Moon Rose Fish: a handwriting practicing tool for children

Trudie Bosse
Eindhoven University of Technology
Den Dolech 2
5600 MB, Eindhoven
g.o.h.m.@student.tue.nl

Josefine Funnekotter
Eindhoven University of Technology
Den Dolech 2
5600 MB, Eindhoven
j.o.funnekotter@student.tue.nl

Kaz Voeten
Eindhoven University of Technology
Den Dolech 2
5600 MB, Eindhoven
k.k.voeten@student.tue.nl

Abstract

To make learning to handwrite more playful we came up with a concept. In this concept children learn their handwriting skills in a game. By making use of a sigmoid function working on the Neural Network, this game can recognize the letters that are written and give feedback on the writing. From the practical experiment we can conclude that the concept is feasible. Yet, we do to validate our concept in context, for example by talking with teachers, children and parents or doing a user test.

Authors Keywords

Machine Learning; Neural Network; Reinforcement Learning

ACM Classification Keywords

H: Information Systems; H1.2: User Machine Systems

Introduction

Many can probably remember their writing classes in elementary school. Repeating a letter, in order to make it perfect. Although the booklets were usually illustrated, the activity always stayed the same.

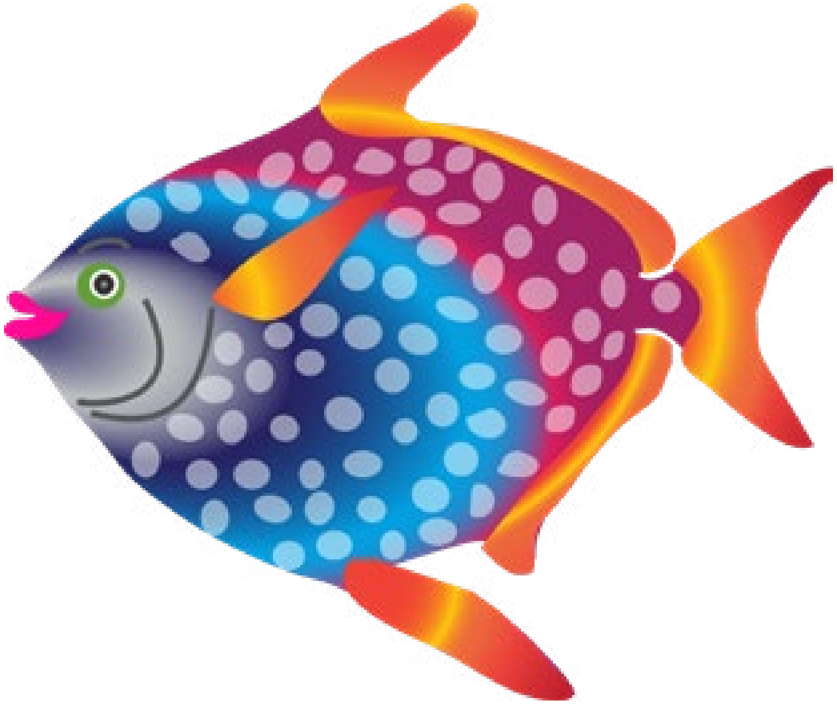
Along with the world, education is digitizing as well. Schools have started using laptops instead of books, starting a very young age already. With these fast developments, it seems like, in years, it would not even be necessary to train the skill of handwriting. Though, practicing handwriting remains really important. The performances of young children in school can be indicated by their handwriting skills, as they show clearly what kind of skills are lacking. [1] This is because handwriting involves many important skills, e.g. visual perceptual skills, understanding left to right progression, visual motor skills and many more. These will help children improve their communication skills, reading and written expression. [2]

Also, studies showed that children who had stronger fine motoric skills, developed by writing, tend to become more active when they are adults. [3] Being active is beneficial for your health and therefore it seems that handwriting affects you positively in this area, too.

Concept

Learning handwriting with MoonFish!

Our final concept is a tool to practice Handwriting playfully. This game can be played at home to give children more opportunities to learn how to write, or at school to provide the teachers with more time for other personal feedback.
In a game, Children learn to handwrite with the help of their buddy Moon Rose Fish.
In this game, children write different letters with a pencil on a pad. The game detects if the letter is written correctly by using the neuron network.
The Moon Fish gives support and feedback. If a letter is written wrong, the system shows how it is supposed to be.
If the game detects that a child keeps writing a letter in a way that it looks like another, the exercises are adapted to learn how to distinguish the two letters.
The progress bar shows how far the Children are with their exercises. When the player can write a letter, the Moon Fish celebrates dancing.



Reinforcement learning algorithm

In our concept, we used reinforcement learning. This chapter describes what the inputs and outputs are and which algorithm is used.

