Machine learning

Chapter · November 2022

DOI: 10.1016/b978-0-12-818630-5.14013-8

4 authors:

Valerie J. Shute
Florida State University

G. Curt Fulwider
Florida State University

Zhichun Liu
The University of Hong Kong

Seyedahmad Rahimi
University of Florida

Machine learning

Valerie J. Shute[a], G. Curt Fulwider[a], Zhichun Liu[b], and Seyedahmad Rahimi[a], [a] Florida State University, Tallahassee, FL, United States; and [b] University of Massachusetts, Dartmouth, MA, United States

Introduction

Consider the following examples. Beverly, while starting her car, raises her wrist to her mouth and requests directions to her next appointment. Her request to the digital assistant is not understood at first, but it is after her second attempt. Since she first began using her smart watch, she has noticed a decrease in the times she must repeat herself. The watch is learning her voice and speech patterns. A few weeks later, she raises her wrist to find that her device is already offering to navigate to that location. It has recognized that every Thursday at 4:00 p.m. she travels to the same place. Beverly accepts the directions and heads down the road.

Across town, Jamal (a college sophomore) is figuring out which textbooks to purchase for his coming semester. Opening Amazon, he sees various college-related supplies being recommended; most of which are already monogrammed with his school's mascot. He selects his first textbook and notices that the site is suggesting another book commonly bought with the one he has selected. Checking his list again, he sees that the recommended book is also required. Both go into his shopping cart. After closing the browser, he sees a notification from his favorite social media platform. A "friend" has uploaded a somewhat embarrassing photo of them from the prior evening. The app has recognized Jamal's face, and is asking if he would like to be tagged in the photo. He declines (while also making a mental note to talk with his friend about posting photos online without permission).

These examples all fall under the umbrella of Artificial Intelligence (AI), which increasingly appears in the news because of the breadth of its reach. Part of this breadth comes from the generality of its definition. John McCarthy—who first coined the term in 1955—defined artificial intelligence as, "the science and engineering of making intelligent machines, especially intelligent computer programs" (Stanford University, n.d.). The purpose of AI is to model human thinking, and eventually surpass it. Because it is such a broad term, AI has found application in nearly all areas that make use of technology and data. The specific area within AI where the opening examples fall is called machine learning (ML).

ML has become a popular buzzword in recent years and is often incorrectly used interchangeably with AI. Rather, ML refers to the process by which a computer attempts to find regularities or patterns from a set of data—like speech patterns, purchasing habits, and facial features described in the examples above. Accurate recognition of patterns requires the algorithm to be exposed to large sets of examples. In this way, the algorithm "learns" to identify patterns that it can then recognize in novel data. Even then, the algorithm may, if provided the capacity to, improve itself by learning from feedback loops that inform it of cases which were mistakenly identified.

In the opening illustration, with Beverly's smart watch not recognizing her voice, the ML algorithm embedded within the device might take an initial sample of her speech patterns, but whenever Beverly must repeat herself, the algorithm identifies that as a mistaken interpretation and improves itself for future interactions. The watch may be artificially intelligent, but it is machine learning that allows the watch to improve over time. Another example of ML in use today is how Uber uses ML algorithms to predict how long it will take rides (or meals) to arrive, set pricing, and even determine where best to wait to give a driver the best odds of finding you for pick-up (GeekWire, 2016). These algorithms are specific to their purpose and likely trained using data they mined from their own apps. Finally, it is the ML technology behind driverless cars that can quickly adjust to diverse conditions while driving. The continual self-improvement of the algorithm, however, is what makes it truly "intelligent." ML is not the same as AI, but ML does provide AI with the ability to learn and improve.

International Encyclopedia of Education, Fourth Edition, 2023, 83–91

What is machine learning?

Think of Pavlov's dogs. When he would ring a bell, they started to salivate. Pavlov taught them this pattern: every time the bell rings, food is on the way. The dogs soon learned the pattern and eventually began to respond to the bell *before* the food appeared. To cognitive psychologists, human brains are also natural pattern matchers. Since birth, we are constantly trying to find patterns in what we perceive, then associate new patterns with existing patterns in our memory. This pattern-matching process happens automatically. Through our everyday experiences, our brains are constantly collecting new patterns and adding them to memory. These patterns then influence the inferences and decisions we make in the future.

Inspired by an understanding of how the human brain works, researchers in the field of ML draw on connections to psychology, neuroscience, and the learning sciences. These connections are evident in the use of shared or similar terminology (e.g. reinforcement learning). Tom Mitchell (1997) wrote a widely quoted definition of ML: "A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E" (p. 2). Machine learning focuses on developing algorithms that allow machines to improve their performance on a certain task over time through accumulating experiences. Going back to Beverly, her watch implements an ML algorithm to understand her voiced instructions. Although the initial performance is not fully satisfying (i.e. she had to repeat herself several times), the watch becomes better over time as it experiences more instances of Beverly's voice. Modern companies that build these voice-recognition systems often pre-train the algorithm with massive amounts of voice recordings covering a wide range of accents, volumes, and acoustic environments prior to selling them to the customers. The initial accuracy of the watch's voice recognition should be acceptable, but it is further improved by specifically learning its owner's voice.

