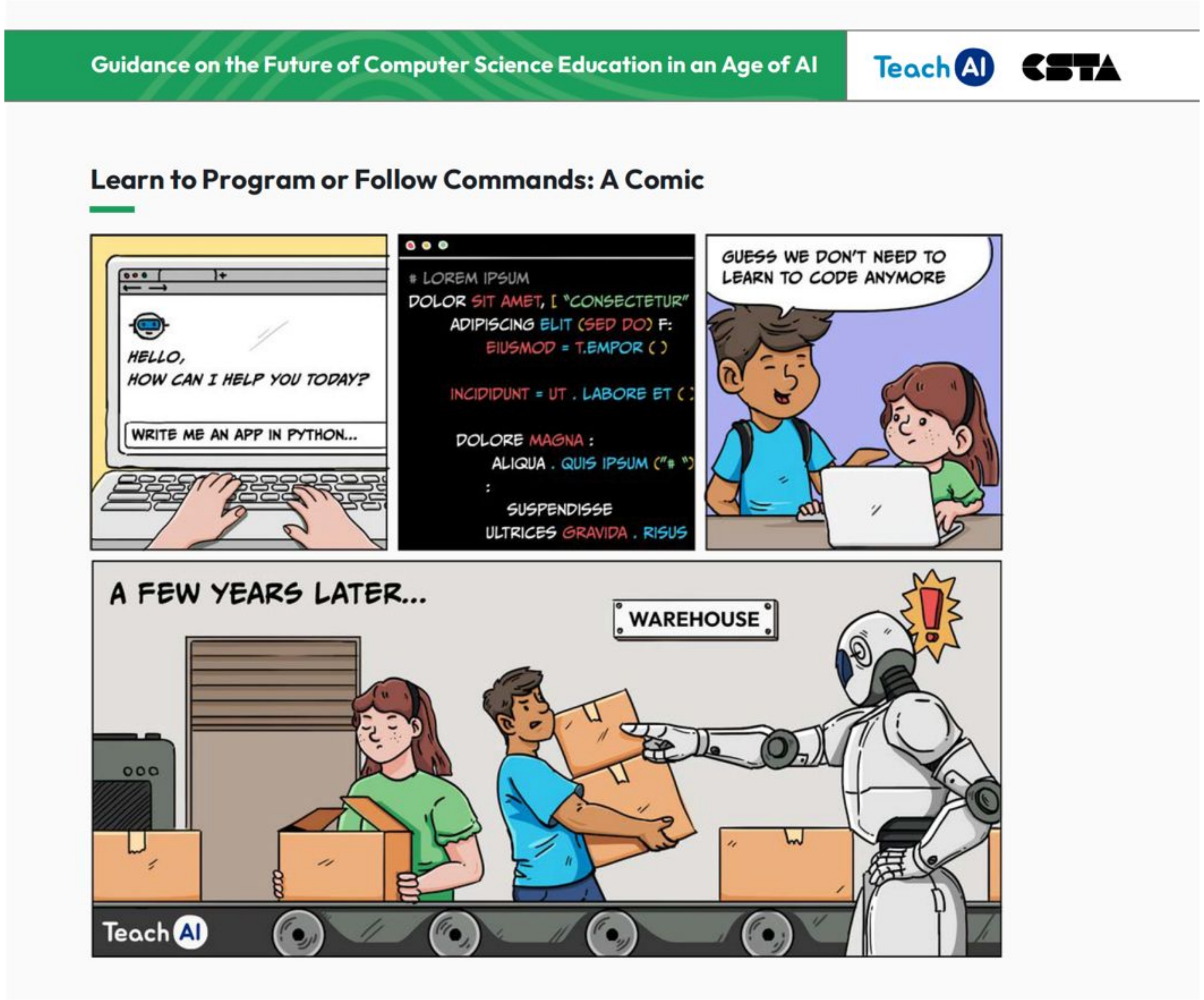


(Embodied Learning for) AI Literacy

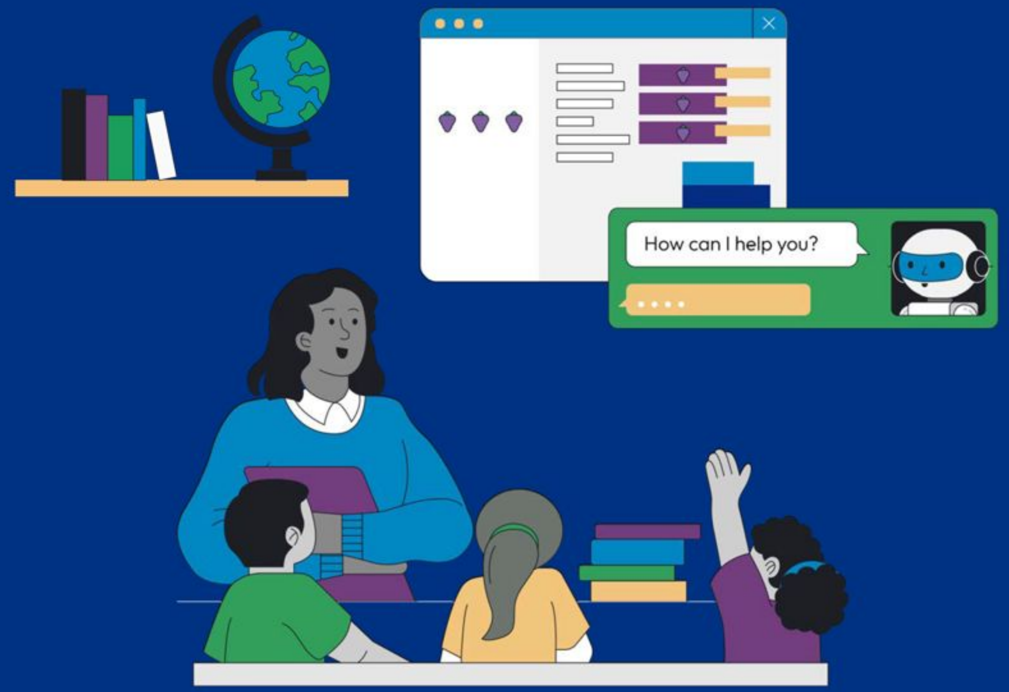
Zhen Bai, Oct 10th, 2024

AI Horizon Learning Series Talk

What do we mean when we talk about
AI Literacy?

(K-12, Higher Ed, Everybody)

Guidance on the Future of Computer Science Education in an Age of AI



teachai.org/cs

New Guidance from TeachAI and CSTA Emphasizes Computer Science Education More Important than Ever in an Age of AI

Voice — The voice of K-12 computer science education and its educators

Posted by CSTA on July 16, 2024

CS Research | CSTA News

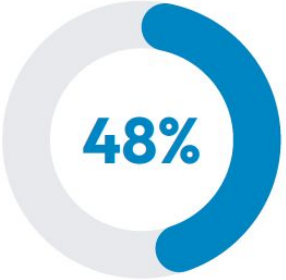
Table of Contents

How Are Computer Science Educators Teaching With and About AI?

Survey Highlights

Guidance on Computer Science Education in an Age of AI is informed by a survey of CS teachers (n = 364 teachers, 24% primary, 76% secondary, 12% international) administered by the CSTA and TeachAI in May 2024.

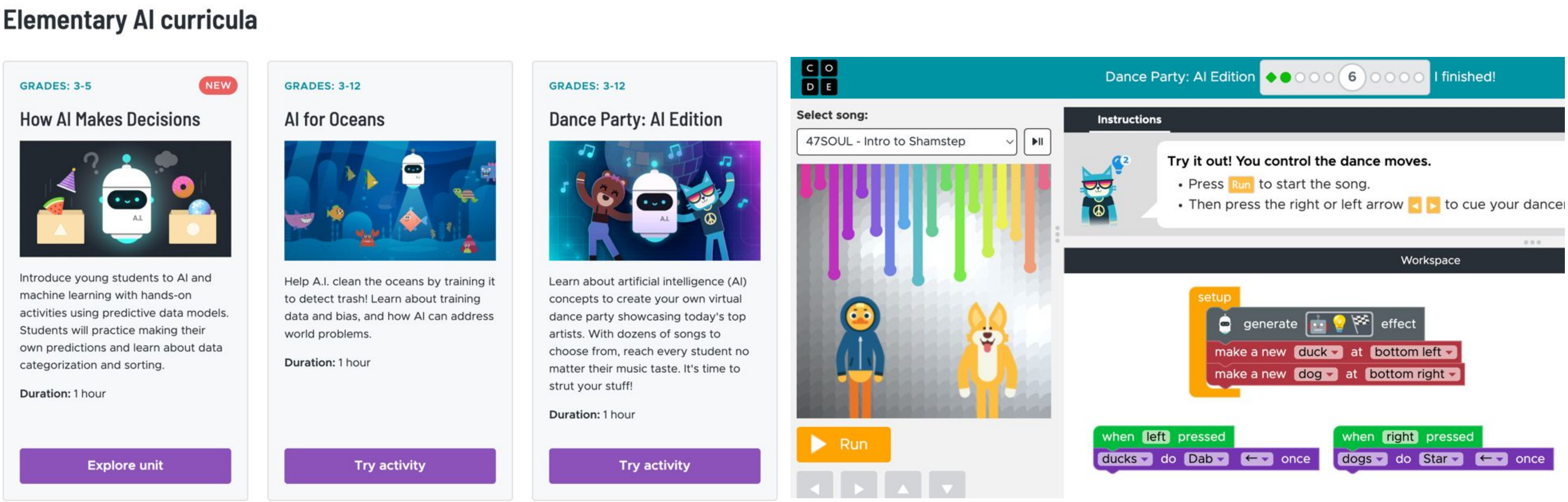
of teachers think students in introductory courses should learn about AI.



of teachers said they feel equipped to teach about AI.

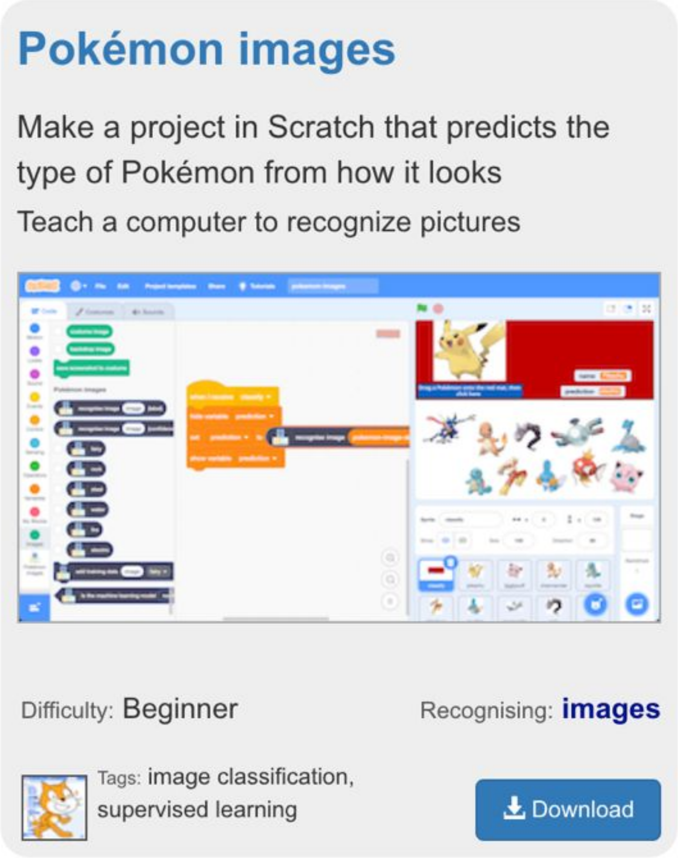
of teachers said they would benefit from professional development to learn how to use and teach about AI.

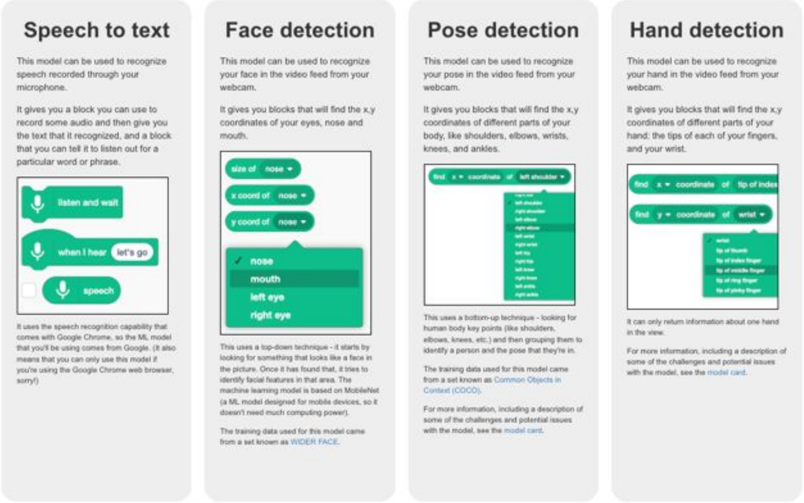
Popular K-12 AI Education Platforms

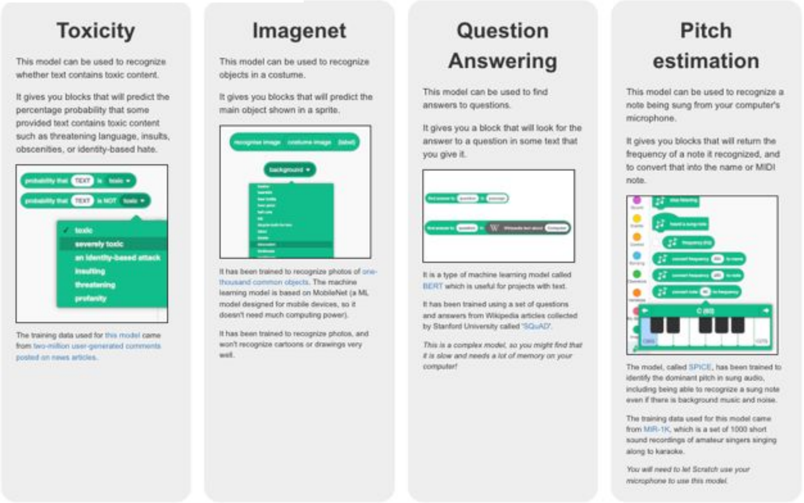


https://code.org/ai

Popular K-12 AI Education Platforms

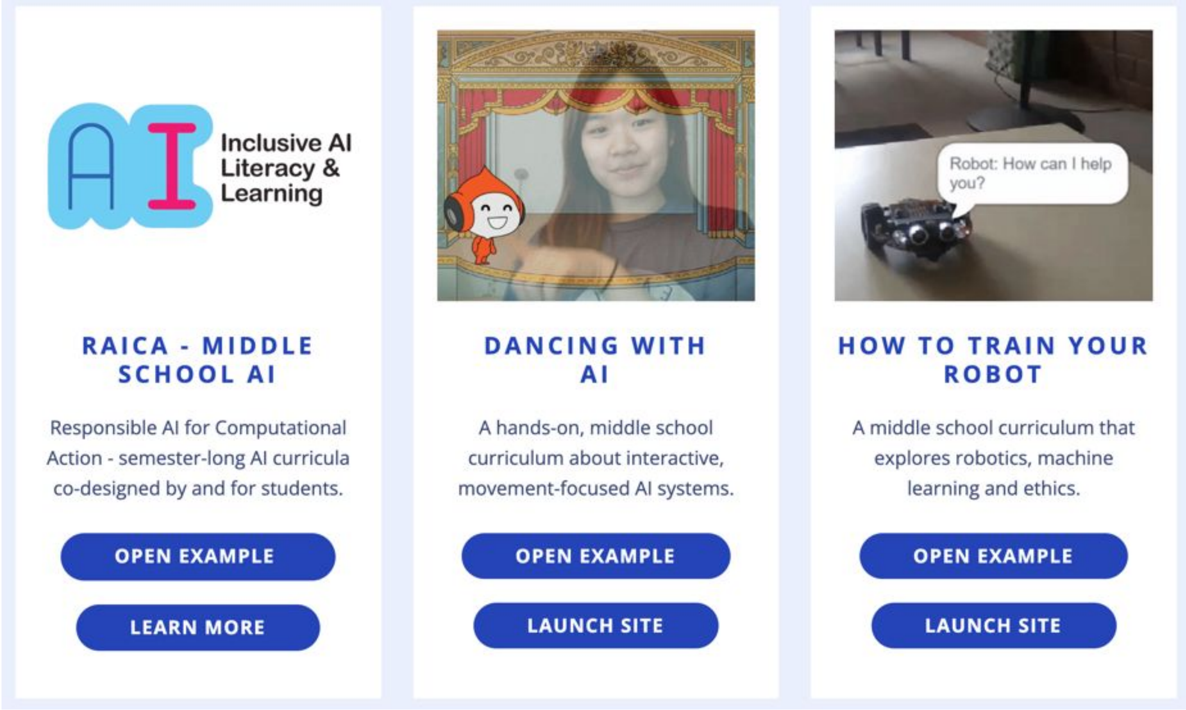






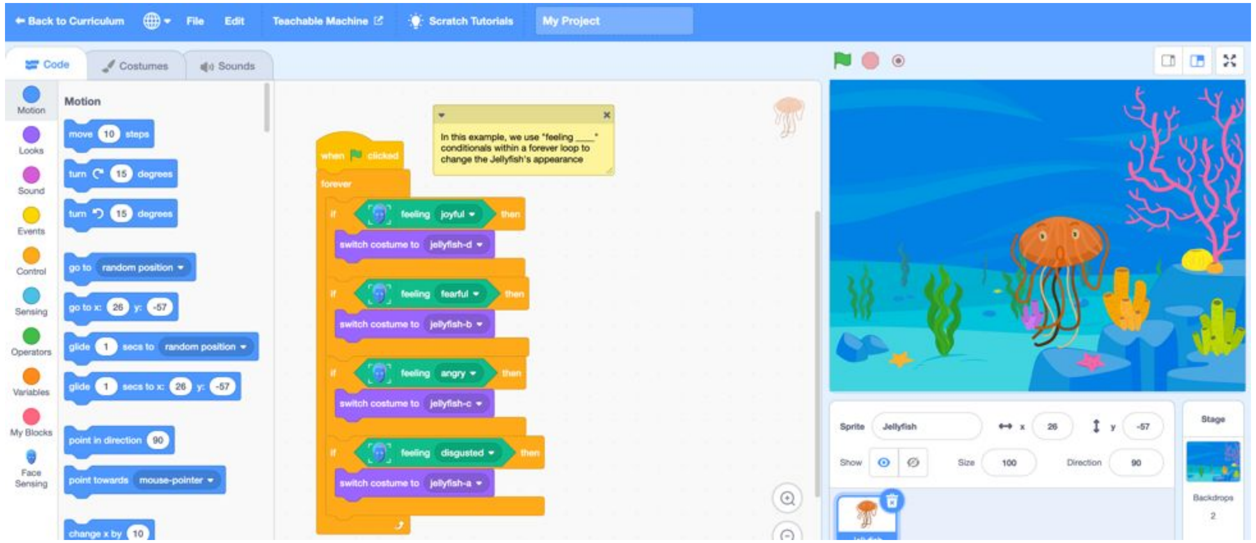
Machine Learning For Kids

Popular K-12 AI Education Platforms



We currently support

- Image classification
- Text classification
- Natural language processing
- Reinforcement learning
- Music generation
- Affective computing
- Gesture recognition
- Social robotics
- Microcontroller robotics
- ...and more!



