Image Classification Using Deep Neural Networks

Prepared and presented by: Caroline Katba

Website: katba-caroline.com

Outline

- What is Deep Learning?
- What are Neural Networks?
- Important Concepts and Terms
- How can we build an Image Classifier?
- Image Classification Example
- Further uses of Image Classification
- Q & A

“Deep Learning: A subset of Machine Learning Algorithms that is very good at recognizing patterns but typically requires a large number of data.”

source

Deep learning is an excellent way to develop a strong program that can recognize objects in images. it’s implemented using several layers of artificial neural networks where each layer is responsible for extracting one or more features of the image. It is the nature in which this network is built that lends it the name deep.

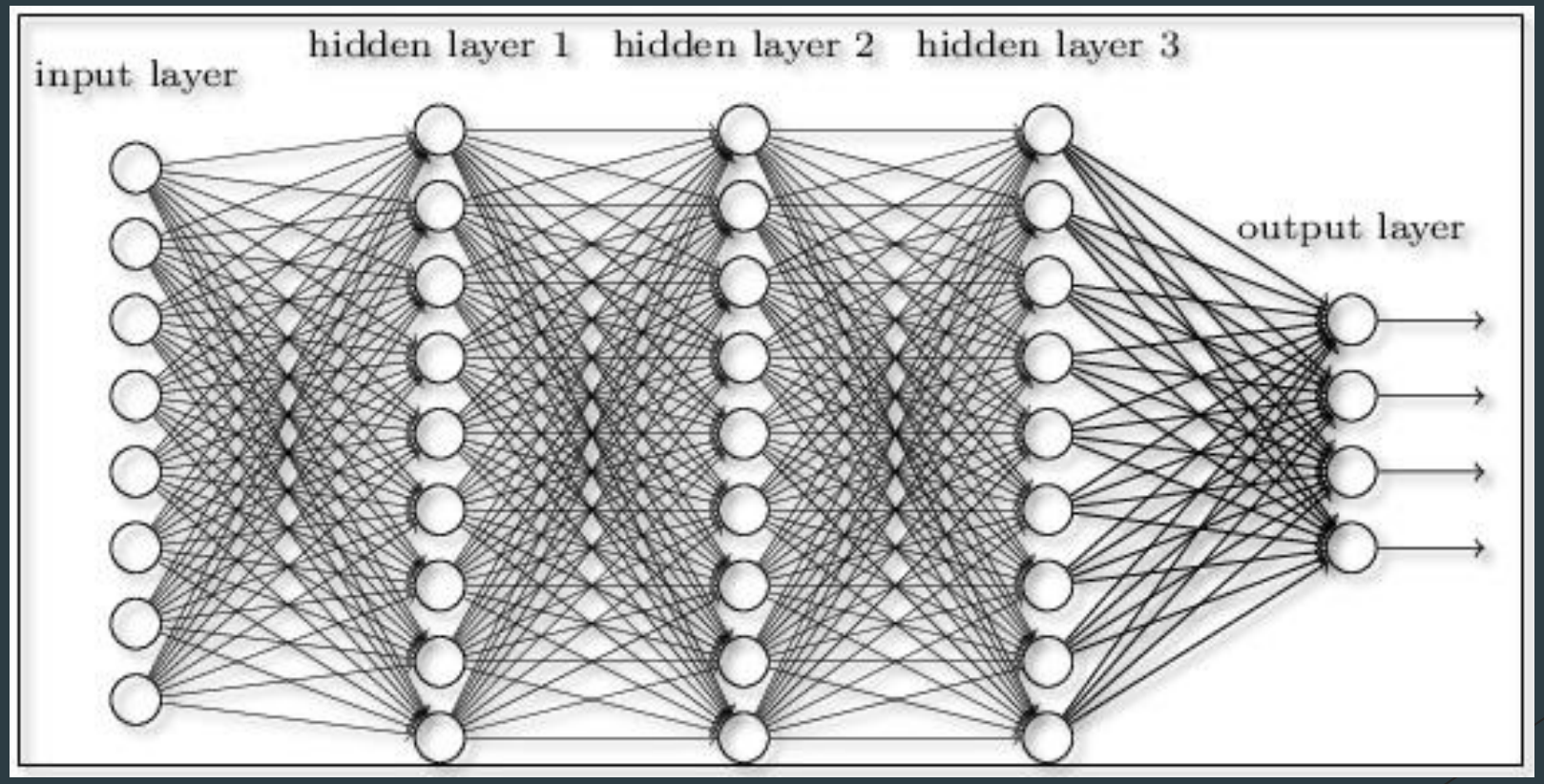
What are some of the practical applications on Deep Learning?

- Beating humans in games like Go and Jeopardy
- Detecting Stock Prices
- Identifying Cancer Cells
  - Teams at UCLA built an advanced microscope that yields a high-dimensional data set used to train a deep learning application to accurately identify cancer cells.
- Improving work place safety
  - by automatically detecting when people or objects are within an unsafe distance of machines.
- Making automated self-driving cars a reality
  - By automatically detecting objects such as stop signs and traffic lights and pedestrians, which helps decrease accidents.
- Read more here: source

