Convolution Neural Network

- A convolutional neural network

  (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery.

Convolution Neural Network

- Facebook uses neural nets for their automatic tagging algorithms
- Google for their photo search
- Amazon for their product recommendations
- Pinterest for their home feed personalization
- Instagram for their search infrastructure.

Image classification

- Most popular, use case of these networks is for image processing.

How Humans identify images??

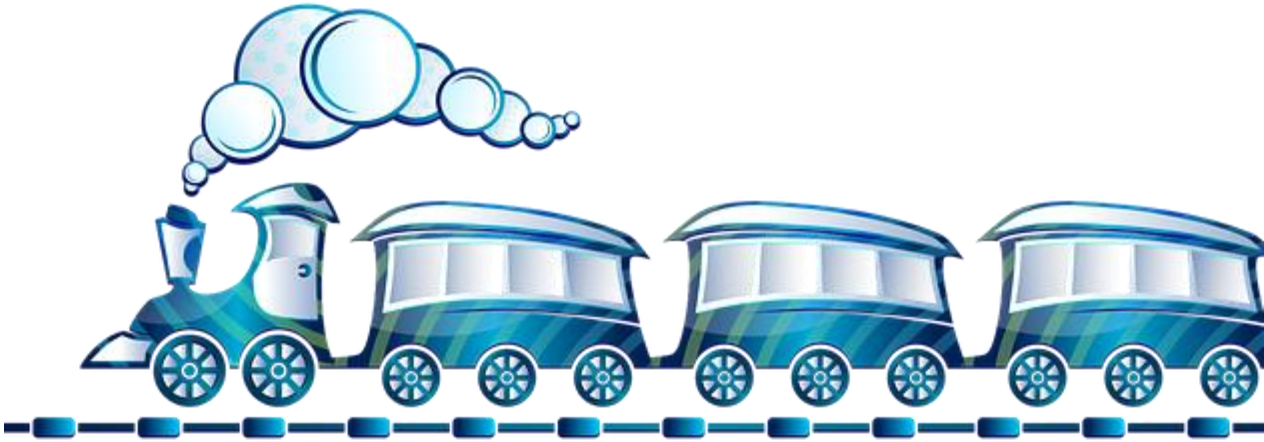
- For humans, this task of recognition is one of the first skills we learn from the moment we are born and is one that comes naturally and effortlessly as adults.
- Most of the time we are able to immediately characterize the scene and give each object a label, all without even consciously noticing.

What's that?

A Train

What's this?

A Train

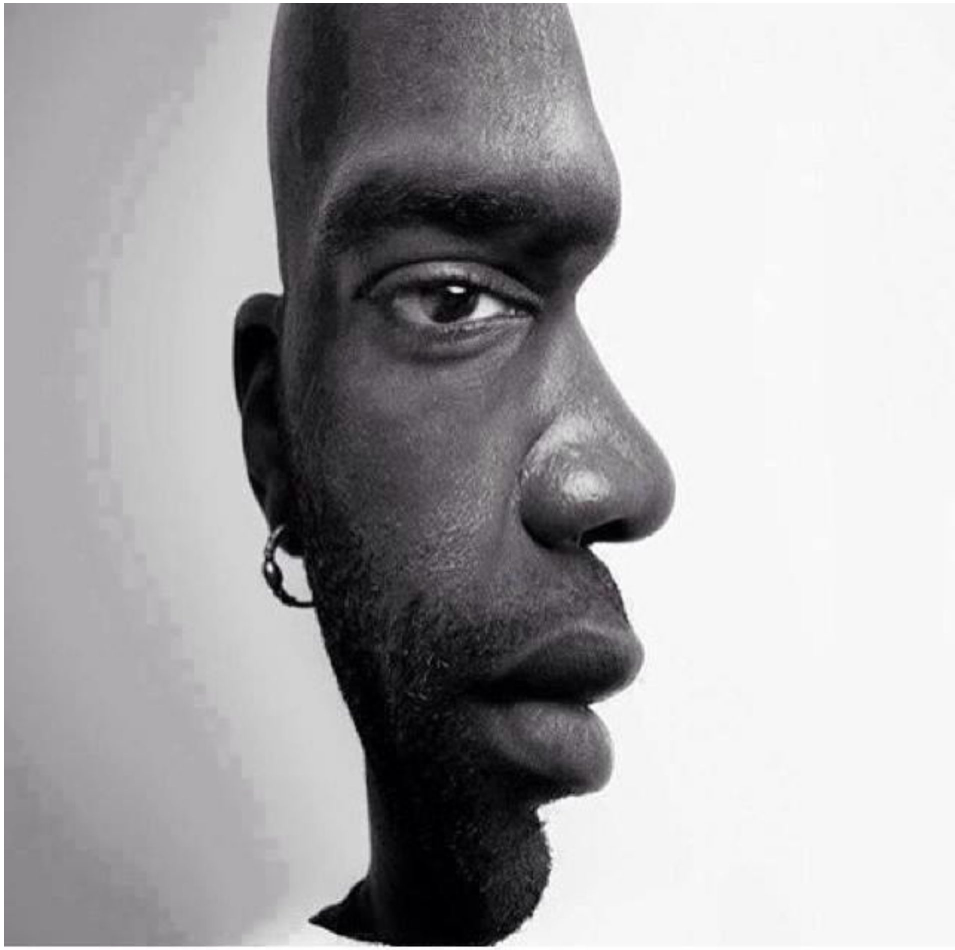


Teach Machine

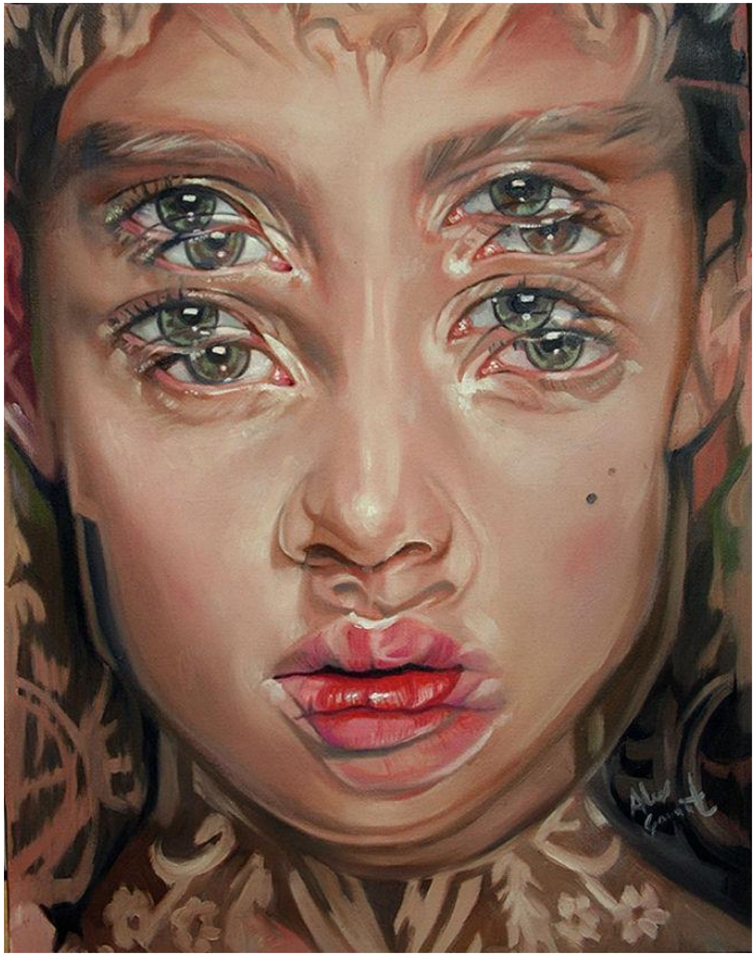
Human skills of being able to :

