Enhance your Java applications with Deep Learning using Deep Netts

Dr Zoran Sevarac, Deep Netts CEO

DEEP NETTS

Demystifying Machine Learning

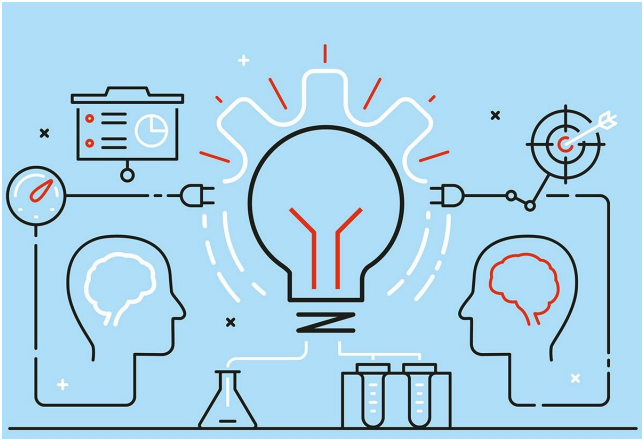
A computer algorithm that is capable to

automatically configure internal parameters (learn)

by looking at a set of examples given as data set,

in order to perform specific task on a similar data with

the highest possible accuracy.



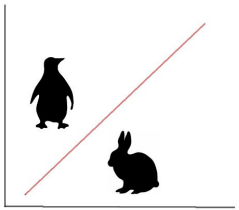
What it can do

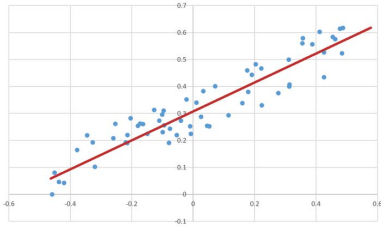
It can learn from examples to:

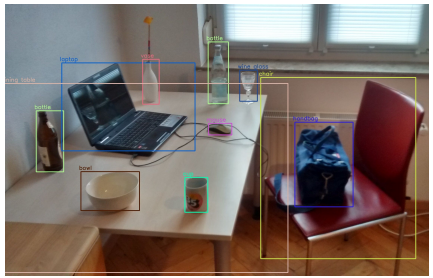
1) Classify data
   classification task

2) Estimate numeric value
   regression task

3) Recognize image
   classification + regression(position)







DEEP NETTS

How can that be usefull

Classification

Predict Banner Click

Detect Spam or Fraud

Classify web pages

Classify support ticket

Classify users

Regression

Predict application performance for given load

Optimize configuration parameters

Predict sales or cost

Image Recognition

Image classification (automate content management)

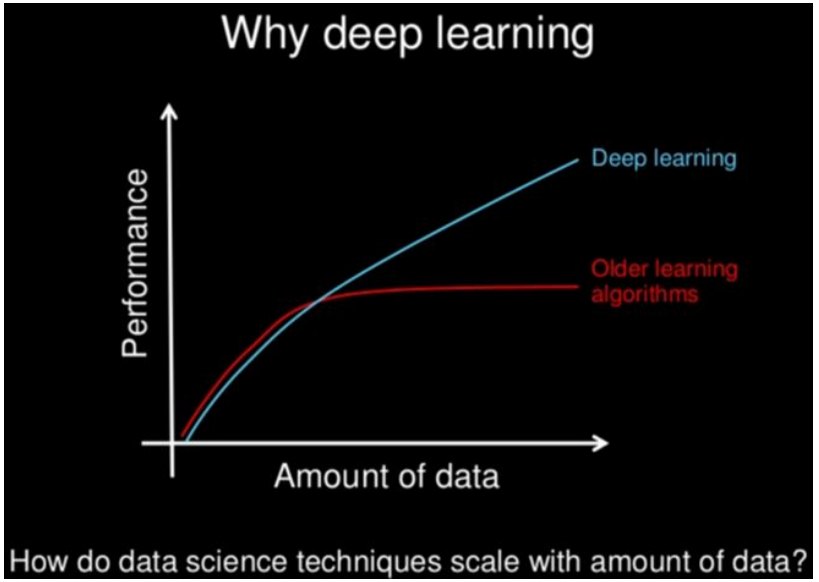
Visual Inspection

Object Detection (recognize multiple objects in image)

DEEP NETTS

Why Deep Learning

- Higher accuracy with more data
- Successful with high dimensional data (large number of inputs)
- Automates steps from traditional machine learning



DEEP NETTS

Challenges for Java Developers

- Complex to get started with and learn
  - Many algorithms - which one to use?
  - Many settings - what are these for?