MIT RAISE

Computers and Education: Artificial Intelligence 6 (2024) 100211

Contents lists available at ScienceDirect

Computers and Education: Artificial Intelligence

journal homepage: www.sciencedirect.com/journal/computers-and-education-artificial-intelligence

A systematic review of AI education in K-12 classrooms from 2018 to 2023: Topics, strategies, and learning outcomes

Sang Joon Lee [a,*], Kyungbin Kwon [b]

[a] Industrial Technology, Instructional Design, and Community College Leadership, Mississippi State University, Box 9730, Mississippi State, MS, USA

[b] Learning, Design, and Adult Education, Indiana University, 201 N. Rose Ave., Bloomington, IN, USA

ARTICLE INFO

Keywords:
Artificial intelligence
AI education
Systematic review
K-12

ABSTRACT

AI education aims to teach AI concepts, essential knowledge, and skills related to the fundamental ideas in AI. As AI becomes increasingly prevalent in our daily lives, schools and educators have started to recognize the importance of AI education in K-12 schools. However, there have been a limited number of studies reporting on the implementation of AI education in classrooms. This systematic review aimed to provide an overview of the current state of AI education in K-12 schools, exploring topics, instructional approaches, and learning outcomes. Twenty-five peer-reviewed journal articles published between 2018 and 2023 were selected for this systematic review. The findings highlighted that various topics were covered in K-12 AI education, including fundamental AI concepts, different types of AI, AI applications, and ethical considerations related to AI. To facilitate meaningful learning experiences, educators frequently integrated hands-on activities and project-based learning. The findings supported the benefits of AI education in enhancing students' AI literacy, problem-solving skills, and ethical reflections on AI's societal impact. Furthermore, it fostered motivation, positive attitudes toward AI, and an interest in technology while inspiring career aspirations. It is recommended to develop tailored AI curricula, instructional strategies, and appropriate tools and resources that seamlessly integrate into various subjects within the standard school curriculum.

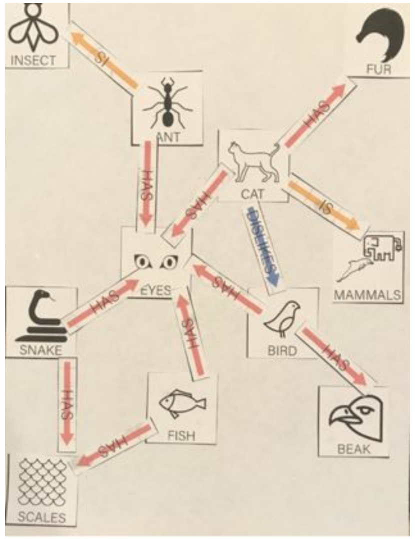
Table 4
Sample tools used in AI education.

Programming languages and environment	Machine learning and AI platforms	Educational platforms and tools
• Python (Jagannathan and Komives, 2019; Oskotsky et al., 2022; Tsai et al., 2022) • Jupyter Notebook (Chiu et al., 2022; Kaspersen et al., 2022) • Anaconda Python (Monteith et al., 2022) • Blockly (Chiu et al., 2022) • Scratch (Alonso, 2020; Estevez et al., 2019; Fernandez Martinez et al., 2021; Jang et al., 2022; Shamir and Levin, 2021, Shamir and Levin, 2022; Xia et al., 2022) • Google CoLab (Oskotsky et al., 2022)	• AWS (Jagannathan and Komives, 2019) • Scikit Learn (Jagannathan and Komives, 2019) • Cognimates (Fernandez Martinez et al., 2021) • Machine Learning for Kids platform (Shamir and Levin, 2022) • Google's Teachable Machine AI training primer (Monteith et al., 2022) • TensorFlow/Keras (Aung et al., 2022; Monteith et al., 2022) • Magenta (Monteith et al., 2022) • OpenCV (Monteith et al., 2022) • Google's Teachable Machine 2 (Vartiainen et al., 2021) • Nvidia's GauGan (Monteith et al., 2022) • Single-Neuron toolkit (Shamir and Levin, 2022) • ExpliClas (Alonso, 2020) • Quickdraw (Ng et al., 2022) • AI modeling games (Ng and Chu, 2021) • AI for oceans (Shamir and Levin, 2021) • VotestratesML (Kaspersen et al., 2022) • IBM Watson engine (Shamir and Levin, 2022)	• Edmodo (Ng and Chu, 2021) • Code.org (Ng and Chu, 2021; Ng et al., 2022; Shamir and Levin, 2021, Shamir and Levin, 2022) • Fischer Technik kits (Marrone et al., 2022) • Mitsuku website (Shamir and Levin, 2021, Shamir and Levin, 2022) • StoryQ (Jiang et al., 2022, Jiang et al., 2023) • Storyjumper (Ng et al., 2022) • AIThaiGen (Aung et al., 2022) • School-Book (Ali et al., 2021) • Micro:bit (Park and Kwon, 2023; Wu and Yang, 2022) • Teachable machine, AIY Voice Kit from Google, Huskylen, or drone (Xia et al., 2022)

- Over ***75%*** of K-12 schools in the US do not offer Computer Science (CS) curriculum including programming (Wang et al., 2016).
- Young children of ***low SES*** experience more difficulty in understanding AI concepts than high SES peers due to lack of programming skills and experience interacting with AI technologies (Druga et al., 2019).
- Young novice programmers with greater programming skills benefits more from using code generator tools (Kazemitabaar et al., 2023).

Table 4
Sample tools used in AI education.

Programming languages and environment	Machine learning and AI platforms	Educational platforms and tools
• Python (Jagannathan and Komives, 2019; Oskotsky et al., 2022; Tsai et al., 2022) • Jupyter Notebook (Chiu et al., 2022; Kaspersen et al., 2022) • Anaconda Python (Monteith et al., 2022) • Blockly (Chiu et al., 2022) • Scratch (Alonso, 2020; Estevez et al., 2019; Fernandez Martinez et al., 2021; Jang et al., 2022; Shamir and Levin, 2021, Shamir and Levin, 2022; Xia et al., 2022) • Google CoLab (Oskotsky et al., 2022)	• AWS (Jagannathan and Komives, 2019) • Scikit Learn (Jagannathan and Komives, 2019) • Cognimates (Fernandez Martinez et al., 2021) • Machine Learning for Kids platform (Shamir and Levin, 2022) • Google's Teachable Machine AI training primer (Monteith et al., 2022) • TensorFlow/Keras (Aung et al., 2022; Monteith et al., 2022) • Magenta (Monteith et al., 2022) • OpenCV (Monteith et al., 2022) • Google's Teachable Machine 2 (Vartiainen et al., 2021) • Nvidia's GauGan (Monteith et al., 2022) • Single-Neuron toolkit (Shamir and Levin, 2022) • ExpliClas (Alonso, 2020) • Quickdraw (Ng et al., 2022) • AI modeling games (Ng and Chu, 2021) • AI for oceans (Shamir and Levin, 2021) • VotestratesML (Kaspersen et al., 2022) • IBM Watson engine (Shamir and Levin, 2022)	• Edmodo (Ng and Chu, 2021) • Code.org (Ng and Chu, 2021; Ng et al., 2022; Shamir and Levin, 2021, Shamir and Levin, 2022) • Fischer Technik kits (Marrone et al., 2022) • Mitsuku website (Shamir and Levin, 2021, Shamir and Levin, 2022) • StoryQ (Jiang et al., 2022, Jiang et al., 2023) • Storyjumper (Ng et al., 2022) • AIThaiGen (Aung et al., 2022) • School-Book (Ali et al., 2021) • Micro:bit (Park and Kwon, 2023; Wu and Yang, 2022) • Teachable machine, AIY Voice Kit from Google, Huskylen, or drone (Xia et al., 2022)



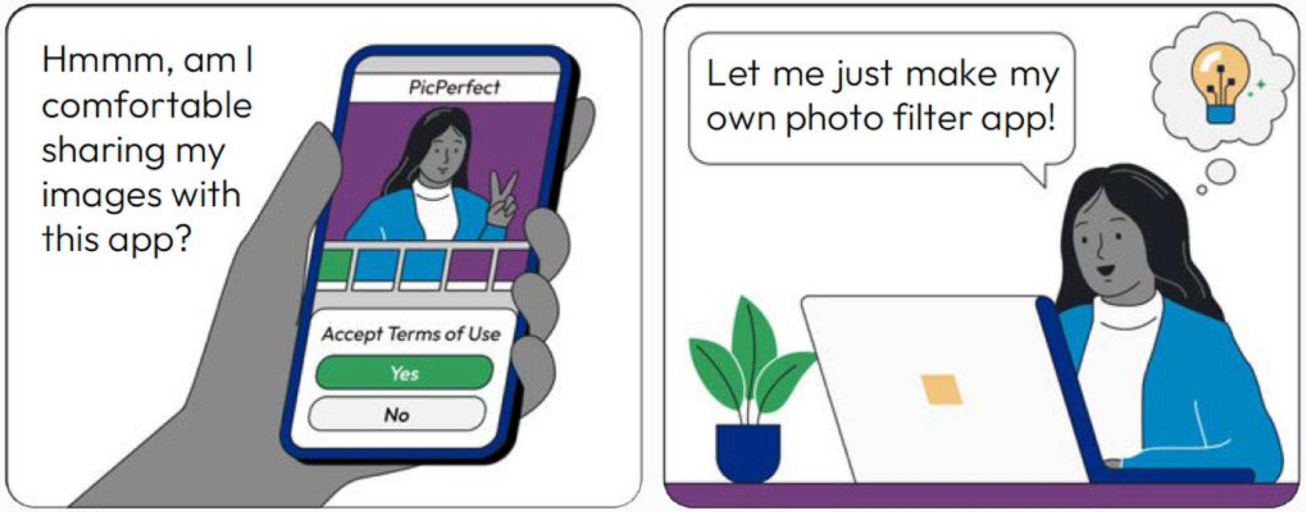
Unplugged activities (Long et al., 2021)

"[T]here is a tendency to make AI seem either magical, sentient, infallible, or overly human . . . Since such (mis)representations are rife in mainstream discourse, K-12 education needs to work extra hard to address this challenge through approaches to demystify AI and lift the hood on how it works ." (Grover, 2024)

- **Shuchi Grover**, Director, Looking Glass Ventures and Edfinity

To what level do we need to demystify AI for young learners?

Critical Consumers, Responsible Creators



Information Accuracy and Integrity

Transparency and Accountability

Fairness and Justice

Privacy Rights

Ethical Design

AI4ALL

AI4ALL co-founders Dr. Fei-Fei Li and Dr. Olga Russakovsky at SAILORS (now Stanford AI4ALL) in 2015

AI capabilities

Try a "Byte of AI"

AI & Drawing
Length: 1 hour
Skills/knowledge you'll gain: Understanding of what AI is, privacy and bias concerns
MORE INFO
TEACH THIS LESSON

AI & Facial Recognition
Length: 1 hour
Skills/knowledge you'll gain: Ethics
MORE INFO
TEACH THIS LESSON

AI & Deepfakes
Length: 1 hour
Skills/knowledge you'll gain: Critical thinking
MORE INFO
TEACH THIS LESSON

AI & the Environment
Length: 1-2 hours
Skills/knowledge you'll gain: Conservation
MORE INFO
TEACH THIS LESSON

AI & Dance
Length: 1-4 hours
Skills/knowledge you'll gain: Movement, collaboration
MORE INFO
TEACH THIS LESSON

AI & Ethics
Length: 10 hours
Skills/knowledge you'll gain: Human-centered design, ethics
MORE INFO
TEACH THIS LESSON

AI mechanism

Learn More About "How It Works"

How Neural Networks Work
Length: 2-4 hours
Pairs with: Everything
MORE INFO
TEACH THIS LESSON

How GANs (Generative Adversarial Networks) Work
Length: 2-4 hours
Pairs with: AI & Deepfakes
MORE INFO
TEACH THIS LESSON

How CNNs (Convolutional Neural Networks) Work
Length: 2-4 hours
Pairs with: AI & The Environment, AI & Dance, AI & Facial Recognition
MORE INFO
TEACH THIS LESSON

How RNNs (Recurrent Neural Networks) + Transformers Work
Length: 2-4 hours
Pairs with: AI & Ethics, AI & Drawing
MORE INFO
TEACH THIS LESSON

https://ai-4-all.org/about/our-story/ https://ai-4-all.org/resources/

Five Big Ideas in Artificial Intelligence v.2

1. Perception

Computers perceive the world using sensors. Perception is the process of extracting meaning from sensory signals. Making computers "see" and "hear" well enough for practical use is one of the most significant achievements of AI to date.

2. Representation & Reasoning

Agents maintain representations of the world and use them for reasoning. Representation is one of the fundamental problems of intelligence, both natural and artificial. Computers construct representations using data structures, and these representations support reasoning algorithms that derive new information from what is already known. While AI agents can reason about very complex problems, they do not think the way a human does.

3. Learning

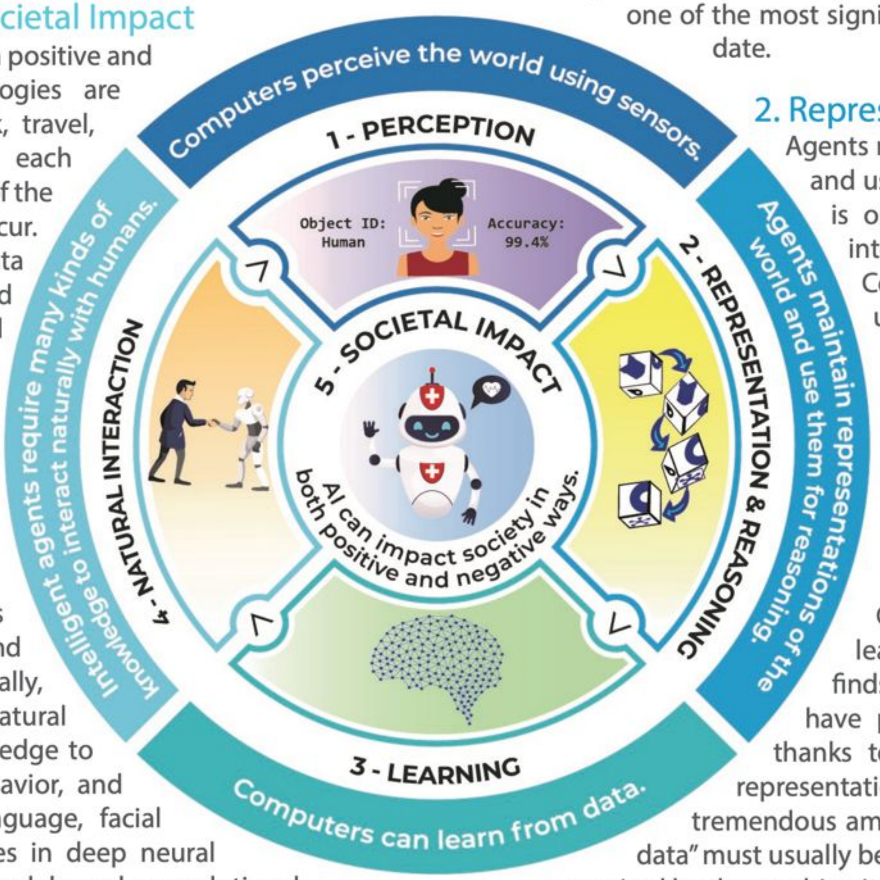
Computers can learn from data. Machine learning is a kind of statistical inference that finds patterns in data. Many areas of AI have progressed significantly in recent years thanks to learning algorithms that create new representations. For the approach to succeed, tremendous amounts of data are required. This "training data" must usually be supplied by people, but is sometimes acquired by the machine itself.

4. Natural Interaction

Intelligent agents require many kinds of knowledge to collaborate and interact naturally with humans. Ideally, agents will converse with us using natural language, draw upon cultural knowledge to infer intentions from observed behavior, and respond appropriately to body language, facial expressions, and emotions. Advances in deep neural networks such as large language models and convolutional neural networks are making this possible.

5. Societal Impact

AI can impact society in both positive and negative ways. AI technologies are changing the ways we work, travel, communicate, and care for each other. But we must be mindful of the harms that can potentially occur. For example, biases in the data used to train an AI system could lead to some people being less well served than others. Thus, it is important to discuss the impacts that AI is having on our society and develop criteria for the ethical design and deployment of AI-based systems.