Approaches to training

Although all ML approaches share the goal of automatically improving performance on a task based on accumulated experiences, they differ largely on the target task, the nature and availability of training data, and the measure of performance. Given the complexity and variability of typical problems addressed with ML, there is no one-size-fits-all approach to training. Although not a comprehensive list, the three largest categories are: supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning

In supervised learning, the ML algorithms are trained using large collections of examples that are *labeled* with the correct "answer." Similar to how one studies for a test by completing endless practice problems, the algorithm "learns" how properly to identify or classify a given input by processing large, labeled datasets until it can do so with an elevated level of confidence—the larger the training set with rich and diverse examples, the more accurate the algorithm.

A simple example of supervised learning involves training an algorithm to differentiate between images of cats and dogs. The training dataset consists of a huge selection of different images showing either a cat or dog, along with its correct label. The algorithm "studies" these examples by identifying patterns within the pixels. Then, when provided with a novel (unlabeled) image, the algorithm will look for similar patterns and provide an answer—although "best guess" might be more appropriate—and a level of confidence. Now, what happens when the algorithm, trained on nothing but cats and dogs, is given an image of a rabbit? There are certainly similarities that could be drawn between rabbits, cats, and dogs, but the algorithm can only choose between the latter two animals. It will have to choose whichever category the rabbit image most closely resembles, but the confidence of the result will likely be low.

The training data are critical to a successful supervised learning approach, but reliance on the training data is also a limitation. If high-quality training datasets (i.e. containing accurate human coders' labels with unbiased and diverse training examples) are not available, it can take hundreds of hours for humans to collect and label a new training set. Moreover, if humans are doing the labeling, they must be very careful as each error will influence a dataset's quality and the overall performance of any ensuing predictive model.

Supervised learning primarily deals with problems of prediction and forecasting. Supervised learning algorithms look at existing data to make an inference about new instances or predict the outcome of a yet-to-be-seen example. And while identifying the species of pet photos on the Internet may be fun, supervised learning can be applied to data beyond images, such as using numerical data for predicting and forecasting. For example, stock prices or real estate trends, due to the potential financial gains and the ample amounts of historical data, are popular real-world examples of supervised learning. The prediction of a single house price, for instance, might use data that include several properties of individual houses in the dataset, called features. Each house might include features describing earlier sales prices, the neighborhood, square footage, number of bedrooms, and so forth. Another algorithm may be trained on patterns leading up to major shifts in the market to predict major trends in the future. Costly errors can be made when an algorithm, meant to predict stock prices, is trained on data that neglect the potential of dramatic swings in the market due to unique events (e.g. a natural disaster or war). Thus, supervised learning algorithms can have powerful predictive and discriminatory capabilities—provided the availability of enough quality training data that is also sufficiently representative of the targeted population.

Unsupervised learning

When well-labeled data is unavailable and the task of creating a labeled dataset is neither physically nor financially feasible, or when one is without preconceived categories (or unsure what to even look for), unsupervised learning may be the ideal approach. The unsupervised approach allows the algorithm to detect patterns that may be too complex for a human to recognize. The algorithm reads the unlabeled data and discovers hidden patterns within.

For instance, the construction of a commercial airliner involves millions of mechanical parts. Each part of the aircraft will have a degree of error in length, weight, or performance due to simple variation in quality control—all pieces meeting a minimum standard, but not perfect. Even though all parts individually may work without perceivable error, in a complex machine like an aircraft, the accumulated error may be catastrophic. Traditionally, technicians must manually conduct multiple systemic evaluations of the plane, which cost tremendous amounts of time and money. Using an unsupervised approach, such as Anomaly Detection, and data regarding the quality of each part, an algorithm can holistically assess the amount of accumulated error and determine to what extent the accumulated errors affect the quality of the plane's airworthiness before the plane even leaves the warehouse.

Reinforcement learning

Both supervised and unsupervised learning are powerful approaches, but sometimes real-world problems that are natural to humans can be too complex or dynamic for machines to label or model (e.g. driving a car on a highway). Compared to supervised learning, where the answers are provided upfront, reinforcement learning tackles a task with parameters for successful completion. One might draw a simplistic analogy comparing supervised learning to rote memorization and reinforcement learning to discovery learning. The algorithm works by making random attempts, each of which is provided immediate feedback, until it can successfully complete the task.

Self-driving cars are a well-known example of reinforcement learning. When training an algorithm to drive a car, one central task is to determine the optimal combination and sequence of gas pedal, brake, and angle of the steering wheel. Instead of labeling the behaviors of 1000 different drivers, reinforcement learning can happen in a *simulated* environment with a wide variety of driving situations.

