its Connection to Body Image." *BMC Women's Health*, vol. 22, no. 1, 27 June 2022, pp. 1–13, https://doi.org/10.1186/s12905-022-01845-4.

Park, Jihwan, et al. "Predictors of Habitual and Addictive Smartphone Behavior in Problematic Smartphone Use." *Psychiatry Investigation*, vol. 18, no. 2, 2 Feb. 2021, https://doi.org/10.30773/pi.2020.0288.

Romano, Aja. "The Facebook Data Breach Wasn't a Hack. It Was a Wake-up Call." *Vox*, 20 Mar. 2018, www.vox.com/2018/3/20/17138756/facebook-data-breach-cambridge-analytica-explained.

Shewale, Rohit. "70 Machine Learning Statistics 2023 (Market Size & Trends)." *Demandsage*, 6 Oct. 2023, www.demandsage.com/machine-learning-statistics/.

Singh, Seema. "Cousins of Artificial Intelligence." *Medium*, 1 June 2018, towardsdatascience.com/cousins-of-artificial-intelligence-dda4edc27b55.

Sutton, Richard S, and Andrew Barto. *Reinforcement Learning : An Introduction*. Cambridge, Ma ; Lodon, The Mit Press, 2018.

AI and Recommender Systems

By Tex McGinley

Lesson overview

- What is AI?
- Recommender Systems
- Reinforcement learning
- Epsilon Greedy Algorithm
- Introduction to the Game
- Game
- Discussion
- Questions

What do you know about AI?

Recommender systems

Fig 1. Source YouTube

Fig 2. Source Instagram

Fig 3. Source TikTok

Fig 4. Source Netflix

Discussion

- Can you think of any other areas where recommender systems are used?

Introduction to Machine learning

- Crosswalks vs. Stop Lights
- Labeled data
- Learning features and patterns
- Generalizing to unseen data
- Learning from the data

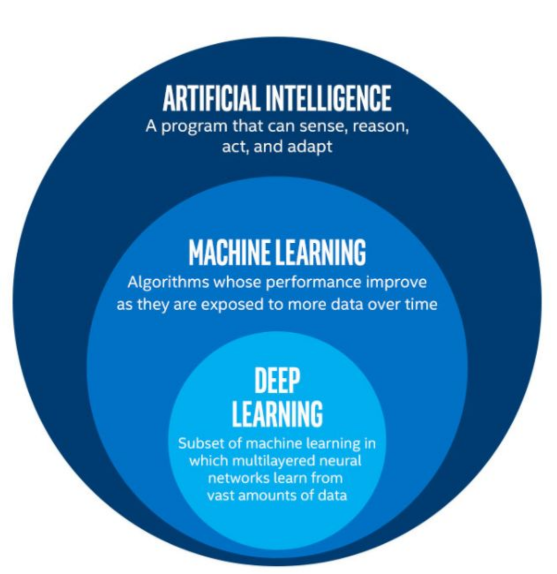

Fig 5. AI subcategories ([Source](#))

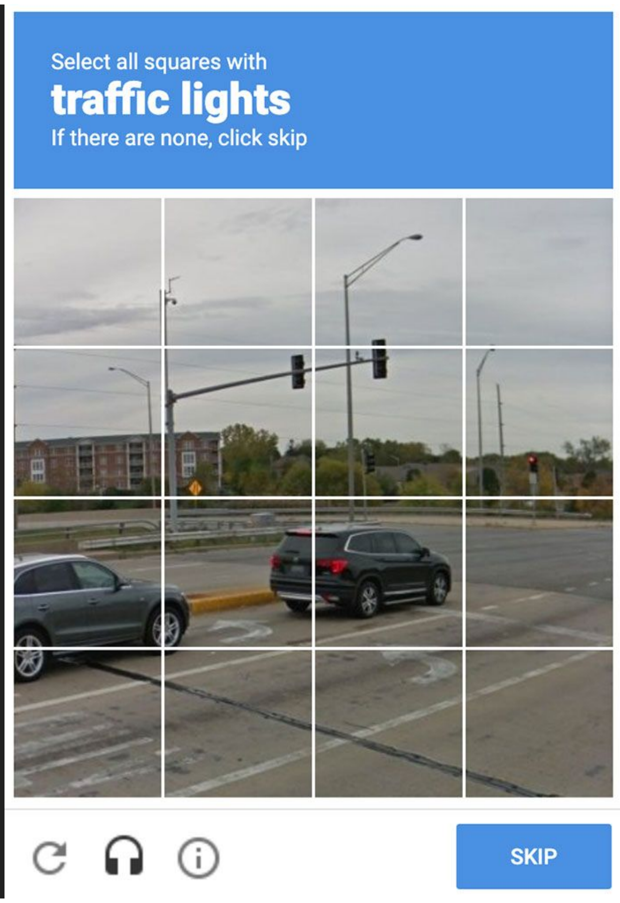

Fig 6. CAPTCHA traffic lights (Source)