F: X -> Y Unknown target function
The function should recognize the letter which is given as input (X). It should see if it is written correctly and return the letter that is written (Y). In our prototype, we work with only 3 letters. Therefore the return is 1,2 or 3. If we work with the whole alphabet, the return would be 1,2, ………, 26.

Training examples
The algorithm trains itself with the dataset that defines how a letter should be written. This dataset is made by writing the letters. In our prototype, this is done at the start of the programme. In our concept, the creation of this dataset is done as part of the development of the game.

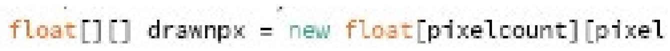
Learning Algorithm
The learning algorithm that we used is the sigmoid function working on the Neural Network.

Final Hypothesis
The final results of the algorithm are a value (1,2,3 for the prototype and 1,2,3,….26 in the concept) and the percentage for this value. The percentage indicates how accurate the estimated value is.

The Code

In order to define our letters in code we setup a dataset as follows:

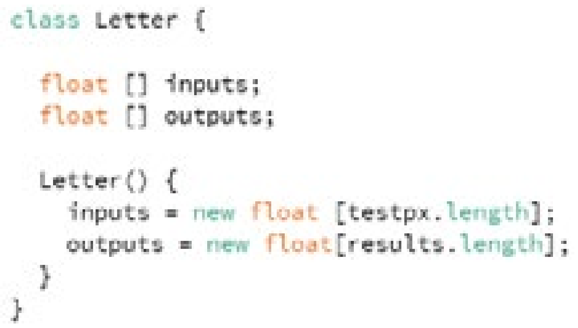
Firstly we create a grid of 15 rows and 15 columns. Where 15 is the amount of pixels horizontally and vertically on the grid (var pixelcount). Giving us a total of 15 * 15 = 225 pixels. We represent said grid in the code as a two dimensional array:



```
float[][] drawnpx = new float[pixelcount][pixel
```

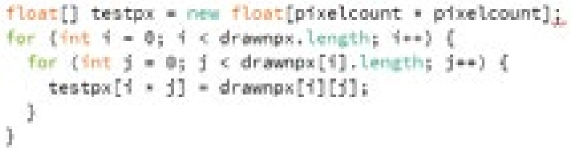
By default we set each value to -1.0 in this array. This represents a white pixel on our grid. If the mouse button is held down, we check if the mouse is in any block of this grid. If that's the case we set the value to 1.0. This way we can draw in our grid.

The Neural network we use utilizes objects with a dataset and answer for training purposes. The object we called 'Letter' holds two float arrays:



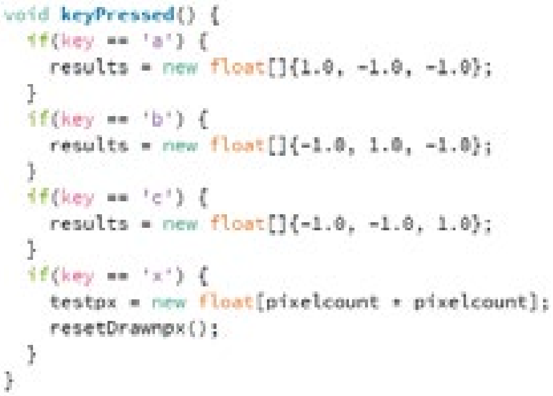
```
class Letter {

  float [] inputs;
  float [] outputs;

  Letter() {
    inputs = new float [testpx.length];
    outputs = new float[results.length];
  }
}
```

Here the input array is the dataset. We construct this dataset from our drawnpx grid by converting it to a singular linear array by iterating over each row's (i) array's indexes (j) and casting the value into an array that's linear. The process can be represented as follows:



```
float[] testpx = new float[pixelcount * pixelcount];
for (int i = 0; i < drawnpx.length; i++) {
  for (int j = 0; j < drawnpx[i].length; j++) {
    testpx[i * j] = drawnpx[i][j];
  }
}
```

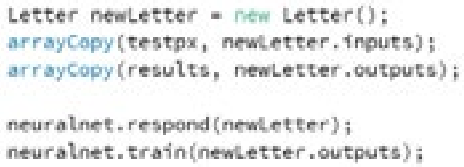
This results in the array 'testpx' which we set as the 'inputs' array in our Letter object. As for the outputs in our Letter object we construct a float array based on the letter trained. For each letter from "a" to "c" we have an array where the correct answer is 1.0 and the wrong answers are -1.0. This is set manually though the following function:



```
void keyPressed() {
  if(key == 'a') {
    results = new float[]{1.0, -1.0, -1.0};
  }
  if(key == 'b') {
    results = new float[]{-1.0, 1.0, -1.0};
  }
  if(key == 'c') {
    results = new float[]{-1.0, -1.0, 1.0};
  }
  if(key == 'x') {
    testpx = new float[pixelcount * pixelcount];
    resetDrawnpx();
  }
}
```