The AI for K-12 Initiative is a joint project of the Association for the Advancement of Artificial Intelligence (AAAI) and the Computer Science Teachers Association (CSTA), funded by National Science Foundation award DRL-1846073

AI4K12.org

Draft Big Idea 1 - Progression Chart

www.AI4K12.org

Big Idea #1: Perception	*Computers perceive the world using sensors.*	**Perception is the extraction of meaning from sensory information using knowledge.**	**The transformation from signal to meaning takes place in stages, with increasingly abstract features and higher level knowledge applied at each stage.**	**LO** = Learning Objective: what students should be able to do. **EU** = Enduring Understanding: what students should know.
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Sensing (Computer Sensors) **1-A-ii**	**LO:** Locate and identify sensors (camera, microphone) on computers, phones, robots, and other devices. **EU:** Computers "see" through video cameras and "hear" through microphones.	**LO:** Illustrate how computer sensing differs from human sensing. **EU:** Most computers have no sense of taste, smell, or touch, but they can sense some things that humans can't, such as infrared emissions, extremely low or high frequency sounds, or magnetism.	**LO:** Give examples of how intelligent agents combine information from multiple sensors. **EU:** Self driving cars combine computer vision with radar or lidar imaging. GPS measurement, and accelerometer data to form a detailed representation of the environment and their motion through it.	**LO:** Describe the limitations and advantages of various types of computer sensors. **EU:** Sensors are devices that measure physical phenomena such as light, sound, temperature, or pressure. **Unpacked:** Cameras have limited resolution, dynamic range, and spectral sensitivity. Microphones have limited sensitivity and frequency response. Signals may be degraded by noise, such as a microphone in a noisy environment. Some sensors can detect things that people cannot, such as infrared or ultraviolet imagery, or ultrasonic sounds.
Processing (Sensing vs. Perception) **1-B-i**	**LO:** Give examples of intelligent vs. non-intelligent machines and discuss what makes a machine intelligent. **EU:** Many machines use sensors, but not all use them intelligently. Non-intelligent machines are limited to simple sensing. Intelligent machines demonstrate perception. **Unpacked:** Cameras and phones can record and play back images and sounds, but extracting meaning from these signals requires a computer with artificial intelligence.	**LO:** Use a software tool such as a speech transcription or visual object recognition demo to demonstrate machine perception, and explain why this is perception rather than mere sensing. **EU:** Perception is the extraction of meaning from sensory signals. **Unpacked:** speech recognition and face detection are examples of perception. An automatic door activated by a pressure pad or ultrasonic sensor does not exhibit perception because it is just reacting to the raw signal rather than using kowledge to extract meaning from the signal.	**LO:** Give examples of diferent types of computer perception that can extract meaning from sensory signals. **EU:** There are many specialized algorithms for perceptual tasks, such as face detection, facial expression recognition, object recognition, obstacle detection, speech recognition, vocal stress measurement, music recognition, etc.	**LO:** Explain perception algorithms and how they are used in real-world applications. **EU:** Many devices and services rely on specialized perception algorithms, e.g., license plate readers, zip code readers, face-based phone unlocking, tagging people in Facebook posts, object identification (e.g., Google Lens), or voice-based customer service.
Domain Knowledge (Types of Domain Knowledge) **1-C-i**	**LO:** Describe some things an intelligent agent must "know" in order to make sense of a question. **EU:** To understand spoken requests, computers must know our vocabulary and pronunciation conventions, and they must be able to distinguish a question from a command. **Unpacked:** Understanding a spoken query such as "Will it rain today?" requires all the above knowledge.	**LO:** Demonstrate how a text to speech system can resolve ambiguity based on context, and how its error rate goes up when given ungrammatical or meaningless inputs. **EU:** Speech recognition systems are trained on millions of utterances, allowing them to distinguish common from uncommon sequences of words, which helps them select the most likely interpretation of the signal. **Unpacked:** Compare the transcription of "the jockey reined in the horse" vs. "the king reigned in the horse". Or test the system on "which witch is which" or "two ways to go is one too many". To explore grammatical influences, compare the transcription accuracy of a sentence read with normal word order vs. the same sentence read with the word order (not the individual words) reversed, e.g., "see the view" vs. "view the sea".	**LO:** Classify a given image (e.g., "traffic scene", "nature scene", "social gathering", etc.) and then describe the kinds of knowledge a computer would need in order to understand scenes of this type. **EU:** Domain knowledge for vision includes knowing what kinds of objects are likely to appear in a scene, where they are likely to appear in relation to other objects, and how occlusions and shadows can alter object appearances. **Unpacked:** In a traffic scene, cars appear on roads, some traffic signs appear alongside of roads but not in the road, some signs appear above the road, and pedestrians appear on sidewalks, in crosswalks, and occasionally on roads. In a nature scene, the top of the image is likely to be blue sky and the bottom of the image is likely to be green grass or trees.	**LO:** Analyze one or more online image datasets and describe the information the datasets provide and how this can be used to extract domain knowledge for a computer vision system. **EU:** Domain knowledge in AI systems is often derived from statistics collected from millions of sentences or images. **Unpacked**: sample image databases: ImageNet: https://image-net.org/ Coco: http://cocodataset.org/#explore Word prediction when typing texts or emails is an example of the use of statistical prediction similar to what is found in high level perception systems. Analyzing large collections of images produces statistics about what kinds of objects are likely to co-occur in a scene.

Big Idea #2: Representation and Reasoning		*Computers maintain representations of the world and use them for reasoning.*	LO = Learning Objective: What students should be able to do. EU = Enduring Understanding: What students should know. Unpacked descriptions are included when necessary to illustrate the LO or EU	
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Representation *(Feature vectors)* **2-A-iv**	**LO:** Identify the features that make each object in a collection unique. and create a table of features to organize the objects. **EU:** Objects can be described in terms of the features they possess. **Unpacked:** This could be as simple as Legos bricks of different shapes, sizes, and colors, or features that distinguish different types of animals: cats, dogs, chickens, goldfish, penguins, etc., e.g., does it have fur, does it fly, etc. Another option is features that describe face emojis indicating different emotional states.	**LO:** Construct a feature vector representation for a set of objects and show how similar objects are close together in feature space. **EU:** Recommender systems represent things like movies, books, consumer products. or social media posts using feature vectors. **Unpacked:** Feature vectors represent concepts as sequences of numbers. The distance between two feature vectors can be measured by counting the number of positions at which they disagree, so similar objects lie closer together in feature space. Feature vectors can be constructed by hand, but they can also be constructed automatically using machine learning. **Example:** in the Pasta Land exercise students develop a discrimination tree for recognizing different types of pasta. The questions that make up the nodes of the tree can provide the features for a binary feature vector representation of the pasta types.	**LO:** Explain how word embeddings (which are feature vectors) represent words as sequences of numbers. **EU:** Word embeddings are a key part of neural natural language processing, including machine translation (e.g., Google Translate) and text generation systems (BERT, GPT3, etc.) **Unpacked:** Each word is a point in a feature space with many dimensions, organized so that words with similar meanings are close to each other in the feature space. See this Word2VecDemo.	**LO:** Describe how a transformer network operates. **EU:** Transformer networks map sequences of input words to sequences of output words, where words are represented as feature vectors. **Unpacked:** Neural network natural language processing applications such as machine translation or question answering are driven by word embedding representations, which are feature vectors. Words are fed in one vector at a time, and the network delivers its output one vector at a time. **Activity:** https://app.inferkit.com/demo
Search *(State spaces and operators)* **2-B-i**	**LO:** Illustrate a next possible state in the game of tic-tac-toe given a starting state. **EU:** A game such as tic-tac-toe can be described as a sequence of states, where each move transitions from a state to a successor state. **Unpacked:** Each state should be drawn as a separate tic-tac-toe board. Answers may vary depending on which move the student chooses to make. **Resource:** online tic-tac-toe games: https://playtictactoe.org/ or https://www.coolmathgames.com/0-tic-tac-toe	**LO:** Illustrate how a computer can represent the playing of a game such as tic-tac-toe or nim by drawing the linear sequence of board positions produced by the players' moves. **EU:** Computers play games and solve puzzles by creating a sequence of states (board positions) connected by legal moves, using an algorithm to choose their next move at each step. **Unpacked:** The state space (or search space) of a game is the set of all board states reachable from the start state (illustration), and the operators are the set of possible moves a player can make that adhere to the rules of the game. A particular game (linear sequence of board positions: illustration) is one path through this state space.	**LO:** Illustrate how a computer can solve a maze, find a route on a map, or reason about concepts in a knowledge graph by drawing a search tree. **EU:** Computers solve mazes, find driving routes, and reason about concepts in knowledge graphs using graph search algorithms, which construct search trees. **Unpacked:** The search space of a graph search problem is the set of all paths originating from the designated start node of the graph. The operators used for solving a maze move one node north, south, east, or west. In the more general case of graph search the operators extend a path by adding a new node at the end. Legal moves add a node that is reachable by a direct link in the graph. Legal states are those reachable by a sequence of legal moves. **Example:** to determine whether a kangaroo is a mammal, we search for a path in the knowledge graph from the "kangaroo" node to the "mammal" node that is comprised of "is-a" links. Quick intro article.	**LO:** Identify types of real-world problems that are search problems and describe their states and operators. **EU:** Computers can solve many types of problems using search techniques if the problem can be described in terms of finding a path from a start state to a goal state. **Unpacked:** Examples include task planning problems, scheduling problems, and resource allocation problems. A search algorithm determines which operators to apply, in which order. Finding a sequence of legal moves (operators) to reach a goal state can be used even with problems whose solution is not a sequence. For example, if the problem is to pack a collection of objects of various sizes into a set of containers with various capacities, a solution is an assignment of objects to containers such that no container is overfilled and no object is left out. This can be formulated as a search problem where an operator places one object in a container that can hold it, and a goal state has all objects placed. For this type of problem, the sequence in which the operators are applied does not matter.
Reasoning *(Types of reasoning problems)* **2-C-i**	**LO:** Identify problems as either classification problems or search problems. **EU:** In classification problems we decide what kind of thing we have based on its features. In search problems we find a path from a start to a goal, such as finding a route on a map or exploring possible moves in a game.	**LO:** Categorize problems as either classification problems or search problems. **EU:** Classification problems assign each input to one of a predetermined set of classes. Search problems construct answers by applying operators to states to generate new states. **Unpacked:** Labeling images as dog photos or cat photos (as in Teachable Machine) is a classification problem. Finding the board positions that can be reached in one move from a given starting position is an example of a search problem.	**LO:** Categorize problems as classification, prediction, combinatorial search, or sequential decision problems. **EU:** Prediction problems are similar to classification problems except they estimate a continuous value, such as height or daily temperature. Sequential decision problems choose the next move for any given state in order to maximize overall reward. **Unpacked:** Sequential decision problems are covered in Big Idea 3; they are addressed using reinforcement learning. **Examples:** spam vs. not-spam (classification), tomorrow's high temperature (prediction), solving puzzles such as the wolf, goat, and cabbage problem (combinatorial search), and playing a video game such as Super Mario (sequential decision problem).	**LO:** Categorize real-world problems as classification, prediction, sequential decision problems, combinatorial search, heuristic search, adversarial search, logical deduction, or statistical inference. **EU:** Reasoning problems can be categorized based on the types of inputs supplied, the types of outputs to be produced, and the characteristics of the search space, if applicable. **Unpacked:** Heuristic search is needed when the state space is too large to examine all possible states. uses a rule of thumb (heuristic) to limit the search by focusing on the most promising states. In adversarial search, used in game playing, the algorithm alternates between finding the best move for the player and finding the best response for the opponent, which would be the worst move from the player's perspective. Adversarial search may require heuristics if the game is complex, such as chess or go. In logical deduction, the reasoner starts with a set of facts and derives new facts by applying inference rules. Logical deduction can be done using formal logic such as propositional or predicate logic, or ad hoc inference rules used with semantic networks or the IF-THEN rules found in expert systems. Statistical inference involves reasoning with probabilities.