- Great flexibility BUT difficult to use - what do I do with all of this?

  Well you just need to learn some basic linear algebra, calculus, optimization, statistics, basic machine learning,…

  ...and then you'll be able to start to understand deep learning

- And you have great choice of many machine learning platforms

What is special about Deep Netts?

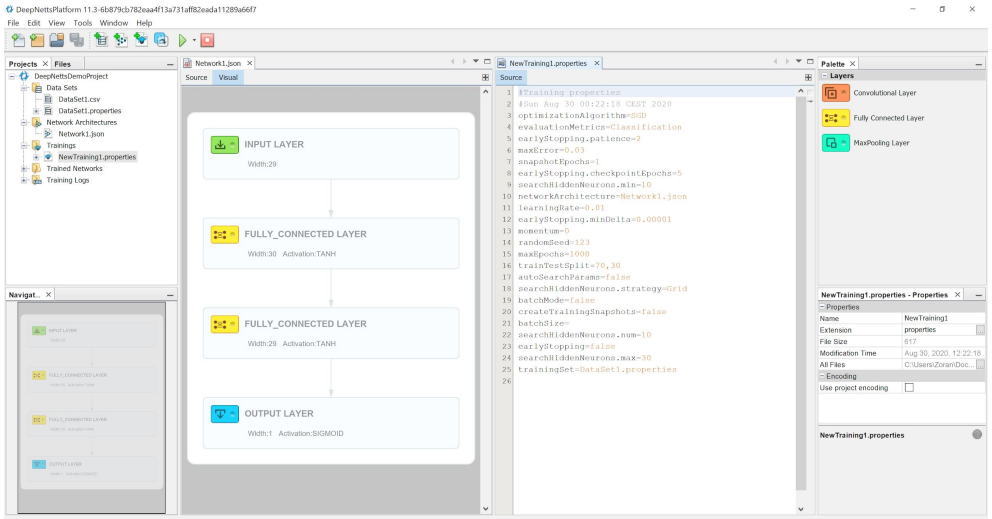
If you know basic Java, you can apply deep learning using Deep Netts.

After learning Java during the lockdown, a 14-year-old created deep learning for his sudoku game in Java using Deep Netts - means no PhD required, no heavy math!

Full article at:
https://www.deepnetts.com/blog/how-14-year-old-learned-deep-learning-in-java-using-deep-netts.html

About Deep Netts

- Pure Java Deep Learning Toolkit
- Library and Developer friendly tool
- Community Edition (Open Source)
- Official Java Standard - JSR 381