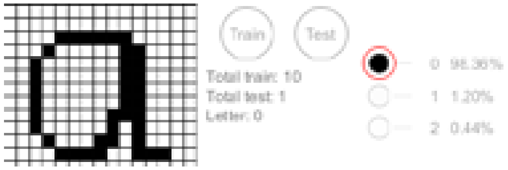
Where the array 'results' is the outputs array in our 'Letter' object. We use this letter object to train our Neural Network. This is a relatively simple process. After setting up our Neural Network, which is done by initializing it with 3 variables. 1: the length of the input dataset, 2: the size of the hidden layer, 3: the size of the outputs. We're using a NeuralNet by Alasdair Turner, so we don't have to program the inner workings of this network and only configure it properly for our purpose. For our hidden layer we chose a size of 100. Through trial and error we found this to be a good number to quickly reach accurate results with relatively few training. After training roughly 150 letters we started getting accurate results from the network. In theory adding more layers could even further improve accuracy but might take more computational power slowing down the program. With a hundred we noticed no delay and had satisfying results.

Training is as simple as drawing the letter, pressing the letter it represents on the keyboard and clicking the 'Train' button. A Letter object will then be constructed and fed to the network like so:



```
Letter newLetter = new Letter();
arrayCopy(testpx, newLetter.inputs);
arrayCopy(results, newLetter.outputs);

neuralnet.respond(newLetter);
neuralnet.train(newLetter.outputs);
```

For guessing you just press the "Test" button instead of the "Train" button in our program:



Total train: 10
Total test: 1
Letter: 0

● 0 98.36%
○ 1 1.20%
○ 2 0.44%

The amount of pixels, results and hidden layers can easily be expanded in order to create a more complex system. The current setup however proves that this concept is feasible and can after a little trainer very well guess which letter has been drawn.

Video of concept and prototype
https://vimeo.com/252685711

Conclusion

We can conclude from our practical experiment that our concept is very much feasible. The results after small training effort (drawing letter for about 10 minutes) are rather accurate already. Meaning that in a final product we can afford a few days training with different handwriting and have a system that pretty accurately estimates which letter from the alphabet has been drawn. This means our game concept can certainly work. By adding more complexity through pixel count (allowing more detailed drawings) and increasing the size of the hidden layer we can maintain his accuracy in a full scale program. This confirms that selecting a Neural Network as the answer to our machine learning problem statement was the correct decision. As it resulted in a situation where the machine can predict the answer from a rather complex dataset very accurately.

Discussion

To make the game work, a lot of data should be acquired, and that could take a lot of time. It is possible to have a big group of people write letters for one day, to get many different types of handwriting. Although it is faster than training the system with one person, it is still not the most efficient way. In addition, you would need to be sure that at least a part of the group has a neat handwriting. The goal is to teach children to write the letters correctly, thus it would be more convenient to get data from an online data set, including different types of correct letters. Such data sets are available online, so having found a fitting one, this would ease the workings of our prototype a lot.

To come to this concept, we didn't speak to any teachers to find out where they stand. If we proceeded with it, a good starting point would be to validate the concept with teachers and children. Further, the product should be tested with a user-test, to see how children would use it. One of the elements of good writing where the game does not give feedback on is someone's posture. Because this is an essential element of learning how to write, it might be worth to look into this feature of writing to see if it can be part of the game.
Comment on this concept could be that not many people have excess to a pen and a pad. Therefore people need to buy or rent one, and the question is if they would find it worth it to buy one.

Acknowledgements

We would like to thank Rong-Hao Liang for his lectures, materials and guidance.

References

1. Integrated Learning Strategies, Learning Corner (2015). Handwriting Results Show Importance of

Child’s Fine Motor Skills Development. Retrieved Jan 17, 2018, from: https://ilslearningcorner.com/2015-10-what-your-childs-handwriting-and-fine-motor-skills-may-tell-you/

2. H. Greutman (2017). Which Fine Motor Skills Are Important For Handwriting. Retrieved Jan 17, 2018, from: https://www.growinghandsonkids.com/which-fine-motor-skills-important-for-handwriting.html
3. [3] L. M.Barnett M.P.H. (2008). Childhood Motor Skill Proficiency as a Predictor of Adolescent Physical Activity. Retriever Jan 17, 2018, from https://www.sciencedirect.com/science/article/pii/S1054139X08002954#!