Beyond Black-Boxing: Building Intuitions of Complex Machine Learning Ideas Through Interactives and Levels of Abstraction

Brian Broll [a], Shuchi Grover [b], Derek Babb [c]

[a] Vanderbilt University, [b] Looking Glass Ventures, & [c] University of Nebraska Omaha

Why Teach Machine Learning Concepts in High School?

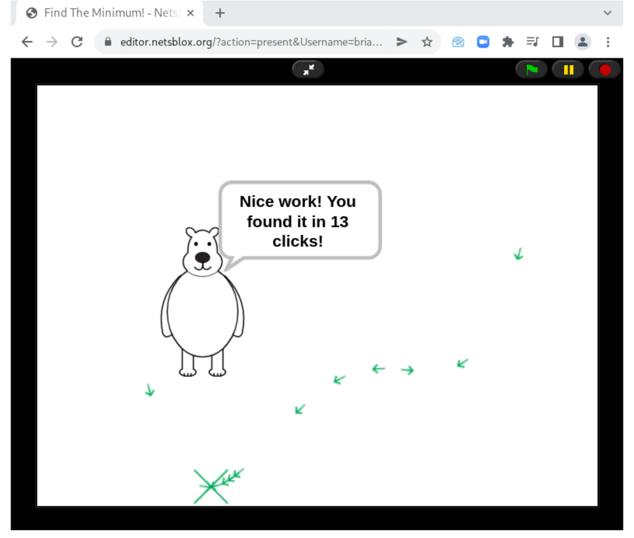
- Artificial Intelligence and Machine Learning (AI/ML) are ubiquitous and impact our daily lives
- AI/ML is related to many social issues and has far-reaching ethical & political implications
- Developing a better intuition of fundamental ML concepts enables students to be better equipped to reason about the AI/ML-powered tools they interact with every day
- This should better prepare them for making well-informed decisions about the inclusion/usage of AI/ML in their daily life including both the benefits and potential pitfalls

Design Approach

- We draw inspiration from past work in turtle geometry by pioneers such as Abelson and diSessa that made sophisticated ideas in mathematics and physics accessible to young learners
- Introduce core ML ideas such as classification, optimization, gradient descent, and adversarial examples
- Activities include unplugged, non-programming, and programming activities in NetsBlox (an extension of Snap!)

Pre-programmed Games

- Used to introduce fundamental ML topics in an interactive, engaging way prior to programming activities
- One example is the "Find the Minimum" game (shown below) in which students embody an optimization algorithm and try to find the minimum value of an unknown function. The goal is to develop preliminary intuition about how the gradient descent algorithm works before the subsequent coding activities.