The driving algorithm is given a straightforward goal: drive from point A to point B without accident and before time runs out. Any behaviors that progress toward meeting the goal are rewarded while the rest are penalized. The algorithm starts with random actions and, for a while, might not even move forward. However, after thousands of random trials, the algorithm begins to use the gas pedal to move forward (not backwards) because moving forward closes the gap between point A and point B, but it may not have learned yet how to use the steering wheel to turn. After running off the road, hitting stationary objects, making U-turns, and another million failed trials, the algorithm finally begins to guide the car to its destination. The cost of reinforcement learning is computational resources and time in simulated environments, which is relatively inexpensive and more efficient compared to crashing cars on a real road or collecting data on even 100 real drivers.

Supervised learning, unsupervised learning, and reinforcement learning are currently the most well-known approaches in ML. Other more complex approaches derive from these basic approaches. For example, semi-supervised learning and self-supervised learning combine both supervised and unsupervised learning techniques, such as using the results from unsupervised learning to inform labeling in supervised learning datasets. Additional ML approaches (e.g. ensemble learning, active learning, transfer learning) strive strategically to improve the performance of the basic approaches. These techniques do not have to be supervised, unsupervised or reinforcement learning, but they do employ strategies to improve the performance of any single model. For instance, ensemble learning combines multiple classifiers and then outputs more robust results compared to simpler approaches. Active learning can involve a human coder to resolve uncertain cases. And transfer learning, in some instances, can use pre-tuned model parameters to shorten the training time or deal with problems with fewer training examples needed.

In summary, ML is an applied field and different approaches are inspired by diverse practical problems across various disciplines. Therefore, these approaches do not necessarily all share the same theoretical foundation and historical root. Each approach to ML has developed a multitude of different algorithms for different purposes. This diversity began during the mid-20th Century and continues to evolve today.

A brief history of machine learning

As early as 1947, the British mathematician Alan Turing envisioned a machine that could, based on an initial set of instructions, modify its own operating instructions over time. The new instructions created by the machine were the result of responses to recently added information. He likened the process to students who have learned basic information from their teachers and went on to develop new knowledge on their own.

> In such a case one would have to admit that the progress of the machine had not been foreseen when its original instructions were put in. It would be like a pupil who had learnt much from his master but had added much more by his own work.
>
> Turing (1995, p. 13).

Turing's idea was too theoretical and abstract to implement in 1947, but today's development of ML is largely inspired by his vision. With modern technology, Turing's ideas have come to fruition.

One of the earliest practical examples of learning machines is often attributed to Arthur Samuel. In the 1950s, Samuel developed a computer program that could learn to play the game Checkers, a strategic board game where two players make diagonal moves and capture pieces by jumping over the opponent. Samuel's program evaluated each position by looking ahead at all the possible moves to determine the best next move. Later, he developed a simple learning method (i.e. rote learning) where, through repetitive self-play, the program memorized optimal moves given a set of conditions observed over thousands of games. Although the algorithms he used were relatively slow and limited by the computer memory at that time, the program was able to improve its performance continuously and reached the level of "better-than-average novice."

Around the same time, researchers came up with the idea for artificial neural networks (ANN) that imitate human neurons to deal with complex computing often involved in machine learning. This idea was initially discussed in the early days of machine learning (e.g. Perceptron invented by Frank Rosenblatt at Cornell Aeronautical Laboratory in 1957). In the 1970s and 1980s, classic structures of ANN were invented (e.g. Convolutional Neural Networks for image recognition and Long Short-Term Memory Recurrent Neural Networks for natural language processing). However, these efforts did not make a substantial impact outside of research labs due to the lack of computational power and limited data at the time. ANN has regained attention in recent years in the form of deep learning (i.e. multi-layered neural networks). Modern deep learning applications range from classic speech recognition and image recognition to complicated music composition, bioinformatics, drug discovery, and so on. Many applications reach the level of "experienced amateur" and some even surpass human performance. To illustrate, from 2015 to 2017, AlphaGo—a neural-network-based AI system—defeated every world champion in a strategic board game called Go (like chess). Originating in ancient China, Go is a vastly more complicated game with 361 potential opening moves compared to Checkers (four moves) or Chess (twenty moves). Brute-force assessment of every possible next move in Go, as is done in simpler algorithms used for Checkers and Chess, is beyond the computational power of any computer in existence. However, with the successful implementation of ML algorithms (e.g. Monte Carlo tree search and deep neural network), AlphaGo can play against itself just as Arthur Samuel envisioned, evolving over time in search of the optimal strategy.

Besides rote learning and artificial neural networks, ML has been influenced by many different disciplines and the field of ML has branched out to multiple lines of research through its history. For example, Email systems now routinely have spam filters to block junk emails. These filters are usually powered by algorithms like Naïve Bayes classifiers, based on Bayesian probabilistic inferences, invented in the 1970s. Later, around the 1990s, models like Support-vector machines (SVM; Cortes and Vapnik, 1995) were introduced to the ML field. They can be used for both regression and classification tasks and are typically used for classification purposes. SVMs provide an alternative, non-probabilistic method for classification compared to Naïve Bayes. Inspired by evolutionary theory, genetic algorithms were also invented around the same time as SVMs (Holland, 1992). Genetic algorithms were once widely used to solve complex and dynamic problems like scheduling optimization for flights and trains and later informed the development of reinforcement learning. The theoretical foundations and data structure of different algorithms may vary, but they all share a similar goal—identifying an optimal solution based on existing and incoming new data. Now, having laid a foundation about ML in general, we next describe how ML can be used in education.