- Quickly recognize patterns
- Generalize from prior knowledge

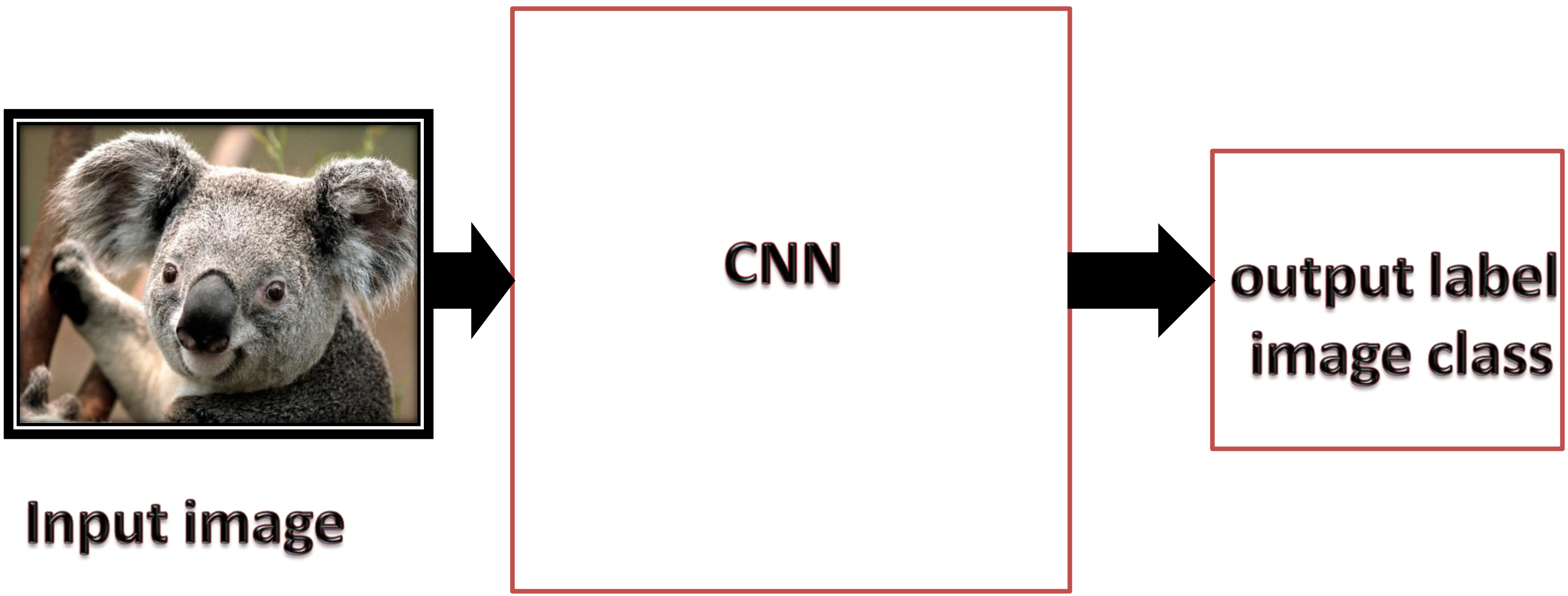
illusions



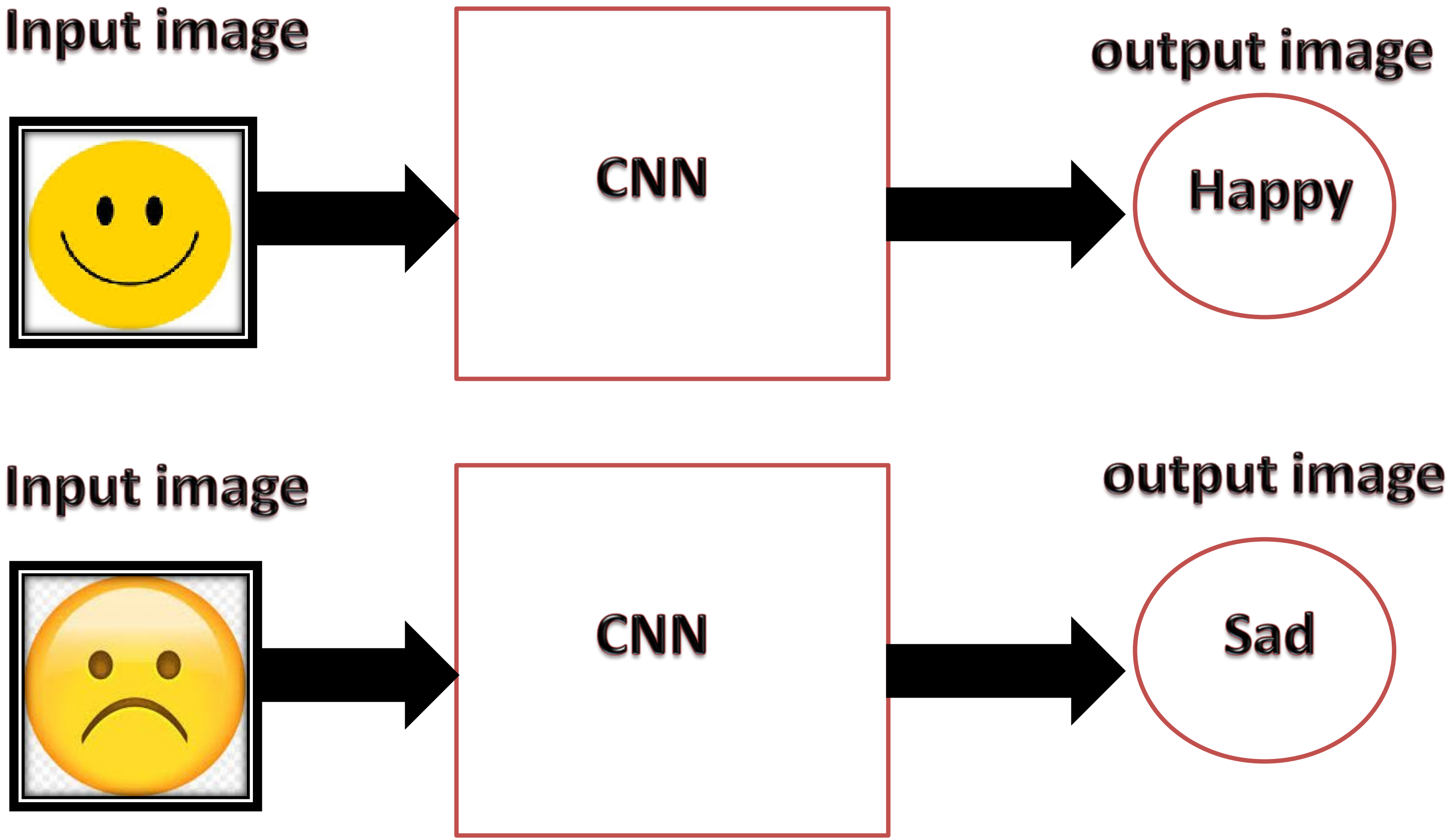
illusions



Convolution Neural Network



Convolution Neural Network



How Does Computer See an image??

When a computer sees an image ,it will see an array of pixel values. Depending on the resolution and size of the image, it will see a 32 x 32 x 3 array of numbers (The 3 refers to RGB values).

What we see

What computers see

How Does Computer See an image??

What we see

What computers see

These numbers, when we perform image classification, are the only inputs available to the computer.

The idea is that you give the computer this array of numbers and it will output numbers that describe the probability of the image being a certain class

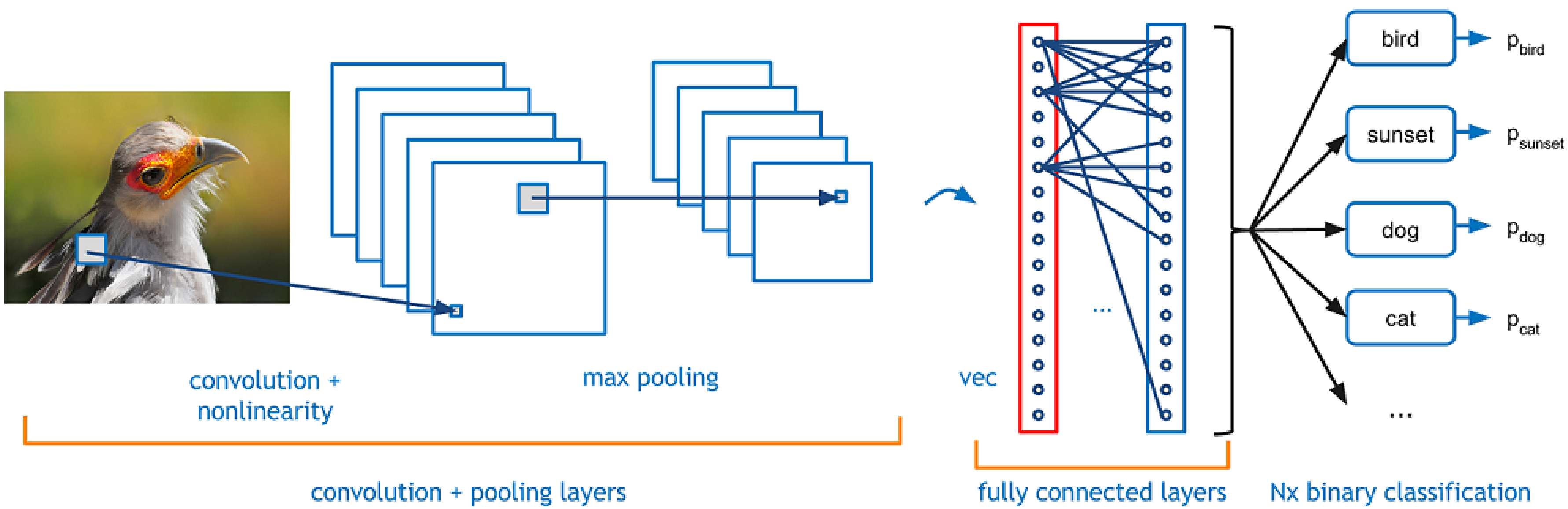
(80% for flower, 15% for sky etc).

What We Want the Computer to Do

- we want the computer to do is to be able to differentiate between all the images it’s given and figure out the **unique features** that make a dog a dog , that make a cat a cat or that make a flower a flower .
- The computer is able to perform image classification by looking for low level features such as edges and curves, and then building up to more abstract concepts through a series of **convolutional layers**.

Convolution Neural Network

CNNs take the image, pass it through a series of convolutional, nonlinear, pooling (downsampling), and fully connected layers, and get an output.



How computer sees an image

Humans see this

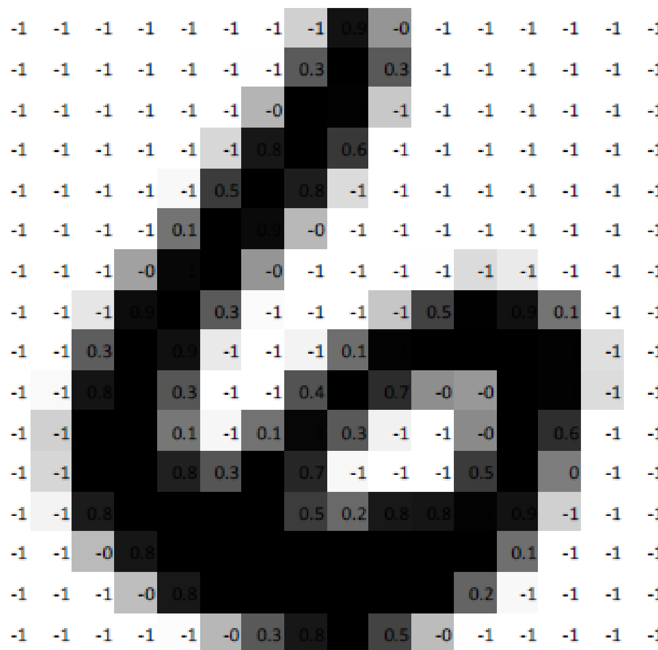
-1	-1	-1	-1	-1	-1	-1	-1	0.9	-0	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	0.3	1	0.3	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-0	1	1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	0.8	1	0.6	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	0.5	1	0.8	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	0.1	1	0.9	-0	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-0	1	1	-0	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	0.9	1	0.3	-1	-1	-1	-1	0.5	1	0.9	0.1	-1	-1
-1	-1	0.3	1	0.9	-1	-1	-1	0.1	1	1	1	1	1	-1	-1
-1	-1	0.8	1	0.3	-1	-1	0.4	1	0.7	-0	-0	1	1	-1	-1
-1	-1	1	1	0.1	-1	0.1	1	0.3	-1	-1	-0	1	0.6	-1	-1
-1	-1	1	1	0.8	0.3	1	0.7	-1	-1	-1	0.5	1	0	-1	-1
-1	-1	0.8	1	1	1	1	0.5	0.2	0.8	0.8	1	0.9	-1	-1	-1
-1	-1	-0	0.8	1	1	1	1	1	1	1	1	0.1	-1	-1	-1
-1	-1	-1	-0	0.8	1	1	1	1	1	1	0.2	-1	-1	-1	-1
-1	-1	-1	-1	-1	-0	0.3	0.8	1	0.5	-0	-1	-1	-1	-1	-1