Basic Machine Learning Tasks with Deep Netts Wizard

Classification

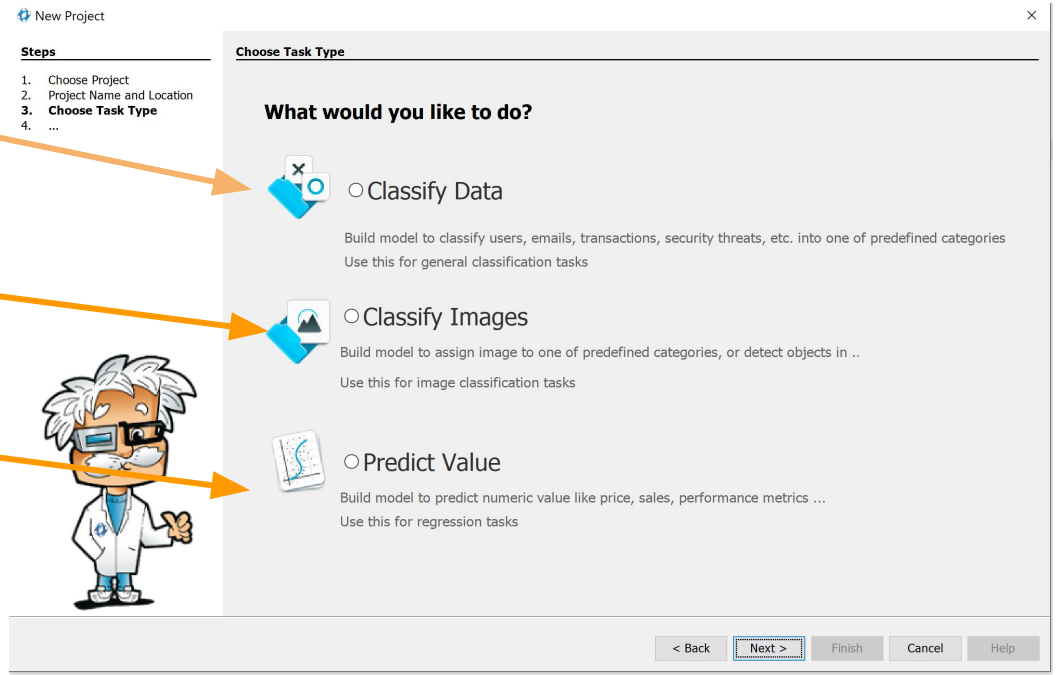
Assign item to category

Image Recognition

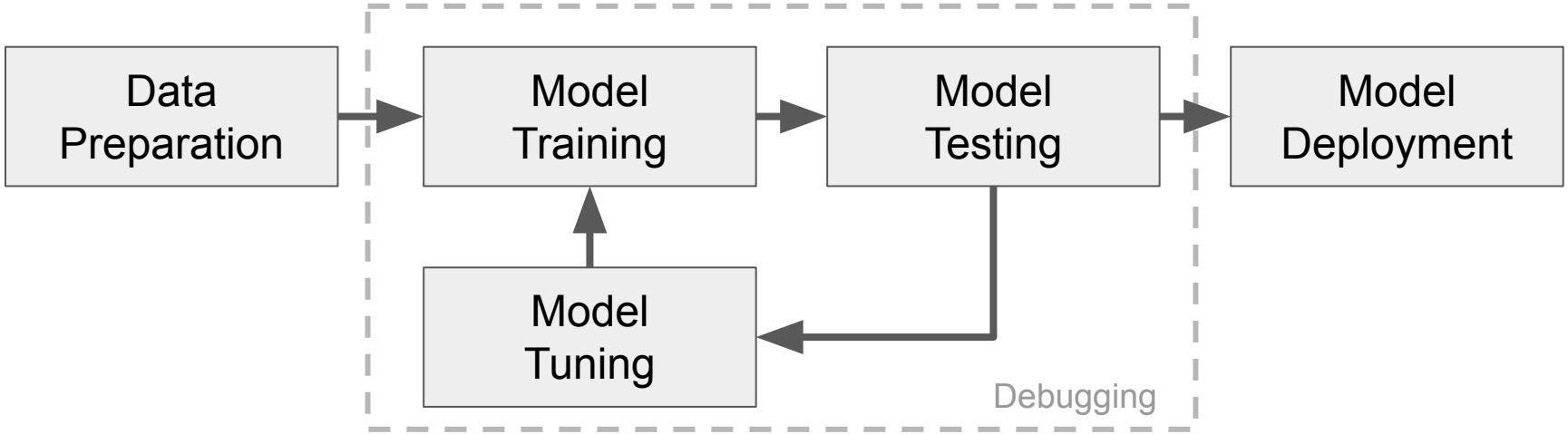
Assign image to category

Regression

Predict numeric value



Machine Learning Workflow



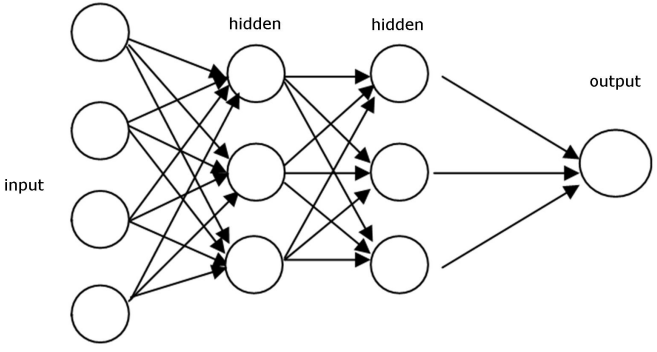
Using Deep Netts:

Import data from file or directory, cleanup and prepare for training	Quick and easy experimentation with various settings	JAR with model and Deep Netts library JAR