Machine learning in education

At its core, the purpose of ML is to solve problems and offer insight into issues too complex for human analysis, or to automate routines too costly for human labor. Successful implementation of ML algorithms reduces the reliance on human effort by automating complicated or tedious tasks. In education, there are many such tasks that could benefit from ML. For instance, a report published by the Organization for Economic Cooperation and Development (OECD) in 2015 found that teachers spend, on average, about half as much time teaching as they do on non-teaching tasks including lesson planning, grading, administrative work, and collaboration with other teachers. Moreover, teachers struggle to provide equal attention to all students in addition to preparing high quality instructional activities to classrooms of 30+ students. Struggling students are often left alone to grapple with difficult subjects before the class moves on to the next topic. Such unaddressed gaps in learning may plague students for years.

The opportunities for ML to benefit the field of education are bountiful; however, change in education is often slow. Developing and implementing ML-powered systems requires significant financial investment, expertise in both education and machine learning techniques, and extra caution when dealing with vulnerable populations. For these reasons, intelligent machine aids are only slowly making their way into schools worldwide. We now describe some current ML applications that are helping both teachers and students.

Instructor aids

Teachers stand to benefit greatly from ML systems, such as in time saved from completing routine tasks that are easily automated. Such ML-powered systems can be employed to save teachers' time by grading homework assignments and exams that are in essay format. For example, the Educational Testing Service has been using an ML-based system called *c-rater* for years to score participants' essays (Sukkarieh and Blackmore, 2009).

Tools to enable facial recognition have already been shown to be helpful in simple daily tasks like taking attendance (e.g. Okokpujie et al., 2017; Sanli and Ilgen, 2018). Although checking which student is present may take between two to 10 min, depending on the class size, even just 2 min can add up to a total of 6 h across a 180-day school year. These systems, which have been tested in limited classrooms already, record the classroom at certain times throughout the day. Student faces are captured, and the facial recognition software identifies the student and marks them as present.

Attendance may seem trivial, but facial recognition algorithms can also help with the arduous task of noticing students' subtle behaviors during class. ML systems have demonstrated the ability to track facial expressions that can indicate students' engagement. Systems like these can notify teachers when students are not engaged, which in turn could prompt the teacher to change up a lesson, call for a break, or help a struggling student. For example, Sümer et al. (2021) used ML to analyze the audiovisual recordings from a classroom of students over a period of one and half months. They trained their engagement classifier models using two features of engagement (attention and affect). The results were promising. That is, their engagement classifier achieved an acceptable level of accuracy, which is notable because this study was the first in this area (i.e. multimodal engagement analysis in a physical classroom) and because of the relatively small amount of student-specific data used. Results from this type of research can help teachers identify whether their students are engaged or not so they can modify their teaching (e.g. slow down and elaborate, ask a thought-provoking question), as needed. Granted, facial recognition in classrooms raises issues of privacy and potential abuse. We address those concerns later.

Identifying real-time indicators of engagement is one benefit of ML, but long-term data collection can yield even more valuable insights. For example, researchers have used multiple ML algorithms to predict high-school students' risk of dropping-out based on records of coursework, attendance, and performance (Coleman et al., 2019). By recognizing patterns that lead to dropping out of courses, teachers can improve retention rates by offering interventions before the student has even seriously considered quitting, such as suggesting a student come for extra help after school prior to an important test. Moreover, ML has been used to model complicated socio-cognitive processes such as knowledge sharing and negotiation in computed-supported collaborative problem-solving contexts (Stewart et al., 2019). In general, receiving real-time data—especially that which has been accumulated over time—can provide recommendations so that teachers are able to make more informed decisions on subsequent learning activities. But ML can also benefit students directly.

Student aids

In addition to supporting teachers, ML has incredible potential for enhancing students' learning experiences. One early vision for implementing ML was to develop intelligent tutoring systems which were meant to emulate the role of a human teacher and respond to students thoughtfully as an expert tutor (Self, 1990; Shute and Zapata-Rivera, 2010). In this case, ML can be used to (a) identify students' learning processes and behaviors, (b) diagnostically assess their weaknesses and strengths, and (c) provide individualized cognitive and affective supports as needed. As part of interacting within such digital learning environments, students demonstrate observable behaviors relative to targeted knowledge and skills. These behaviors could then be collected as information traces and used to train ML models, for example, identifying behavioral patterns that are successful and those that are not for various outcome measures. The models can then infer students' evolving knowledge and skill levels based on an accumulation of observed behaviors. Using these inferences, the intelligent learning environment can recommend the most appropriate learning supports or subsequent content to optimize students' learning experience. Diagnostic information and intervention recommendations can also be shared with the teacher.

