Neural Networks and Handwriting Recognition

Steven Sloss
Math 164

26 April 2007

Presentation Outline

Artificial Intelligence

- Today's computers can perform many computations much, much faster than a human being can.

Example

- Integrate

$$\int_0^1 \sqrt{1-x^2}\,dx$$

- My laptop: 0.3867... seconds.
- Me: $\sim$1.3 minutes

Artificial Intelligence, Contd.

- There are many areas where computers fall short, however.

Artificial Intelligence, Contd.

- There are many areas where computers fall short, however.

Example

- Find the swingset:

- My 2 year old neighbor: $\sim$1.2 seconds
- A computer: ???

Artificial Intelligence, Contd.

- Artificial Intelligence is the field of mathematics and computer science that tries to give computers human-like cognitive abilities.
- **Neural Networks** are an important way to do this.

Presentation Outline

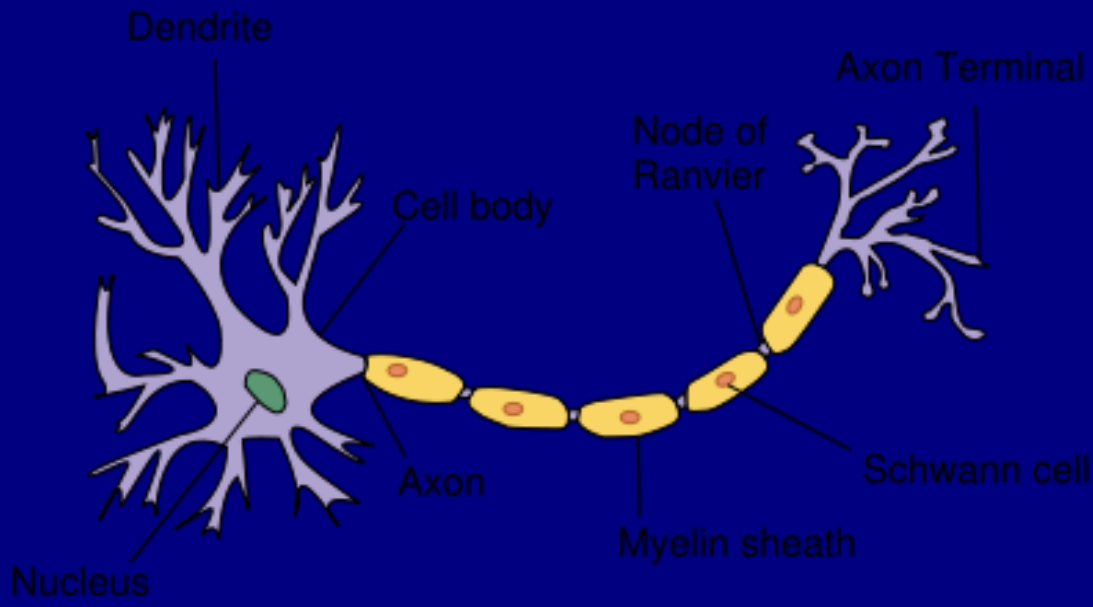
Neural Networks

- Neural networks - teaching a computer to do pattern recognition like a human brain!

Biological Neural Networks versus Artificial Neural Networks

- Lots of parallels between artificial and biological neural networks.
  - Both biological and artificial neural networks use neurons.

Biological Neural Networks versus Artificial Neural Networks



- **Biological neurons**:
  - Accepts signal from Dentries.
  - Upon accepting a signal, that neuron may fire
  - If it fires, a signal is transmitted over the neuron's axon, leaving the neuron over the axon terminals
  - This signal is then transmitted to other neurons or nerves

Biological Neural Networks versus Artificial Neural Networks

- **Artificial neurons**: Artificial neurons are based on digital systems (computers) rather than analogue systems (dentries),
  - Receives a number of inputs (from other neurons or the program itself)
    - Each input has a weight
  - Each neuron has a activation threshold

Solving Problems with Neural Networks

Problems not suited to neural networks

- Deterministic problems
- Programs that can be written with a flowchart
- Where the logic of the program is likely to change
- Where you must know how the solution was derived

Solving Problems with Neural Networks

Problems suited to neural networks

- Problems that can't be solved as a series of steps
- Pattern recognition
- Classification

The Neuron

The basic building block of the neural network.