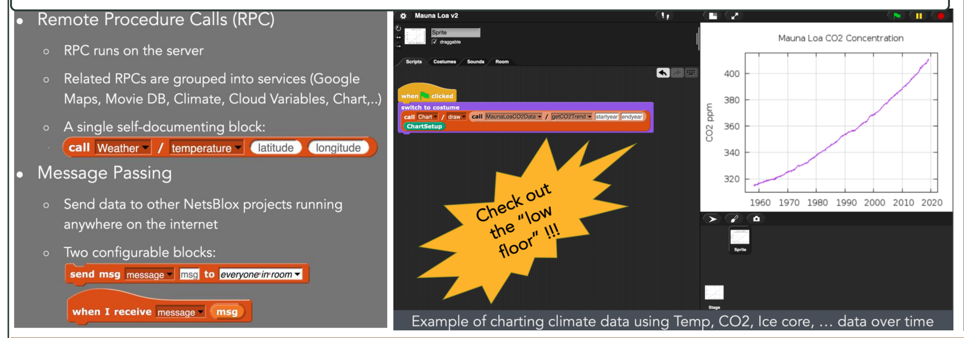


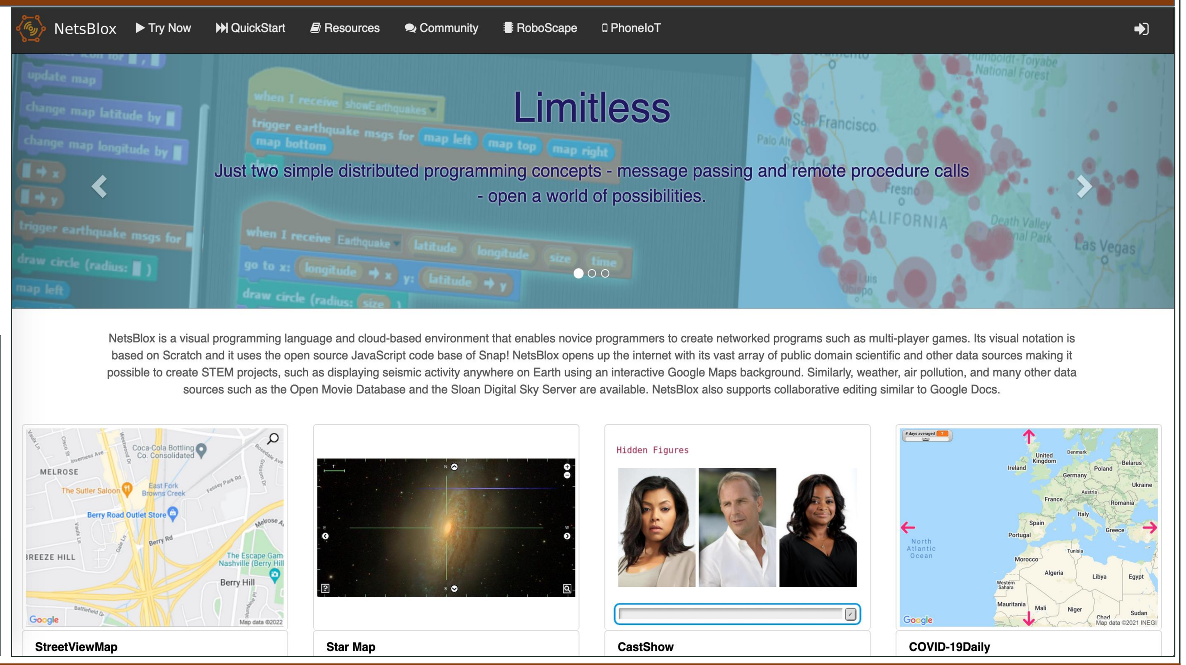
Can complex ML concepts be made more accessible through interactive tools, pre-programmed games, and carefully scaffolded activities?

NetsBlox as a Tool to Explore Advanced Ideas in Computing

About NetsBlox (netsblox.org)

- Blocks-based programming environment based on Snap!
- Open-source, web browser-based
- Provided distribute computing blocks
  - Remote Procedure Calls (RPCs)
  - Message passing
- Low-floor, high-ceiling, wide-walls