Recently, the National Science Foundation granted $20 M to an AI-institution project led by the University of Colorado, Boulder (Strain, 2020). This five-year project is targeted at middle- and high-school students and aims to improve teamwork. In the classroom envisioned by the project, ML algorithms will be used to recognize who is speaking, what is being said, and then react in real-time (e.g. provide feedback) as needed. The data collected over time will be used to create individual models of each student that can be used for formative and summative assessments. ML and similar techniques comprise the building blocks of such complex AI-based classrooms.

The costs of designing, developing, and testing intelligent systems are high, thus making them cost-prohibitive for most schools. However, many of the functions of such systems can be emulated, with the help of ML, by more familiar environments, like educational games. For instance, *Physics Playground* (Shute et al., 2019) is an educational game that incorporates stealth assessment (Shute, 2011) to estimate students' conceptual physics understanding in real-time. Stealth assessment is embedded into the game code to continually track a player's progress and support learning during gameplay using an automated statistical scoring and accumulation approach (Bayesian inference networks). Moreover, the game includes a quit-prediction model. This model can predict if a student is frustrated and is likely to quit playing. The game then can use this information (i.e. the probability of quitting for each student at any given time) to provide appropriate supports (e.g. affective supports such as motivational messages) to encourage persistence in solving game levels. The quit-prediction model uses 37 variables, also known as features, to determine the probability of a student quitting. These features were extracted from student-generated log files through an iterative ML process called feature engineering (Shute et al., 2020; Slater et al., 2017a,b).

Thus far, we have looked at how a machine can assess and support student learning, both of which are dependent on ML approaches that focus on pattern recognition. Reinforcement learning provides another benefit that goes beyond monitoring and reporting on students' knowledge and skill levels. That is, reinforcement learning algorithms can provide individualized content and experiences that surpass what a single teacher could achieve. The ITSs mentioned earlier can reach a new level of sophistication when paired with reinforcement learning. That is, earlier ITSs were dependent on rule-based, manually coded, and generalized responses to student interactions. The manual approach is limited by how many rules the designer can think of and integrate into the system. Using reinforcement learning, the ITS can adapt to learners' behaviors and select the optimum next step for learners based on their prior performance (Chi et al., 2011). Every student receives a unique learning experience, based on the insights gathered from the ML-based systems.

Taking ML's ability to generate tailored content in another direction via reinforcement learning to train and build simulated environments, individual students may interact with these environments to experiment and learn. What could only be explained with diagrams and hypotheticals can now be demonstrated in real time (or in accelerated time for slow-moving processes) as the student observes. Economic markets can be generated that mimic modern-day markets (e.g. Zheng et al., 2020). Students can adjust parameters to see how simple changes can have massive impacts. Tiny creatures can be modeled, created by students, and left to live out the perils of natural selection as students observe how they survive (or not) over long periods of time. Not only can students observe these complicated systems that have such powerful influences on our lives, but they can also create their own systems and see how they respond to different inputs. This is putting the world in students' hands and letting discovery learning take place on a whole new scale.

What does the future hold for ML in education? We believe that the influence of ML will continue to grow in educational systems around the world. Following are some ideas and examples of what we expect to see happen with ML and education in the future.

Future of ML in education

Although modern programming tools and programming libraries already make the workflow of structuring ML projects easier than before (Slater et al., 2017a,b), ML is far from accessible to the layperson. We do expect to see that, in the future, ML will become a fairly ubiquitous influence in the educational world by making insights and answers readily available to all stakeholders. For instance, educators will be able to actively use ML tools to support instructional decisions and students can use ML to diagnose their own performance without going through hundreds of hours of training only to get a single score. This accessible ML future first relies on the advancement of technology (e.g. increases in computational power). That is, although the main theoretical foundations of ML were established in the last century, applications in this field were largely restricted by the costs and limits of technology. Recent advances in computing—like quantum computing, Optical Neural Networks, and distributed computing—continue to increase computing performance and lower cost. Before long, computing resources required for more advanced ML applications, rather than being a luxury of well-funded computing labs, will be within the budgets of local school systems—making the benefits of ML more prevalent in education.

In addition to rising power and decreasing costs, this envisioned future depends on raising the bar for "basic" computer literacy. Just as the ability to read was once a luxury of the upper class, and not long-ago creating content on the Internet was limited to only the technically savvy, the minimum threshold for what is considered "basic knowledge" of statistics, computer science, and engineering practices (e.g. simulation and modeling) must be raised. A transformation in society's computing literacy will take time, but we are optimistic. Again, just 30 years ago, when the Internet was young, creating a simple website required advanced programming skills. But today, anyone can create sophisticated and beautiful websites without any programming knowledge. As ML technology continues to increase in availability and become more prevalent in our everyday lives, we fully expect ML to become a basic programming skill and, one day, be a part of our children's core science curricula.