Computer sees this

How computer sees an image

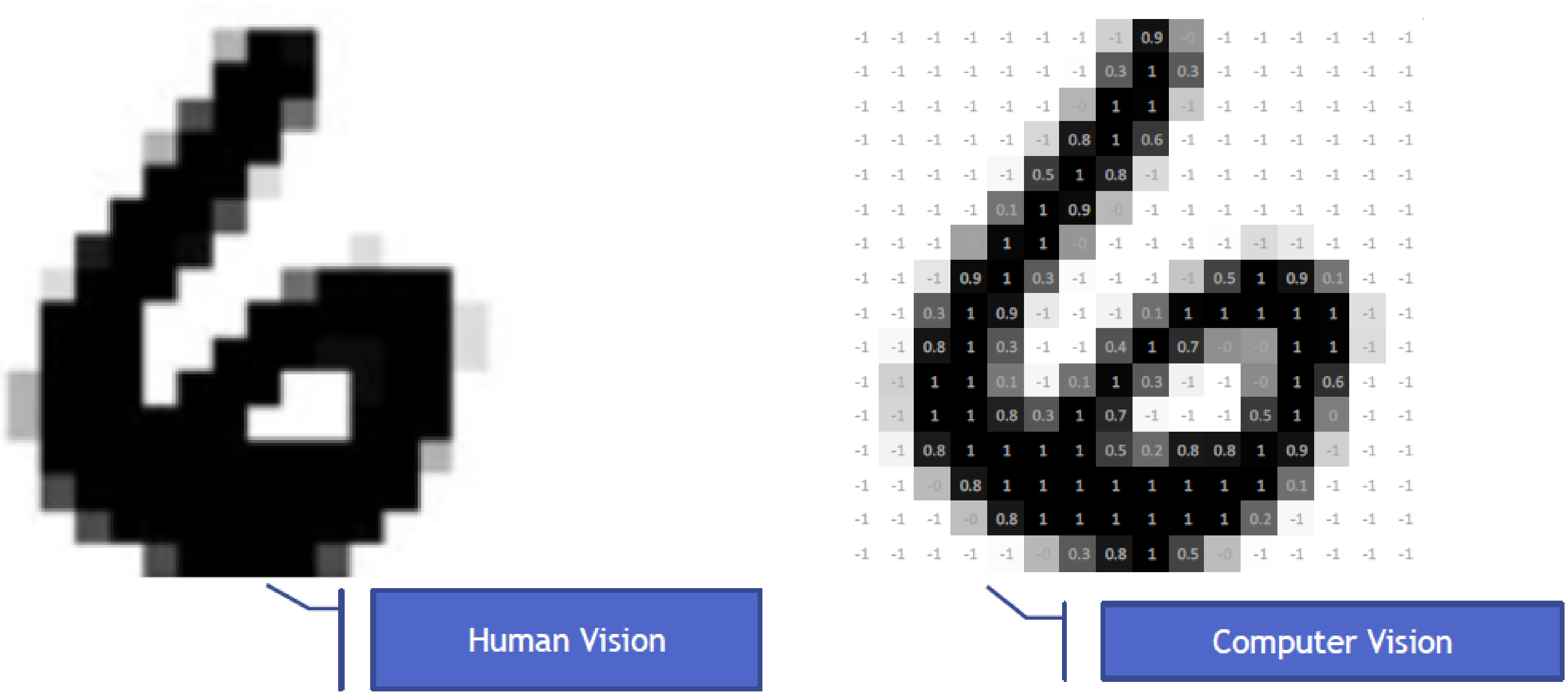
-1	-1	-1	-1	-1	-1	-1	-1	0.9	-0	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	0.3	1	0.3	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-0	1	1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	0.8	1	0.6	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	0.5	1	0.8	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	0.1	1	0.9	-0	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-0	1	1	-0	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	0.9	1	0.3	-1	-1	-1	-1	0.5	1	0.9	0.1	-1	-1
-1	-1	0.3	1	0.9	-1	-1	-1	0.1	1	1	1	1	1	-1	-1
-1	-1	0.8	1	0.3	-1	-1	0.4	1	0.7	-0	-0	1	1	-1	-1
-1	-1	1	1	0.1	-1	0.1	1	0.3	-1	-1	-0	1	0.6	-1	-1
-1	-1	1	1	0.8	0.3	1	0.7	-1	-1	-1	0.5	1	0	-1	-1
-1	-1	0.8	1	1	1	1	0.5	0.2	0.8	0.8	1	0.9	-1	-1	-1
-1	-1	-0	0.8	1	1	1	1	1	1	1	1	0.1	-1	-1	-1
-1	-1	-1	-0	0.8	1	1	1	1	1	1	0.2	-1	-1	-1	-1
-1	-1	-1	-1	-1	-0	0.3	0.8	1	0.5	-0	-1	-1	-1	-1	-1

Computer sees this



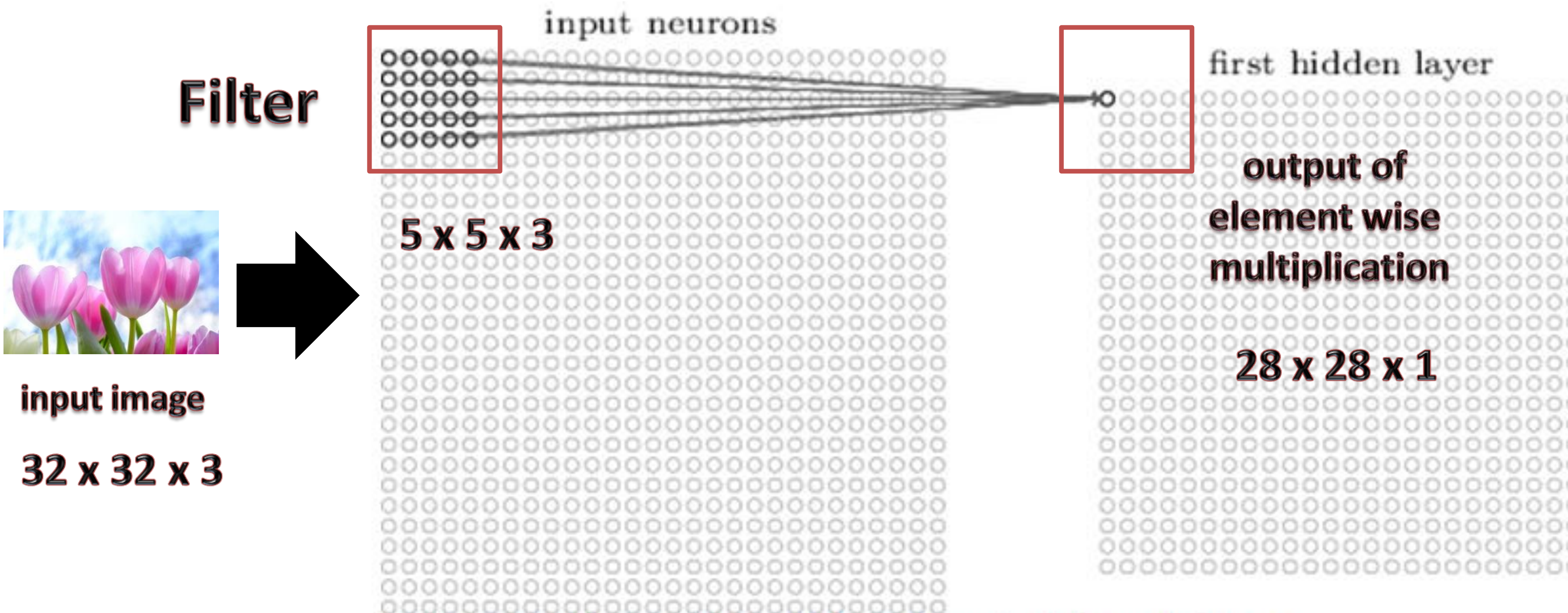
Same matrix, highlight the cells based on cell value

How computer sees an image



Convolution Neural Network

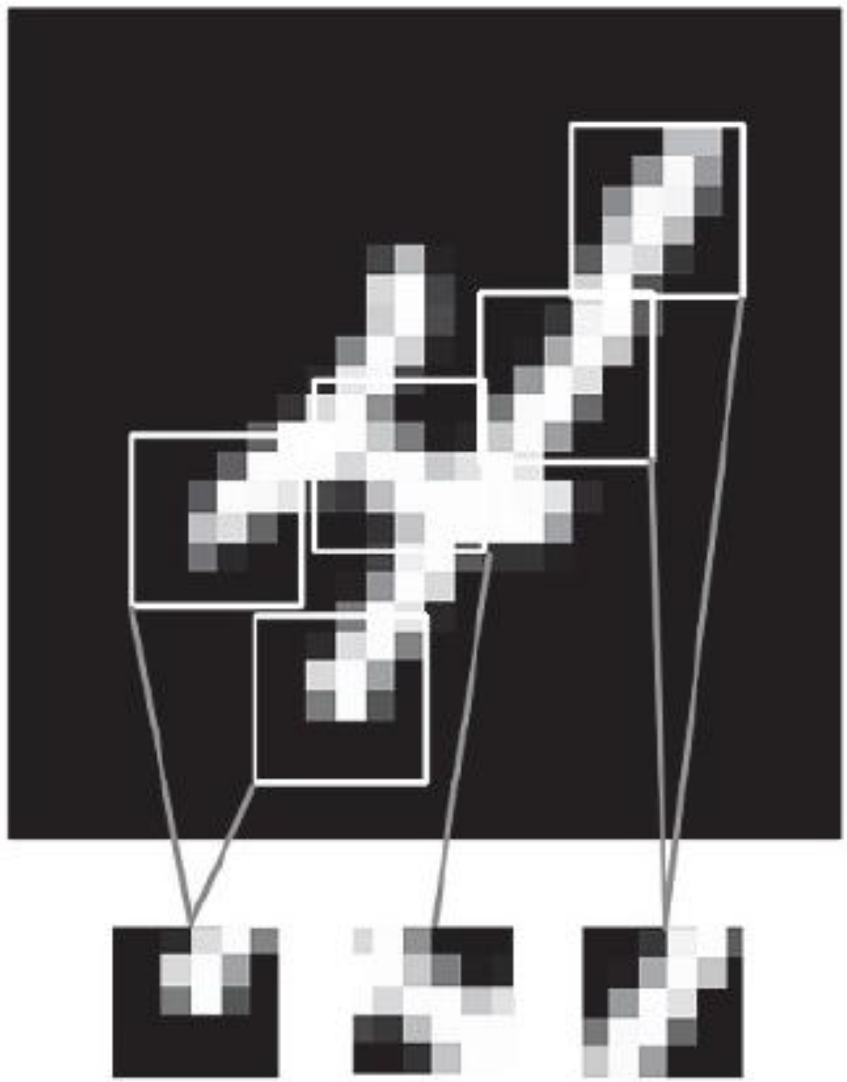
- **First Layer – Math Part**
  - The first layer in a CNN is always a **Convolutional Layer**.



Visualization of 5 x 5 filter convolving around an input volume and producing an activation map

Convolutional Layer

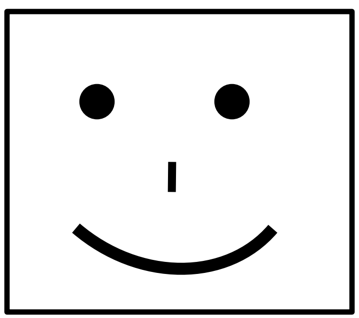
- The fundamental difference between a **densely-connected layer** and a **convolution layer** :
  - **Dense layers learn global patterns in their input feature space**
  - **Convolution layers learn local patterns i.e. in the case of images, patterns found in small 2D windows of the inputs**.