- Individual neurons are connected to one another
- Each connection is assigned a weight.
- These connection weights determine the output of the neural network

The Neuron

Input/Output

- Receives input from other neurons or the user's program
- Sends output to other neurons or the user's program

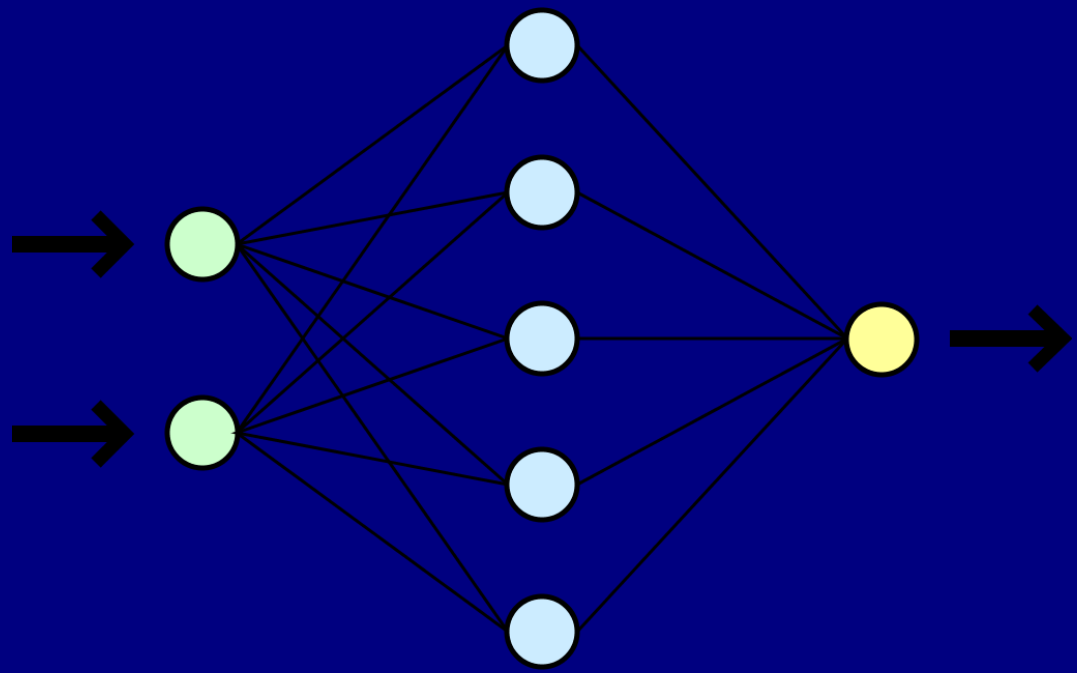
A neuron "fires" or "actives" when the sum of its inputs

$$f(x) = K\left(\sum_i w_i g_i(x)\right)$$

is high enough. We may use one of many activation functions, like

- $$y = 1$$
- Tanh:
$$\tanh(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$$
- Sigmoid:
$$y = \frac{1}{1 + e^{-x}}$$

Neural Network Structure



Split into two parts, neurons and layers

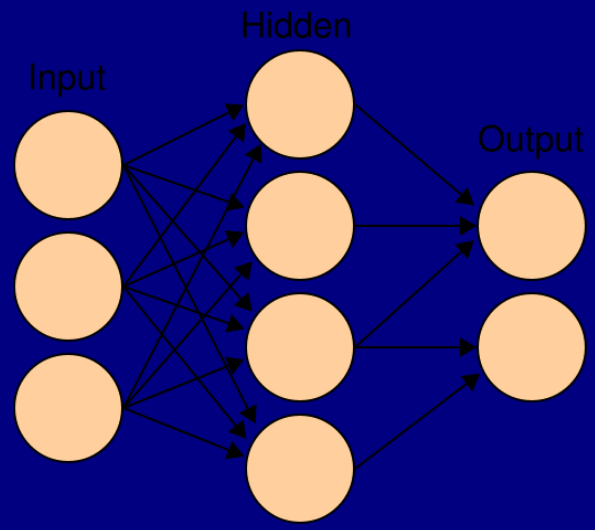
- Neurons - basic element. Interconnected, with each connection having a weight.
- Layers - groups of neurons.

