Teaching by Induction: Project-Based Learning for Silicon Valley

Ravi Krishnan Jagannathan[1], Claire Komives[2]

[1]Vice President, Data Science, Plethy Inc.
[2]Chemical & Materials Engineering, San Jose State U.
[1]Ravi@soumya.com
[2]Claire.komives@sjsu.edu

Abstract: It is well known that there continues to be a shortage of women in technology jobs. The teaching methods and gender bias in K-12 education continue to favour boys progressing to engineering and technology positions. The project initiated by Ravi Krishnan Jagannathan in Silicon Valley, California, sought to prepare middle school and high school girls for success in high tech. The strategy used was to teach math and science inductively. The use of computers for teaching the girls both fundamental science and math in the middle school laid the foundation for teaching high school girls to apply Artificial Intelligence (AI) to develop solutions to current medical and ergonomic problems. This paper will discuss the inductive teaching strategy with the aim of motivating both parents and teachers to adopt a similar project that can help address the shortfall of women in tech.

Keywords: Inductive teaching, Women in technology, Artificial Intelligence (AI)

A. Introduction

There persists an imbalance between men and women in the high tech workforce. Only about 20% of the high tech workforce in the US is composed of women, even though about 50% of the workforce is women (Bose 2018). In India the percentage of women in tech is only slightly higher at 28%(Srinivas and Bansal 2018). In a recent interview with Catherine Ashcraft, the Director of Research and Senior Research Scientist for the National Center for Women & Information Technology (NCWIT), the causes of the shortage of women were cited to be societal influences and bias, workplace systems and also teaching methods at the primary and secondary levels (Welson-Rossman 2018).

Primary and secondary education continues to follow primarily the rote teaching method which is unfavourable to most learning styles in young people (Welson-Rossman 2018). In addition, teachers tend to call on boys more often than girls. For college admissions, a new trend is to reward students for demonstrated experience rather than potential. Thus, programs for girls at the middle school and high school levels should provide the opportunity to gain experience that can equip them for success in achieving their aspirations for a high tech career.

A recent article summarized the literature on teaching strategies in Science, Technology, Engineering and Mathematics (STEM) education that support the highest success for student learning (Thibaut et al. 2018). Open ended problems that the students themselves identify are the gold standard for learning STEM, and in particular the T in STEM. However, it has been shown to be more effective to lead the students up an inclined plane of project/problem complexity. When teachers present projects initially that are somewhat structured and utilize the students' skills, they are more apt to succeed at more challenging and open ended projects.

From 2014 to 2019, Ravi K. Jagannathan gave private classes to a group of mostly female students with the aim of giving them experience in programming and Artificial Intelligence (AI). The goal was to open the opportunity for the girls to pursue a career in high tech. Two of the participants were accepted to University of California Berkeley with the intent to study computer engineering.

In addition to various on-line tools described throughout the text, on-line courses were used to help design the program (EdX 2019; Dwarknath 2019; Sra 2019; Amazon 2019b).

B. Methodology

Students in middle school were introduced to python programming using the turtle function. Turtle is a built in

Corresponding Author
Ravi Krishnan Jagannathan, Plethy, Inc., San Jose, CA
Ravi@soumya.com

library that does not require any installation. It can be moved on the screen using simple go up, go down, go left, go right commands. By moving the turtle on the screen, students were taught variables, functions, lists, objects, and for-loops that exist in all programming languages. Basic Cartesian Geometry was also needed to solve some of the problems. (Papert 1980).

In addition, students were enabled to explore the basic concepts of Newtonian physics, also with the turtle. Through this course, they learned how to represent the real world in the form of equations. Using the same turtle, but with a focus on velocity and movements by writing programs to simulate throwing the turtle and watching it fall using Newton's laws of Motion, students learned physics inductively. They then combined the programming and the physics to develop an *Angry Birds* game as a final project.

The advanced programming involved coding a "Hangman game," which showed the students concepts of programming a server hosted in the cloud and playing a multi-player game, via a browser over the internet. The course also covered networking concepts such as the use of IP addresses and routers (Downey 2019).