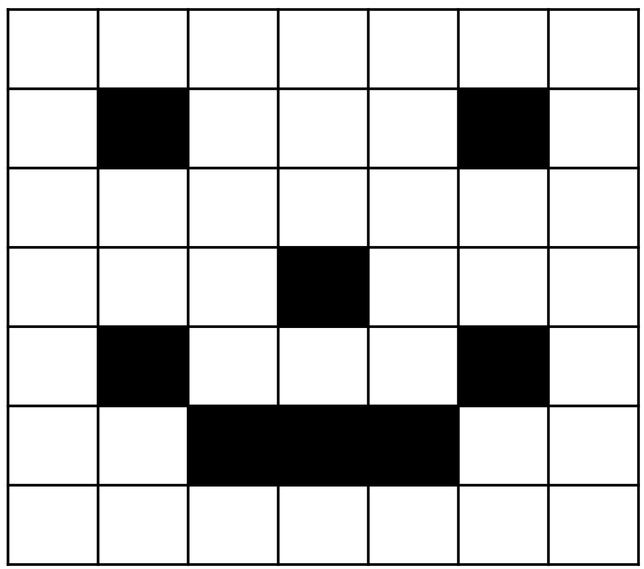


Convolutional Layer

- The patterns they learn are *translation-invariant :*
  - ***after learning a certain pattern in the bottom right corner of a picture, a convnet is able to recognize it anywhere, e.g. in the top left corner.***
  - ***A densely-connected network would have to learn the pattern anew if it appeared at a new location***
- This makes convnets very data-efficient when processing images : they need less training samples to learn representations that have generalization power

CNN





0	0	0	0	0	0	0
0	**1**	0	0	0	**1**	0
0	0	0	0	0	0	0
0	0	0	**1**	0	0	0
0	**1**	0	0	0	**1**	0
0	0	**1**	**1**	**1**	0	0
0	0	0	0	0	0	0

Step1-Convolution

0	0	0	0	0	0	0
0	**1**	0	0	0	**1**	0
0	0	0	0	0	0	0
0	0	0	**1**	0	0	0
0	**1**	0	0	0	**1**	0
0	0	**1**	**1**	**1**	0	0
0	0	0	0	0	0	0

Input image

X

0	0	1
1	0	0
0	1	1

Feature Detector

=

0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Feature Map1

Kernel Matrix examples

1	1	1
1	1	1
1	1	1

Unweighted 3x3 smoothing kernel

0	1	0
1	4	1
0	1	0

Weighted 3x3 smoothing kernel with Gaussian blur

0	-1	0
-1	5	-1
0	-1	0

Kernel to make image sharper

-1	-1	-1
-1	9	-1
-1	-1	-1

Intensified sharper image

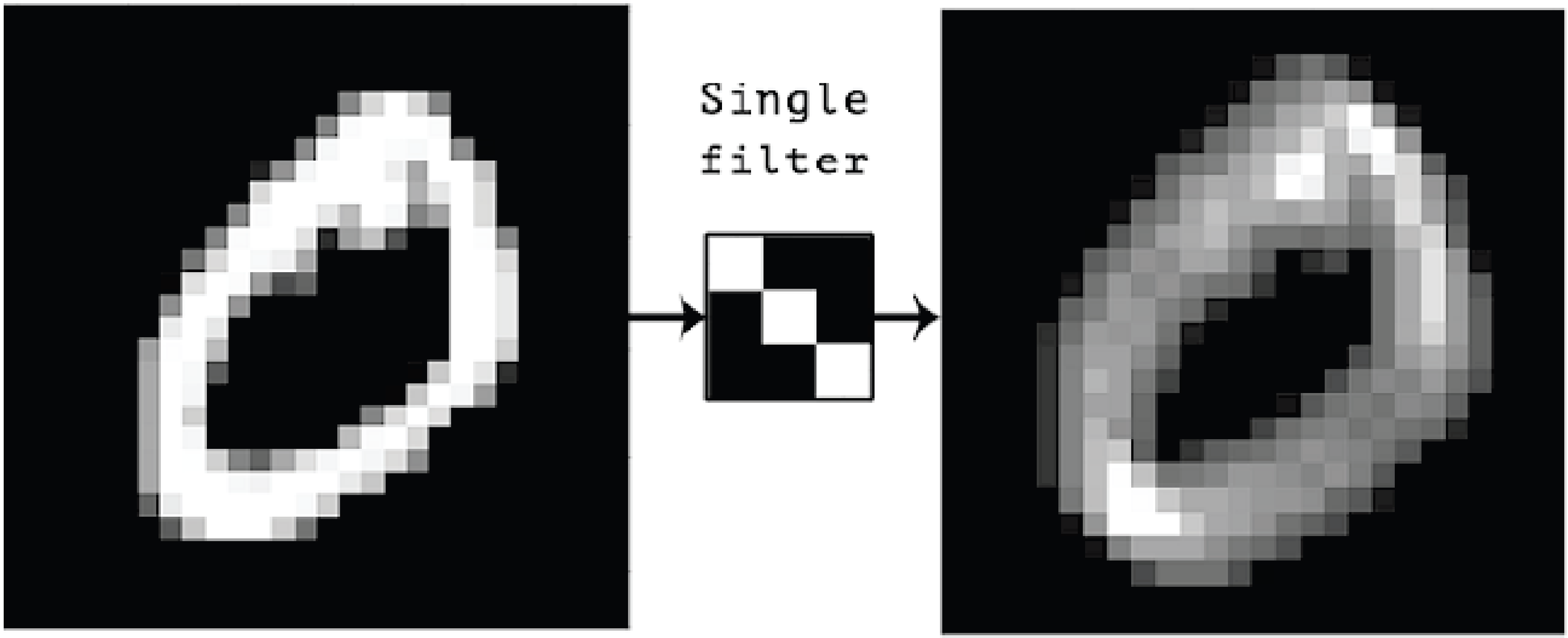
Gaussian Blur

Sharpened image

Many Filters in Convolution Layer

- Each Kernel matrix captures a specific type of pattern in input data
- Apply different filters by changing the kernel function
- There are many kernel functions/filters
  - Filter for detecting the curves
  - Filter for detecting the circles
  - Filter for detecting the sharp edges
  - Filter for detecting the straight lines

Feature Map / Response Map



- convolution is most typically done with 3x3 windows and no stride (stride 1).

Step1-Convolution

0	0	0	0	0	0	0
0	**1**	0	0	0	**1**	0
0	0	0	0	0	0	0
0	0	0	**1**	0	0	0
0	**1**	0	0	0	**1**	0
0	0	**1**	**1**	**1**	0	0
0	0	0	0	0	0	0

Input image

0	1	0
0	0	0
1	1	1

Feature Detector

0	0	0	0	0
1	1	1	1	1
1	1	0	1	1
1	2	4	2	1
1	0	0	0	1

Feature Map2

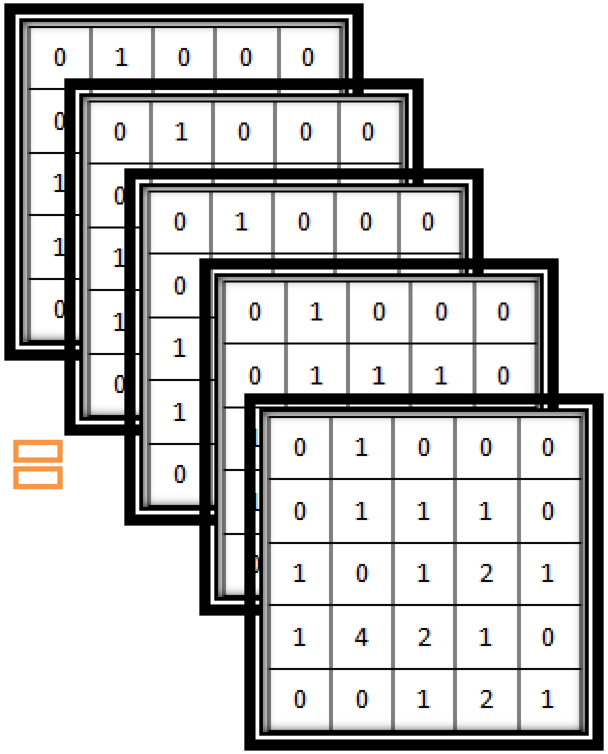
Step1-Convolution

0	0	0	0	0	0	0
0	**1**	0	0	0	**1**	0
0	0	0	0	0	0	0
0	0	0	**1**	0	0	0
0	**1**	0	0	0	**1**	0
0	0	**1**	**1**	**1**	0	0
0	0	0	0	0	0	0

Input image

0	0	1
1	0	0
0	1	1

Feature Detector



Feature Map

Effective Convolution with Kernel Matrix

1	1	1	0	0	1	1
1	1	0	1	0	1	1
1	0	0	1	1	1	0
0	0	0	0	0	1	1
1	1	0	0	0	0	1
1	0	0	1	1	1	0
1	1	1	0	0	1	1

kernel matrix

1	1	1
1	1	1
1	1	1

6	5			

- Use Kernel
  - To enhance intensity
  - or to make the image smooth
  - or to make it sharp

Effective Convolution with Kernel Matrix

1	1	1	0	0	1	1
1	1	0	1	0	1	1
1	0	0	1	1	1	0
0	0	0	0	0	1	1
1	1	0	0	0	0	1
1	0	0	1	1	1	0
1	1	1	0	0	1	1

kernel matrix

0	1	0
1	1	1
0	1	0

- There are different versions of kernels

3				

Many Filters in Convolution Layer

1	1	1	0	0	1	1
1	1	0	1	0	1	1
1	0	0	1	1	1	0
0	0	0	0	0	1	1
1	1	0	0	0	0	1
1	0	0	1	1	1	0
1	1	1	0	0	1	1