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Nature of Learning (Finding patterns in data) **3-A-ii**	**LO:** Identify patterns in labeled data and determine the features that predict labels. **EU:** Classes can be defined in terms of feature values. The relevant features can be inferred by examining labeled examples. **Unpacked:** To give students a feel for the problem of learning to classify we must ask them to learn a class that's not intuitively obvious, e.g., learn "poisonous fish" by examining cartoon fish images labeled "poisonous" or "not poisonous". They can then be asked to describe which features indicate a fish is poisonous, e.g., red fish with square heads. Using images as input simplifies the task because the features are intuitive, even though the classification rule should not be.	**LO:** Model how supervised learning identifies patterns in labeled data. **EU:** When learning to classify labeled data, the patterns (or rules) that are discovered can be expressed as weights in a neural network or nodes in a decision tree. **Unpacked:** This extends the K-2 version by having students draw a decision tree instead of merely verbalizing their proposed rule. In addition, the task can be made richer in 3-5 by increasing the number of classes or by making the class definitions more complex. For example, a fish could be poisonous if it is either red with a square head or blue with a round head or purple with pointy spines and any shape head. Each node of the decision tree can test one feature value, e.g., color, so complex features require deeper trees.	**LO:** Model how unsupervised learning finds patterns in unlabeled data. **EU:** Unsupervised learning is useful when we don't know in advance what classes exist. It discovers patterns (or classes) in data by grouping nearby points into clusters. Once a set of clusters has been found, new points can be classified based on distance from the cluster boundaries. **Unpacked:** This can be done graphically using points in the plane and visually constructing cluster boundaries by outlining (e.g., drawing an ellipse around) each cluster.	**LO:** Model how machine learning constructs a reasoner for classifcation or prediction by adjusting the reasoner's parameters (its internal representations). **EU:** Supervised learning adjusts the parameters of a mathematical model (selected in advance by a human) to generate correct classifications or predictions. This model could be a simple linear equation, a high-degree polynomial, or an even more complex nonlinear equation such as a deep neural network. The internal representations that encode the relationship between inputs and outputs express the "patterns" found in the data. **Unpacked:** In regression, we pick a mathematical model such as a linear equation y=mx+b and then adjust its parameters to fit a set of data points as best we can. The model can then be used to predict a y value for any x value. Linear regression can be done with a ruler by eyeballing the distance between the line and the points. Students can model polynomial or logistic regression by giving them a graphical display with sliders to control the parameter values. They can manually adjust the sliders to reach what they perceive as a best fit to the data. More advanced students can be shown how quality of fit can be measured mathematically using mean squared error. For classification problems the Y value is either 1 for "in class" or 0 for "not in class" and the decision boundary is the line or surface y=0.5.
Nature of Learning (Training a model) **3-A-iii**	**LO:** Demonstrate how to train a computer to recognize something. **EU:** Computers can learn from examples. **Unpacked:** With instructor assistance, Teachable Machine could be used to recognize hand gestures or sounds.	**LO:** Train a classification model using machine learning, and then examine the accuracy of the model on new inputs. **EU:** Computers can learn to classify instances or predict values by being shown labeled examples. If the results on new inputs are unsatisfactory, additional training may be required to improve the accuracy. **Activity:** Using Teachable Machine or Machine Learning for Kids, training examples can be supplied by webcam input or collected from an image search on the web, and the model can be trained on a task such as recognizing pictures of cats.	**LO:** Train and evaluate a classification or prediction model using machine learning on a tabular dataset. **EU:** Computers can learn to classify instances or predict values by examining feature values. If the results on new inputs are unsatisfactory, additional training may be required to improve the accuracy. **Unpacked:** Within a tabular dataset, each training example is a row in the table and is described by a set of feature values; the features are the columns of the table. Classification assigns each example to one of a discrete set of classes (e.g., cat or dog); prediction outputs a continuous value, such as predicting a person's height from their age. The learning algorithm is likely to be a decision tree learner rather than a neural network. **Activity:** Sites like MachineLearningForKids and eCraft2Learn include decision tree learning. The learning algorithm figures out which are the relevant features and what values they should have for each class.	**LO:** Use either a supervised or unsupervised learning algorithm to train a model on real world data, then evaluate the results. **EU:** In supervised learning the model is trained on a training set to produce the correct labels for labeled data. We evaluate the results by measuring the percent of items in a test set that are labeled correctly. In unsupervised learning, the model is trained to assign each input to a cluster of similar inputs. The clusters are determined by the learning algorithm since there are no labels attached to the training data. We evaluate the results by examining the clusters to see if they capture useful distinctions in the dataset. **Unpacked:** Both supervised and unsupervised learning algorithms find patterns in data. In supervised learning, the "pattern" is the relationship between feature values and class labels. In unsupervised learning the pattern is the way that data is grouped into clusters. Real world data sets are now widely available on the web. In earlier grade bands students might test their trained models on a few new data points, but in this grade band students are asked to quantiatively measure the performance of a trained model on a nontrivial test set.
Nature of Learning (Adjusting internal representations) **3-A-v**	**N/A**	**LO:** Analyze a game where one constructs a decision tree, describing the organization of the tree and the learning algorithm used to add nodes. **EU:** In a decision tree learning game, the tree's branch nodes are questions and the leaf nodes are classes. The learning algorithm moves through the tree by asking the questions at the branch nodes (testing features of the input) until it arrives at a leaf node. If that leaf node's class is incorrect, the node is replaced by a branch node with a new question, and the leaf node is reattached at that branch. **Activity:** the "guess the animal" game, troubleshooting problems, and the Pasta Land activity are good choices for demonstrating decision tree learning.	**LO:** Compare how a decision tree learning algorithm works vs. how a neural network learning algorithm works. **EU:** In decision tree learning, each step adds a new node, which tests a single feature value. In neural network learning, each step makes a small change to every weight in the network. **Unpacked:** A decision tree's internal representations are the nodes, the feature each node examines, and the value the feature is compared against. A neural network's internal representations are the weights. Decision tree learning algorithms try to find, for each new node they create, the most informative feature to examine. Changing the training data can result in a different choice of feature to examine next. With a neural network, changing the training data will lead to different weight adjustments as the algorithm tries to reduce the error signal.	**LO:** Describe how various types of machine learning algorithms learn by adjusting their internal representations. **EU:** Decision tree learners build decision trees by adding nodes one at a time. Neural net learning algorithms adjust weights. Regression algorithms adjust equation parameters. Reinforcement learners update value predictions or policies.
Nature of Learning (Learning from experience) **3-A-vi**	**N/A**	**LO:** Explain how reinforcement learning allows a computer to learn from experience (i.e., trial and error). **EU:** Computers can learn from experience if there is a "reinforcement" signal indicating whether the computer's actions are leading to good or bad outcomes. **Unpacked:** Computers can learn to play games using a reinforcement signal that indicates whether the computer won or lost the most recent game, or how many points it scored. The computer may have to play hundreds of thousands of games to become an expert player. **Demonstration:** Reinforcement learning can be illustrated using an agent navigating through a grid world with obstacles and hazards; the task is to learn the best path to a goal location. At each grid square, the allowable actions are to move N/S/E/W. Over repeated trials, the agent learns the best move to make in each square.	**LO:** Explain the differences between supervised learning and reinforcement learning. **EU:** Supervised learning tells the agent what output it should produce for each input; reinforcement learning only tells the agent how well it's doing as it chooses actions to take. **Unpacked:** In supervised learning, the teacher indicates the correct output for each training example, so the learning algorithm can see what it's doing wrong. In reinforcement learning, the reinforcement signal indicates how well the model is performing, but does not tell the learning algorithm what actions the model should have chosen to do better. This must be discovered by trial and error, so it may take hundreds of thousands of trials to reach expert level performance. For example, when playing a video game, the reinforcement signal could be the number of points scored. Because the computer learns from its own experience, reinforcement learning can find solutions to problems where there is no teacher who could tell it the best action to take.	**LO:** Select the appropriate type of machine learning algorithm (supervised, unsupervised, or reinforcement learning) to solve a reasoning problem. **EU:** Major types of learning algorithms and the kinds of reasoning problems they are used to solve are: supervised learning, used for classification and prediction; unsupervised learning, used for clustering; and reinforcement learning, used for sequential decision making. **Unpacked:** Both supervised and unsupervised learning algorithms find patterns in data. Supervised algorithms use labeled training data and adjust the reasoning model's parameters to try to produce the correct labels. They are used for classification or prediction problems. Unsupervised learning algorithms, which use unlabeled data, try to group similar data points together. They are used to discover classes in the data. Reinforcement learning algorithms are used for sequential decision problems. They learn policies for choosing actions that maximize the reinforcement the model will receive. Reinforcement learning can be slow because learning must proceed by trial and error; there is no teacher telling the algorithm the best action at each step. But having a computer learn from its own experience has the advantage that it can discover solutions to problems where it's not known in advance which is the best action.
Neural Networks *(Structure of a neural network)* **3-B-i**	**N/A**	**LO:** Illustrate how a neural network of 1 to 3 neurons is a function that computes an output. **EU:** A neural network uses one or more neurons working together to form a function. Each neuron takes a set of numbers as input and produces a single number as its output. **Unpacked:** A neural network is a collection of neurons that are connected to each other. Every neuron has a set of input connections, each with an attached weight. Each input connection carries a value. The neuron multiplies each input value by the connection weight to produce a weighted input. The sum of all the weighted inputs is compared to the neuron's threshold value. If the sum is above the threshold value, the neuron outputs a 1; otherwise it outputs a 0. The output value can be used as an input for other neurons. **Actrivity:** Calculate the output of a single neuron with multiple inputs, or a network of two multi-input "hidden" neurons feeding a single output neuron. Such networks can compute simple functions such as "AND", "OR", or "at least 2 out of 3". For a quick tutorial on neural nets for grades 3-5, see https://docs.google.com/document/d/1bYs0tTiL44sQhsMADgU2bDmVjWaVKVI2pT_SYZTzEwI/edit#heading=h.g640mybwbie6	**LO:** Illustrate the structure of a neural network and describe how its parts form a set of functions that compute an output. **EU:** Neural networks are organized as layers of units (input, hidden, and output layers), with weighted connections between units in successive layers. Each unit computes the sum of its weighted inputs. It passes that sum through a transfer function to produce a numeric output. **Unpacked:** A neural network maps input patterns to output patterns in a complex way. Each neuron computes a function, and the network as a whole computes a complex function that can be considered a very wiggly mathematical function.	**LO:** Describe the following neural network architectures and their uses: feed-forward network, 2D convolutional network, recurrent network, generative adversarial network. **EU:** Feed-forward networks can learn arbitrary functions and are used for both classification an regression. 2D convolutional networks learn small "kernels" that are convolved with the input, and max-pooling layers to reduce image resolution; they are used for image analysis. Recurrent networks have feedback connections and are used for language processing. Generative adversarial networks have generator and discriminator modules and are used to create deepfakes.
Neural Networks *(Weight adjustment)* **3-B-ii**	**N/A**	**LO:** Demonstrate how weights are assigned in a neural network to produce a desired input/output behavior. **EU:** The behavior of a neural network can be altered by adjusting its weights.	**LO:** Demonstrate how a learning rule can be used to adjust the weights in a one-layer neural network. **EU**: During training, weights are adjusted in response to errors in the network's output, so that an error will be less likely when the input is seen again. **Unpacked:** Training can be done using binary units and a simple learning rule for adjusting the weights (such as the perceptron learning rule in the "Will this dog bite me?" exercise).	**LO:** Train a multilayer neural network using the backpropagation learning algorithm and describe how the weights of the neurons and the outputs of the hidden units change as a result of learning. **EU:** A neuron's weights start out as small random values and evolve to a more precise pattern through learning. The changes in the neuron's weights are computed by a learning rule driven by a back-propagated error signal. The neuron's weight pattern determines the features that the neuron detects. **Unpacked:** Students are not expected to know the details of the backpropagation learning algorithm, only that error is propagated backward from later layers to earlier ones. **Activity:** An online demo such as TensorFlow Playground can be used to visualize the changes in weights during learning.
Datasets *(Bias)* **3-C-iii**	**LO:** Examine a labeled dataset and identify problems in the data that could lead a computer to make incorrect predictions. **EU:** How well a computer learns to classify depends on the data used to train it. **Unpacked:** If examples of healthy foods are broccoli, green beans, peas, and spinach (all green), and unhealthy foods are donuts, cake, and candy bars, what will the computer conclude about green gummy bears?	**LO:** Examine features and labels of training data to detect potential sources of bias. **EU**: Machine learning algorthims require a representative collection of data in order to build an accurate model. Training datasets drawn from historical data may reflect pre-exisiting human and societal biases. **Unpacked:** Amazon's resume sorter learned a bias against female applicants because it was trained to mimic the statistics of past hiring history.	**LO:** Explain how the choice of training data shapes the behavior of the classifier, and how bias can be introduced if the training set is not properly balanced. **EU:** Bias can result if the model is asked to classify inputs that don't resemble the training data, or if the training data contains irrelevant correlations we don't want the classifier to rely on. **Unpacked:** A classifier trained on only Caucasian faces will do poorly on Black or Asian faces. A classifier trained on a loan application dataset where most of the rejected applicants lived in Pleasntville might decide to never make a loan to anyone who lives in Pleasantville.	**LO:** Investigate imbalances in training data in terms of gender, age, ethnicity, or other demographic variables that could result in a biased model, by using a data visiualization tool. **EU:** Machine learning algorithms will take advantage of any imbalances or correlations in the training set that help lower the error rate. If the dataset is not representative, those correlations can be misleading **Unpacked:** Data exploration to help students uncover imbalances or correlations can be done using histograms in Excel, or using any number of data visualization tools such as Pandas (for Python).

Big Idea #4: Natural Interaction

Intelligent agents require many types of knowledge to interact naturally with humans.

LO = Learning Objective: What students should be able to do.
EU = Enduring Understanding: What students should know.
Unpacked descriptions are included when necessary to illustrate the LO or EU

Concept	K-2	3-5	6-8	9-12
Natural Language *(Structure of language)* **4-A-i**	**LO:** Demonstrate knowledge of the structure of language through tasks such as (a) generating plausible and implausible novel words, or (b) reordering the words in a scrambled sentence so that it makes sense. **EU:** Human languages have rules for how words and sentences are constructed, and computers can use these rules to help them figure out what people are saying. **Unpacked:** A plausible novel word is "flurg"; an implausible one is "fnurg" because English doesn't allow words to begin with "fn". Turning to word order, the words "ate pancakes John" only make sense in the order "John ate pancakes". **Activity:** Use this SpeechDemo to see how Google responds to *"I ordered a large **flurg** with whipped cream."* vs. *"I ordered a large **fnurg** with whipped cream."*	**LO:** Demonstrate that human language is infinite by showing how any sentence can be repeatedly extended to form a more complex sentence. **EU:** Human language can express an infinite number of ideas and form an infinite number of sentences. This property makes it impossible to pre-program a computer with a response to every sentence. Thus, to understand a new sentence a computer must recognize how the words combine into phrases and clauses to communicate complex ideas. **Unpacked:** Sentences can be lengthened by adding new phrases or clauses,with no limit. Repeatedly extending a sentence will yield something like, "John said that Mary knew that Peter saw that Lisa gave Megan the book about hamsters from outer space that Harry recommended." **Activity:** Have the class take turns extending a sentence until it is exceptionally long. Then run it through the Berkeley Neural Parser to see if it can correctly recognize the phrase structure.	**LO:** Demonstrate a computer's grasp of grammar by using a parser program to display the syntactic structure of a sentence, and explain what the nodes represent. **EU:** Parse trees are a way of representing the syntactic structure of a sentence, showing the relationships between words. Computers can use syntax trees to both analyze and generate sentences. **Unpacked:** The grammatical structure of a sentence is key to understanding its meaning. For example, if the task is question answering, then we need to understand the meaning of the question to perform the appropriate query. If the task is to get a robot to do something, the robot must understand what is being requested. Parsers produce syntax trees whose nonterminal nodes are grammatical categories such as NP (Noun Phrase), VP (Verb Phrase), and PP (Prepositional Phrase). "Put the cup on the saucer" vs. "put the saucer on the cup" illustrates the importance of syntax. Both sentences contain the same words, but the words have different syntactic relationships. That's why changing the word order changes the meaning. In addition, "cup" can be either a noun or a verb, but in this context the syntax indicates it's being used as a noun. **Resource:** The Berkeley Neural Parser demonstrates both POS (part of speech) tagging and parse tree generation. The tags come from the Penn Treebank project.	**LO:** Identify portions of a text that would be difficult for a computer to understand, and explain why. **EU:** Computers have difficulty understanding text that makes use of metaphor, imagery, hyperbole, sarcasm, or humor, or word play. **Unpacked:** Currently, we don't have satisfactory formal explanations of metaphor, imagery, hyperbole, sarcasm, humor, or word play. This is the focus of current AI, linguistics, and cognitive science research.
Natural Language *(Reasoning about text)* **4-A-iii**	**LO:** Demonstrate how a computer can produce different forms of a verb, such as present or past tense. **EU:** In order for a computer to speak naturally with humans it must be able to understand how words are constructed and put words in the proper form. **Unpacked:** Verbs can take different forms in first, second, and third person, singular or plural, and present or past tense. For a computer to respond correctly to the question "When did dinosaurs live?" it should say "Dinosaurs lived over 65 million years ago." **Activity 1:** List all the different forms a verb can take, then check your results using this online verb conjugator. **Activity 2:** This past tense converter is not very good. Experiment to see where it succeeds and where it fails. **Activity 3:** Set up Google Translate for English -> Spanish and enter a sentence like "I walk to school". Then try different forms of the sentence ("I walked", "I am walking", "I was walking" "I had been walking", etc.) and observe how the translation changes.	**LO:** Experiment with a speech to text system to see if it resolves alternative word choices correctly based on context. **EU:** Speech recognition systems can use grammar and context to resolve ambiguous words but they don't always get it right. **Activity:** Use this SpeechDemo to see alternative interpretations of *"the dishes' size and weight were impressive"*, or homophones such as "which/witch" in *"I couldn't tell which of the witches was the witch with the broomstick."*	**LO:** Illustrate how word embeddings can be used to reason about the meaning of words. **EU:** Word embeddings represent words with similar meanings as nearby points in a semantic feature space, and allow us to reason about words by doing arithmetic. **Unpacked:** An example of reasoning about word meanings is **outlier detection**, e.g., which word in the list "breakfast", "lunch", "banana", "dinner" does not fit with the others? Similarity of words can be measured as distance in feature space; the outlier will be farther from the other words. Another example of reasoning about meaning is **analogy completion**, e.g., "man is to king as woman is to queen". This can be solved by calculating the feature vector "king" - "man" + "woman" and finding the word closest to the result. The real benefit of word embeddings is that they are used as inputs to transformer networks which perform much more complex reasoning operations on text. *See 2-A-iv.6-8 For more on word embeddings and 2-A-iv.9-12 for transformer networks.* **Activity 1:** Experiment with WordEmbeddingsDemo to explore realistic word embeddings. **Activity 2:** Use knowledge of word embeddings to explain how the Semantris game works (Blocks activity).	**LO:** Demonstrate how a small context-free grammar can be used to parse or generate simple sentences. **EU:** Context-free grammars describe how words are combined into phrases and clauses, and can represent much of the syntactic structure of natural language, but don't handle things like subject-verb agreement well. **Unpacked:** An example of an agreement constraint is that subjects and verbs must agree on number, e.g., "he says" but "they say". Expressing this in a context-free grammar would require separate rules for singular and plural subjects. This gets cumbersome when sentences become more complex, e.g., coordinating conjunctions may require agreement on tense as well as number. See 4-A-ii.6-8 for examples of syntactic ambiguity, which leads to multiple syntactic parse trees that must be disambiguated using semantic knowledge. Even a simple grammar, if it's recursive, can generate an infinite number of sentences; see 4-A-i.3-5. **Resource:** This Stanford Context Free Grammar tool can be used to construct parse trees, generate sentences, and verify the grammaticality of a sentence. See this reference for creating a context free grammar.
Natural Language *(Applications)* **4-A-iv**	**LO:** Demonstrate the kinds of tasks an intelligent assistant can and cannot perform. **EU:** Intelligent assistants (e.g., Siri, Alexa) are computers designed to respond to a limited set of requests. They cannot engage in a conversation like a human. **Unpacked:** Siri and Alexa can answer questions, play music, set alarms, and make lists. But intelligent agents are not people. They have limited conversational abilities despite their ability to recognize spoken language. One reason is that they have trouble maintaining context, i.e., remembering what was said before to help them understand what is being said now. **Activities:** Talk with an intelligent agent such as Alexa. What are the kinds of things they can do well or not so well? What are the limits of its understanding?	**LO:** Demonstrate some types of questions that a search engine or intelligent assistant can answer, and some types that it cannot answer. **EU:** Search engines (e.g., Google) and intelligent assistants (e.g., Siri, Alexa) have a collection of specialized and general purpose modules they draw upon to answer different types of questions. **Unpacked:** Examples of queries handled by specialized modules include: definitions of words; unit conversions (e.g., inches to millimeters, dollars to euros); current time and weather anywhere in the world; biographical and geographical facts (birth date of Abraham Lincoln, capital of Belgium); store hours and driving directions; and airline, train, and bus schedules. Current intelligent agents do less well when asked to reason about relationships between entities (e.g., is an alligator bigger than an ostrich), or when asked to produce explanations rather than simple facts (e.g.,Why didn't Shakespeare write about airplanes?). When no specialized module can handle a query, search engines fall back on keyword search, but the results are often unsatisfying.	**LO:** Describe some NLP (Natural Language Processing) tasks computers can perform, and explain how they work. **EU:** NLP (Natural Language Processing) tasks include text summarization, text generation, sentiment analysis, question answering, machine translation, and conversational interaction.	**LO:** Describe several approaches to Natural Language Processing, ranging from simple to more sophisticated. **EU:** Simple NLP approaches include keyword matching, dictionary lookup, and template matching, while newer, more sophisticated but less transparent approaches use deep neural networks and machine learning. **Unpacked:** Taking chatbots as an example, the simplest approach looks for keywords in the user input to decide what response to give. A slightly more sophisticated approach uses templates to describe all the variations a phrase might take rather than looking at single phrases. More sophisticated chatbots use deep neural networks for "intent recognition", which detects when the *meaning* of an input matches a template, rather than looking for specific words or phrases. **Activities:** Students should be able to describe when these approaches might be helpful or not. Explain the limitations.
Commonsense Reasoning **4-B-i**	**LO:** Give commonsense explanations of human behavior or events that might be useful for a robot to know. **EU:** Choosing the most likely ending for a story is an example of commonsense reasoning, which requires knowledge about people and things, and how they behave, that computers may not have. **Activities:** (1) Choose the most likely ending to a story given a list of alternatives, and explain your choice; (2) Explain why a character in a story took a particular action; (3) Suggest an appropriate action for a character in a particular situation, and explain why it is appropriate.	**LO:** Explain what knowledge would be required for a computer to understand a story. **EU:** AI has trouble understanding stories because it doesn't have the knowledge that humans have about everyday life. **Unpacked:** "Everyday life" includes both cultural knowledge (what is an umbrella for) and naive physics (dropped objects will fall due to gravity). **Activity:** Compose a story that may be hard for a computer to understand, and explain what makes it hard.	**LO:** Explain the knowledge a computer would need to solve a naive physics reasoning problem. **EU:** Computers can reason about physical phenomena using *naive physics* inference rules, which formalize our intuitive understanding of concepts such as mass, volume, forces, and motion. At present, computers are not very good at this. **Unpacked:** Naive physics includes knowledge of conservation laws. Pouring a large container of water into a smaller container will cause an overflow because small containers have less volume and the amount of water doesn't change. By grades 6-8 students should have mastered conservation laws; see this article on Piaget conservation tasks. Naive physics also includes inference rules for reasoning about forces, such as "When a heavy thing hits a light thing, the light thing moves." **Activity:** Propose a reasoning problem that requires knowledge of naive physics to solve, and say what that knowledge is.	**LO:** Explain the cultural and naive physics knowledge required for a computer to correctly interpret a fable or fairytale. **EU:** For humans or computers to infer the meaning of a story requires understanding of cultural knowledge, naive physics, and folk psychology. This is still difficult for computers. **Unpacked:** Fables and fairy tales are challenging for computers because they can incorporate cultural knowledge, complex human motivations, imagery, humor, and metaphor. Folk psychology refers to our everyday ability to attribute mental states to other people, including their beliefs, desires, and intentions. **Activity:** Explain what knowledge is required to correctly resolve the referent of the pronoun in a Winograd sentence. An example Winograd sentence is "Anna did a lot [better/worse] than Lucy on the test because she had studied so hard." Here, the required knowledge is that more studying for a test leads to better performance.
Understanding Emotion **4-C-i**	**LO:** Demonstrate how computers recognize emotions in faces. **EU:** Computers recognize emotions in faces by looking at the shapes of the mouth, eyes, and eyebrows. **Activity:** Use a face demo that recognizes facial landmarks (e.g., eyebrows, eyes, and mouth) to infer emotional state. The activity can be introduced using icons to demonstrate different emotions.	**LO:** Illustrate how computers can judge the emotional tone of text. **EU:** Computers can recognize positive and negative statements about a topic using natural language processing techniques known as "sentiment analysis". **Unpacked:** Sentiment analysis is used for a variety of tasks, such as: - **Movie reviews:** Analysing online movie reviews to get insights from the audience about the movie. - **Restaurant reviews**: analyzing reviews of restaurants to measure customer satisfaction. - **News coverage:** Analyzing news coverage of an event, person, or company to assess media opinion. - **Social media analysis:** analyzing the sentiments of Facebook, Twitter, or Instagram posts to assess public opinion about events, persons, or products. **Activities**: Use appropriate AI services (e.g., Scratch plugins) to create artifacts that demonstrate sentiment analysis. Use a web-based sentiment analysis tool to analyze the polarity of text that is marked subjective.	**LO:** Describe how computers use different types of cues to recognize human emotional states. **EU:** Computers can recognize human emotional states by looking at facial expressions, gaze, gestures, body language, tone of voice, and choice of words. **Unpacked:** For example, body language can indicate engagement, hostility, anxiety, or boredom. Body language includes such factors as how a person holds their arms, how they position their feet, whether they are leaning in or leaning back, and how they are tilting their head. Computers can extract this pose information from webcam images.	**LO:** Identify ways AI applications can modify their behavior to respond to people's emotional states. **EU:** Computers can respond appropriately to human emotions by acknowledging what the human is feeling and responding in a way that the human finds supportive and socially appropriate. **Unpacked:** For example, AI-based tutoring systems monitor student behavior to identify frustration, boredom, and tiredness so they can adapt the instruction or prompt the student to take a break. Automated customer service agents that detect human emotions could adapt their responses accordingly, e.g., by adjusting their speech rate and tone of voice. **Activity:** Given that computers can use AI to identify human emotions, discuss how should they respond to people in a way that we think is supportive and socially appropriate.