Reinforcement learning

- Action in an environment
- Outcome from action
- Reward
- Maximize rewards over time

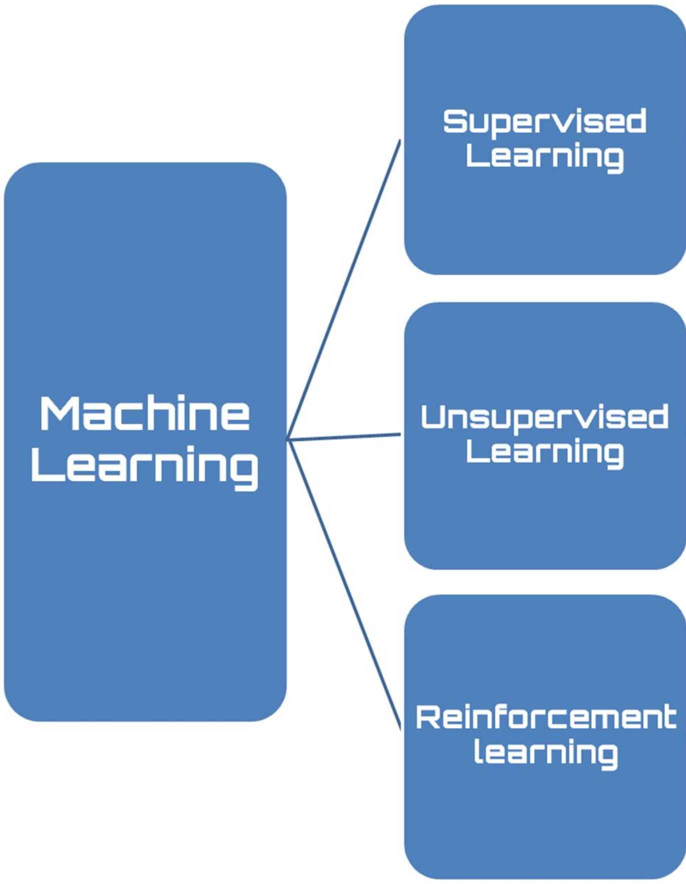

Fig 7. Machine Learning Sub-categories (Adapted from Source)

Introduction to Multi-Armed Bandits (MAB) problem

Fig 8. MAB Slot Machine Example
(Adapted from Source)

Epsilon Greedy introduction

- Exploration vs exploitation
- Explore a new state
- Exploit what we know
- Epsilon = probability to explore
- Maximize long term rewards

Epsilon Greedy

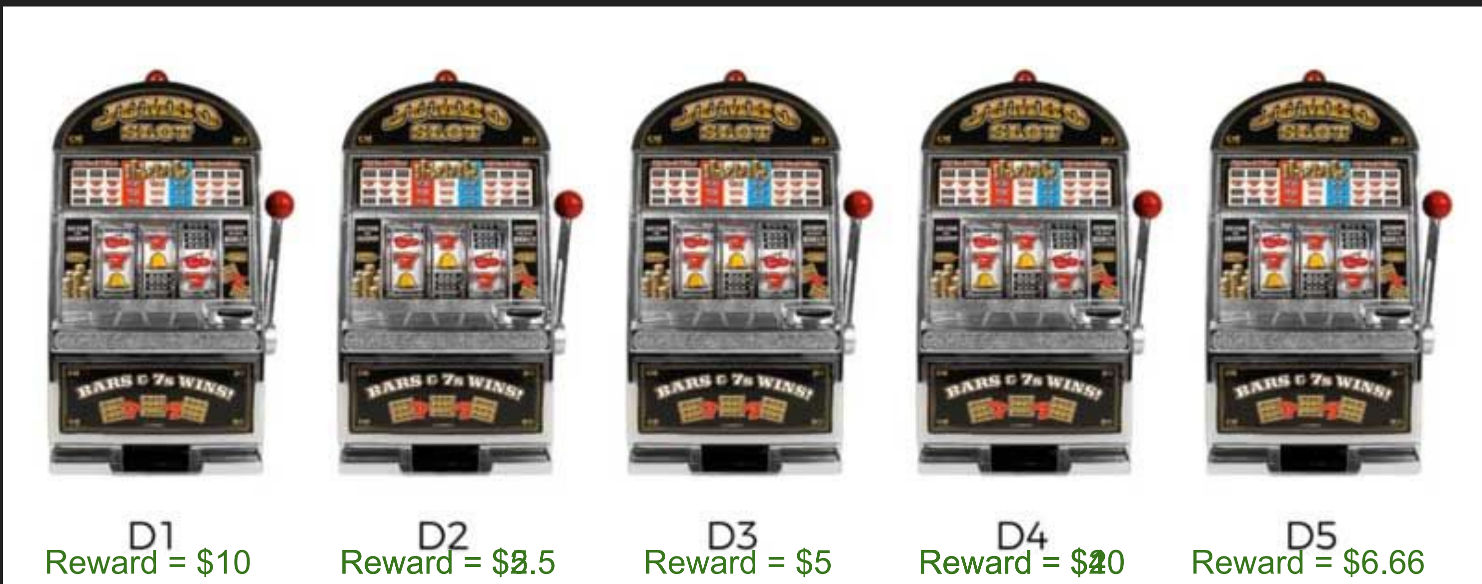

Fig 9. MAB Slot Machine Example
(Adapted from [Source])