Finally, a course on logical argument was taught using games as a basis for teaching truth tables and propositional logic. All of the courses were entitled *Fun with X* to engage the students. (Smullyan 1992)

Starting in 2016, with their basic understanding of computer programming, physics, and logic, the students were given an introduction to the interface of hardware and software. They learned the fundamentals of Internet of Things (IOT) and wearable devices. This work prepared them to design and assemble their final project. They were able to develop a medical system comprising a wearable device. The device was to measure physical variables such as acceleration and temperature and was to be used to diagnose or treat a real medical problem.

The girls were asked to identify a real life problem, and then to build a small wearable Arduino device with sensors to address the problem. The instructor introduced to the adafruit.com sensor package which has various low-cost sensors that could measure temperature, capacitance, acceleration, air pressure, magnetic direction (compass), and moisture. A number of projects done by other hobbyists on the website were described to the girls. They were then left alone to discuss among themselves to identify a real world problem that was solvable with the given kit.

The girls came up with the following interesting problem. Babies under one year old often feel too hot or cold and start crying. Poor communication results in parents misdiagnosing the problem, such as to feed the child. The goal of the project was to communicate the actual source of the discomfort related to temperature.

The instructor-led discussion thereafter focussed on collecting the parts to build an effective solution. A flora board was used as shown in Figure 1, which has a small CPU, a built in LED, and a sensor to read the temperature.

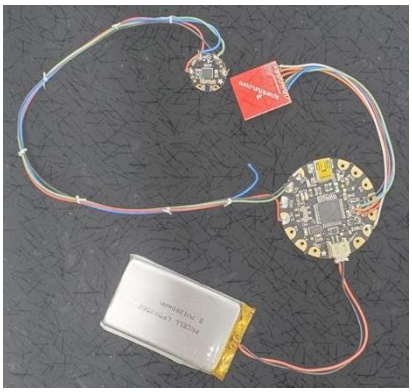


Figure 1. Sensor system

A program written in Arduino-C running on the CPU is capable of reading temperature measured by the sensor. The same program could blink the LED. The solution to the Baby problem, was to program the board so that the LED would blink fast (twice per second) when hot and steadily when cold. The students followed the instructions on the Arduino site to set up the device. They were able to program the correct temperature thresholds: greater than 80°F that identified hot and less than 55°F as cold. We tested this by bringing the device close to hot water and ice. The instructor identified the Adafruit platform containing numerous components, and then the girls correctly identified the 3-4 components required to solve the problem. The first version of programming was done by the instructor but the program was then modified by the students for setting temperature thresholds. Thus the instructor enabled the students by providing the learning opportunity for the students and initiating the coding.

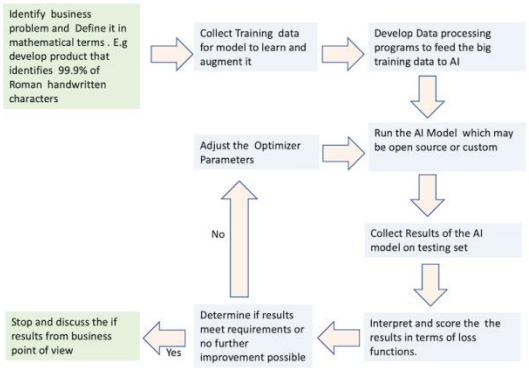
In addition to the technical development, the students learned some of the business aspects. Defining minimum viable product (MVP) or refining a concept to its simplest form were example exercises. The students were presented with basic challenges such as whether or not a battery is required and if so, what size of battery is needed? Discussions about sales options were also carried out. For example, should just one device with one temperature threshold or rather five devices with five different thresholds be sold?

Students also practiced market research by taking their product to parents and friends and asking them how they would use it. Did they see a need for it and if so, how much would they pay for it?