Big Idea #5: Societal Impact

AI can impact society in both positive and negative ways.

LO = Learning Objective: What students should be able to do.
EU = Enduring Understanding: What students should know.
Unpacked descriptions are included when necessary to illustrate the LO or EU

Concept	K-2	3-5	6-8	9-12
Ethical AI *(Ethical Design Criteria)* **5-A-ii**	**LO:** Discuss the characteristics of systems that are fair and unfair and the impact on people when a system is not fair. **EU:** AI systems should be designed to benefit people. Creators of these systems should make sure that their systems treat everyone fairly. **Activity:** Facilitate an in-class exercise focused on fair distribution of limited goods, such as deciding how to share 10 peices of candy among 15 students, to help students think personally about what fairness and unfairness feel like. **Resources:** Fairness teaching guide, Fairness & Justice	**LO:** Evaluate how an AI system meets the design criteria of transparency and explainability. **EU:** An AI system is transparent if we know what data and decision making criteria it uses. Part of transparency is having the system provide *explanations* for its decisions. **Resources:** https://appinventor.mit.edu/about/termsofservice https://teachablemachine.withgoogle.com/faq#Basics https://quickdraw.withgoogle.com/data https://experiments.withgoogle.com/quick-draw (see article and video)	**LO:** Evaluate how an AI system meets the design criteria of accountability and respect for privacy. **EU:** Accountability in AI systems means designers and decision makers take responsibility for the system's actions. Respect for privacy means the system does not act in ways that violate people's privacy rights.	**LO:** Analyze an AI system to determine whether it satisfies ethical design criteria. **EU:** To ensure that AI systems are helpful and not harmful, ethical design criteria include: fairness, transparency, explainability, accountability, respect for privacy, and adherance to societal values. **Unpacked** Fairness means treating people equally. Transparency means disclosing what information a system uses and how it uses it. Explainability means being able to justify the decisions a system makes. Accountability means being clear about who is responsible for the actions an AI system takes. Respect for privacy means not acting in ways that could undermine people's privacy. Adherence to societal values means not acting in ways counter to those values. **Resources:** Catalog of resources on AI ethics: https://aiartists.org/ai-ethics Model Cards (promoting transparency and accountability): https://modelcards.withgoogle.com/about
AI & Culture (*AI in Daily Life)* **5-B-i**	**LO:** Identify devices in daily life that use AI technologies. **EU:** AI technologies are part of any device that includes speech recognition or computer vision, such as smart phones, intelligent home assistants, and modern automobiles.	**LO:** Describe how AI-powered services are used in daily life. **EU:** AI-powered services are used to look up information, to provide voice interfaces to many kinds of apps, to make recommendations based on a person's interests, and to make cars safer. **Unpacked:** AI provides the ability to understand the meaning of people's requests and develop models of their interests and goals. Search engines use AI to understand search queries and the contents of web pages. Recommender systems that learn users' interests include Netflix for movies, Amazon for shopping, social media for news feeds, and advertising networks for ad selection. Autonomous vehicles use computer vision to predict the intentions of other drivers.	**LO:** Examine an aspect of daily life that is predicted to change due to the introduction of AI technologies. **EU:** AI technologies are changing daily life as intelligent machines find new roles in society. **Unpacked:** Aspects of daily life include topics such as how we communicate, how we learn, how we interact socially, and the makeup of our daily routines. Consider the many changes to transportation likely to result from the adoption of self-driving cars. Drivers will have to learn how to interact with other cars that have no driver. Parents will have to decide when it's appropriate to let their children ride unaccompanied in a self-driving car, and the cars themselves might be adapted to facilitate safe transport and dropoff of children. Automated ride-hailing services may lead to reductions in car ownership and the need to learn how to drive, and also reduce demand for parking space. New types of guided tour services may combine a chatbot with a self-driving vehicle.	**LO:** Explain the kinds of debates that might arise as AI technology continues to evolve and is further woven into our culture. **EU:** Some new AI technologies will pose challenges for cultural norms and expectations that society must identify and wrestle with. **Unpacked:** Some potential debates include: What is acceptable behavior when interacting with intelligent assistants or robots? How should we regard text or artwork that was created by or in collaboration with AI programs? When is it acceptable for students to use large language models for help with homework assignments? What rights should machine learning engineers have to use people's personal data, or to use publicly available data in ways the copyright holder hasn't authorized? Many more such controversies should be expected.
AI & Culture (*Trust and Responsibility)* **5-B-ii**	N/A	**LO:** Analyze deepfake images or videos and identify the flaws that reveal them as deepfakes. **EU:** Current deepfakes have detectable flaws, but as the technology improves it will undermine our trust in digital media. **Resource:** Deepfake faces: https://thispersondoesnotexist.com/	**LO:** Critique uses of AI technology that can be used to surveil people or violate their privacy. **EU:** AI technologies endanger privacy by reducing the cost of implementing widespread surveillance and enabling new types of surveillance. **Unpacked:** Types of surveillance include not only face recognition in public spaces and license plate trackers on public roads, but also monitoring of search activities and behavior on web sites that can be used to build user profiles that include sensitive personal information.	**LO:** Identify areas where it is appropriate to regulate use of AI technologies and evaluate regulations that have been proposed. **EU:** Legal regulation of AI technologies is appropriate in areas where there are societal values that require protection. **Unpacked.** Potential areas for regulation include use of facial recognition technologies for policing or surveillance, privacy protection for customer data accumulated by AI applications, and safety of autonomous vehicles. **Activities:** (1) Read an article about proposed legislation on AI technology and create an informed critique of the legislation. (2) Write a letter to an elected official about a legislative issue related to AI. **Resources:** Center for AI and Digital Policy: https://www.caidp.org/resources/ai-policy-frameworks/ Autonomous vehicle regulations: https://www.ncsl.org/research/transportation/autonomous-vehicles-self-driving-vehicles-enacted-legislation.aspx
AI & the Economy (*Impacts of AI on Sectors of Society)* **5-C-i**	**LO:** Identify current uses of AI and how they have impacted society. **EU:** Society has undergone changes because of AI and this will continue in the future. **Unpacked:** AI currently affects things like how we get questions answered, how we get directions, and how we find entertainment.	**LO:** Identify changes in how sectors of society operate due to the introduction of AI. **EU:** Every sector of society is changing (or will change) as a result of the introduction of AI. **Unpacked:** Sectors of society include manufacturing, retail, agriculture, food, hospitality, transportation, housing, environment, education, entertainment, healthcare, finance, government, public safety, social services, and law enforcement. An example of change: manufacturing is taking advantage of increased automation using AI to reduce costs and improve quality. **Activity:** Research a story and describe how an AI transformative change impacted society positively and potentially negatively.	**LO:** Compare the changes AI is bringing to society with those of previous industrial revolutions. **EU:** AI is causing societal advances and disruptions comparable to earier industrial revolutions. **Unpacked:** The first industrial revolution was based on mechanical power, the second on electricity and mass production, and the third on computers and networking. The fourth will be based on AI, robotics, Internet of Things, and genetic engineering technologies.	**LO:** Predict how a sector of society is likely to change in the short and intermediate term as a result of AI technology. **EU:** Anticipating and planning for the changes new technology brings is important for the healthy advancement of society. **Unpacked:** Two types of impacts associated with AI technoogy are increased levels of automation and new types of services. Roy Amara, past president of The Institute for the Future, coined Amara's Law which states:" We tend to overestimate the effect of a technology in the short run and underestimate the effect in the long run." **Activities:** (1) Discuss possible new services that can evolve due to AI. (2) Identify and explain an unintended consequence in society that resulted from an AI system.
AI & The Economy (*Effects on Employment)* **5-C-ii**	**LO:** Describe some jobs that no longer exist due to advances in technology. **EU:** New technology changes the types of jobs that are available for people. **Unpacked:** The automobile reduced our reliance on horses, which eliminated jobs for farriers and horse trainers but created jobs for auto mechanics. Factory automation enabled mass production, which reduced the need for blacksmiths, yarn spinners, and weavers but created jobs for people who build and maintain the factories.	**LO:** Describe how a job will change due to the introduction of AI or robotic technologies. **EU:** As AI and robotic technologies are adopted in the workplace, the ways people perform their jobs will change. **Activity:** Students can read grade-appropriate articles that describe jobs being upated with the use of AI technologies and robots, e.g., warehouse workers working alongside robots.	**LO:** Predict a new type of job that might arise, or how an existing type of job might change or go away, as a result of the adoption of AI technologies. **EU:** Cultures change as new technologies are adopted, and as a result some types of jobs are reduced and new types of jobs appear. **Activity:** Develop a "job description" of the future for a given profession - what will working with AI and robotic systems look like? What skills will be required?	**LO:** Investigate the skills needed for AI-enabled careers. **EU:** AI-aligned skills will be relevant throughout the workforce, not just for programmers. Most types of work will involve some interaction with AI technologies. **Unpacked:** As new technologies are adopted, the nature of work will change over a person's lifetime. People can expect to learn continually throughout their careers. AI-aligned skills that are becoming important include: collecting and curating datasets for machine learning; interacting with intelligent agents that help people do their jobs; training robots to complete specific tasks; use of AI-powered creative tools for image creation and manipulation; and knowledge engineering for AI systems.
AI for Social Good (*Democratization of AI Technology)* **5-D-i**	N/A	**LO:** Describe and use some of the AI extensions or plugins available in a programming framework familiar to you. **EU:** AI is becoming part of everyone's toolbox through extensions or plugins that support development of AI applications serving the needs of many different communities. **Unpacked:** Examples for Scratch include speech to text, text to speech, face recognition, sentiment analysis, question answering, and visual classifier extensions.	**LO:** Create a novel application using some of the AI extensions or plugins available in the programming framework of your choice. **EU:** AI is becoming part of everyone's toolbox through extensions or plugins that support development of AI applications serving the needs of many different communities. **Unpacked:** Examples for Scratch include speech to text, text to speech, face recognition, sentiment analysis, question answering, and visual classifier extensions. There is a similar list for MIT App Inventor. Calypso has many of these features built in.	**LO:** Create a novel application using some of the AI tools available in the programming framework of your choice. **EU:** AI tools are becoming commonplace and freely available, and can be used by people without advanced degrees or expensive equipment.
AI for Social Good (*Using AI to Solve Societal Problems)* **5-D-ii**	**LO:** Describe how AI can be used to solve a societal problem **EU:** AI can be used to create a classifier that solves a problem important to society. **Unpacked:** Classifiers can be trained to distinguish wildlife from manufactured items, recyclables from non-recyclables, or healthy from diseased plants. **Activity:** Use an AI for Social Good application to contribute to a solution to a societal problem. **Resource:** Code.org's AI For Oceans - https://studio.code.org/s/oceans/	**LO:** Design a solution to a societal problem that makes use of AI technology **EU:** AI is being used to solve societal problems such as environmental protection, energy conservation, and improved public health.	**LO:** Research a societal problem and describe how AI technologies can be used to address that problem. **EU:** AI technologies for perception, reasoning, and machine learning can be applied to many types of societal problems. **Resources:** UN's 16 Sustainable Development Goals - https://sdgs.un.org/goals Google's AI for Social Good page https://ai.google/social-good	**LO:** Evaluate an AI for Social Good project in terms of the problem it is addressing and the project's actual or potential impact. **EU:** "AI for social good" is the use of AI technologies to solve societal problems. **Unpacked:** "Social good" or common good seeks to provide the greatest benefit to the greatest number of people, to make the world a better place. This includes goals such as energy conservation, environmental protection, protection of endangered species, better public health, and prevention of human trafficking. **Resources:** Google blog post on social good projects: https://blog.google/technology/ai/30-new-ai-for-social-good-projects/ Nature article on AI for social good: https://www.nature.com/articles/s41467-020-15871-z

The Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24)

Artificial Intelligence in the CS2023 Undergraduate Computer Science Curriculum: Rationale and Challenges

Eric Eaton[1] and Susan L. Epstein[2,3]

[1]Department of Computer and Information Science, University of Pennsylvania
[2]Department of Computer Science, Hunter College of The City University of New York
[3]Department of Computer Science, The Graduate Center of The City University of New York
eeaton@seas.upenn.edu, sepstein@hunter.cuny.edu

Abstract

Roughly every decade, the ACM and IEEE professional organizations have produced recommendations for the education of undergraduate computer science students. These guidelines are used worldwide by research universities, liberal arts colleges, and community colleges. For the latest 2023 revision of the curriculum, AAAI has collaborated with ACM and IEEE to integrate artificial intelligence more broadly into this new curriculum and to address the issues it raises for students, instructors, practitioners, policy makers, and the general public. This paper describes the development process and rationale that underlie the artificial intelligence components of the CS2023 curriculum, discusses the challenges in curriculum design for such a rapidly advancing field, and examines lessons learned during this three-year process.

AI's prevalence has made it a key focus in the education of computer scientists. For the past few decades, the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers – Computer Society (IEEE-CS) have jointly issued curriculum guidelines for undergraduate CS education. ACM released its first set of CS curricular guidelines in 1968, with a subsequent update in 1978 (Hemmendinger 2007). In 1991, IEEE-CS joined this process; together they published significant updates in 2001, 2008, and 2013. These curricula heavily influence CS major requirements and courses worldwide, and thereby guide the development of the next generation of researchers and practitioners.