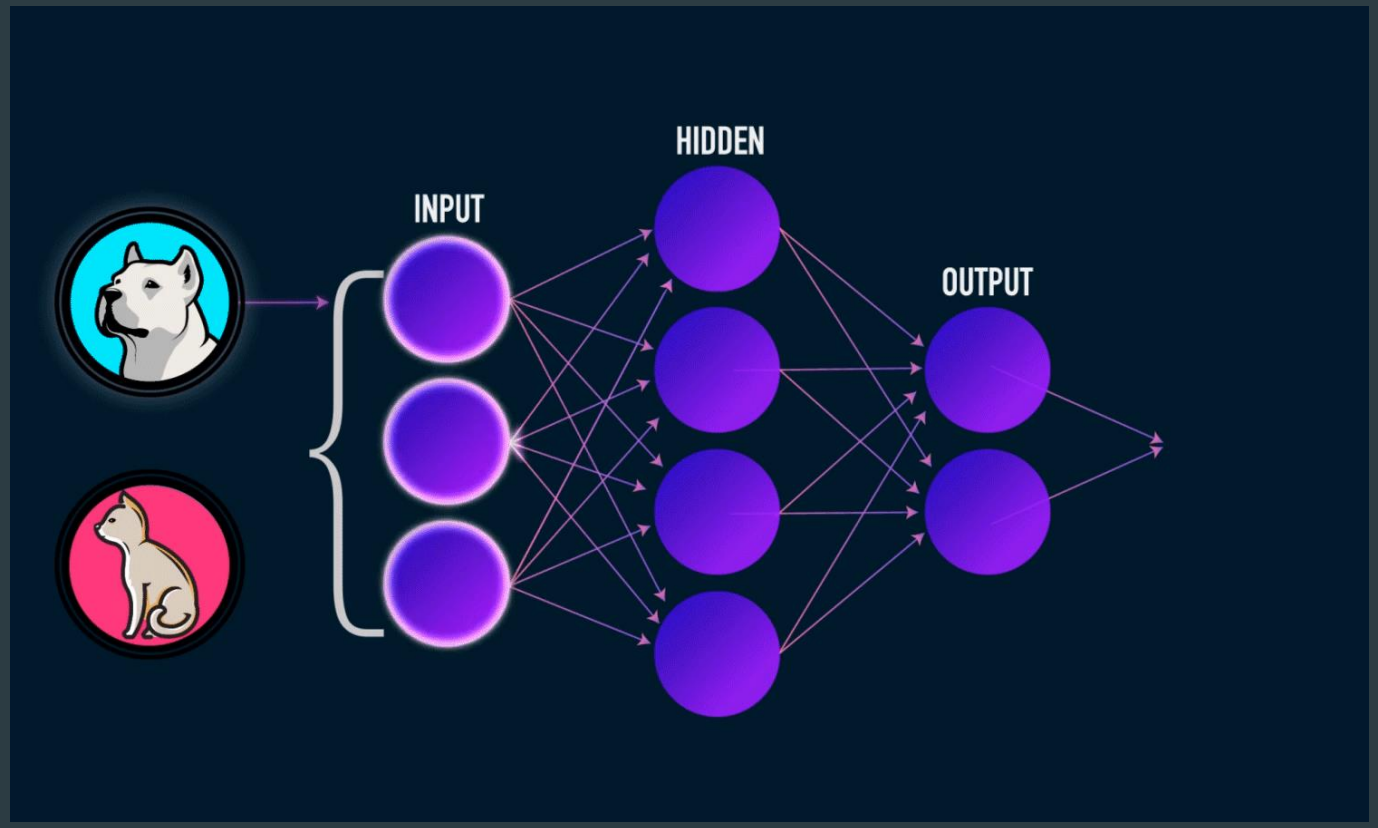
Neural Networks: The Heart (or Brain) of Deep Learning

- Neural networks are called that because they vaguely imitate the brain operations, where many nodes are connected to one another and firing off information to make a decision.
- The term “deep” usually refers to the number of hidden layers in the neural network. Traditional neural networks only contain 2-3 hidden layers, while deep networks can have as many as 150. source

Here is what a Neural Network could look like:



Here’s what it’s actually doing:



Important Concepts and Terms

- A very brief introduction to the nuts and bolts of a Neural Network:
  - Perceptron
  - Sigmoid Function
  - Softmax Function
  - Cross-Entropy
  - Feedforward
  - Backpropagation