DEEP NETTS

Basic Deep Learning Model - FeedForward Neural Network

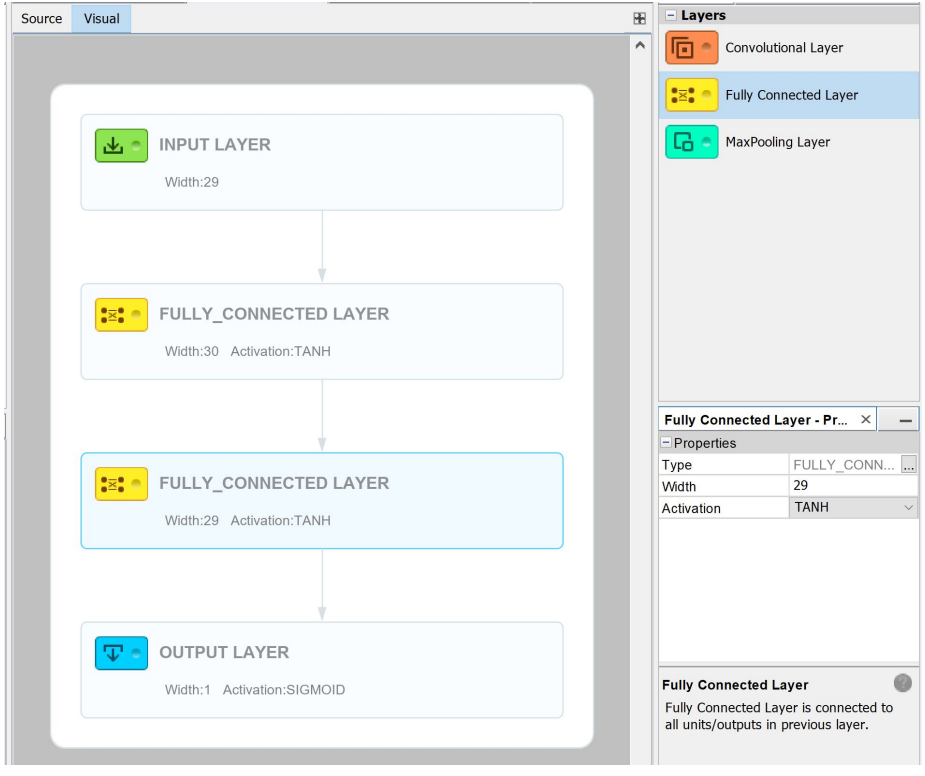
Directed graph in which each node performs some computation



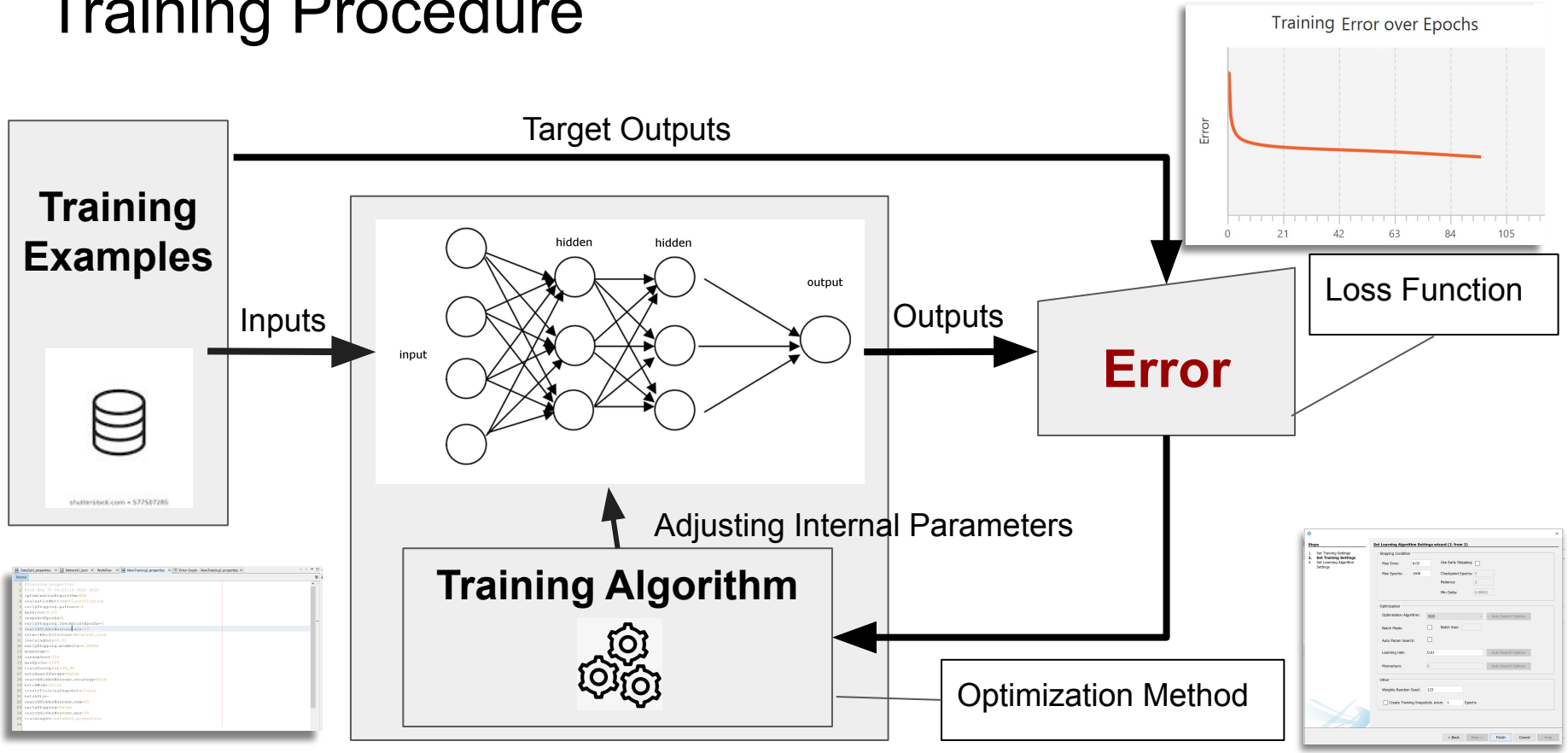
Nodes are grouped into layers to determine the order of computation.