ACM and IEEE-CS recognize that those who develop and deploy AI-enhanced technology must understand and con-

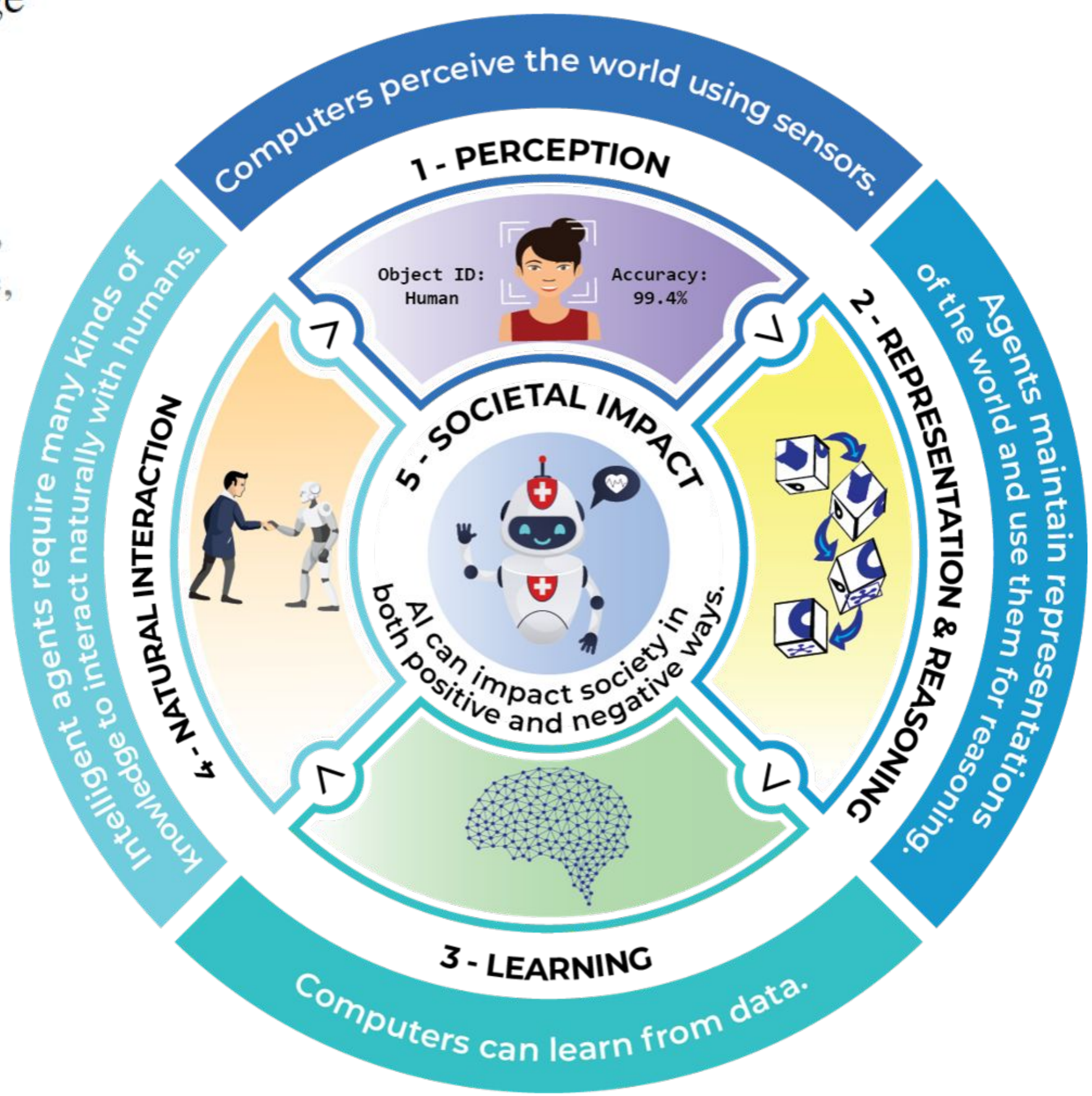
Structure. The CS2023 Artificial Intelligence knowledge area is divided into 12 knowledge units:

CS Core, KA Core, Elective:

- Fundamental Issues
- Search **Big Idea 2**
- Fundamental Knowledge Representation and Reasoning (KRR) **Big Idea 2**
- Machine Learning **Big Idea 3**
- Applications and Societal Impact **Big Idea 5**

Elective only:

- Probabilistic Representation and Reasoning **Big Idea 2**
- Planning **Big Idea 2**
- Logical Representation and Reasoning **Big Idea 2**
- Agents and Cognitive Systems **Big Idea 4**
- Natural Language Processing **Big Idea 4**
- Robotics **Big Idea 4**
- Perception and Computer Vision **Big Idea 1**



What is AI Literacy? Competencies and Design Considerations

Duri Long
Georgia Institute of Technology
Atlanta, GA, USA
duri@gatech.edu

Brian Magerko
Georgia Institute of Technology
Atlanta, GA, USA
magerko@gatech.edu

ABSTRACT
Artificial intelligence (AI) is becoming increasingly integrated in user-facing technology, but public understanding of these technologies is often limited. There is a need for additional HCI research investigating a) what competencies users need in order to effectively interact with and critically evaluate AI and b) how to design learner-centered AI technologies that foster increased user understanding of AI. This paper takes a step towards realizing both of these goals by providing a concrete definition of *AI literacy* based on existing research. We synthesize a variety of interdisciplinary literature into a set of core competencies of AI literacy and suggest several design considerations to support AI developers and educators in creating learner-centered AI. These competencies and design considerations are organized in a conceptual framework thematically derived from the literature. This paper's contributions can be used to start a conversation about and guide future research on AI literacy within the HCI community.

Design and education both play a role in contributing to public misunderstandings about AI. Black-box algorithms (i.e. algorithms with obscured inner-workings) can cause misunderstandings about AI [55]. On the other hand—even with more transparent technologies—a lack of technical knowledge on the part of the user can lead to misconceptions [25]. There is a clear need for a better understanding of this space from the perspectives of both learners and designers.

Researchers in the HCI community have begun to address public misconceptions of AI by investigating how people make sense of AI (e.g. [46]) and exploring how to design more understandable technology (e.g. [67]). However, there is a need for additional research investigating what new competencies will be necessary in a future in which AI transforms the way that we communicate, work, and live with each other and with machines. We refer to this set of competencies as *AI literacy*.

Emerging research is exploring how to foster AI literacy in audiences without technical backgrounds. Within the past

What is AI Literacy? Competencies and Design Considerations

More general than K-12 and Higher Ed

Competency 1 (Recognizing AI)
Distinguish between technological artifacts that use and do not use AI.
Supporting References: [10,18,54,55,57,73,116,124,138,145]

Competency 2 (Understanding Intelligence)
Critically analyze and discuss features that make an entity "intelligent", including discussing differences between human, animal, and machine intelligence.
Supporting References: [21,64,69,100,115,116,125]

Competency 3 (Interdisciplinarity)
Recognize that there are many ways to think about and develop "intelligent" machines. Identify a variety of technologies that use AI, including technology spanning cognitive systems, robotics, and ML.
Supporting References: [64,115,117,145]

Competency 4 (General vs. Narrow)
Distinguish between general and narrow AI.
Supporting References: [57,58,64]

Competency 5 (AI's Strengths & Weaknesses)
Identify problem types that AI excels at and problems that are more challenging for AI. Use this information to determine when it is appropriate to use AI and when to leverage human skills.
Supporting References: [10,22,106,124,125,130]

Competency 6 (Imagine Future AI)
Imagine possible future applications of AI and consider the effects of such applications on the world.
Supporting References: [6,43,143,145]

Competency 7 (Representations)
Understand what a knowledge representation is and describe some examples of knowledge representations.
Supporting References: [30,72,78,92,113,130]

Competency 8 (Decision-Making)
Recognize and describe examples of how computers reason and make decisions.
Supporting References: [29,30,72,78,113]

Competency 9 (ML Steps)
Understand the steps involved in machine learning and the practices and challenges that each step entails.
Supporting References: [45,117,125,145]

Competency 10 (Human Role in AI)
Recognize that humans play an important role in programming, choosing models, and fine-tuning AI systems.
Supporting References: [22,125]

Competency 11 (Data Literacy)
Understand basic data literacy concepts such as those outlined in [107].
Supporting References: [36,68,107]

Competency 12 (Learning from Data)
Recognize that computers often learn from data (including one's own data).
Supporting References: [36,68,107,130]

Competency 13 (Critically Interpreting Data)
Understand that data cannot be taken at face-value and requires interpretation. Describe how the training examples provided in an initial dataset can affect the results of an algorithm.
Supporting References: [6,36,68,107,130,145]

Competency 14 (Action & Reaction)
Understand that some AI systems have the ability to physically act on the world. This action can be directed by higher-level reasoning (e.g. walking along a planned path) or it can be reactive (e.g. jumping backwards to avoid a sensed obstacle).
Supporting References: [42,115,131]

Competency 15 (Sensors)
Understand what sensors are, recognize that computers perceive the world using sensors, and identify sensors on a variety of devices. Recognize that different sensors support different types of representation and reasoning about the world.
Supporting References: [94,114,115,131,132]

Competency 16 (Ethics)
Identify and describe different perspectives on the key ethical issues surrounding AI (i.e. privacy, employment, misinformation, the singularity, ethical decision making, diversity, bias, transparency, accountability).
Supporting References: [3,6,8,35,93,108,130,145]

Competency 17 (Programmability)
Understand that agents are programmable.
Supporting References: [45,47,79,80]

From Primary Education to Premium Workforce: Drawing on K-12 Approaches for Developing AI Literacy

Magnus Høholt Kaspersen
magnushk@cc.au.dk
Aarhus University
Denmark

Line Have Musaeus
lh@cs.au.dk
Aarhus University
Denmark

Karl-Emil Kjær Bilstrup
keb@cs.au.dk
Aarhus University
Denmark

Marianne Graves Petersen
mgraves@cs.au.dk
Aarhus University
Denmark

Ole Sejer Iversen
oiversen@cc.au.dk
Aarhus University
Denmark

Christian Dindler
dindler@cc.au.dk
Aarhus University
Denmark

Peter Dalsgaard
dalsgaard@cavi.au.dk
Aarhus University
Denmark

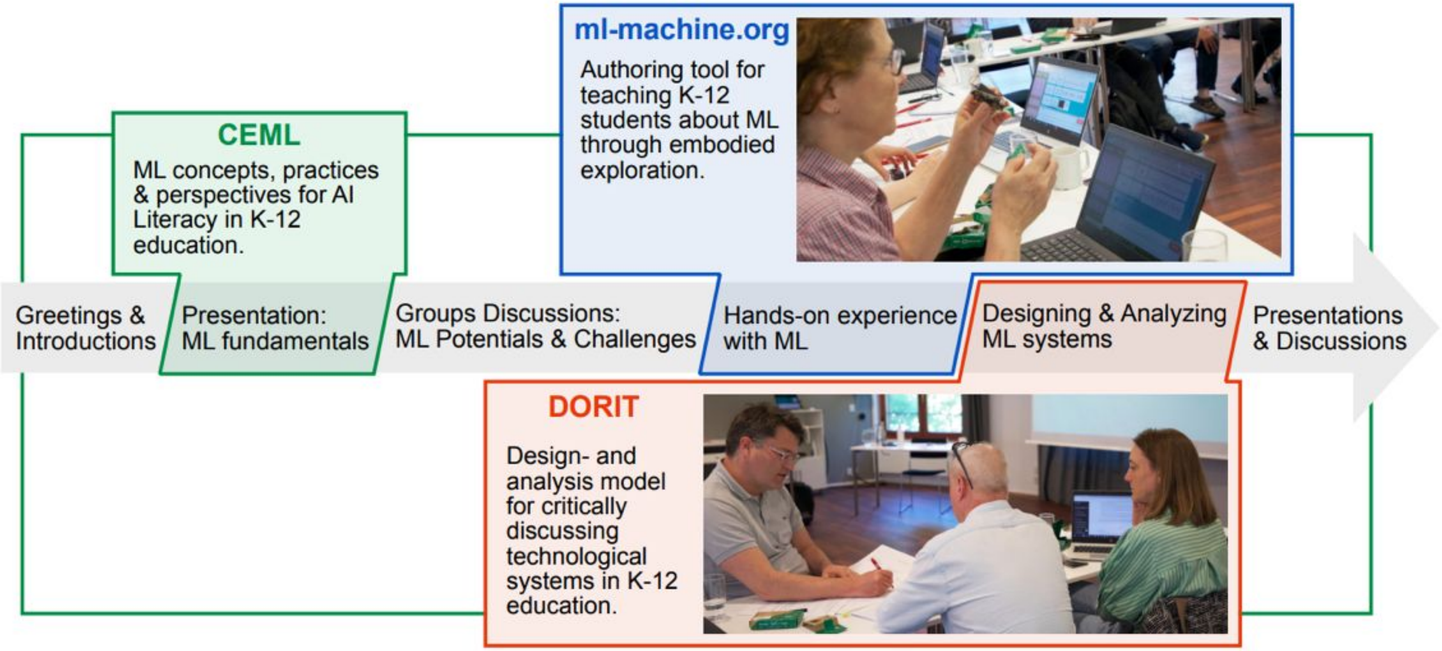


Figure 1: Format for the workplace-oriented AI-literacy workshop. The figure highlights where Child-Computer Interaction approaches were used and what for. The approaches used are the CEML-model [37], the DORIT-model [22], and ml-machine.org [8]. All participants were consented regarding appearing non-anonymized in the paper.

- Greater understanding of ML
- No improvement in self-efficacy, computational thinking, empowerment

Generative AI Literacy: Twelve Defining Competencies

RAVINITHESH ANNAPUREDDY*, Idiap Research Institute, Martigny, Switzerland and EPFL, Lausanne, Switzerland
ALESSANDRO FORNAROLI*, Idiap Research Institute, Martigny, Switzerland
DANIEL GATICA-PEREZ, Idiap Research Institute, Martigny, Switzerland and EPFL, Lausanne, Switzerland

This paper introduces a competency-based model for generative artificial intelligence (AI) literacy covering essential skills and knowledge areas necessary to interact with generative AI. The competencies range from foundational AI literacy to prompt engineering and programming skills, including ethical and legal considerations. These twelve competencies offer a framework for individuals, policymakers, government officials, and educators looking to navigate and take advantage of the potential of generative AI responsibly. Embedding these competencies into educational programs and professional training initiatives can equip individuals to become responsible and informed users and creators of generative AI. The competencies follow a logical progression and serve as a roadmap for individuals seeking to get familiar with generative AI and for researchers and policymakers to develop assessments, educational programs, guidelines, and regulations.

CCS Concepts: • **Social and professional topics → Computing literacy**; • **Computing methodologies → Philosophical/theoretical foundations of artificial intelligence**; • **Human-centered computing → HCI theory, concepts and models**.

Additional Key Words and Phrases: Generative AI Literacy, AI Literacy, Data Literacy, Generative AI, Prompt engineering, AI competencies, AI skills

1 Introduction

With the rapid spread of Artificial Intelligence (AI) systems across all domains, the concept of *AI literacy* ("a set of competencies that enables individuals to critically evaluate AI technologies"[61]) has become increasingly important and necessary in the past few years. The legislative work on the European AI Act [32] done by the European Parliament and European Commission has also contributed to increasing attention towards the risks and challenges posed by systems and tools based on AI models, as well as viable ways to regulate them [41, 42].

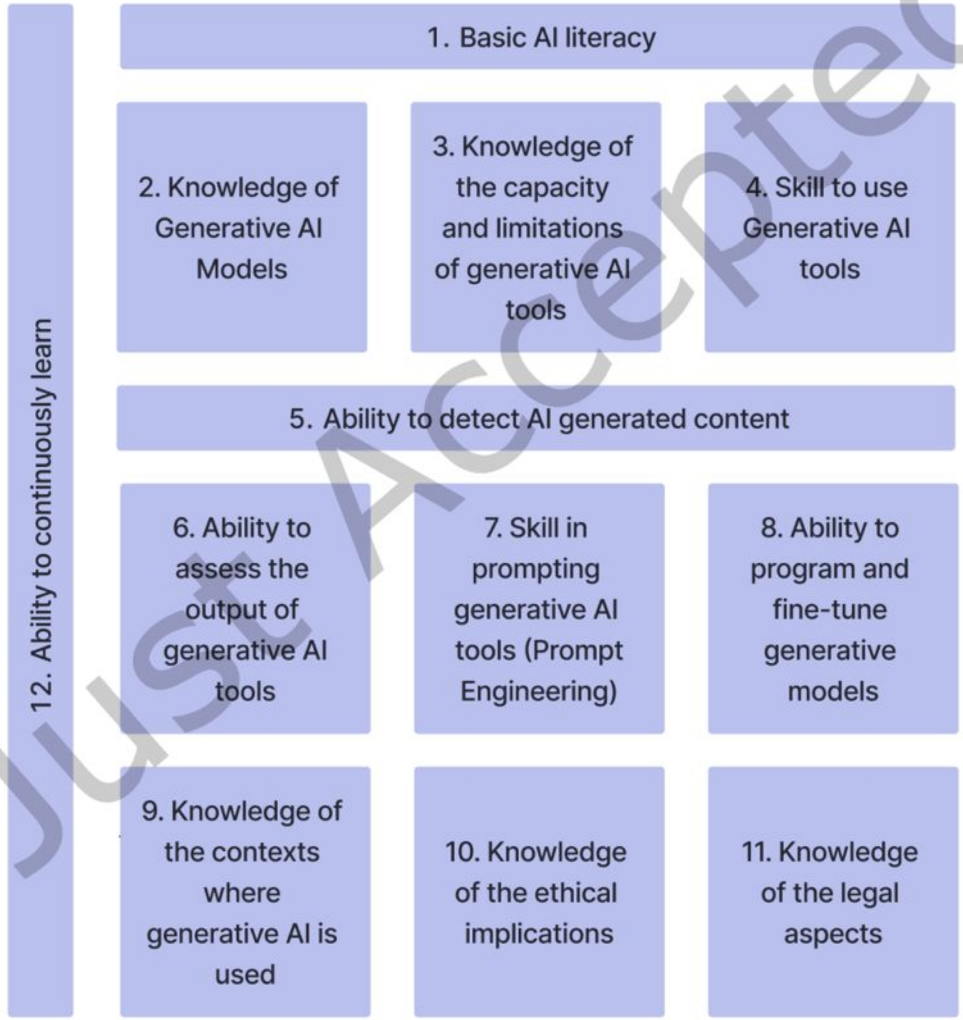
Generative models have found applications across many sectors, reflecting their adaptability and potential impact [58, 78, 81]. As governments worldwide increasingly digitalize their operations and services, there is a growing intersection between AI and governance. Generative AI technologies can transform communication, public engagement, and decision-making processes with and within governmental bodies [13]. Understanding the implications, challenges, and opportunities presented by generative AI is vital for researchers and practitioners in the field of digital government to make informed decisions about its usage and adoption. At the same time, a literate workforce is better equipped to identify and mitigate potential risks, ensuring that the deployment of generative AI in government processes is accompanied by risk assessment and mitigation strategies. With the

*Both authors contributed equally to this research.