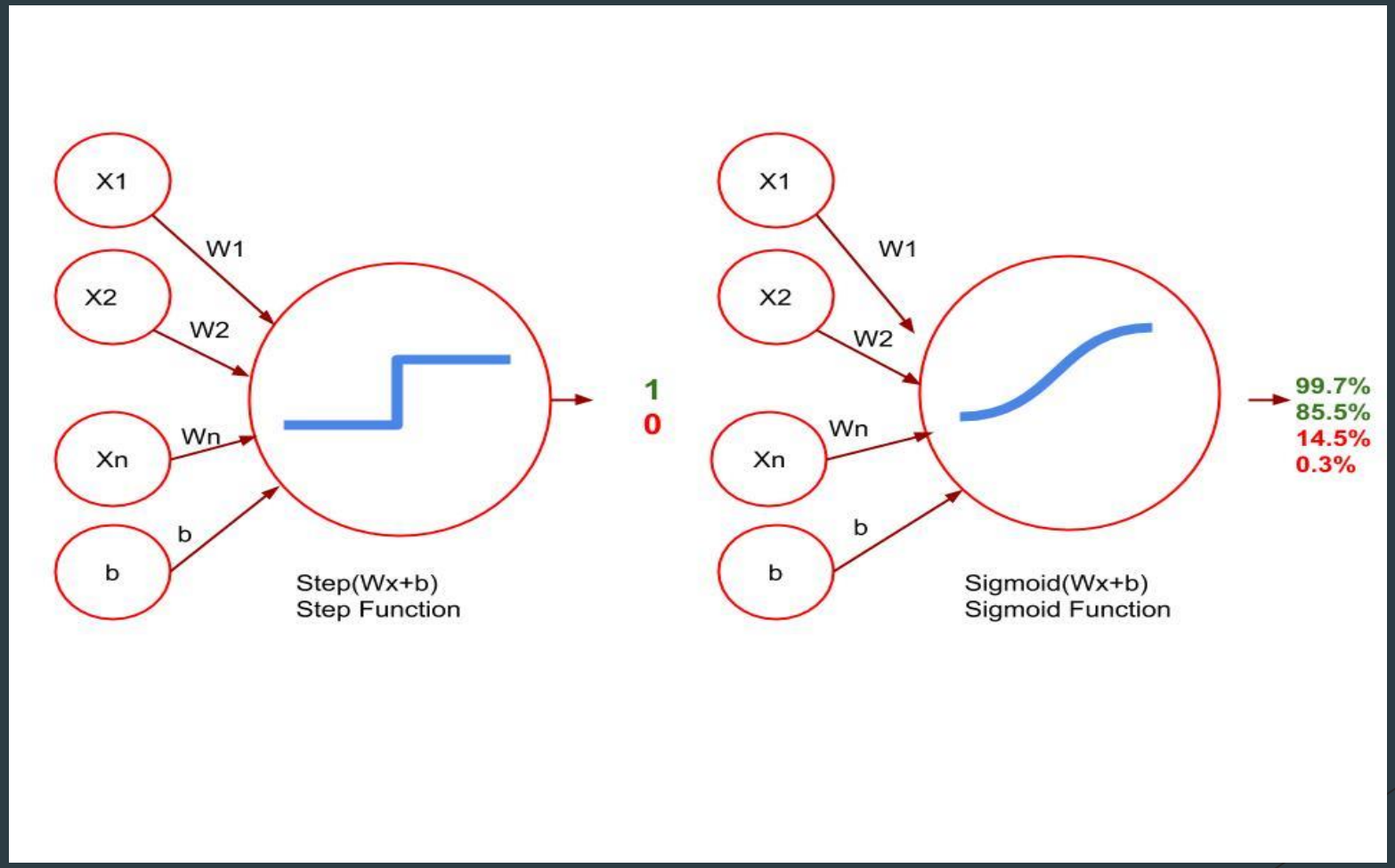
Perceptron: Building Block of Neural Networks

- A perceptron takes several binary inputs, and produces a single binary output.
- To compute output, simply use Weights
  - Weights are real numbers that express the importance of each input relative to the output.
  - The neuron's output, 0 or 1, is determined by whether the weighted sum is less than or greater than some threshold value.

- Concern: any small tweak to our neural network could completely throw off our entire model and could change the output from 0 to 1 or vice versa.
- Read more here: source

Sigmoid Smooth is the way to go

Now our predictions are a probability rather than a 1 or 0.
Continuous predictions rather than Discrete.



Softmax: activation function for more than 2 classes

- Softmax is the equivalent of the sigmoid activation function, but for when we have a multiclass classification problem.
- "Softmax computes the **exponential (e-power)** of the given input value and the **sum of exponential values** of all the values in the inputs. Then the ratio of the exponential of the input value and the sum of exponential values is the output of the softmax function."
- Main Softmax advantage: output probabilities range.
  - The range is **0 to 1**, and the sum of all the probabilities will always **equal to one**.
    - No negative probabilities
  - In multi-classification models, softmax returns the probabilities of each class and the target class will have the high probability.
- Read more here: source