Each node performs computation based on it's inputs and set of weight coefficients.

Learning based on error minimization of total network error for given data set.



Training Procedure



https://www.deepnetts.com/blog/from-basic-machine-learning-to-deep-learning-in-5-minutes.html

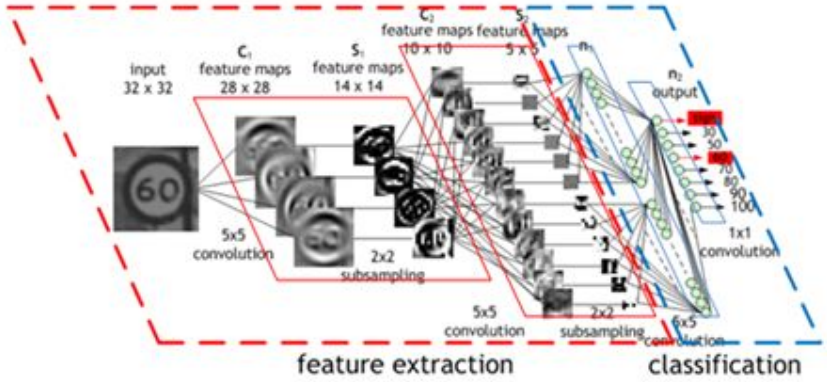
DEEP NETTS

Image Recognition using Convolutional Neural Network

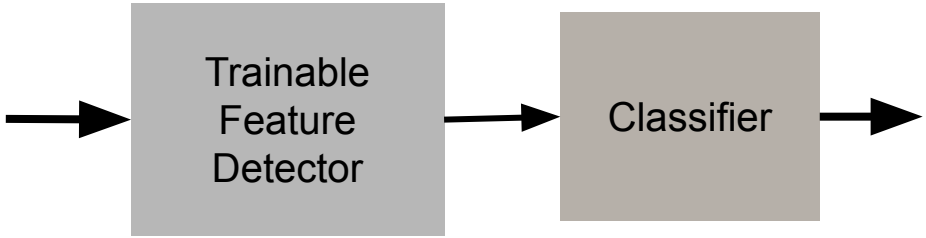
- Extension of a feed forward network specialized for image classification/recognition

- Introduces convolutional layers with adaptive image filters capable for learning and detecting shape and color features

- Reduces manual image preprocessing and feature extraction - it's learned during the training