0	0	1
0	1	1
1	1	1

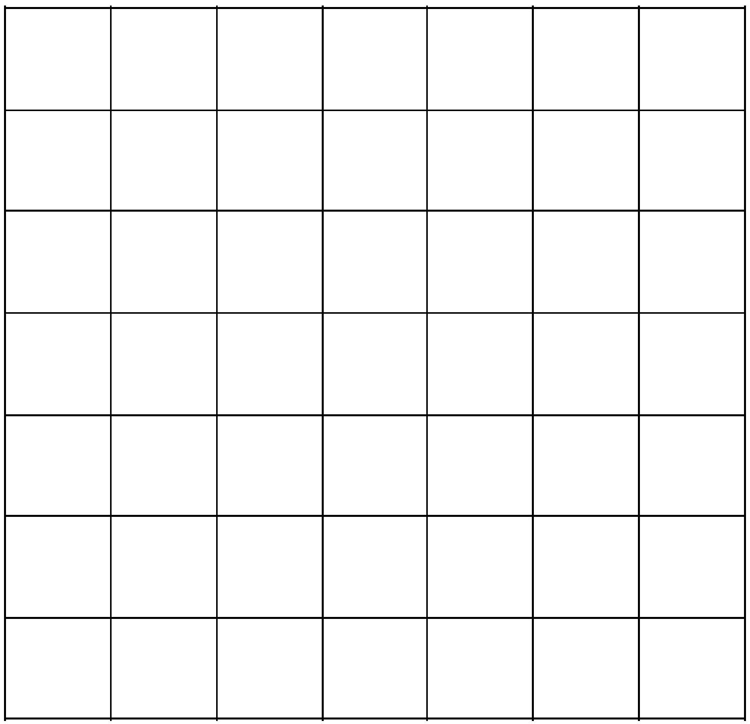
1	1	1
1	1	0
1	0	0

0	0	15
0	15	0
15	0	0

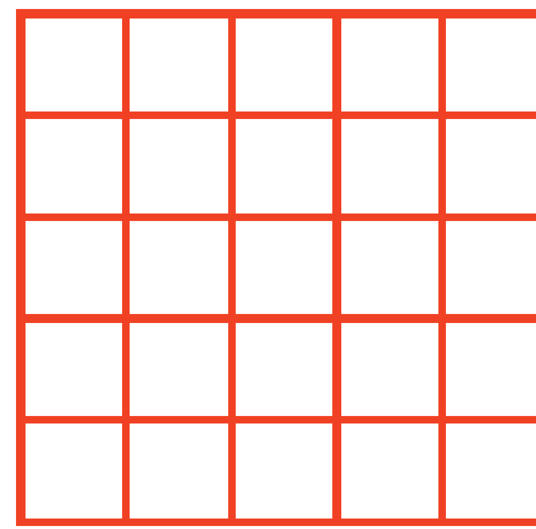
0	10	0
10	10	10
0	10	0

Different kernel matrices

First Hidden Layer in CNN is Convolution Layer

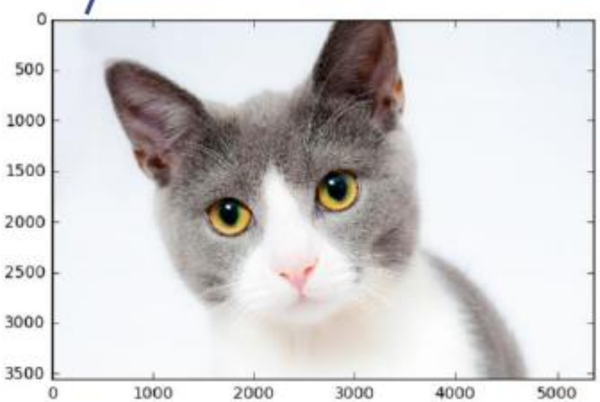


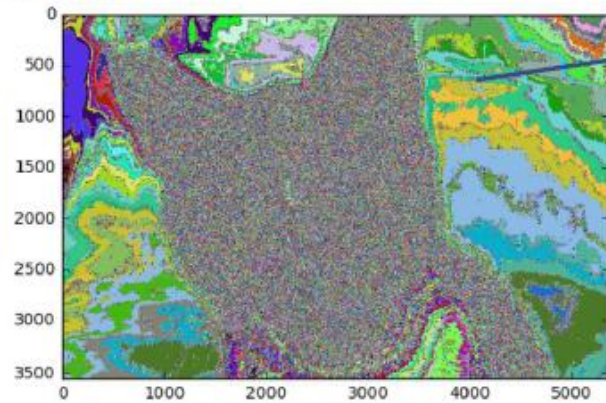
- By Applying filters and kernel we are creating new convoluted features
- The hidden layer in ANN is now the convoluted layer
- Convoluted features keep the input data integrity intact with lesser data points



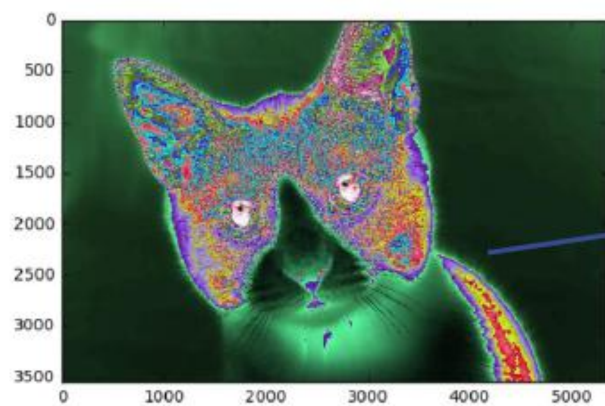
Convoluted features

Different filters capture different features





A filter to capture overall shape and edges



A filter to ignore white spaces on image

The depth in convolution layer

- Every filter gives us a resultant matrix (activation map)

0	0	0	0	0	0	0	0	0
0	1	1	1	0	0	1	1	0
0	1	1	0	1	0	1	1	0
0	1	0	0	1	1	1	0	0
0	0	0	0	0	0	1	1	0
0	1	1	0	0	0	0	1	0
0	1	0	0	1	1	1	0	0
0	1	1	1	0	0	1	1	0
0	0	0	0	0	0	0	0	0

1	1	1
1	1	1
1	1	1

4	5	4	2	3	4	4
5	6	5	4	6	6	5
3	3	3	3	6	6	5
3	3	2	2	4	5	4
3	3	2	2	4	5	4
5	6	4	3	4	5	4
3	4	3	3	4	4	3

The depth in convolution layer

- With two filters we will get two activation maps i.e depth=2

0	0	0	0	0	0	0	0	0
0	1	1	1	0	0	1	1	0
0	1	1	0	1	0	1	1	0
0	1	0	0	1	1	1	0	0
0	0	0	0	0	0	1	1	0
0	1	1	0	0	0	0	1	0
0	1	0	0	1	1	1	0	0
0	1	1	1	0	0	1	1	0
0	0	0	0	0	0	0	0	0

1	1	1
1	1	1
1	1	1

1	1	1
1	-1	1
1	1	1

4	5	4	2	3	4	4

3	4	3	2	3	3	3
4	5	5	3	6	5	4
2	3	3	2	5	5	5
3	3	2	2	4	4	3
2	2	2	2	4	5	3
4	6	4	2	3	4	4
2	3	2	3	4	3	2

The depth in convolution layer

- If we apply 10 different filters then we will get 10 resultant matrices. The depth is 10

0	0	0	0	0	0	0	0	0
0	1	1	1	0	0	1	1	0
0	1	1	0	1	0	1	1	0
0	1	0	0	1	1	1	0	0
0	0	0	0	0	0	1	1	0
0	1	1	0	0	0	0	1	0
0	1	0	0	1	1	1	0	0
0	1	1	1	0	0	1	1	0
0	0	0	0	0	0	0	0	0

1	1	1
1	-1	1
1	1	1

3	4	3	2	3	3	3
4	5	5	3	6	5	4
2	3	3	2	5	5	5
3	3	2	2	4	4	3
2	2	2	2	4	5	3
4	6	4	2	3	4	4
2	3	2	3	4	3	2

Max Pooling

Pooling layer

- Pooling helps in further downsizing the data
- Pooling is used to avoid overfitting and increase the robustness
- Reduces lot of computation time

What happens in pooling layer

- Down sampling of the data.
- Since we preserved lot of information in each convolution layer, we can comfortably down sample it, in this pooling layer
- Yes, we do loose some information, but we will not loose the overall integrity of the data
- It is still good enough for a classification model
- Generally we try max pooling with 2X2 filter with stride 2

How many parameters in pooling layer

- No parameters
- We just perform down sampling, that's it. No further activation and no further parameters
- Number of parameters in pooling layer=0

Max Pooling

0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Feature Map

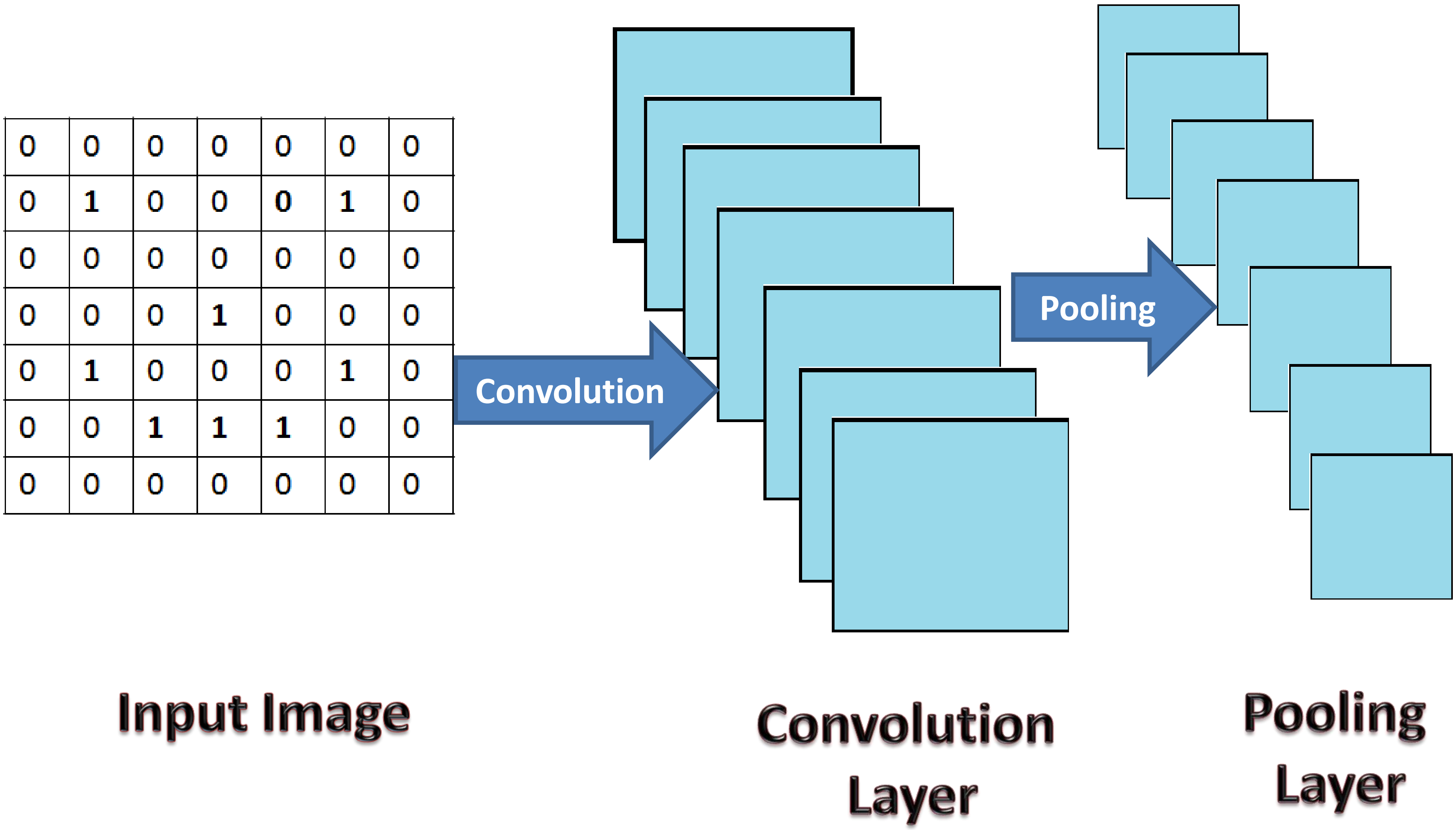
Pooling

1	1	0
4	2	1
0	2	1

Pool Feature Map

Max pooling is usually done with 2x2 windows and stride 2, so as to downsample the feature maps

Max Pooling


0 0 0 0 0 0 0
0 1 0 0 0 1 0
0 0 0 0 0 0 0
0 0 0 1 0 0 0
0 1 0 0 0 1 0
0 0 1 1 1 0 0
0 0 0 0 0 0 0
Convolution
Pooling
Input Image
Convolution Layer
Pooling Layer