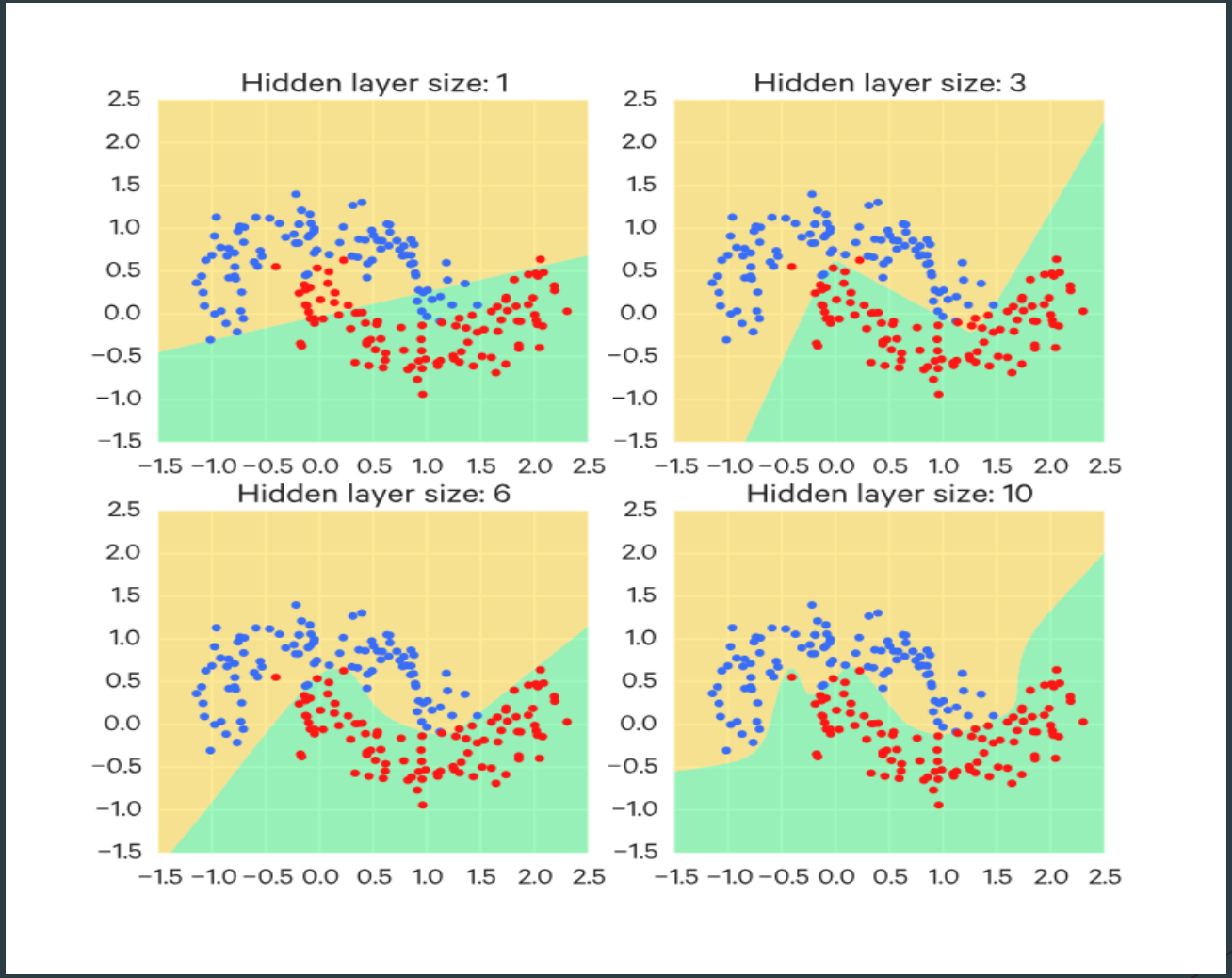
Cross Entropy

- Cross Entropy looks at the likelihood that events happen based on the probabilities:
  - If very likely, then small Cross-Entropy
  - If very unlikely, then large Cross-Entropy
  - A good model will give us low cross-entropy, a bad model will give us high cross-entropy
- In Machine Learning models: The cross-entropy compares the model's prediction with the label which is the true probability distribution.
  - The cross-entropy goes down as the prediction gets more and more accurate.
  - It becomes zero if the prediction is perfect.
  - Therefore, the cross-entropy can be a loss function to train a classification model.
  - Read more here: source

Feedforward

- Not all models are linear, Neural Networks are excellent at solving classification problems that cannot be solved by drawing a straight line.
- Feedforward is simply the process neural networks use to turn the input into an output.
- Feedforward is what neural networks do:
  - They take the input vector
  - Apply a sequence of linear models and sigmoid functions
  - Then when combined, become highly non-linear map.

Example of Feedforward on non linear problems.



Backpropagation

- Backpropagation consists of the following steps:
  - Doing a feedforward operation.
  - Comparing the output of the model with the desired output.
  - Calculating the error. (how bad was a point misclassified?)
  - Running the feedforward operation backwards (backpropagation) to spread the error to each of the weights.
  - Use this to update the weights, and get a better model.
  - Continue this until we have a model that is good.