Neuron Connection Weights

- Neurons are connected together by weighted connections
- These weights allow the neural network to recognize patterns
  - If you adjust the weights, the neural network will recognize a different pattern.
- **Training** a neural network is merely adjusting the weights between the neurons until we get the desired output

Layers of Neurons

- Neurons are commonly grouped in layers
- Layers - groups of neurons that perform similar functions
- Three types
  1. Input layer - receives input from the user
  2. Output layer - sends data to user
  3. Hidden layer - neurons connected only to other neurons



Training

- Remember: neurons are connected via weighted connections, these weights determine the output of the network
- Training methodology:
  1. Assign random numbers to weights
  2. Determine validity of neural network (see next slides)
  3. Adjust weights according to validation results

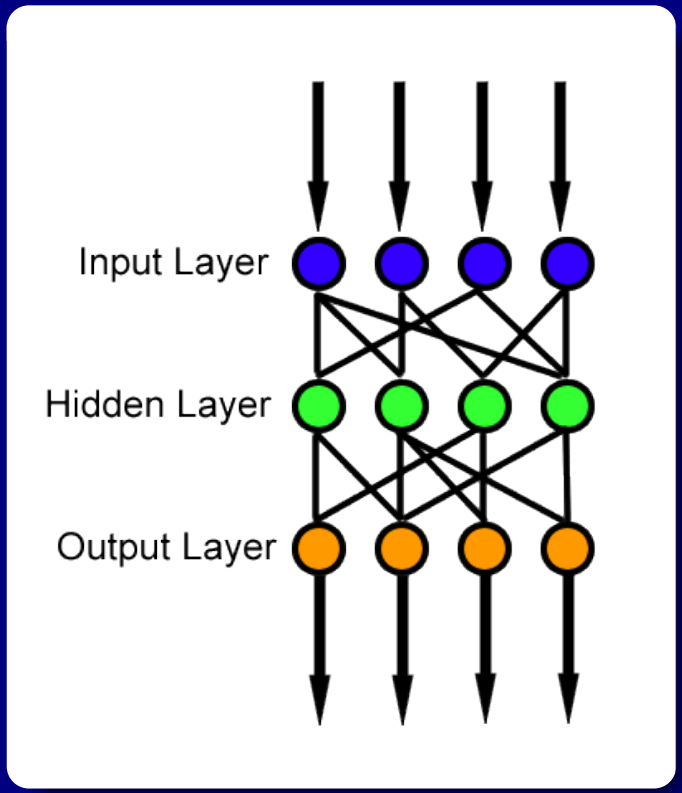
Training Methods

- Supervised Training
  - Most common form of neural network training
  - Give the neural net a set of sample data with anticipated outputs for each sample.
  - Progresses through several iterations (*epochs*) until the actual neural network matches the anticipated output within error.
- Unsupervised Training
  - Give the neural network a set of sample data without anticipated outputs
  - Used when the neural network needs to classify the inputs into several groups
- Hybrid Approaches

Validation

- We must check that training has gone correctly!
- We determine whether we need further training
- Validation data is separate from training data
  1. Use half the data to train the network
  2. Use other half to make sure neural network's weights correspond to correct solutions.

Feed-Forward Networks



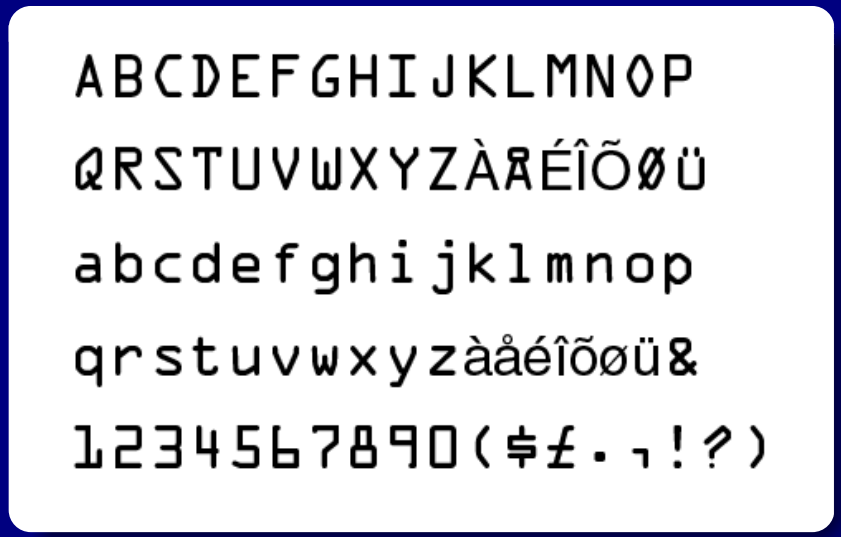
A feed-forward neural network.

- A **feed-forward** neural network is one where connections between neurons *do not* form a directed cycle.