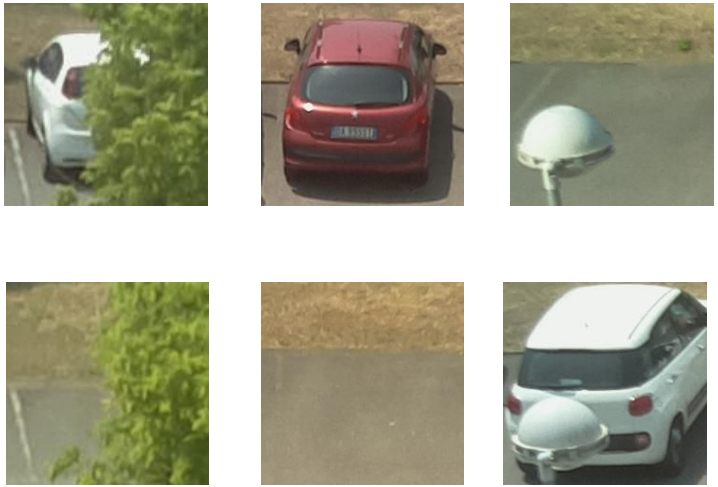
https://developer.nvidia.com/discover/convolutional-neural-network



DEEP NETTS

Example: Parking Lot Occupancy Detection

Image classification using convolutional neural network



A Dataset for Visual Occupancy Detection of Parking Lots: CNRPark+EXT
Giuseppe Amato, Fabio Carrara, Fabrizio Falchi, Claudio Gennaro, Claudio Vairo
Deep learning for decentralized parking lot occupancy detection