Authors' Contact Information: Ravinithesh Annapureddy, Idiap Research Institute, Martigny, Valais, Switzerland and EPFL, Lausanne, Vaud, Switzerland; e-mail: ravinithesh.annapureddy@epfl.ch; Alessandro Fornaroli, Idiap Research Institute, Martigny, Valais, Switzerland; e-mail: alessandro.fornaroli@alumni.epfl.ch; Daniel Gatica-Perez, Idiap Research Institute, Martigny, Valais, Switzerland and EPFL, Lausanne, Vaud, Switzerland; e-mail: daniel.gatica-perez@epfl.ch.

ACM 2639-0175/2024/8-ART
https://doi.org/10.1145/3685680

A Competency-Based Model for Generative AI Literacy



International Society of the Learning Sciences

Children's Overtrust and Shifting Perspectives of Generative AI

Jaemarie Solyst, Carnegie Mellon University, jsolyst@andrew.cmu.edu
Ellia Yang, Carnegie Mellon University, elliay@andrew.cmu.edu
Shixian Xie, Carnegie Mellon University, shixianx@andrew.cmu.edu
Jessica Hammer, Carnegie Mellon University, hammerj@andrew.cmu.edu
Amy Ogan, Carnegie Mellon University, aeo@andrew.cmu.edu
Motahhare Eslami, Carnegie Mellon University, meslami@andrew.cmu.edu

Abstract: The capabilities of generative AI (genAI) have dramatically increased in recent times, and there are opportunities for children to leverage new features for personal and school-related endeavors. However, while the future of genAI is taking form, there remain potentially harmful limitations, such as generation of outputs with misinformation and bias. We ran a workshop study focused on ChatGPT to explore middle school girls' (N = 26) attitudes and reasoning about how genAI works. We focused on girls who are often disproportionately impacted by algorithmic bias. We found that: (1) middle school girls were initially overtrusting of genAI, (2) deliberate exposure to the limitations and mistakes of generative AI shifted this overtrust to disillusionment about genAI capabilities, though they were still optimistic for future possibilities of genAI, and (3) their ideas about school policy were nuanced. This work informs how children think about genAI like ChatGPT and its integration in learning settings.

Table 1
Guessing Game: User Inputs and ChatGPT Outputs

User Input	ChatGPT Output	Answer	Why We Selected
(Q1) "Does [study city] have the most bridges in the world?"	There are other cities with more bridges.	Correct	Grounded example in learners' context
(Q2) "Compute 32874*34918"	1147010 (shows multiplication process)	Incorrect	Common example from news
(Q3) "List related papers on machine learning"	A list of seven papers with title, author, year of publication	Incorrect	Common example from news
(Q4) "How do I add a line break to a comment in Google docs?"	A four-step process on what keys to press in what order.	Incorrect	Exploratory procedural question
(Q5) "When did Ohio fight Pennsylvania?"	Ohio and Pennsylvania have never fought a war.	Correct	Incorrect or ambiguous input

- **Use text-based generation tool**
  - Educational slides + demo
  - Generate generate a gift list for girls and boys
- **Limitation**
  - Guessing game
- **Imagining Future Applications with GenAI**
- **Policy discussion**
  - Accessibility of ChatGPT

Text-to-Image Generation

"A pretty girl with a strawberry hat" "African people at work" "European people at work"

Figure 1: Images created using TTIG algorithms demonstrating the algorithms' harmful stereotypes

Co-design workshop with Teachers
(Ali et al., 2024)

- **Use Text-to-Image Generation tools**
  - Prompt techniques
  - Create visual stories, self-portrait, dreams
- **Technical understanding**
  - Animation of diffusion & CLIP
  - Explore database
- **Ethical implications**
- **Develop AI learning materials** focusing on TTIG for their students.

Middle and High school AI curricula

GRADES: 6-12 NEW

Generative AI for Humanities

This unit includes two standalone lessons — one on writing with AI and one on researching with AI—designed for Humanities classes to demystify ethical and effective chatbot use, encouraging students to explore and reflect on these technologies.

Duration: Two, 45-minute lessons

Explore unit

GRADES: 6-12

Coding with AI

The Coding with AI unit teaches strategies for using AI to simplify complex concepts, guide problem-solving, and even generate code, empowering students to become informed and ethical future coders.

Duration: Five 45 minute lessons

Explore lessons

GRADES: 7-12

Societal Impact of Generative AI

Investigate the impact of generative AI from different perspectives, then collaborate as a team to come up with guidelines that address the most needs from all participants.

Duration: 1 hour

View lesson plan

https://code.org/ai

Choose from the following activities:

a Essay Introduction

Use your AI tool to write an introduction for an essay about the benefits of learning a second language.

b Story Idea Generation

Generate three creative story ideas for a science fiction short story.

c Grammar and Style Correction

Improve the grammar and style of the following sentence: "Running fast the dog through the park chasing the frisbee."

d Argumentative Essay Points

List three strong arguments against the use of single-use plastics.

e Poem Creation

Write a short poem about the changing seasons.

f Research Summary

Summarize the key findings of a recent study on the effects of screen time on children.

g Email Draft

Draft a formal email to a teacher requesting an extension on a project deadline.

h Creative Writing

Write a creative scene where a character discovers a hidden door in their house.

▾ Lesson 2: Research Process

In this lesson, students explore the use of AI chatbots and search engines for research purposes, comparing their effectiveness in finding quotes, verifying historical claims, and answering scientific questions. They engage in hands-on activities to discern the strengths, limitations and ethical considerations of both tools. Through interactive reflection stations, students critically evaluate when and why to use each tool, fostering their skills as informed digital citizens.

- 1 Verifying Quotes
- 2 Using a Search Engine to Find Information
- 3 Using an AI Chatbot to Find Information
- 4 End of Lesson Survey

2. **Crafting your prompt:** Start by deciding on the specific aspects you want the AI to focus on. These might include themes (e.g., survival, exploration), settings (e.g., distant planets, future Earth), or characters (e.g., robots, aliens).

   Consider how specific you need to be to get the type of story ideas you want.

5. **Evaluate the response:** Consider the following questions when evaluating the AI's response:

6. **Refining your prompt:** If the AI's response is not as expected, think about how you can improve your prompt. You might need to add more detail or be more specific about what you want.

Foundations of Generative AI

This unit aims to build a foundational understanding of text-based generative AI models, focusing on core concepts over technical skills. Students will demystify generative AI models by exploring their internal structures through the familiar lens of input, storage, process, and output. They will gain insights into how these models represent language, the impact of training data on model performance, and the potential for bias. Using this knowledge they will be presented with scenarios throughout the unit where they can help educate individuals who feel powerless or lack agency in how AI is impacting their lives, or respond to individuals who have only read the hype headlines and offer feedback or criticism based on their knowledge of how these AI systems work.

Warm Up

In the glimmering **glump** of Flimflam, every **blibber** clutched their **flagress** blibbertwig. These blibbertwigs, **flagress** in the **glump's** gleam, **zibbled flagressly** under the Flimflam sky. During the **flagress** festival, **blibbers** and blibbertwigs **zibbled** in **flagress** sync, weaving a dance of **flagress** jaggleshot.

1. Where is the glimmering glump?

2. What did the blibber clutch?

3. What did the blibbertwigs do under the Flimflam sky?

4. What happened during the flagress festival?

- **Lesson 1: Introduction to Generative AI**
- **Lesson 2: Input & Training Data**
- **Lesson 3: Bias in the Machine**
- **Lesson 4: Understanding Embeddings**
- **Lesson 5: Embeddings: How They're Created**
- **Lesson 6: Understanding Neural Networks**
- **Lesson 7: Neural Networks: How They're Trained**
- **Lesson 8: Attention Is All You Need**
- **Lesson 9: Outputs & Probabilities**
- **Lesson 10: Hallucinations and Fabrications**
- **Lesson 11: Project: Demystifying Generative AI**

https://studio.code.org/s/foundations-gen-ai-2024

- Lesson 1: Introduction to Generative AI
- Lesson 2: Input & Training Data
- Lesson 3: Bias in the Machine
- Lesson 4: Understanding Embeddings
- Lesson 5: Embeddings: How They're Create
- Lesson 6: Understanding Neural Networks
- Lesson 7: Neural Networks: How They're Tra
- Lesson 8: Attention Is All You Need
- Lesson 9: Outputs & Probabilities
- Lesson 10: Hallucinations and Fabrica
- Lesson 11: Project: Demystifying Gene

Big Idea #2: Representation and Reasoning

6-8	9-12
LO: Explain how word embeddings (which are feature vectors) represent words as sequences of numbers. **EU:** Word embeddings are a key part of neural natural language processing, including machine translation (e.g., Google Translate) and text generation systems (BERT, GPT3, etc.) **Unpacked:** Each word is a point in a feature space with many dimensions, organized so that words with similar meanings are close to each other in the feature space. See this Word2VecDemo.	**LO:** Describe how a transformer network operates. **EU:** Transformer networks map sequences of input words to sequences of output words, where words are represented as feature vectors. **Unpacked:** Neural network natural language processing applications such as machine translation or question answering are driven by word embedding representations, which are feature vectors. Words are fed in one vector at a time, and the network delivers its output one vector at a time. **Activity:** https://app.inferkit.com/demo

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Neural Networks *(Structure of a neural network)* **3-B-i**	**N/A**	**LO:** Illustrate how a neural network of 1 to 3 neurons is a function that computes an output. **EU:** A neural network uses one or more neurons working together to form a function. Each neuron takes a set of numbers as input and produces a single number as its output. **Unpacked:** A neural network is a collection of neurons that are connected to each other. Every neuron has a set of input connections, each with an attached weight. Each input connection carries a value. The neuron multiplies each input value by the connection weight to produce a weighted input. The sum of all the weighted inputs is compared to the neuron's threshold value. If the sum is above the threshold value, the neuron outputs a 1; otherwise it outputs a 0. The output value can be used as an input for other neurons. **Actrivity:** Calculate the output of a single neuron with multiple inputs, or a network of two multi-input "hidden" neurons feeding a single output neuron. Such networks can compute simple functions such as "AND", "OR", or "at least 2 out of 3". For a quick tutorial on neural nets for grades 3-5, see https://docs.google.com/document/d/1bYs0tTiL44sQhsMADgU2bDmVjWaVKVl2pT_SYZTzEwI/edit#heading=h.g640mybwbie6	**LO:** Illustrate the structure of a neural network and describe how its parts form a set of functions that compute an output. **EU:** Neural networks are organized as layers of units (input, hidden, and output layers), with weighted connections between units in successive layers. Each unit computes the sum of its weighted inputs. It passes that sum through a transfer function to produce a numeric output. **Unpacked:** A neural network maps input patterns to output patterns in a complex way. Each neuron computes a function, and the network as a whole computes a complex function that can be considered a very wiggly mathematical function.	**LO:** Describe the following neural network architectures and their uses: feed-forward network, 2D convolutional network, recurrent network, generative adversarial network. **EU:** Feed-forward networks can learn arbitrary functions and are used for both classification an regression. 2D convolutional networks learn small "kernels" that are convolved with the input, and max-pooling layers to reduce image resolution; they are used for image analysis. Recurrent networks have feedback connections and are used for language processing. Generative adversarial networks have generator and discriminator modules and are used to create deepfakes.
Neural Networks *(Weight adjustment)* **3-B-ii**	**N/A**	**LO:** Demonstrate how weights are assigned in a neural network to produce a desired input/output behavior. **EU:**The behavior of a neural network can be altered by adjusting its weights.	**LO:** Demonstrate how a learning rule can be used to adjust the weights in a one-layer neural network. **EU**: During training, weights are adjusted in response to errors in the network's output, so that an error will be less likely when the input is seen again. **Unpacked:** Training can be done using binary units and a simple learning rule for adjusting the weights (such as the perceptron learning rule in the "Will this dog bite me?" exercise).	**LO:** Train a multilayer neural network using the backpropagation learning algorithm and describe how the weights of the neurons and the outputs of the hidden units change as a result of learning. **EU:** A neuron's weights start out as small random values and evolve to a more precise pattern through learning. The changes in the neuron's weights are computed by a learning rule driven by a back-propagated error signal. The neuron's weight pattern determines the features that the neuron detects. **Unpacked:** Students are not expected to know the details of the backpropagation learning algorithm, only that error is propagated backward from later layers to earlier ones. **Activity:** An online demo such as TensorFlow Playground can be used to visualize the changes in weights during learning.

About Us Our Work Opportunities Blog Resources

Why Artificial Intelligence Belongs in English Class

The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)

Exploring Artificial Intelligence in English Language Arts with StoryQ

Jie Chao[1], Rebecca Ellis[1], Shiyan Jiang[2], Carolyn Rosé[3], William Finzer[1], Cansu Tatar[2], James Fiacco[3], Kenia Wiedemann[1]

[1]Concord Consortium, 25 Love Lane, Concord MA 01742

[2]North Carolina State University, Raleigh, NC 27695

[3]Carnegie Mellon University, Pittsburgh, PA 15213

jchao@concord.org,[1] rellis@concord.org,[1] sjiang24@ncsu.edu,[2] cp3a@andrew.cmu.edu,[3] wfinzer@concord.org,[1] ctatar@ncsu.edu,[2] jfiacco@cs.cmu.edu,[3] kwiedemann@concord.org[1]

Abstract

Exploring Artificial Intelligence (AI) in English Language Arts (ELA) with StoryQ is a 10-hour curriculum module designed for high school ELA classes. The module introduces students to fundamental AI concepts and essential machine learning workflow using StoryQ, a web-based GUI environment for Grades 6-12 learners. In this module, students work with unstructured text data and learn to train, test, and improve text classification models such as intent recognition, clickbait filter, and sentiment analysis. As they interact with machine-learning language models deeply, students also gain a nuanced understanding of language and how to wield it, not just as a data structure, but as a tool in our human-human encounters as well. The current version contains eight lessons, all delivered through a full-featured online learning and teaching platform. Computers and Internet access are required to implement the module. The module was piloted in an ELA class in the Spring of 2022, and the student learning outcomes were positive. The module is currently undergoing revision and will be further tested and improved in Fall 2022.

However, in the current school curriculum, opportunities to learn AI concepts and practices are scarce. Computer science (CS) courses, where AI content is considered a natural fit, are only offered in some U.S. high schools. They also have persistent diversity issues (Code.org et al., 2021), mainly because the focus of CS is typically on aspects that rely on advanced math rather than an interdisciplinary approach that would create opportunities for engagement among a more diverse student population. Furthermore, most CS courses do not include an AI unit. Only recently, research groups have started to develop and research AI teaching strategies at the K-12 level (e.g., Glazewski et al., 2022; Lee et al., 2021), and curriculum providers have developed AI content as an optional unit (e.g., Code.org n.d.).

But AI education can extend beyond CS courses. AI is a

Representation: Feature extraction and space

Figure 1: StoryQ visualizes features extracted from any document.

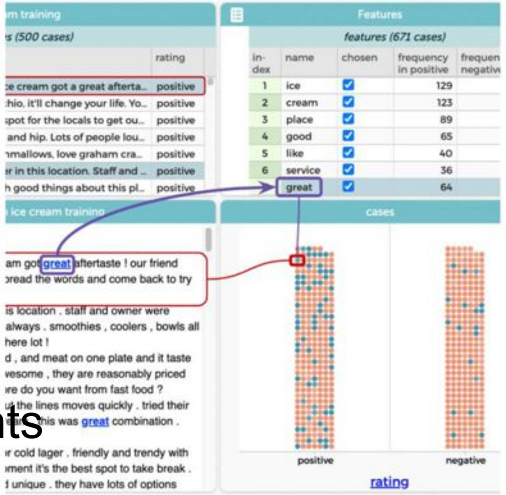


Figure 2: StoryQ generates a feature table that is dynamically linked with training data and feature distribution graph.

Machine Learning: Feature Weights

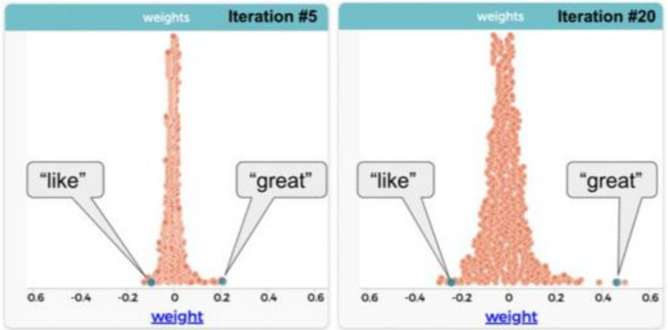


Figure 3: StoryQ visualizes how feature weights change as ML algorithm runs through iterations.

Reasoning:

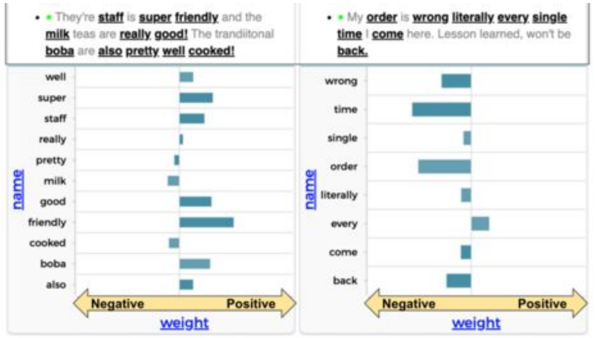


Figure 5. StoryQ visualizes model reasoning as bar graphs.

Learning Accelerators · Published Aug 14, 2024 · 4 min read

Explore new AI features in Learning Accelerators

By Microsoft Education Team

Reading Coach

Give your students engaging, individualized reading fluency practice. Reading Coach provides guidance on the correct pronunciation and syllabification, visuals to help vocabulary recall, and positive reinforcement when students pronounce words correctly.

Learn more

Reading Progress

Tracks students' reading skills, give educators actionable insights quickly, and focus students on specific areas for improvement with Reading Progress. By streamlining the reading assignment creation, review, and analysis process, educators can spend more of their time on active instruction.

Learn more

Reflect

Help students build their emotional vocabulary and express feelings in a safe, fun way, while giving educators the insights they need to provide active support with the Reflect tool in Microsoft Teams.