As a final presentation, the students presented the device to parents and other participants. The presentation involved pitching the device, as a start-up company would, and soliciting initial investment.

1) *Artificial Intelligence (AI) Course*

From 2017, the students were in high school and they were introduced to Artificial Intelligence. The earlier courses were the foundation for the AI courses. As there are numerous areas in AI, too many to cover them all, convolutional neural network (CNN) was chosen, as it deals with pictures. Pictures are more attractive to a wider group of young students than other types of data such as time series data and text. The students were challenged to use their



programming skills and combine it with statistics and mathematics to make an AI model that could recognize pictures.

As in the case of the IOT project, the teaching strategy was to lead the girls to do as much as they could, while the instructor contributed the complex parts. The skills covered in detail are those which allow an AI practitioner in industry to use an AI model, and evaluate the functionality of the model, without knowing all the internal details of the model. The strategy for AI development is shown in Figure 2.

Figure 2. AI Development Stages

The students learned enough to be able to prepare the images needed for training data. The statistics completed by the students included identifying the loss function, for example, if there are 90% correct predictions and 10% wrong, is it acceptable or not. Concepts such as confusion matrix, specificity, sensitivity and accuracy were introduced. Loss functions such as Jaccard, Hinge loss and cross entropy were also taught and programmed. This course gave a comprehensive introduction to the critical set of skills used in real world AI projects.

The actual programming of the CNN AI model was covered as theory, but not actually programmed by the students, as it would have been beyond the scope of the course.

At the end of the course, we reverse engineered an existing Keras AI model that solved diabetic retinopathy. Reverse engineering is to reproduce the product following detailed examination of its construction. From observation, the students understood the mission and strategy of the Keras program although they did not attempt to perform the complete programming. They were able to prepare data (images), feed it into the model and observe the output. Alternatively, there are many Keras models implemented in python that are solutions to various Kaggle competitions. Kaggle is a company that hosts learning tools for computational projects and competitions to solve machine learning problems (Kaggle 2019). The instructor could set up one of the solutions and let the students reverse engineer the solution including data manipulation and input presentation, students discussed various AI products on the market, broke them down to components, and presented how they might have been implemented.

Details of the course by module are described in the following paragraphs.

2) *Modules and Learning Outcomes of the AI Course*

The lesson plan was subdivided into the following modules data augmentation, big data processing, basic linear algebra, optimization, machine learning and cloud computing.

Training Data Augmentation: Developing an AI training set requires manipulation of images. The OpenCV library in python was taught for this purpose. This skill is used to create thousands of images for training the AI model from a smaller sample of hundreds of images. The students learned to select the following operations and perform these on images: Resize, flip, change to black and white, merge two pictures, cut slices out of the pictures. By iteratively performing multiple operations on the same image, they could generate hundreds of training images such as shown in Figure 3.

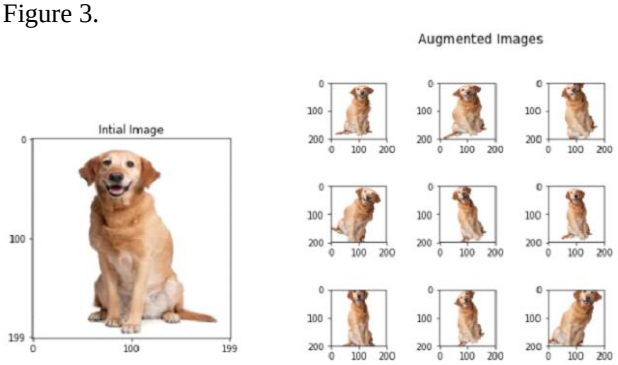


Figure 3. Training images (Contributors 2019a)

Big Data processing: The typical input to an AI or machine learning system is a large CSV file. It could have up to one hundred thousand rows or more of the following type.

"Figure #100, blood sugar level = 40, weight = 100, height = 175cms, male, age = 50, diabetes level = 3 where the last

column is what we are predicting, and the first column refers to picture saved on the disk.