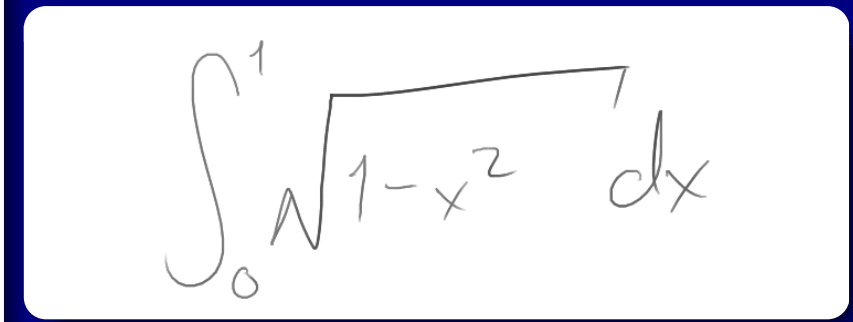
Presentation Outline

Problem Motivation

Computers are now very good at recognizing printed text like this:



But very bad at recognizing text like this:



Problem Motivation

The former is called **Optical Character Recognition** and the latter is called **Intelligent Character Recognition**.

- Optical Character Recognition is a very well-studied problem (see: Google Library Project), and error rates are approaching 1 in several hundred.
- Intelligent character recognition is not very well studied
  - Almost no one has done, properly, handwriting to LaTeX.
  - No one has done it with a Neural Network yet.

Presentation Outline

Problem Definition

Transform handwritten equations into LaTeX form using neural networks.

Presentation Outline

Three Steps

There are three major subproblems inherent in transforming handwritten equations into LaTeX.

1. Single-character recognition
2. Multiple character recognition
3. Positional recognition (i.e. fractions, exponents)

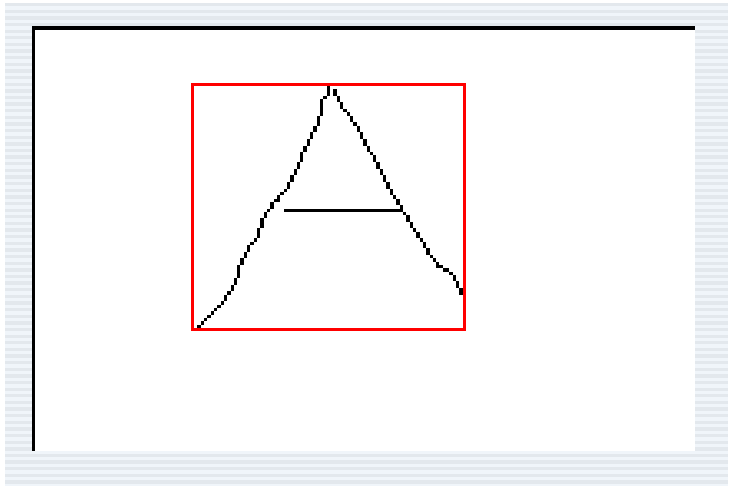
Single-Character Recognition

This can be broken down into several steps:

1. Find bounds of user-drawn letter
2. Downsample letter
3. Feed into Kohonen Neural Network

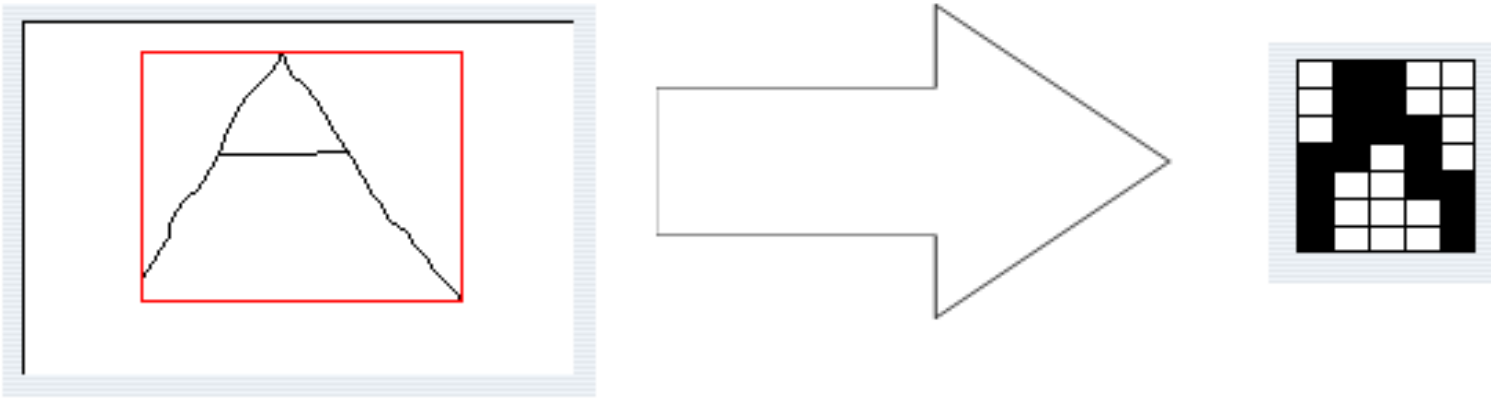
Finding Bounds

- A very easy task!
- Iterate through each direction to see if there is a pixel on that line, if not, keep going
- The final result looks like (graphically)



Downsampling

- We downsample the image to a 5×7 grid
- Graphically looks like:

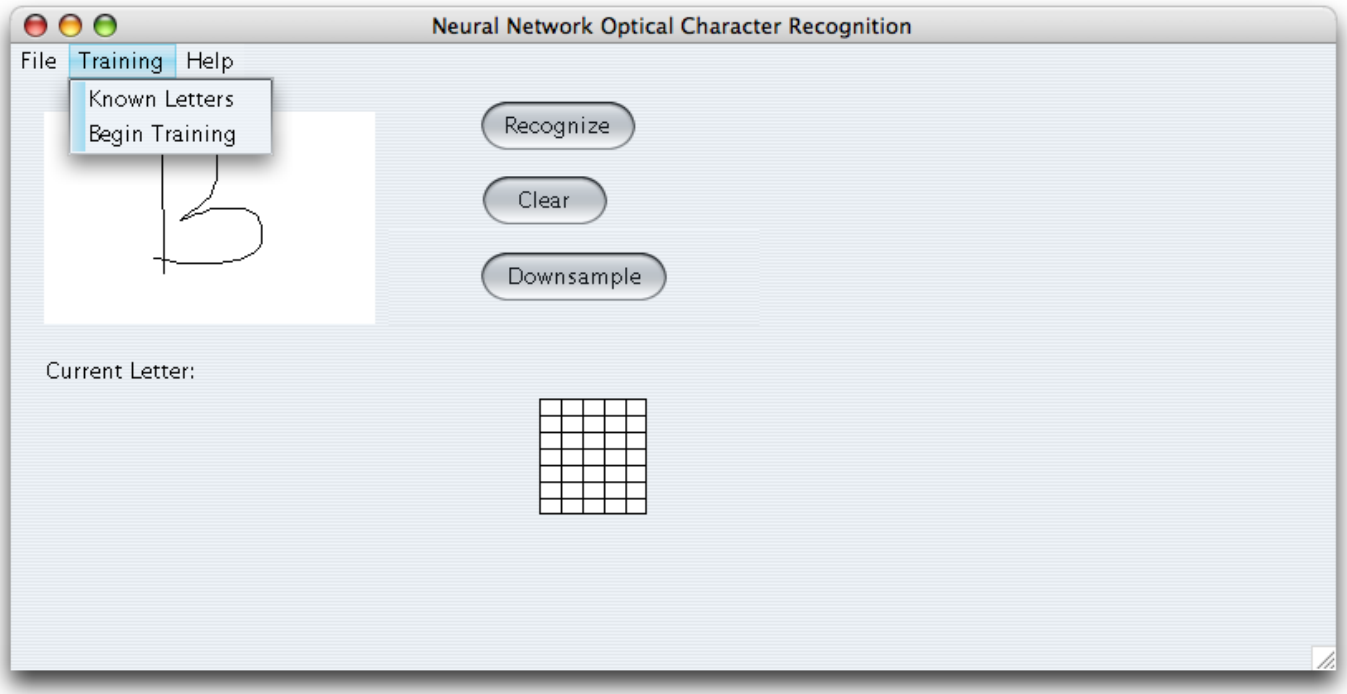


Kohonen Neural Network

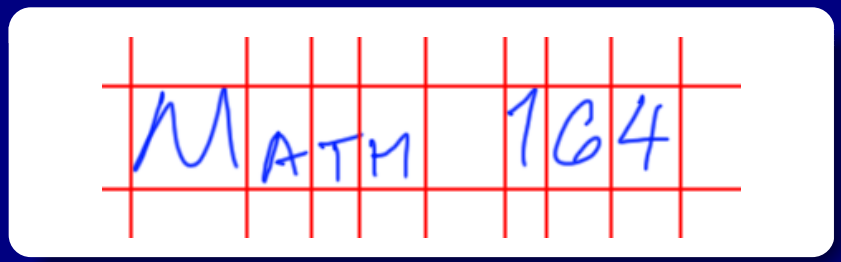
- Single-layer feed-forward network where output neurons are arranged in 2D grid.
- Each input is connected to all output neurons.
- With every neuron there is a weight vector with the same dimensionality as input vectors.