As more and more educational activities happen online, the data available to ML tools used in education will grow exponentially. The increase of data volume is predictable, and we expect the future will include the fusion of dynamic and diverse data being used for ML in education. ML will not have to focus on one model or source of data. Instead, data will come from many sources and channels. Students' academic performances (numerical data), facial expressions (visual data), classroom discourse with peers and instructors (verbal data), and engagement levels (physiological data) can all be merged to provide a holistic and more valid view of students' performance and understanding. We will be able automatically to identify that a student's difficulty in an English class is not because of language ability but due to test anxiety based on affect detection and other physiological data. In addition, data fusion can synthesize the conclusions from multiple sites (e.g. different school districts), which prevent the sharing of individual levels of identifiable digital traces. Ultimately, we can build up a comprehensive and multi-faceted data warehouse to benchmark educational ML performances and inform research and practices.

In the more distant future, we may see analytical systems, powered by ML, help individuals self-actualize by identifying dispositions and paths needed to achieve short- and long-term goals. Such ML-based systems would make the learning process more efficient, self-directed, individualized, and life-long. Through advanced personalized-learning environments, students could pursue their own interests instead of being constrained by one-size-fits-all curricula. Apart from individual support, ML-powered systems could be employed at the group level, for data-driven decision-making purposes. For instance, governments can use school, district, city, and state data to create policies and practices that can dramatically improve education quality with high precision. In this scenario, the unit of analysis is not a student; rather, it is a larger entity made up of students, teachers, educators, parents, administrators, and other stakeholders. Data from all these sources could be used to develop ML-based models to inform policy makers' decisions. Using ML algorithms, governments can build educational success models, starting with feature engineering (i.e. identifying all the important variables predictive of an outcome).

To work effectively, now and in the future, ML requires copious amounts of data, such as online accounts, learning management systems, and school databases. How to balance the use of all these data while protecting students' privacy is a concern as we move forward. The ethical issues of mass data collection and the overall use of ML and AI are current global concerns, but when dealing with vulnerable populations like young students, extra caution must be taken before ML can fully take off in education. Like any major advancement, the future of machine learning, and its benefits, will face hurdles we must surmount.

Concerns moving forward

As we move toward this new world of "smart" machines, the ethics of ML must remain at the forefront of progress. In general, computer scientists, researchers, educators, and educational policy makers who generate, put in use, and advance the technologies that are powered by ML must be mindful of the controversies to prevent their products from doing harm (Garrett et al., 2020). For instance, meticulous care must be given when both creating and training ML algorithms to ensure they remain unbiased. A misconception about AI and ML algorithms is that they are inherently unbiased because humans are not involved. However, humans have biases and humans are the ones writing the algorithms. Even well-intentioned people may possess implicit biases that unintentionally influence their designs (Bertrand and Mullainathan, 2004). More importantly, ML algorithms are trained on massive datasets that reflect the reality of human bias. Therefore, the algorithms can inherit those biases—e.g. racial or gender bias. Beyond that, human judgment is always involved in what is deemed worthy of being modeled by ML, and what is not. This may seem like a small issue, but even the underrepresentation of a single population can have a large impact during application. Consider the case of facial recognition algorithms powered by ML. Recent studies have shown that the accuracy of facial recognition software recognizing "darker" faces was lower compared to the accuracy of recognizing "lighter" faces (e.g. Buolamwini and Gebru, 2018). The reason for this discrepancy was due to the large number of lighter-skinned faces (i.e. about 85%) existing in the databases used to train the ML algorithm. Earlier, we provided two examples of using ML-powered facial recognition in classrooms (one for taking attendance and the other for building an engagement classifier). Such racial biases can sabotage those applications of ML in classrooms by mistakenly marking students of color as absent, or continually misidentifying a student's engagement due to idiosyncrasies associated with cultural differences.

ML algorithms can also be biased relative to gender. For example, Caliskan et al. (2017) found that natural language processing algorithms can inherit gender biases—either from the data that was used to train the algorithms or from ignoring such biases by the programmers. For instance, the word *nurse* is more frequently associated with the word *she* than with the word *he*. A biased model can perpetuate biases in those who interact with the biased environment; even the results of modern image searches can portray gender biases related to occupations (Kay et al., 2015). Allowing these skewed perspectives into the classroom can impair the struggle to reduce gender inequality.

There are some things we can do to reduce bias in data-driven, automated ML approaches. Ntoutsi et al. (2020) have suggested that we need to: (1) Understand bias relative to how it occurs in society and how it finds its way into the data used for training ML algorithms so that we can proactively address biases when creating algorithms and collecting data; and (2) Mitigate bias in each step of the ML decision making process. For example, we need to ensure that the ML algorithms are adequately trained on data from minority populations (i.e. gender- and race-balanced datasets) so that datasets provide a valid representation of all populations. Moreover, several researchers (e.g. Garrett et al., 2020; Saltz et al., 2019) have noted that it is in everyone's best interest to teach ethics in AI as part of computer science courses to make sure that future computer scientists (and others who will be writing the algorithms) are aware of ethical issues. But above all else, those who implement ML algorithms must retain "transparent classifiers which are interpretable on their own and exhibit predictive accuracy close to that of obscure models" (Ntoutsi et al., 2020, p. 8). Aggressive transparency, much like peer-review, helps establish a norm for public auditing of the algorithms that so intimately influence everyone's lives. And although there are ways to reduce the biases ML inherits from humans, we note that ML is not a panacea to all problems, particularly in educational systems. Thoughtful decisions about what to model (and why) need to be made prior to training ML algorithms.