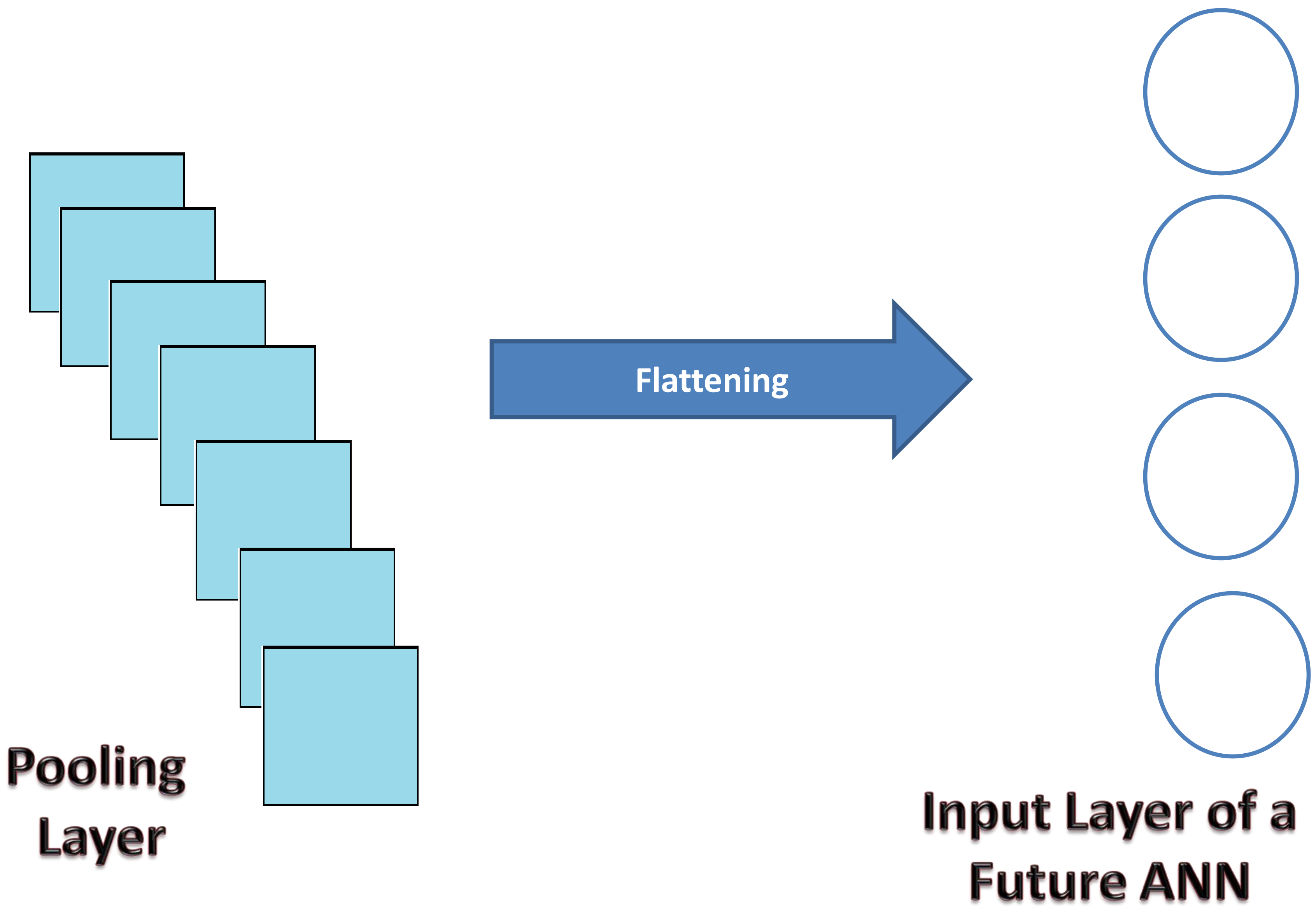
Flattening

1	1	0
4	2	1
0	2	1

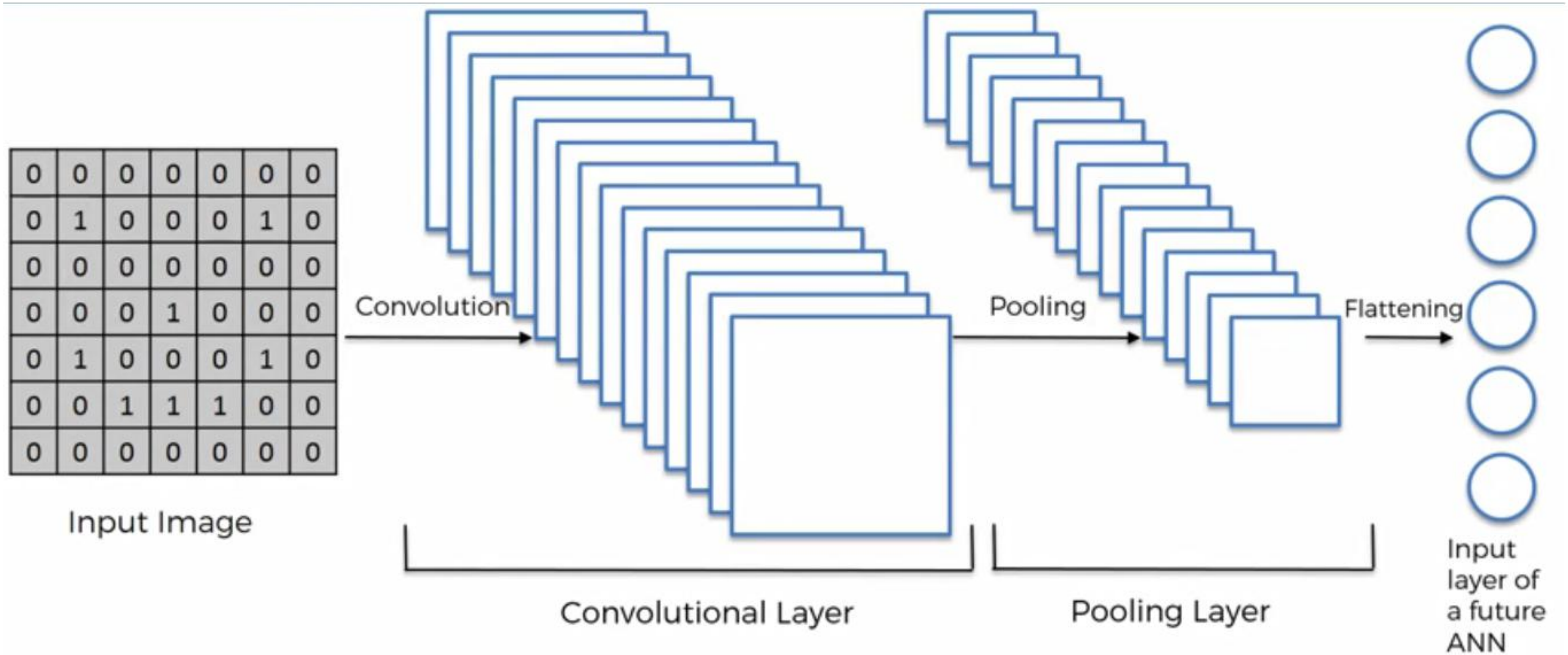
Pool Feature Map

1
1
0
4
2
1
0
2
1

Flattening

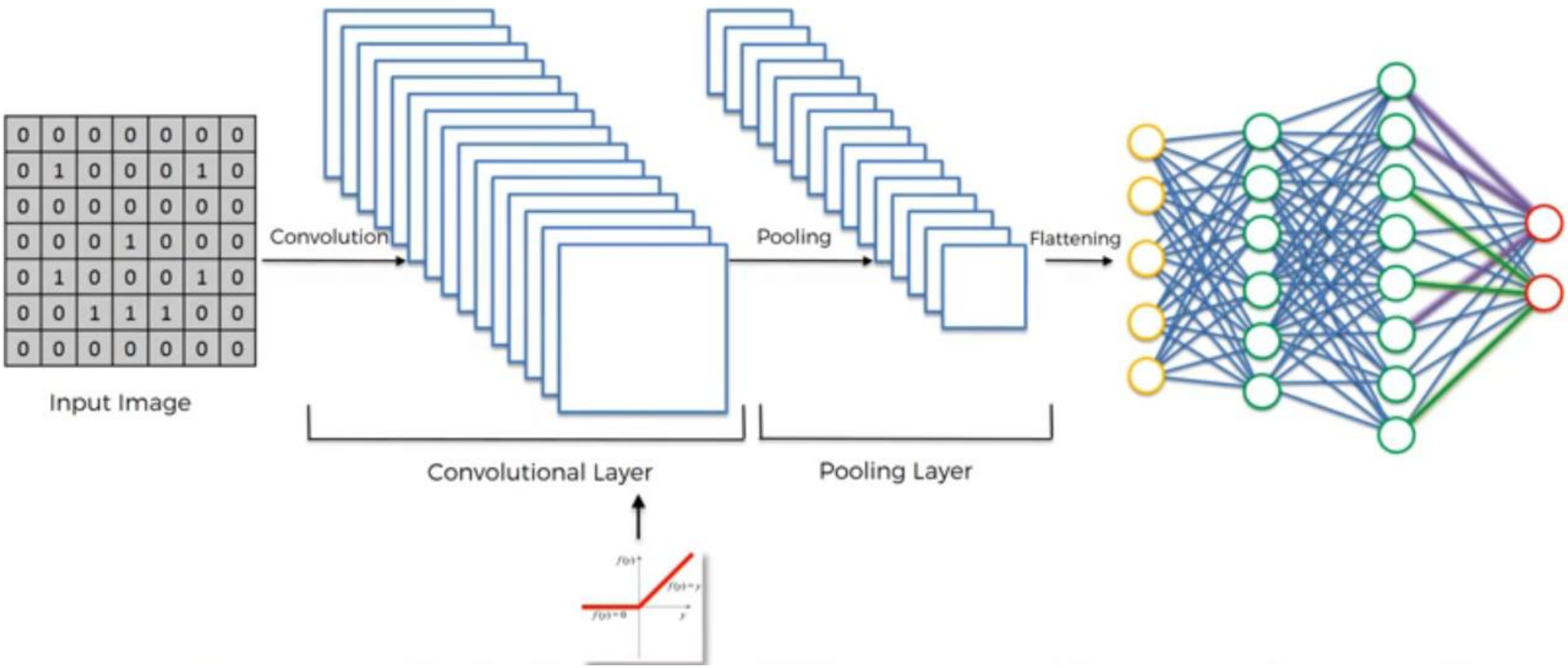


Flattening



0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
Input Image
Convolution
Pooling
Flattening
Convolutional Layer
Pooling Layer
Input layer of a future ANN

Full Connection



Data Augmentation

- *Having too few samples to learn from, rendering us unable to train a model able to generalize to new data.*

- *Given infinite data, our model would be exposed to every possible aspect of the data distribution at hand.*