To manipulate such large data sets, the python Pandas library (https://pandas.pydata.org/) was taught. Pandas allows us to manipulate training data that are formatted as very large CSV files, running to hundreds of thousands of rows. The students were able to take a large CSV file and convert it into a pandas data frame. Given a data manipulation requirement in English, they were able to break down the problem statement and determine the sequence of aggregations needed, and write the code using the pandas manual as reference. Then they were able to write code to perform the aggregation functions Grouping, sum, min, max, sort, call NumPy functions on a groups of columns, and take slices of data defined from starting row, ending row, starting column, ending column and filtering data based within a dataframe based on conditions.

Cloud Computing: Expensive desktops are not available to all. Therefore AWS was taught as an inductive step to overcome the need for each student to have a functional computer(Amazon 2019a). The students were able to explain the difference between cloud and a desktop. They were able to identify the pros and cons of using a cloud server versus a desktop. The learning objectives for this module included creating an AWS account, bringing up a server for a fixed time, installing pandas and open CV tools, as well as utilizing AI software frameworks. The students were able to run the programs developed in the previous two sections on the rented server and shut it down after use.

Matrix Manipulations and basic linear algebra: AI is mostly as massive set of matrix multiplication operations and basic statistics. The matrix manipulation skills are taught using a numerical python NumPy library (SciPy 2019). This skill is needed to read a Keras program and understand optimizers. The learning objective for this module was assembling the rows and columns of n dimensional arrays. They were able to describe a picture (three-dimensions + one colour dimension) as a three-dimensional matrix. They were able to translate the training data of images as N dimensional arrays, for example, an array of three-dimensional pictures, or an array of batches of three-dimensional arrays. The students were able to interpret the required input as a series of indexing, slicing and sorting operations on arrays. In some cases, the student was able to write code to perform the above manipulations. In most cases the students were able to read code and explain it back to the class.

Machine learning tutorials are found in Scikit Learn(Contributors 2019b). Two algorithms were covered as lecture: logistic regression (YES / NO Binary classification); and random forest. The students were taught to identify the correct amount of training data. They were able to discriminate between too few features, too many features, too little data, too much data (almost never), and evaluate the correct amount for that problem set. They were able to identify which problems were suitable for logistic regression and write a python program that ran a logistic regression classifier on given data sets We used the U.C Irvine machine learning repository for this section. They were able to compare the classifier against the human result in the data set. In order to perform the comparison, the concept of loss function was inductively learned. The students were able to compare the functions and demonstrate by implementing them in programs, that each loss function gave a slightly different success score to the classifier.

Optimization Algorithms The goal is to understand an optimizer called Gradient descent (Ruder 2019). The course referred back to turtle, and developed an algorithm that helped the turtle search for hidden food on the two-dimensional computer screen, as shown in Figure 4.

The students were able to extend the two-dimensional food search algorithm into three-dimensions. Gradient descent was taught using animated videos. The students were able to

Do until food is reached
1 Go forward 10 steps
2 If smell is weaker
Turn left 90 degrees
3. If reached food stop

Figure 4. Algorithm for hidden food search

extend the two-dimensional food search algorithm into three-dimensions. Gradient descent was taught using animated videos. The students were able to recognize gradient descent as an N dimensional search, and compare the turtle food search algorithm against Gradient descent. The parameters of Deep learning (epochs and learning rate) were taught. The students once again explained the learning rate parameter as the turtle's stride length. They were able to contrast benefits of using large strides (quicker to reach closer target but may miss it) versus small strides, (turtle gets distracted often). Further they were able to correlate parameters of the turtle algorithm to the parameters of the learning rate decay concept taught in the optimization algorithms section. They explained it back as turtle would move faster when food smell is weak and far off, whereas it would move slowly and cautiously when food is near (Wikipedia 2019).

C. Results and Discussion

The course over the years identified combinations of subjects. This cleared confusion arising from thinking of STEM as silos. True science is not pure math or pure biology or pure physics, it is a combination of subjects. Specifically, biologists could benefit from AI which is programming and math. So biology medicine stream is as interesting as a CS engineering or pure science stream.