The NeuralOCR Application

Screenshot:

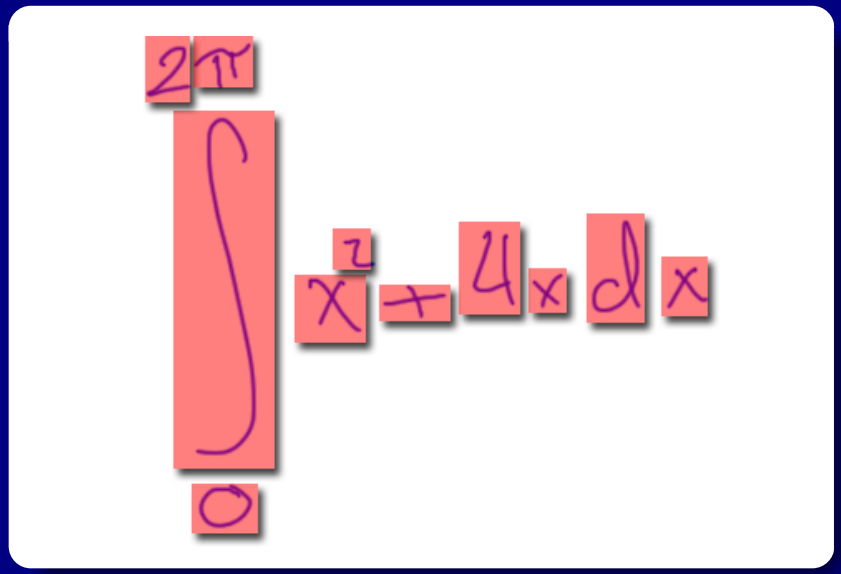


Recognizing Multiple Characters



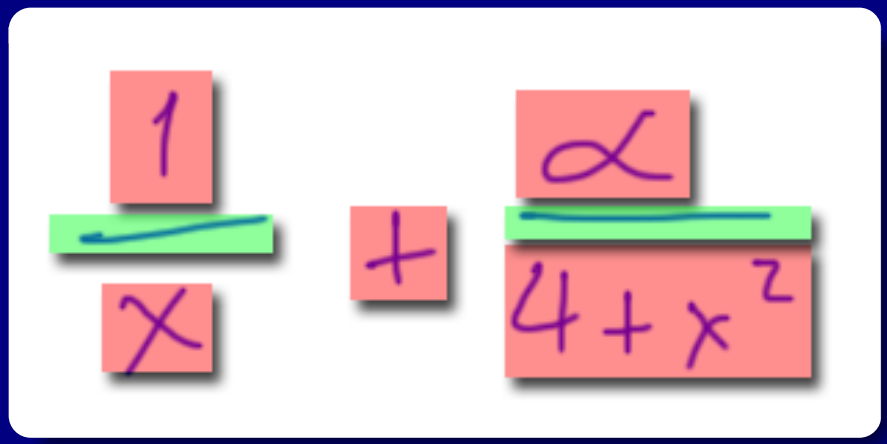
- We can do better - multiple character recognition!
- Multiple bounding boxes - one for each letter
  - Each character gets recognized separately by the neural network
  - This should be a valid assumption for math - people don't write math in cursive.

Recognizing Math!!!



- We can do better still! - multiple character recognition!
- Multiple bounding boxes - one for each character
  - Store position data for each character after you recognize it
  - Use a second neural network to recognize position to determine the difference between:
    - Normal math
    - Subscript
    - Superscript
    - Fractions (more on this later)

Fractions are Hard



- The hangup is recognizing fractions with this method

Presentation Outline

Outlook

Future work:

- Multiple-character recognition (not that hard!)
- Positional recognition

References

All images from Wikipedia (thanks guys!)

Questions?

Quotes (upon hearing research topic)

- *"I will personally pay you lots and lots of money if you get this working."* - Prof. Yong
- *"Dude ... why are you setting yourself up to fail like that?"* - Anonymous Former Neural Nets Student
- *"... hahahahahahaha"* - Anonymous Neural Nets Researcher

Questions?