Security (i.e. protection from hackers) and privacy (i.e. data anonymization) are also issues that need proper attention to ensure a successful future with ML (e.g. Mittelstadt and Floridi, 2016). When personal data on an individual are stored somewhere, people with harmful intentions may access the data (e.g. identity theft). Therefore, it is important to know how the data are stored, who owns the data, and how secure the data are. Regarding privacy, a quick resolution is to remove any identifiable information from the data, but this solution greatly diminishes an algorithm's ability to function as even just one's date of birth and zip code are enough to identify them (Mittelstadt and Floridi, 2016).

As mentioned previously, when facial recognition technology is used in classrooms and on campuses, concerns of privacy and abuse are quick to surface. However, we believe this trend is evident in the literature and the continued integration of ML (e.g. facial recognition) is a double-edged sword, offering great benefits alongside meaningful risks. If quick resolutions are non-solutions, then policies and legislature must ensure the security of students; for example, recommendations tend toward transparency and the ability to "opt-out" of having one's data collected. In short, legislation such as the General Data Protection Regulation passed in Europe and similar legislation passed in California possesses a spirit of security, privacy, and fairness that must extend to some extent to academia.

Finally, because new technologies appearing in the future may be more work and require additional learning for teachers who are already overwhelmed, we need research and development on user-friendly, easy-to-use tools that can attract teachers to use the tools. Moreover, professional education is needed to help teachers understand the strengths and weaknesses of the new tools. Any implementation of ML-powered systems in schools must be accompanied by comprehensive training for teachers to ensure they are fully in command of the power ML can give them, and aware of potential ethical issues in using them.

Conclusion

We first explained what AI and ML mean and how they relate to each other. Then, we focused on machine learning, its definition, history, and use in education, now and in the future. We concluded with a discussion of relevant concerns and issues related to the use of ML, especially in education, moving forward. What makes us confident that the future of education is bright is the amount of progress that has been made in the past several decades in terms of technology, learning sciences, computer science, psychology, psychometrics, and so on. No matter how far we eventually go in using ML in education, we need to remember that nothing can replace human-to-human and particularly teacher-student interactions.

The goal for ML should be to help—not replace—humans in classrooms. The algorithms that improve our daily online experiences can become extensions of the instructor, the administrator, and policymaker. ML can bring to one's fingertips answers to questions that just ten years ago required teams of researchers and decades of experience in the classroom. ML can automate and energize the classroom to the extent that the teacher is free to facilitate learning with a level of precision previously unattainable. Yes, caution must be taken, but with a careful hand, a watchful eye, and forethought informed by history, it is possible that we can press forward, propelled by technology, into a brave new world of education.

Acknowledgment

We would like to thank Ryan Baker and Angela Stewart for reviewing a draft of this paper.

References

Bertrand, M., Mullainathan, S., 2004. Are Emily and Greg more employable than Lakisha and Jamal? A field experiment on labor market discrimination. Am. Econ. Rev. 94 (4), 991–1013.

Buolamwini, J., Gebru, T., 2018. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. Conference on Fairness, Accountability, and Transparency. PMLR, pp. 77–91.

Caliskan, A., Bryson, J.J., Narayanan, A., 2017. Semantics derived automatically from language corpora contain human-like biases. Science 356 (6334), 183–186.

Chi, M., VanLehn, K., Litman, D., Jordan, P., 2011. Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. User Model. User Adapted Interact. 21, 137–180. https://doi.org/10.1007/s11257-010-9093-1. Available at.

Coleman, C., Baker, R.S., Stephenson, S., 2019. Proceedings of the 12th International Conference on Educational Data Mining July 2019, Montreal [online]. In: Lynch, C.F., Merceron, A., Desmarais, M., Nkambou, R. (Eds.), A Better Cold-Start for Early Prediction of Student at-Risk Status in New School Districts, pp. 732–737. Available from. https://eric.ed.gov/?id=ED599170.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297. https://doi.org/10.1007/BF00994018.

Garrett, N., Beard, N., Fiesler, C., 2020. More Than "If Time Allows" the Role of Ethics in AI Education. Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, pp. 272–278.

GeekWire, (2016). https://www.geekwire.com/2016/uber-collapse-without-pattern-finding-computers-says-chief-machine-learning/.

Holland, J.H., 1992. Genetic algorithms. Sci. Am. 267 (1), 66–73.

Kay, M., Matuszek, C., Munson, S.A., 2015. Unequal Representation and Gender Stereotypes in Image Search Results for Occupations. CHI'15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems April 2015, Seoul [online], pp. 3819–3828.