Balancing Exploration vs Exploitation

- Initially favor exploration
- Increasingly favor exploitation
- Epsilon decay rate

Game Introduction

- You will be acting as a movie recommender system
- Come up with the 10 best movies to recommend to the given user profile
- Competing against the Epsilon Greedy algorithm
- 20 moves
- A action is only considered a move if a movie that is not in the recommendation ranking gets recommended
- 5 rounds
- Each round the Epsilon Greedy algorithm will perform better
- How many rounds you can beat the algorithm for?

Game time

Discussion

- What strategy proved the most successful for you?
- Did you notice any features that were more important than others
- How does this system differ from the real world recommender systems?

Connection to real world recommender systems

- Youtube
- Netflix
- Instagram
- Amazon
- Facebook
- TikTok
- Google

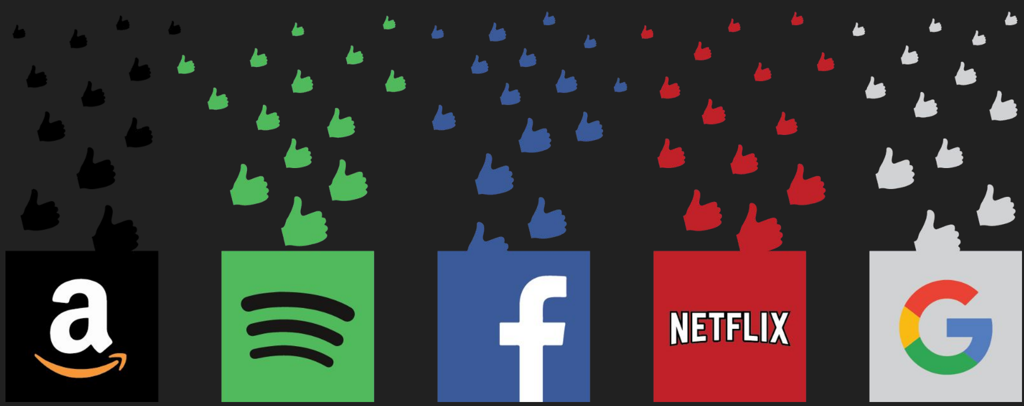

Fig 10. Real World Recommender Systems (Source)

Discussion

- What do you think could be some issues that come from having recommender systems everywhere?
- Have you ever noticed any problems with them?
- What are these recommender systems prioritizing?
- Do you think that recommender systems influence what you believe?

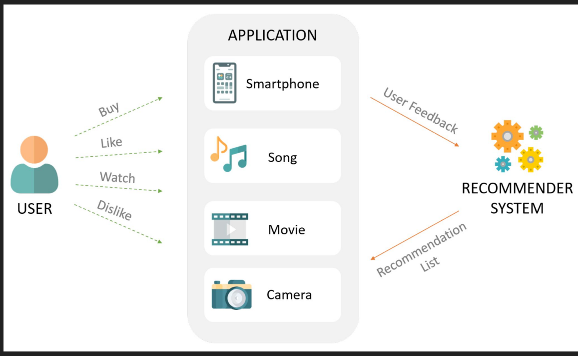

Fig 11. Recommending Content to Users (Source)

Possible issues with recommender systems

- Being shown only one side (polarization)
- The systems learns what results in more engagement from you, and it doesn't care whether the engagement is good or bad
- Behavioral addiction

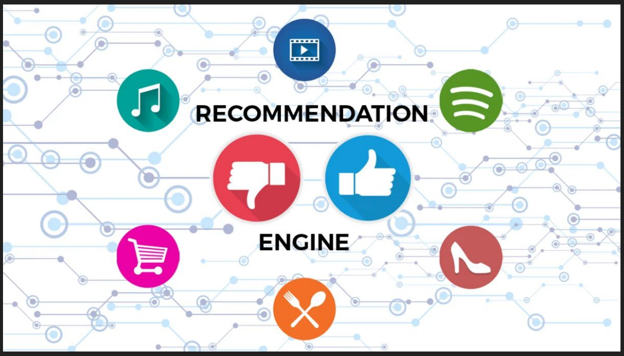

Fig 12. Recommendation Engines (Source)

Discussion

- How can we combat these issues?
- What are some things we can do to, to not fall into some of these traps?
- Can we make better recommender systems?
- What do you think would be good metrics to focus on when making recommendations?

Questions?

Fig 13. Generated using DALL-E 2