Example of backpropagation

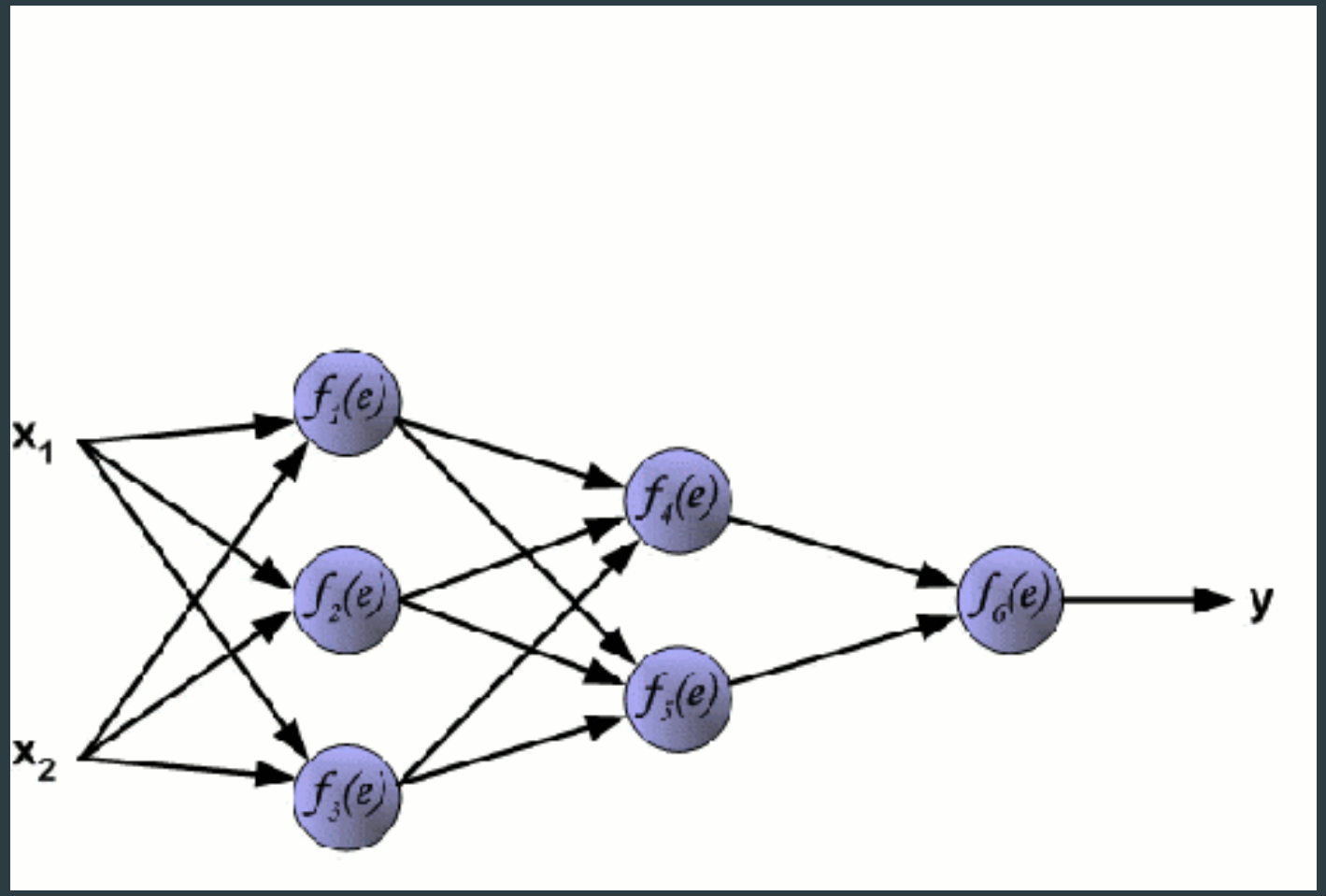
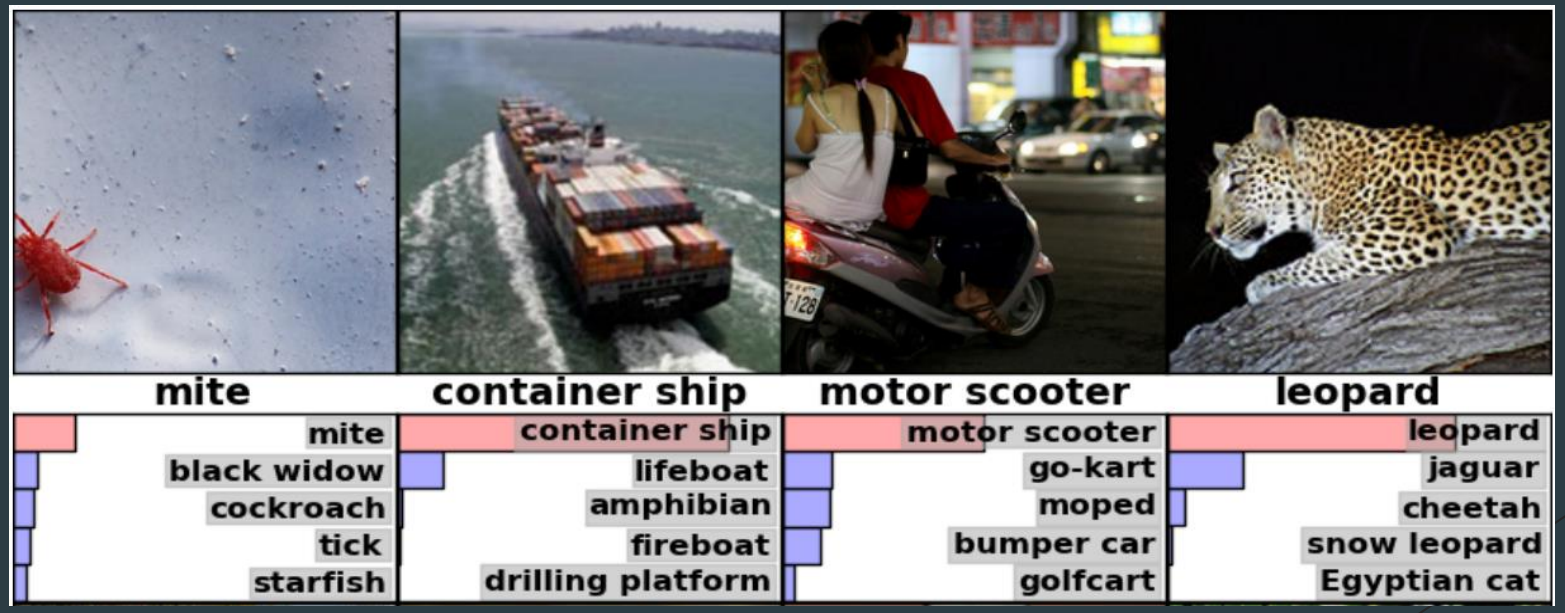


Image Classification

- Teaching a child the difference between a lion and leopard is not hard, teaching a computer that difference is another story.
- Using Neural Networks, we can train models to recognize objects in images accurately and consistently.



source

Human Vision vs. Computer Vision

For a computer, an image is nothing but a bunch of numbers arranged in a grid-like structure which is formally known as an Array.