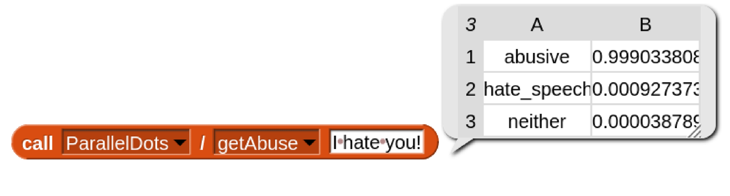




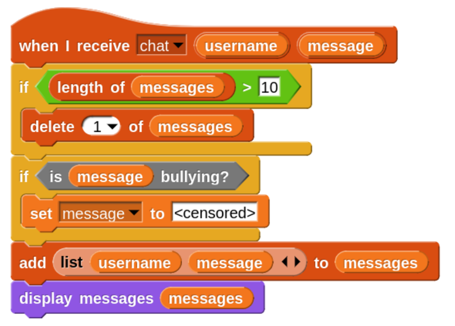
Sample Activities

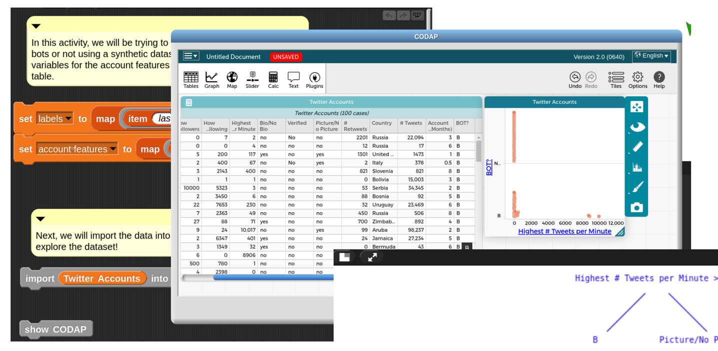
Cyberbullying (Classification)

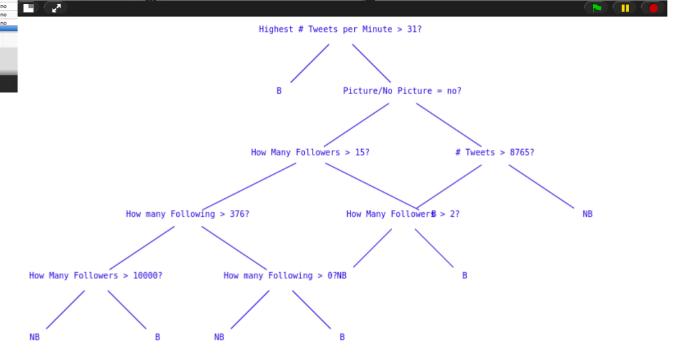
- Extend an existing chat application with moderation features using a pretrained model
- Introduce fundamental concepts including classification, predictions, model confidence, and natural language processing
- Uses the ParallelDots API for NLP which includes abusive text detection, sentiment analysis, emotion detection, and more
- Easily customizable by students into own interests



Using pretrained models from ParallelDots can be invoked with a single block (above). This block can then be used in a bigger script (shown left).



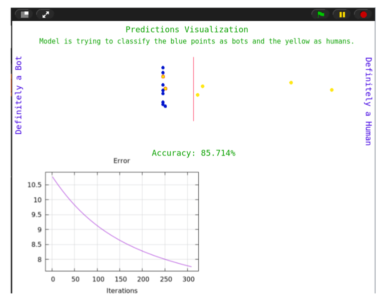


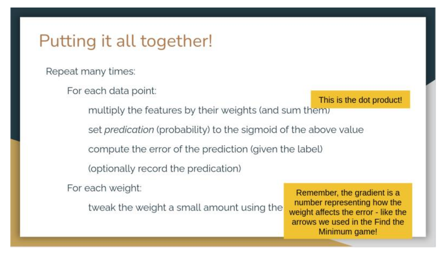


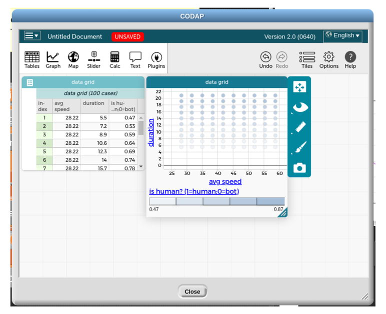
Twitter data can be explored using CODAP (shown above) then a decision tree is learned – and visualized (shown right).

Bot Classification (Decision Trees)

- Given a synthetic Twitter dataset, can you train a model to identify bots?
- Students use CODAP to explore the dataset and write their own rules for identifying bots
- Then, we reflect on our own approach and connect that to decision tree building
- After talking through the pseudocode, students first complete the subgoals and then the low-level implementation.
- Finally, the decision tree is visualized and discussed