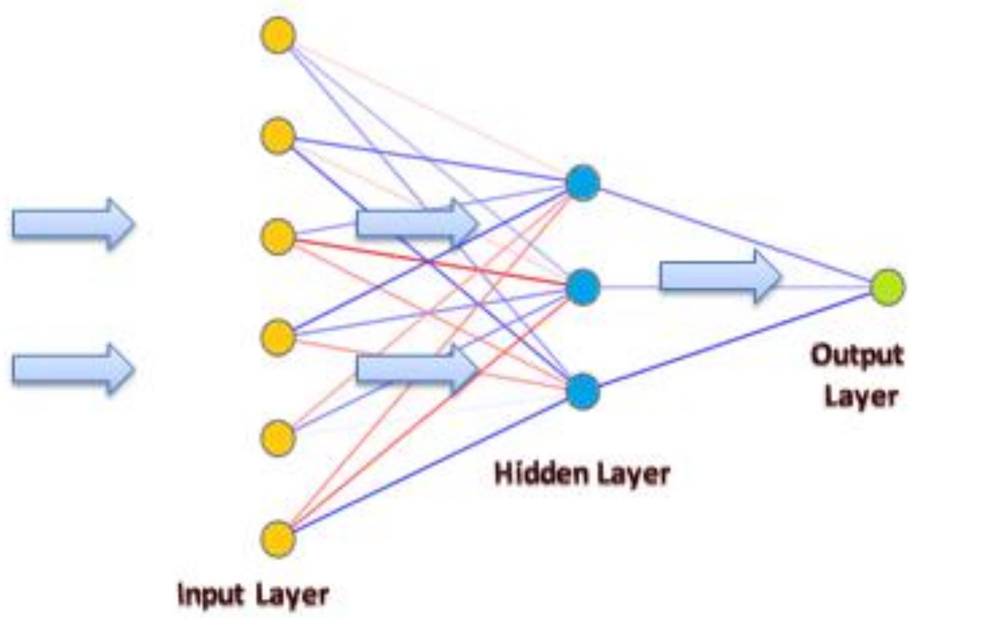
- *Data augmentation takes the approach of generating more training data from existing training samples, by "augmenting" the samples via a number of random transformations that yield believable-looking images.*

- *The goal is that at training time, our model would never see the exact same picture twice. This helps the model get exposed to more aspects of the data and generalize better.*

Generation of cat pictures via random data augmentation

Feedforward networks

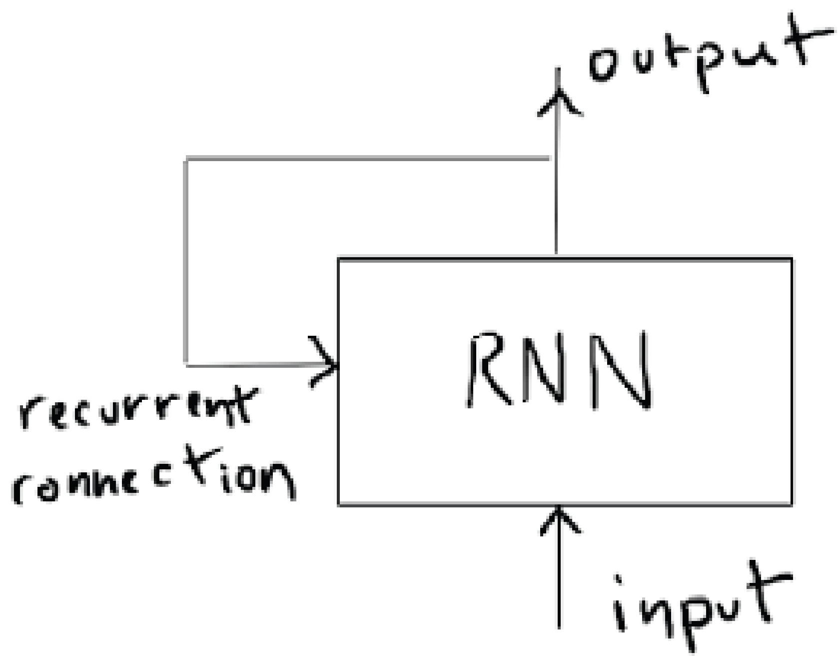
- A major characteristic of all neural networks , such as densely-connected networks and convn ...
- Each i ... sed indepe ... etween inputs
- With s ... ss a seque ... points, we have t ... ne network at once, i.e. turn it into a single datapoint.



RNN

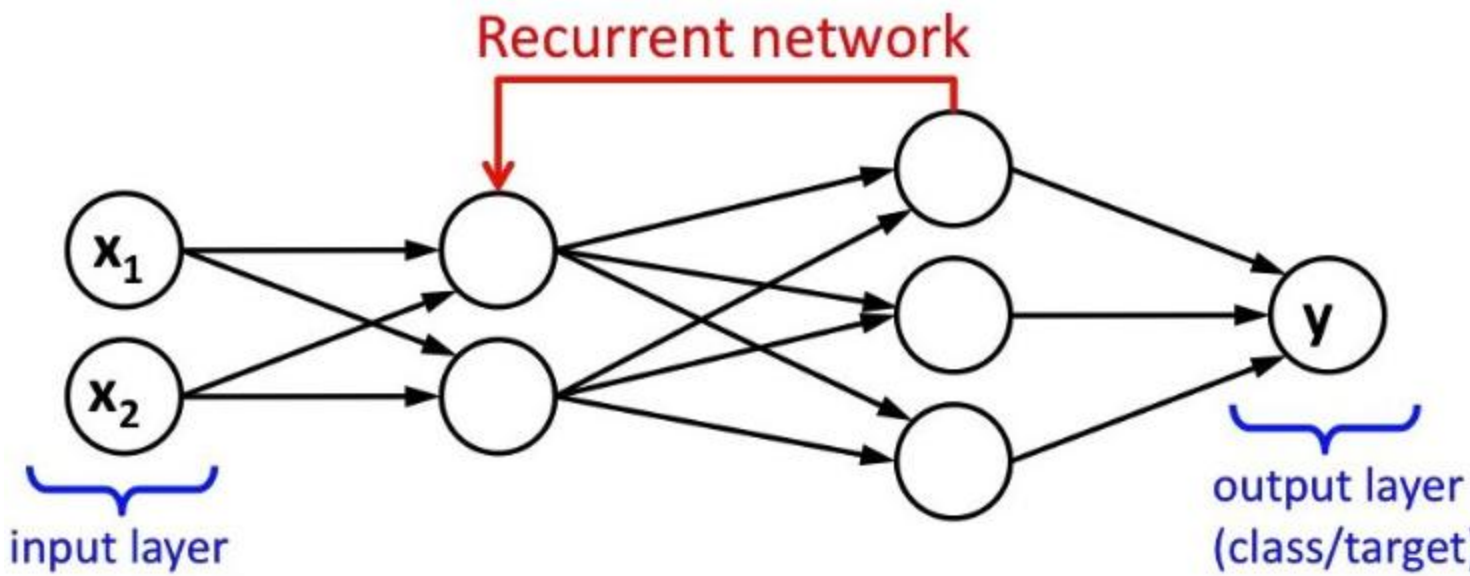
- Biological intelligence processes information incrementally while maintaining an internal model of what it is processing, built from past information and constantly getting updated as new information comes in.
- **Recurrent Neural Networks (RNNs) adopt the same principle**
- They process sequences by iterating through the sequence elements and maintaining a "state" containing information relative to what they have seen so far.

- RNNs are a type of neural network that has an internal loop



Recurrent Neural Network

On recurrent neural networks(RNN), the previous network state also influences the output, so recurrent neural networks also have a **"notion of time".**



Recurrent Neural Network

Multiple ways that you could use a recurrent neural network compared to the forward networks.