Mitchell, T.M., 1997. Machine Learning. McGraw Hill.

Mittelstadt, B.D., Floridi, L., 2016. The ethics of big data: current and foreseeable issues in biomedical contexts. In: The Ethics of Biomedical Big Data, pp. 445–480.

Ntoutsi, E., Fafalios, P., Gadiraju, U., Iosifidis, V., Nejdl, W., Vidal, M.E., et al., 2020. Bias in data-driven artificial intelligence systems—an introductory survey. Wiley Interdiscip. Rev. 10 (3), e1356.

Okokpujie, K., Noma-Osaghae, E., John, S., Grace, K.A., Okokpujie, I., 2017. A Face Recognition Attendance System with GSM Notification, 2017 IEEE 3rd International Conference on Electro-Technology for National Development (NIGERCON). IEEE, pp. 239–244.

Saltz, J., Skirpan, M., Fiesler, C., Gorelick, M., Yeh, T., Heckman, R., et al., 2019. Integrating ethics within machine learning courses. J. Educ. Resour. Comput. 19 (4), 1–26.

Sanli, O., Ilgen, B., 2018. Face Detection and Recognition for Automatic Attendance System. Proceedings of SAI Intelligent Systems Conference. Springer, Cham, pp. 237–245.

Self, J.A., 1990. The defining characteristics of intelligent tutoring systems research: ITSs care, precisely. Int. J. Artif. Intell. Educ. 10, 350–364.

Shute, V.J., Zapata-Rivera, D., 2010. Intelligent systems. In: Baker, E., Peterson, P., McGaw, B. (Eds.), Third Edition of the International Encyclopedia of Education. Elsevier Publishers, Oxford, UK, pp. 75–80.

Shute, V., Almond, R., Rahimi, S., 2019. Physics Playground (1.3) [Computer Software]. https://pluto.coe.fsu.edu/ppteam/pp-links/.

Shute, V.J., Rahimi, S., Smith, G., Ke, F., Almond, R., Dai, C.-P., Kamikabeya, R., Liu, Z., Yang, X., Sun, C., 2020. Maximizing learning without sacrificing the fun: stealth assessment, adaptivity, and learning supports in Physics Playground. J. Comput. Assist. Learn. 37, 127–141. https://doi.org/10.1111/jcal.12473.

Shute, V.J., 2011. Stealth assessment in computer-based games to support learning. In: Tobias, S., Fletcher, J.D. (Eds.), Computer Games and Instruction. Information Age Publishers, pp. 503–524.

Slater, S., Bowers, A., Kai, S., Shute, V.J., 2017a. A Typology of Players in the Game Physics Playground. Proceedings of the Digital Games Research Association Conference, Melbourne, Australia, pp. 1–12. http://www.digra.org/digital-library/publications/a-typology-of-players-in-the-game-physics-playground/.

Slater, S., Joksimović, S., Kovanovic, V., Baker, R.S., Gasevic, D., 2017b. Tools for educational data mining: a review. J. Educ. Behav. Stat. 42 (1), 85–106.

Stanford University, (n.d.). Artificial Intelligence, [online]. Available from. http://jmc.stanford.edu/artificial-intelligence/index.html.
Stewart, A.E.B., Vrzakova, H., Sun, C., Yonehiro, J., Stone, C.A., Duran, N.D., Shute, V., D'Mello, S.K., 2019. Proceedings of the ACM on Human-Computer Interaction. I Say, You Say, We Say: Using Spoken Language to Model Socio-cognitive Processes During Computer-Supported Collaborative Problem Solving, vol. 3. CSCW, pp. 1–19. https://doi.org/10.1145/3359296.
Strain, D., 2020. New $20 Million Center to Bring Artificial Intelligence Into the Classroom. CU Boulder Today. https://www.colorado.edu/today/ai-education.
Sukkarieh, J.Z., Blackmore, J., 2009. c-Rater: automatic content scoring for short constructed responses. In: Lane, H.C., Guesgen, H.W. (Eds.), Proceedings of the Twenty-Second International Florida Artificial Intelligence Research Society Conference. Association for the Advancement of Artificial Intelligence Press, Menlo Park, CA, pp. 290–295.
Sümer, Ö., Goldberg, P., D'Mello, S., Gerjets, P., Trautwein, U., Kasneci, E., 2021. Multimodal engagement analysis from facial videos in the classroom. IEEE Trans. Affect. Comput. https://doi.org/10.1109/TAFFC.2021.3127692.
Turing, A.M., 1995. Lecture to the London Mathematical Society on 20 February 1947. M Comput. 12, 390–404.
Zheng, S., Trott, A., Sunil, S., Nikhil, N., Melvin, G., Parkes, D.C., Socher, R., 2020. The AI economist: improving equality and productivity with AI-driven tax policies. ArXiv 1–46. Available from. https://ui.adsabs.harvard.edu/abs/2020arXiv200413332Z.