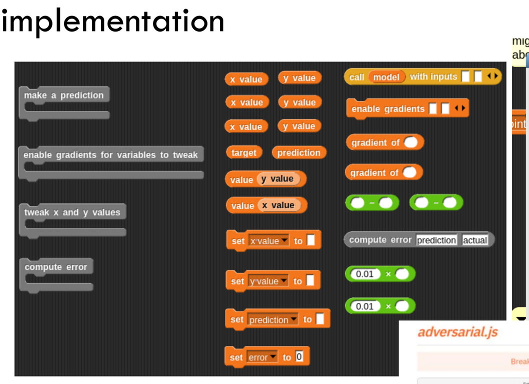


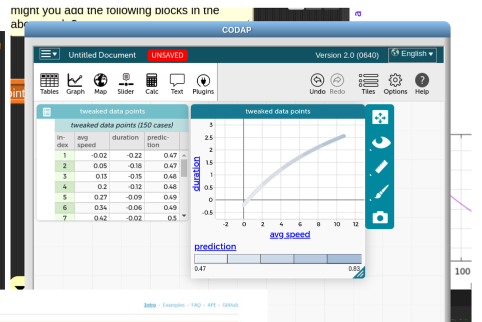


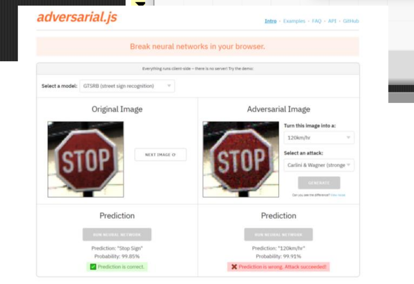
Interactive visualization included in the starter template (shown above). The pseudocode for the algorithm is shown in the top right. After training a model, it is evaluated on a bunch of counterfactual ("made up") points on the right to explore what it learned.

Detecting Bot Account Registrations (Gradient Descent)

- After first creating a bot to create fake accounts on an example website, students train a model to predict if a signup was a human or bot based on mouse movement data
- This follows the "Find the Minimum" game and optimizes a simple function which predicts a class based on weighting the average speed and duration
- As before, we start with pseudocode and talking through the algorithm then proceed to subgoals and low-level implementation







Subgoals and low-level implementation shown above as a Parson's problem. The updates made to the point to fool the model shown in the top right. An interactive web-based tool for generating adversarial examples on traffic sign data shown right.

Fooling Bot Account Detectors (Adversarial Examples)

- Builds on the bot detection activity but optimizes the data point instead of the model
- Follows a similar approach with pseudocode, subgoals, then low-level implementation (shown left)
- Afterwards, we explore an interactive web-based tool for generating adversarial examples with traffic sign data and discuss real-world implications

Integrated Data Exploration with CODAP

- CODAP empowers students to explore data interactively and discover underlying patterns
- Using CODAP from within NetsBlox enables students to leverage the complementary strengths of each: NetsBlox is used for data manipulation and preprocessing; CODAP is used to visualize and inspect the dataset
- An example usage is shown in the decision tree activity (in Sample Activities) where the students manually construct a rule-based classifier using CODAP to identify meaningful rules

Programming at Multiple Levels of Abstraction

- In order to enable learners of all abilities to succeed in engaging with non-trivial ML algorithms, we employ "levels of abstraction" as a scaffolding tool
- An algorithm is first introduced in pseudocode, then subgoal-like design blocks are provided for implementing the algorithm. Finally, Parson's problems are used for code completion at the lowest level
- An example for creating an adversarial example is shown in the sample activities.

Preliminary Feedback

- Feedback was collected following 2 preliminary workshops consisting of a total of 12 high school teachers
- Feedback was positive overall and has helped guide refinement of the activities. For example, the subgoal blocks were well received in the first workshop leading to increased use in other activities.

Example Feedback:

- "Your program is unique in that students really do dig deeper which is what I love."
- "the GANs were tough to grasp."
- "Thank you for doing this! I am not a fan of block coding, but I changed my position by the end of this workshop."

Ongoing Work and Resources

- We are planning on piloting the activities in the fall in HS cybersecurity courses
- For more information, check out:
  - https://netsblox.org
  - https://cyberai4k12.org

This material is based in part upon work supported by the National Science Foundation under Grant 2113803. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.