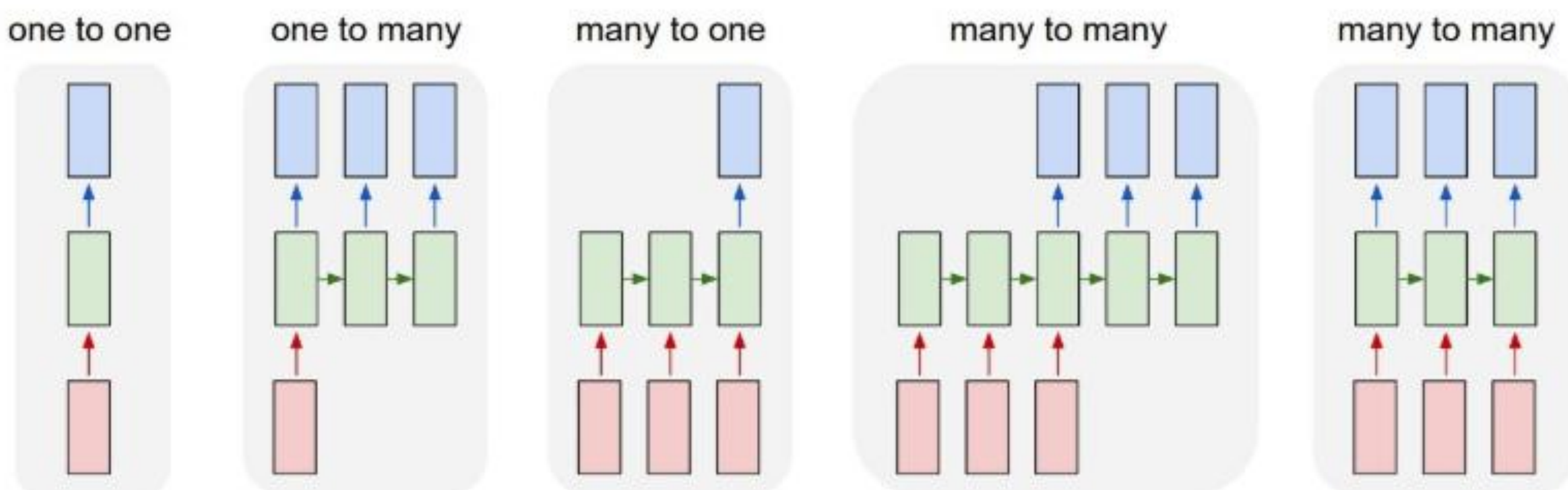


Image in
Label out

Image in
Words out

Words In
sentiment out

English In
French out

Video In
Label out

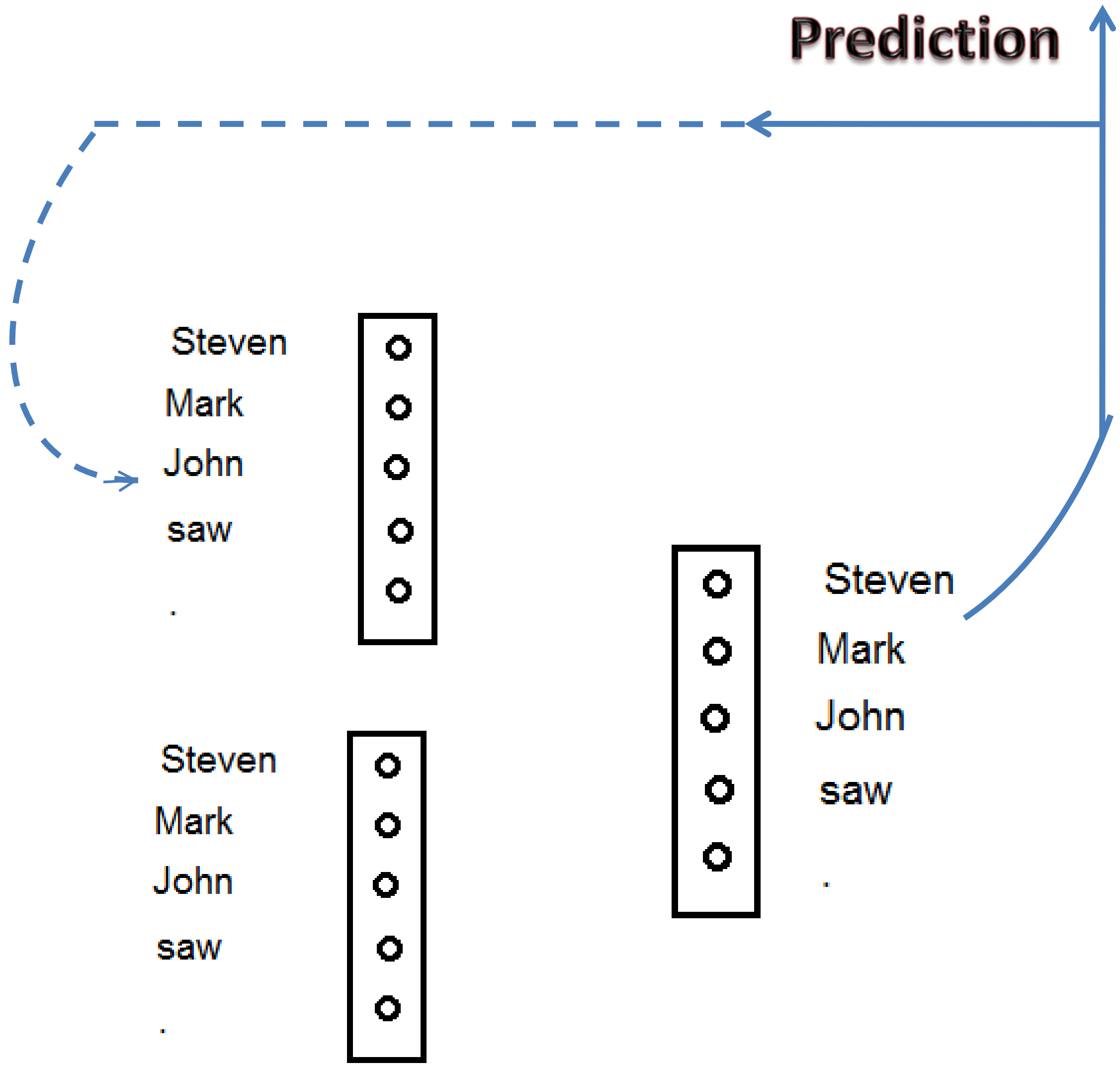
Recurrent Neural Network

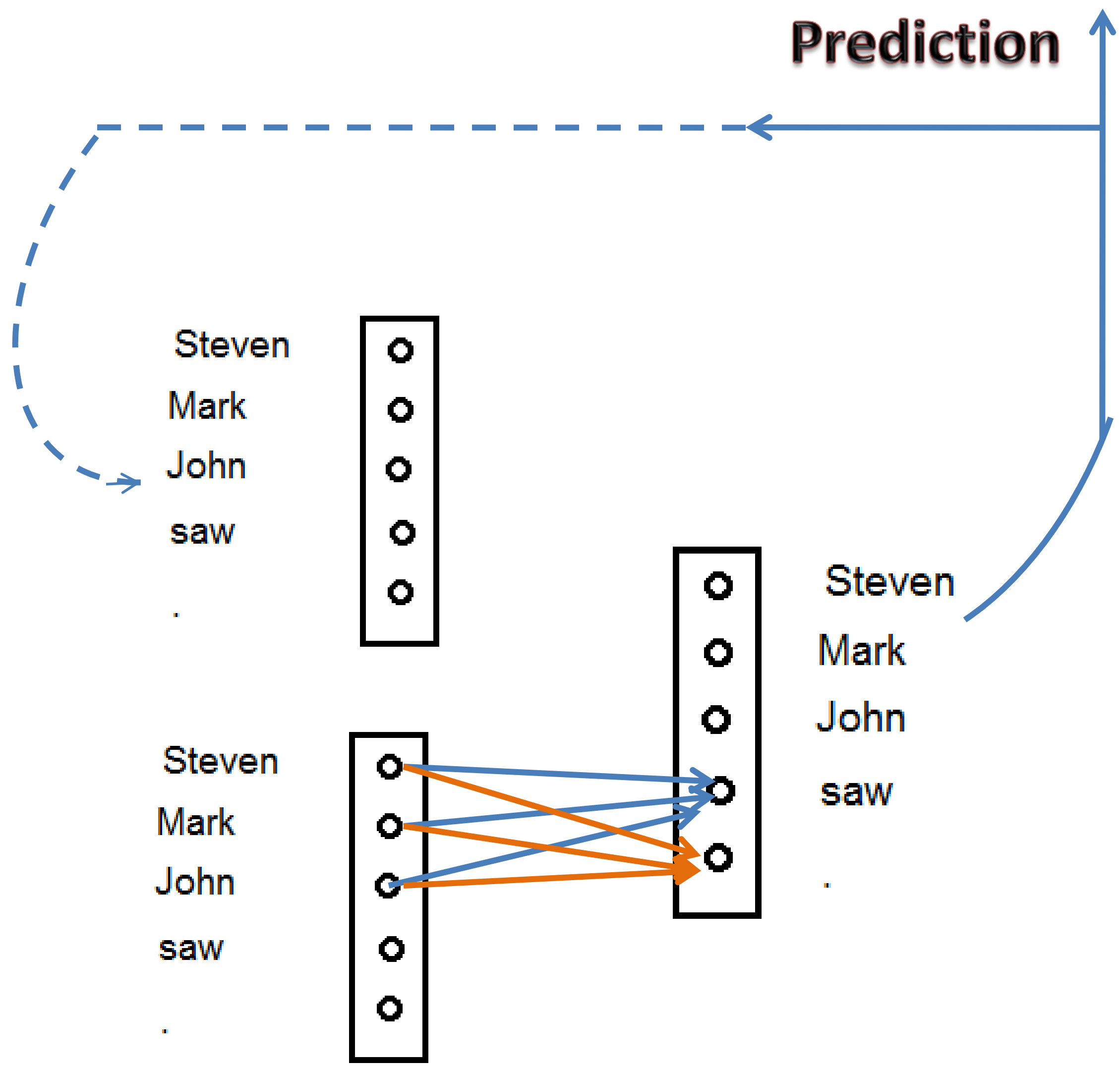
- Steven saw Mark.
- Mark saw John.
- John saw Steven.

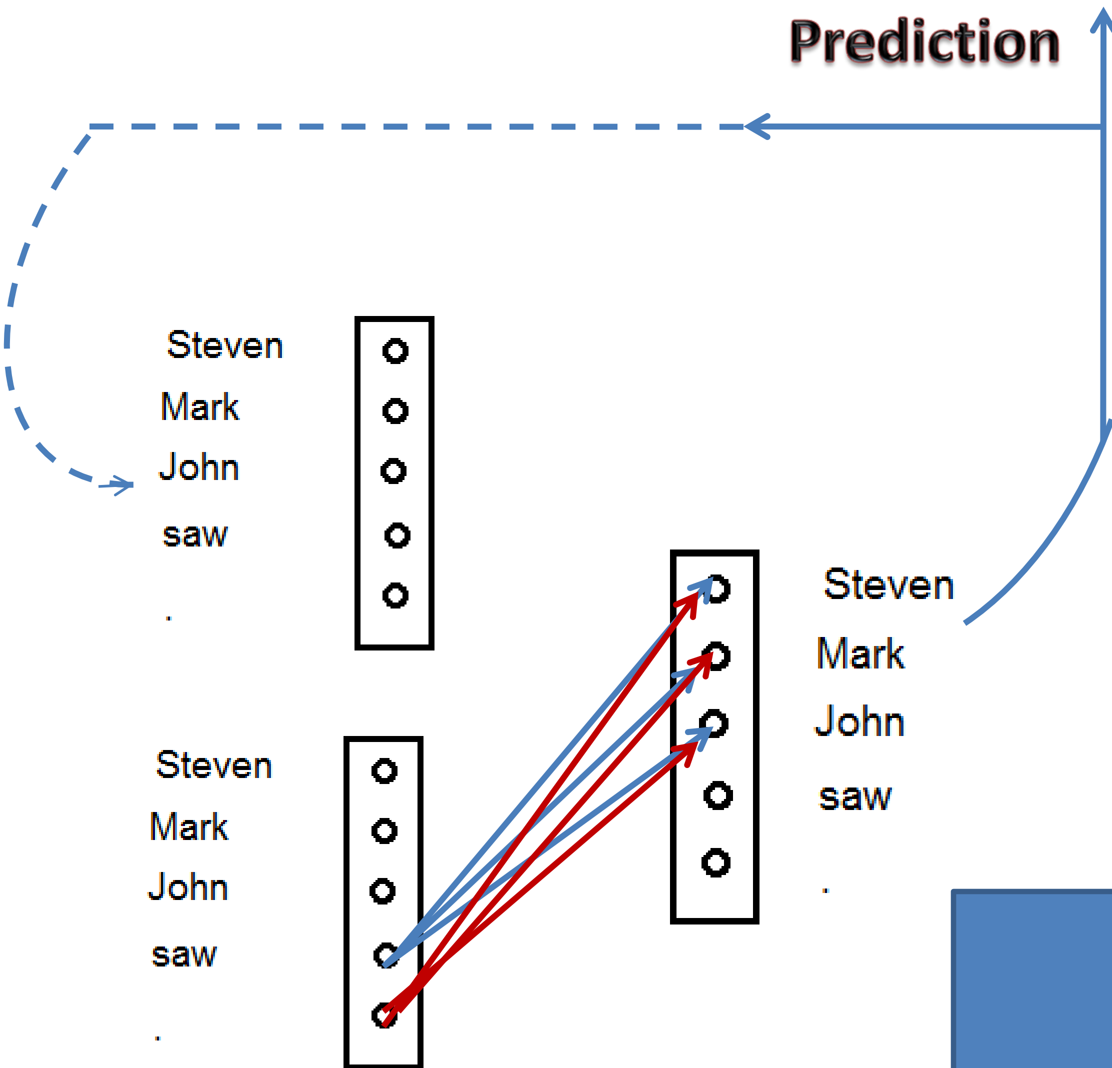
- Small Dictionary of words ::

  {Steven, Mark, John, saw, .}

Steven saw Mark.
Mark saw John.
John saw Steven.




Prediction
Steven
Mark
John
saw
.
Steven
Mark
John
saw
.
Steven
Mark
John
saw
.



Steven saw Mark.

Steven saw Mark saw John saw …

Steven.Mark.John.