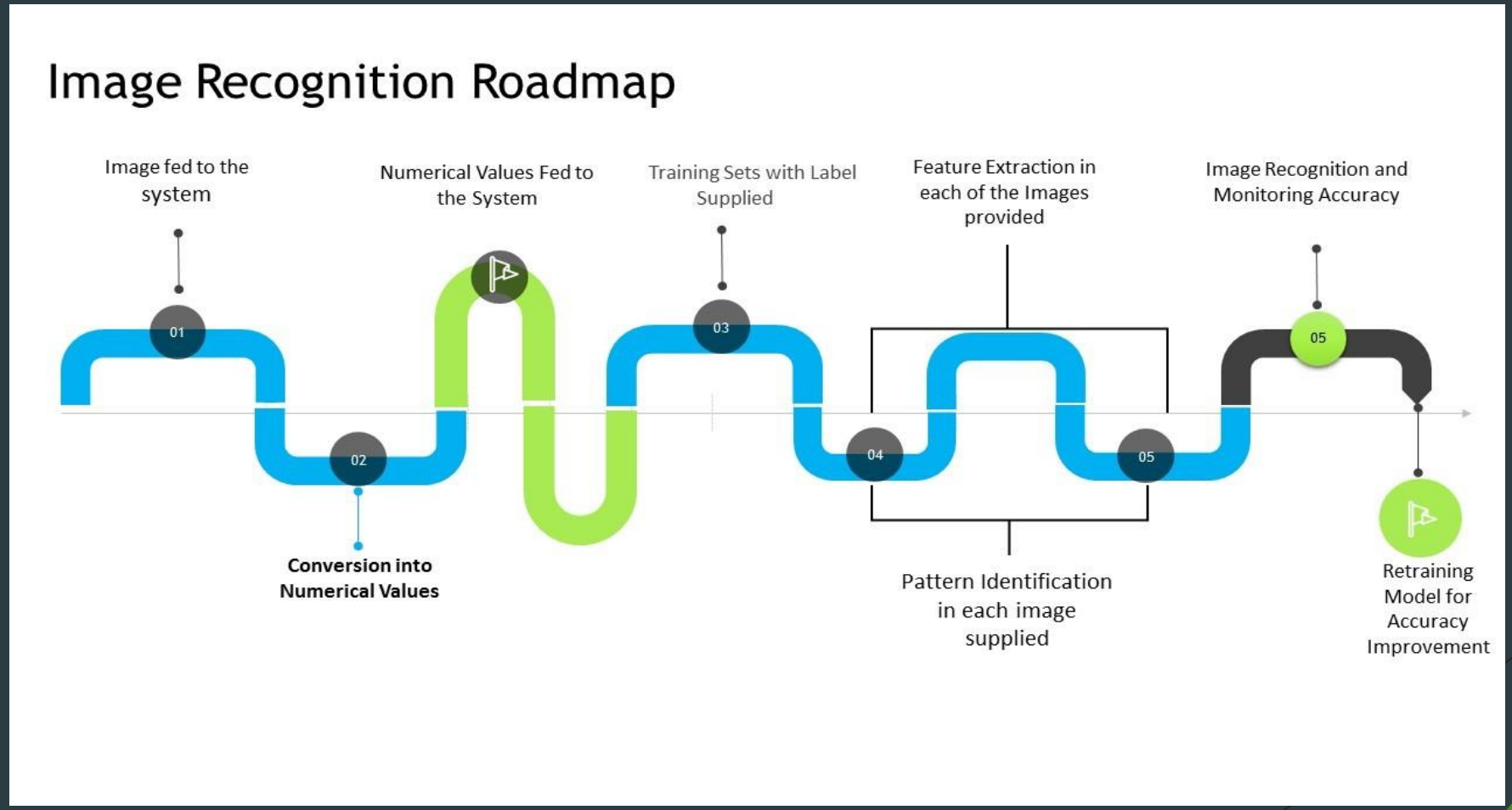
```
[[ 16  15  13  18  16  20  24  33  39  32  32  24  22
  131 141 139  13  53  91  12  13   9  10]
 [ 32  18   9   5   8   9   8   9   7  17  40  45  55
  121 127 134  31  13  98  14   9   8   8]
 [ 21  11  10  12  17  16  30  34  39  37  41  50  57
  111 120 133 114  20  17  57  15  11   8]
 [ 61  25  21  28  54  42  29  36  39  46  36  47  50
   56 128 134 128  27  21  76  13   9  12]
 [232 236 251 226 176  95  42  45  21  41  47  50  51
   55 123 118 106  15  30  10  13  15  27]
 [226 225 250 254 242 255  35  22  45 100 103  30  31
   48 252 246 233 226 221 200 164 101  32]
 [206 227 226 242 238 238  45  70  50  38  20  51  48
   69 237 243 227 227 228 221 112  22 113]
 [215 226 218 231 229 235 210  20  86  26  19  13  18
   85 225 236 226 228 212 204 196 190 161]
 [192 190 190 192 202 222 192  13  24  28  58  35  19
   80 181 192 204 206 206 192 175 175 172]
```

source

How does a computer know what a cat looks like?
Similarity Checks

- We know what a cat looks like because we have see one before.
- For computers, we can use an image of a cat's face as an example image.
  - Or we can use the pixel values in the middle of the array.
- To measure similarity between the picture use Frobenius Inner product:
  - "Frobenius Inner product is nothing but the element-wise multiplication of two arrays followed by the addition operation.
  - In general, it is one of the ways through which you can measure the similarity between two things(matrices).
  - It is nothing but the generalized version of the dot product with only one difference: dot product is defined for vectors and inner product is defined for matrices."
  - Read more here: source