Learn more

Search Coach

Improve information literacy and teach students how to search effectively with Search Coach, built right into Microsoft Teams for Education. Students learn to ask effective questions, find reliable sources, and identify credible sources while safely navigating the web. Educators can get insights into their students' search habits to better inform instruction.

Learn more

Search Progress

Track the development of information literacy skills by evaluating students' search activity and behaviors and identifying ways to improve the quality of their search queries over time using the Search Progress app.

Learn more

Speaker Coach

Evaluate students' public speaking performances and give personalized feedback on details like pitch, use of filler words, and pacing without the stress of an audience using the real-time Speaker Coach tool.

Learn more

Speaker Progress

Monitor data on student public speaking skills in the Speaker Progress app, with analytics from Speaker Coach. Track how presentation skills are improving at the individual, class, grade, and school levels and free up time for active instruction.

Learn more

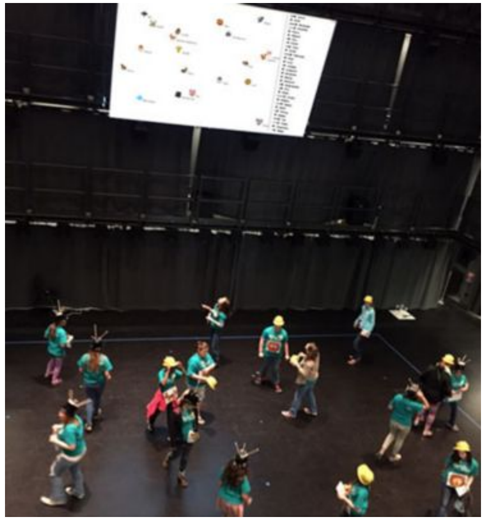
New opportunities of embodied learning for AI literacy

Embodied

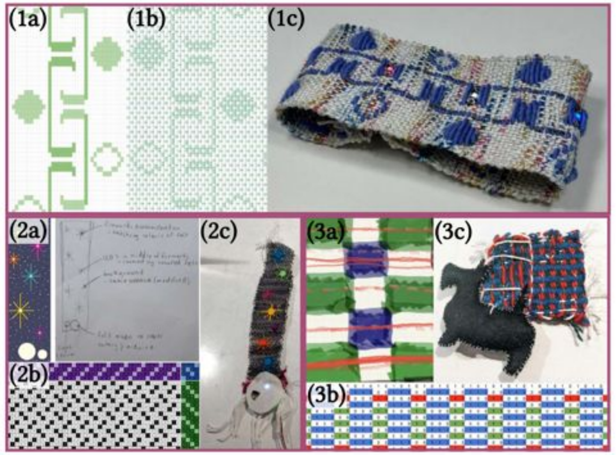
- Connect ***unfamiliar abstract*** concepts with ***familiar sensorimotor*** experiences (Lakoff & Johnson, 1980).

Balance Board Math system
(Tancredi et al., 2022)

Draw2code
(Im & Rogers, 2021)



Embodied visual analytics
(Chen et al., 2017)



SPEERLoom
(Speer et al., 2023)

Tabletop ANN
(De Raffaele et al., 2018)

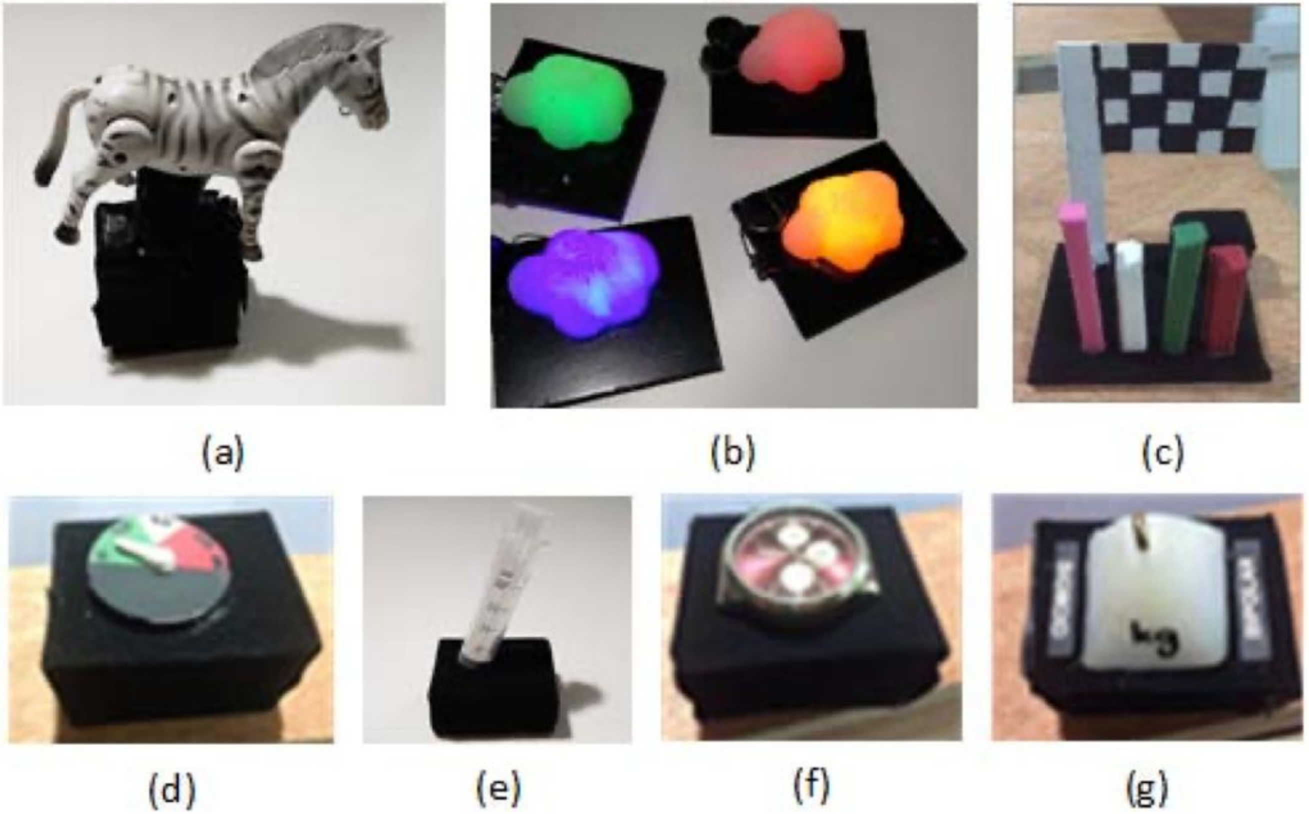


Figure 2. Active tangible objects contextualized for ANN operations including;
a) Horse - Context Simulator Controller,
b) Clouds – Hidden Layer nodes,
c) Finish Podium – Output Visualization,
d) Speedometer – Input Speed Value,
e) Syringe – Input Health Value,
f) Chronograph – Output Time Value,
g) Weight – Synapse Weight Adjustment.

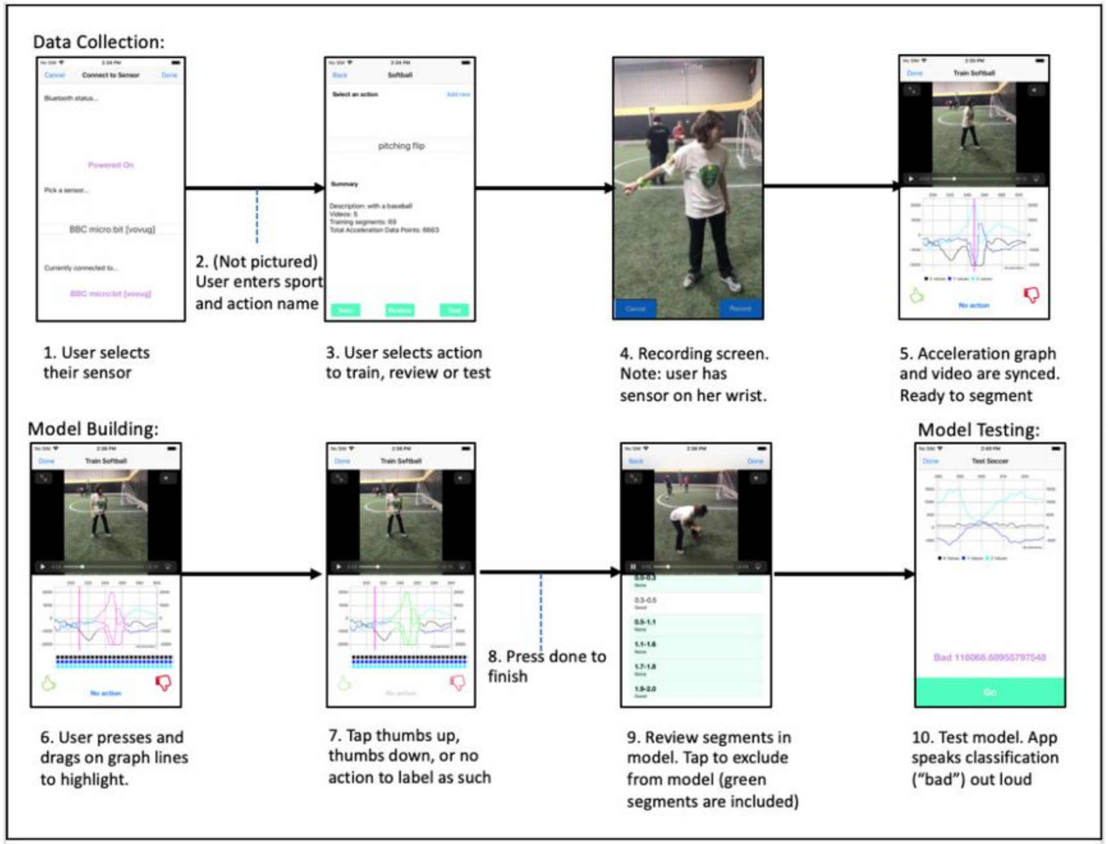


Figure 1: Data Collection, Model Building and Model Testing User Experience

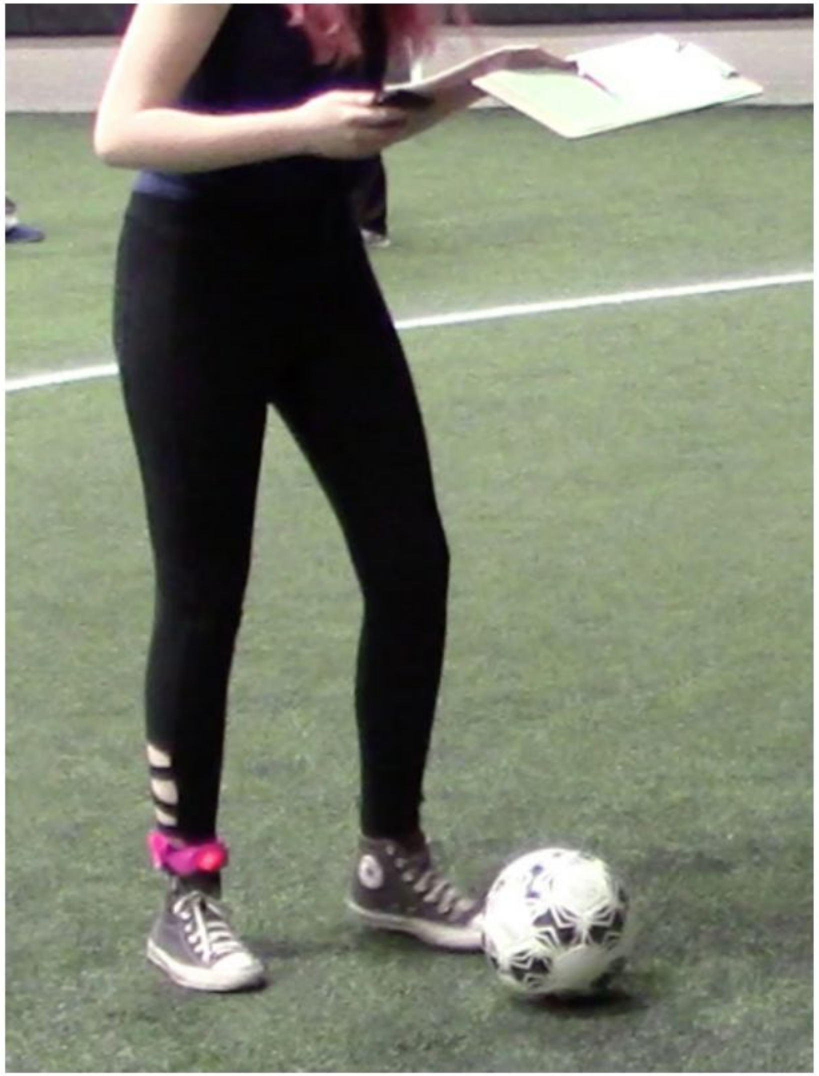
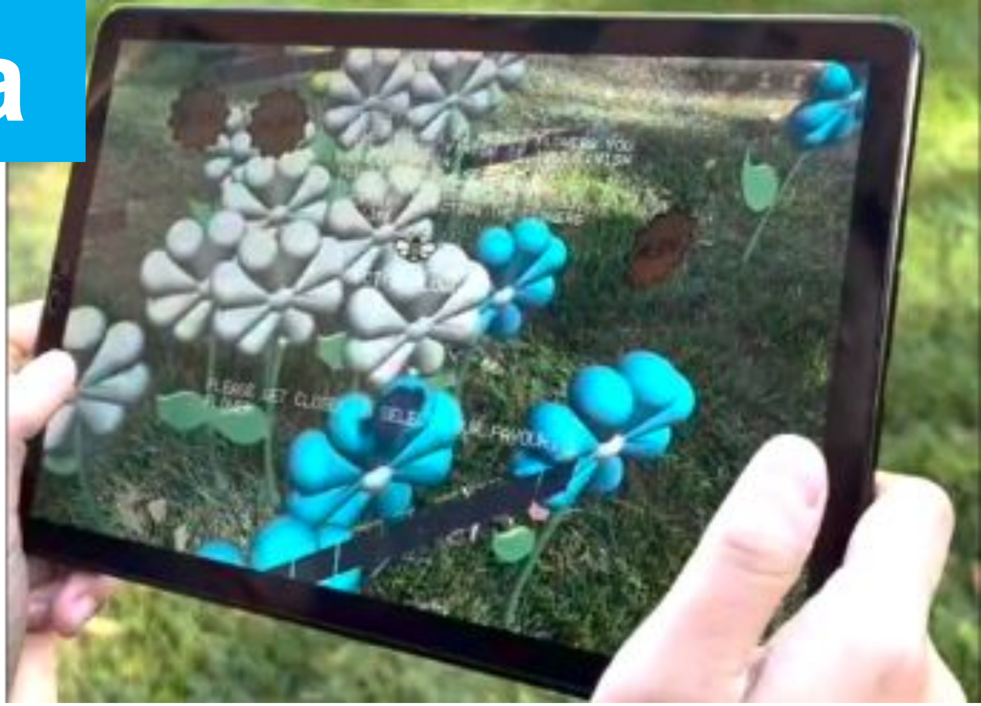


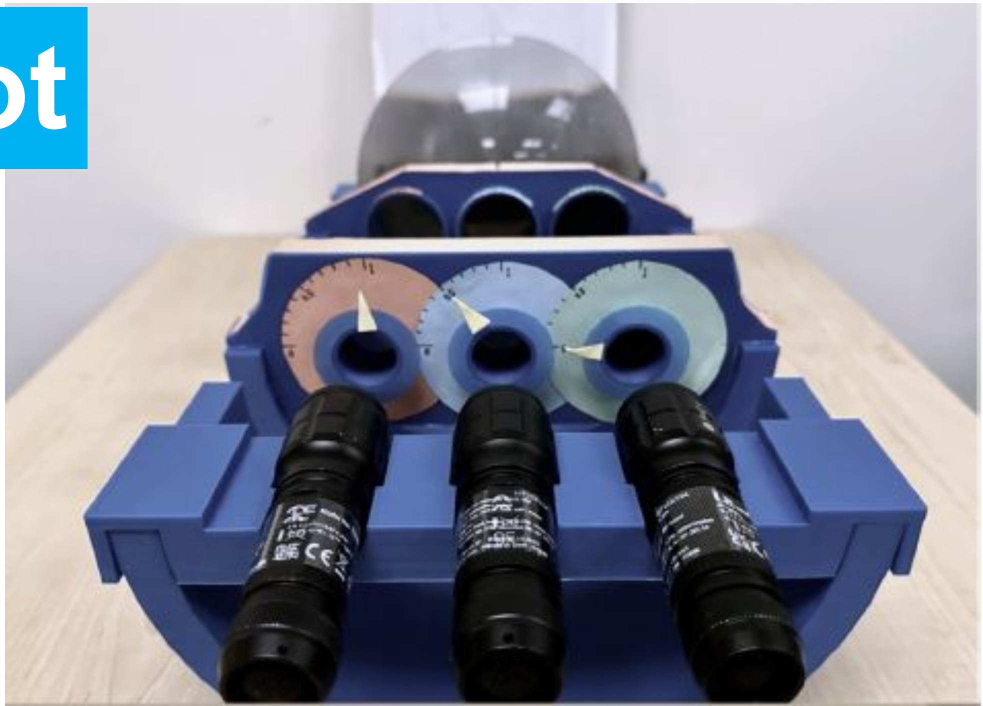
Figure 2: Participant with sensor placed on ankle, clipboard with worksheet, phone with app, and soccer ball during the soccer tutorial.

Athletic Move for ML training
(Zimmermann-Niefield et al., 2019)

BeeTra
p



OptiDot



Discussion

- To what level do we need to demystify (Gen)AI for learners (rapid technological evolution vs. fundamental concepts) ?
- Interdisciplinary AI literacy (math, computing, literacy, art, social science, etc.).
- How can embodiment help address the many opportunities and gaps?