Inductive learning helped students go beyond the textbook, as in "what should I learn" and "how should I learn" rather

than “I learn what is given to me”. This confidence has long lasting effect on the choices the students make in the university.

Instead of trying to solve a new problem, it was more useful to take an existing solution and then guess how it could have been engineered. For example, ask the class how the Amazon face recognition, which is an existing AI solution in the market, may work. Reverse engineering is the process of taking an existing product, breaking it down to its components, and reconstructing and identical product ourselves. It is a key step in inductive learning, leading to building confidence in “I can build anything”. Even though a student may not be ready to do the reconstruction part, they are often ready to deconstruct an existing solution.

Reverse engineering has another benefit as it is likely to succeed. One project that failed aimed to measure the theory that skin conductivity depended on hydration and measure the effect of various moisturizer products. The students summarized the scientific process of exploration by experimentation as an iterative process with no guaranteed result.

As a result of the AI course, students were able to articulate what professions would be replaced by AI and which could not. One such discussion focussed on whether it was worth spending ten years in studying medicine. This form of confident *look-ahead* thinking has a powerful impact on their career paths.

This was a multiyear project for a set of students in an extracurricular program. This content was designed to supplement their regular school work. Self-defined projects guarantee enthusiasm from students. Projects which were carried out only during the summer vacation, had a defined scope. The level of enthusiasm from the students was exceptional.

D. Conclusions

Students were given the opportunity to experience basic AI programming, aiding their entrance into good colleges to study computer engineering.

References

Amazon. "Amazon Web Services." https://aws.amazon.com/.
Amazon. "Training and Certification." https://aws.amazon.com/training/.
Bose, Shubhomita. 2018. "Only 20% of Tech Jobs are Held by Women, How About at Your Business?" In *Small Business Trends*. https://smallbiztrends.com/2018/03/women-in-technology-statistics.html: Technology Trends.
Contributors. "Machine Learning Mastery." https://machinelearningmastery.com/.
Contributors. "scikit-learn: Machine Learning in Python." https://scikit-learn.org/stable/.
Dwarknath, A. "Explore Courses: CME 193: Introduction to Scientific Python." https://explorecourses.stanford.edu/.
EdX. "Computer Vision and Image Analysis." EdX. https://www.edx.org/course/computer-vision-and-image-analysis-3.
Kaggle. "Kaggle Competitions." https://www.kaggle.com/.
Ruder, Sebastian. "An overview of gradient descent optimization algorithms." http://ruder.io/optimizing-gradient-descent/.
SciPy. September 5. "NumPy User Guide." The SciPy community, Accessed 2019. https://numpy.org/devdocs/user/index.html#user.
Sra, Suvrit. "MIT Open Courseware: Lecture 25: Stochastic Gradient Descent." https://ocw.mit.edu/courses/mathematics/18-065-matrix-methods-in-data-analysis-signal-processing-and-machine-learning-spring-2018/video-lectures/lecture-25-stochastic-gradient-descent/.
Srinivas, Arjun, and Samarth Bansal. (2018), "Gender gap in Indian formal sector worse than global average, LinkedIn data shows." In *Hindustan Times*. New Delhi: HT Media Ltd.
Thibaut, Lieve, Stijn Ceuppens, Haydée De Loof, Jolien De Meester, Leen Goovaerts, Annemie Struyf, Jelle Boeve-de Pauw, et al. (2018). Integrated STEM Education: A Systematic Review of Instructional Practices in Secondary Education. Review of. *European Journal of STEM Education* 3 (1).
Welson-Rossman, Tracey. (2018), "The Systems Holding Back Women In Tech." In *Forbes.com*. Forbes Media LLC.
Wikipedia. 2019. "Learning Rate." Wikipedia, The Free Encyclopedia, Accessed September 5. https://en.wikipedia.org/w/index.php?title=Learning_rate&oldid=898255673.