Image Classification Roadmap

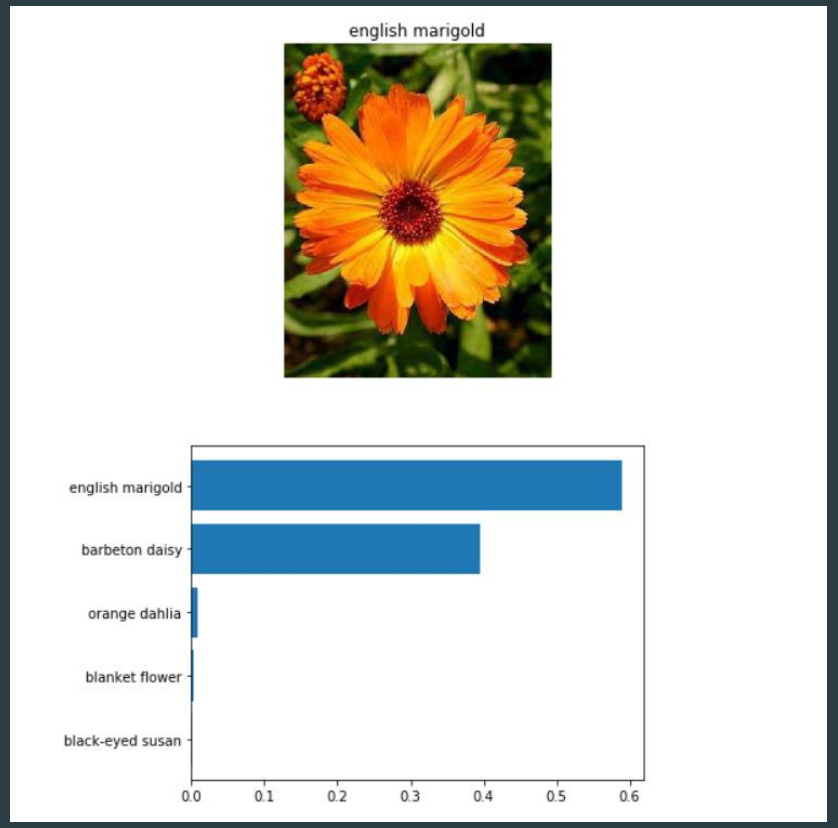


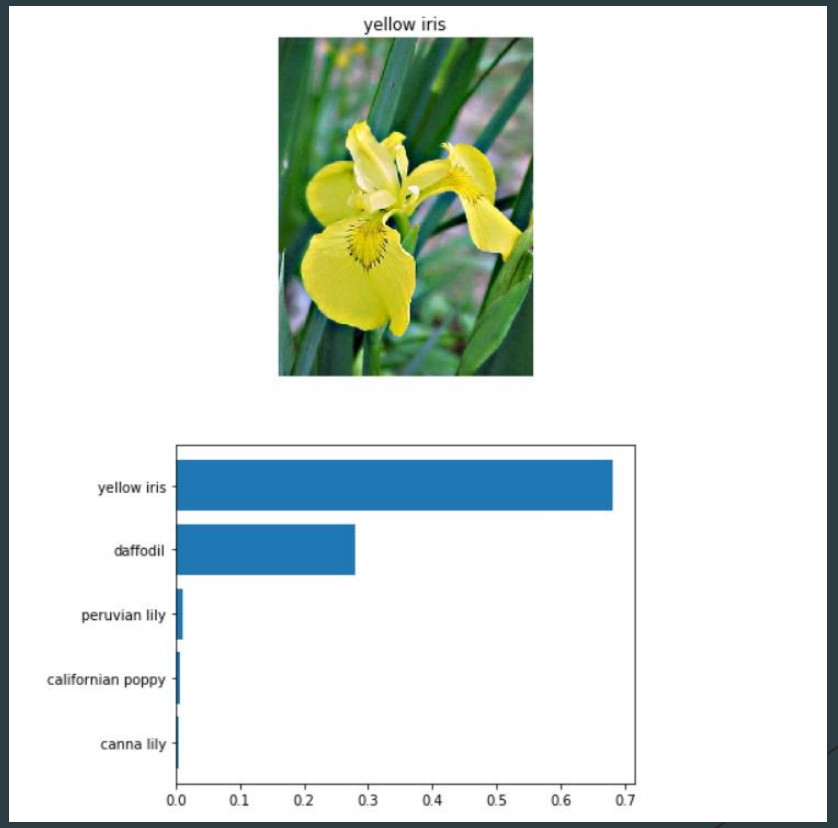
source

Example: Flower Image Classifier

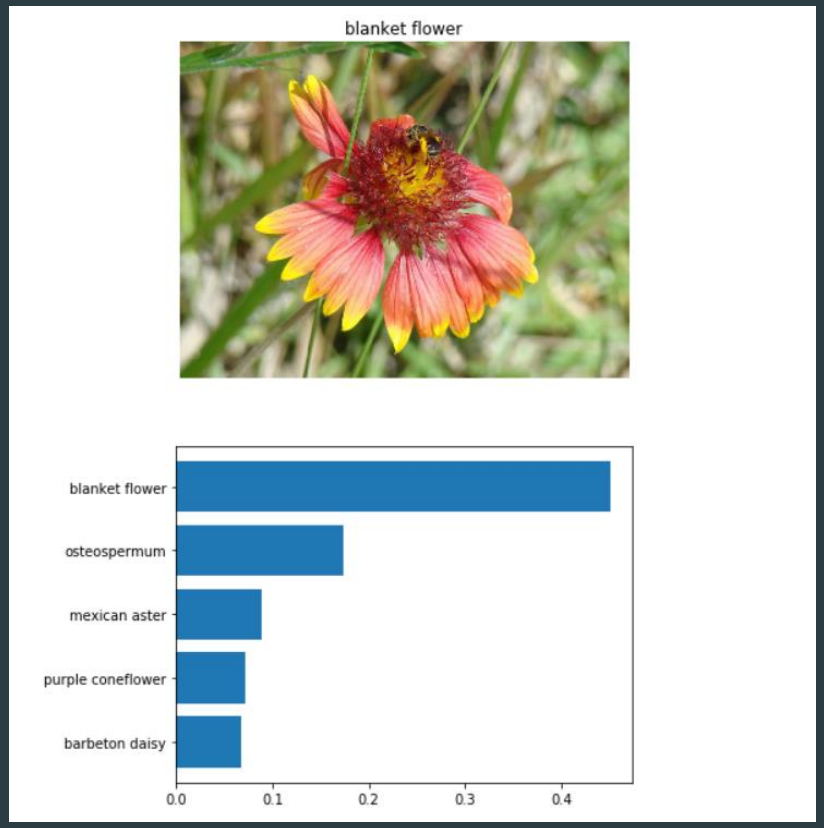
- British flowers data set contains 102 categories of flowers with 40 to 258 images per class, you can read more about it here.
- Using Pytorch (which is an optimized tensor library for deep learning using GPUs and CPUs), I built a deep learning model that utilizes transfer learning to build upon and improve a pre-designed neural network and achieve results with 80% accuracy.
- This model is designed to provide accurate results as well as the probabilities of the top 5 predictions. Once you feed a new picture into the model, it automatically shows you the image with the flower's name as the title, and produces a bar chart that shows the top 5 predicted classes.
- Read more here: source

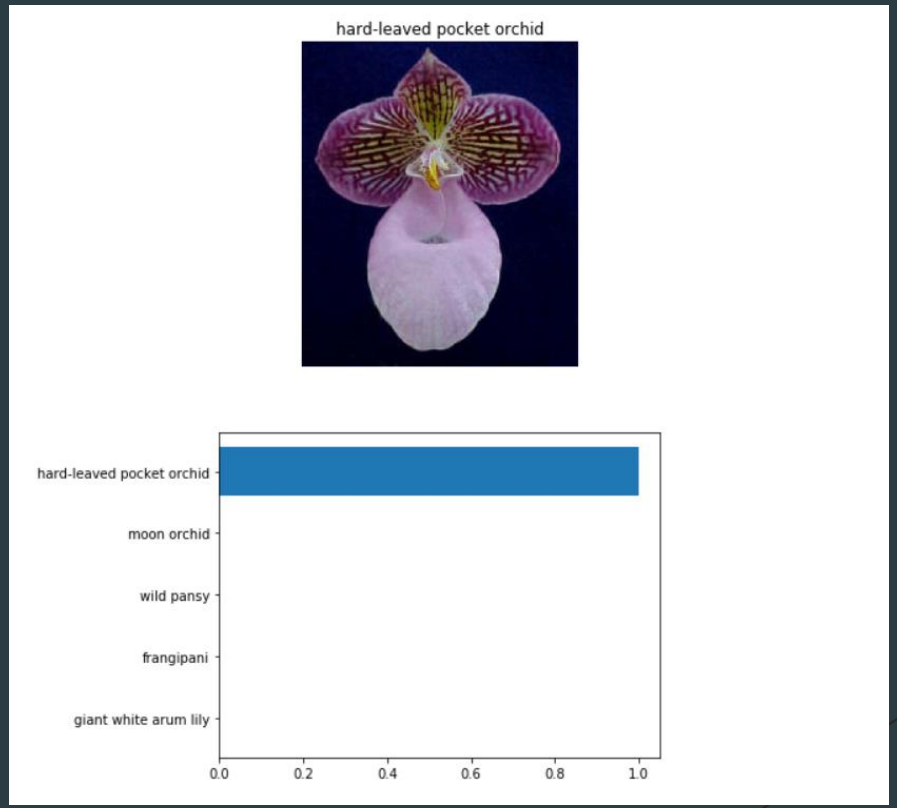
Example: Flower Image Classifier





Example: Flower Image Classifier





Uses of Image Recognition

- Facial recognition for more security or more surveillance.
- Product identification for enhanced marketing and increased sales.
- Drones can provide vision-based automatic monitoring, inspection, and control of assets located in remote areas.
- Product line inspections: “evaluating critical points on a regular basis within the premises. Monitoring the quality of the final products to reduce the defects. Assessing the condition of the workers can help manufacturing industries to have a complete control of different activities in the systems.”
- Autonomous vehicles with image recognition can identify activities on the road and take necessary actions. Mini robots can help logistics industries to locate and transfer the objects from one place to another. It also maintains the database of the product movement history to prevent the product from being misplaced or stolen.
- Unmanned Aerial Vehicles can monitor the forest, predict changes that can result in forest fires, and prevent poaching. It can also provide a complete monitoring of the vast lands, which humans cannot access easily.”
- Read more here: source

Thank you! Questions?

- Caroline Katba
- Webiste: https://katba-caroline.com/
- Email: katba.caroline@gmail.com
- Github: https://github.com/Katba-Caroline