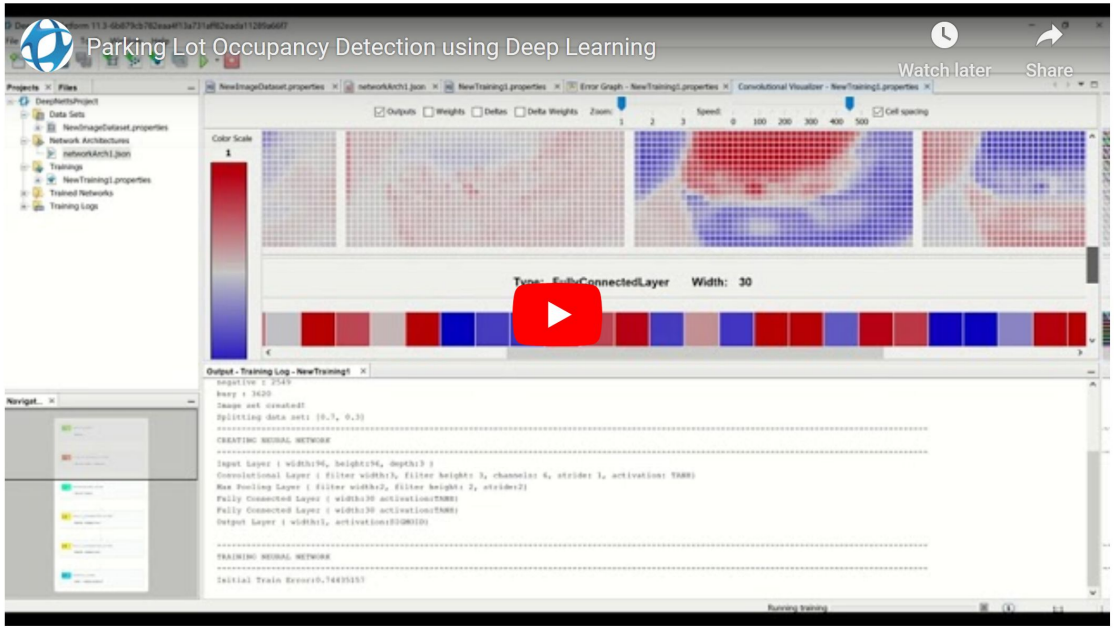
http://cnrpark.it/

Java Code Example for Parking Lot Occupacy

Full source using Deep Netts Community Edition available at GitHub:
https://github.com/deepnetts/parking-lot-occupancy-detection

```
ImageSet imageSet = new ImageSet(imageWidth, imageHeight);
LOG.info("Loading images...");
imageSet.loadLabels(new File(labelsFile));
imageSet.loadImages(new File(trainingFile));
ImageSet[] imageSets = imageSet.split(0.60, 0.40);

// create convolutional neural network
LOG.info("Creating neural network...");
ConvolutionalNetwork legoPeopleNet = ConvolutionalNetwork.builder(,
                                     .addInputLayer(imageWidth, imageHeight, 3)
                                     .addConvolutionalLayer(6, 3, 3, ActivationType.TANH)
                                     .addMaxPoolingLayer(2,2,2)
                                     .addFullyConnectedLayer(30, ActivationType.TANH)
                                     .addFullyConnectedLayer(10, ActivationType.TANH)
                                     .addOutputLayer(1, ActivationType.SIGMOID)
                                     .lossFunction(LossType.CROSS_ENTROPY)
                                     .randomSeed(123)
                                     .build();

// train convolutional network
LOG.info("Training neural network...");
BackpropagationTrainer trainer = legoPeopleNet.getTrainer();
trainer.setLearningRate(0.01f);
trainer.setMaxError(0.05f);
trainer.setMaxEpochs(15);
trainer.train(imageSets[0]);
LOG.info("Done training neural network.");

ClassifierEvaluator evaluator = new ClassifierEvaluator();
EvaluationMetrics  pm =  evaluator.evaluate(legoPeopleNet, imageSets[1]);
```

Demo video



https://www.deepnetts.com/blog/parking-lot-occupancy-detection-using-deep-learning-in-java.html

DEEP NETTS

Loading and Using Trained Network

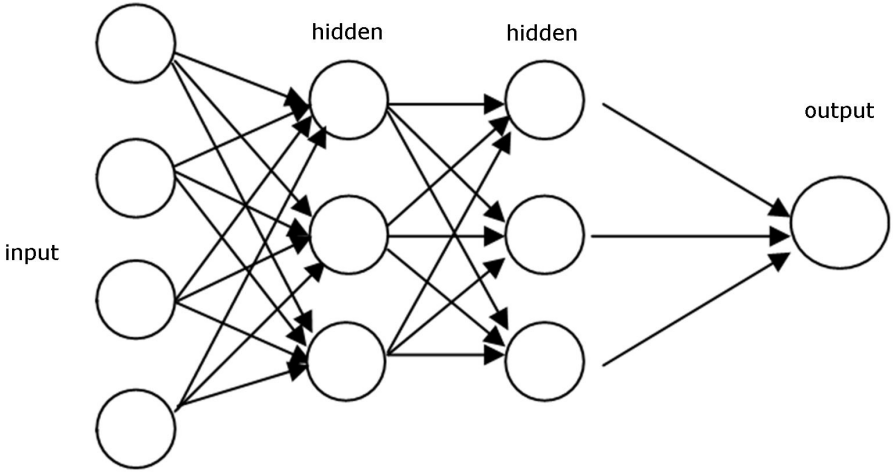
```
// Load trained network
ConvolutionalNetwork neuralNet =
        FileIO.createFromFile("savedNetwork.dnet", ConvolutionalNetwork.class);

// Create image classifier from trained network
ConvolutionalImageClassifier imageClassifier = new
                                        ConvolutionalImageClassifier(neuralNet);

// Classify some image
Map<String, Float> results = imageClassifier.classify(new File("someImage.png"));
```

DEEP NETTS

Data Clasification Example: Email spam

Decide if a given email is spam or not?

Binary classification task



70% chance that given email is a spam
(based on examples that the model was trained with)

DEEP NETTS

Data and Model

word46	word47	word48	char1	char2	char3	char4	char5	char6	capitalAvg	capitalLongest	capitalTotal	isSpam
0	0	0	0	0	0	0.778	0	0	3.756	61	278	1
0	0	0	0	0.132	0	0.372	0.18	0.048	5.114	101	1028	1
0.06	0	0	0.01	0.143	0	0.276	0.184	0.01	9.821	485	2259	1
0	0	0	0	0.137	0	0.137	0	0	3.537	40	191	1

CSV File

```
FeedForwardNetwork neuralNet = FeedForwardNetwork.builder()
                .addInputLayer(numInputs)
                .addFullyConnectedLayer(25, ActivationType.RELU)
                .addOutputLayer(numOutputs, ActivationType.SIGMOID)
                .lossFunction(LossType.CROSS_ENTROPY)
```

Neural network

Tutorial
https://blogs.oracle.com/developers/introduction-to-machine-learning-and-neural-networks-for-java-developers

Complete code example on **GitHub**:
https://github.com/deepnetts/SpamClassifier

Community Edition and VisRec API

Deep Netts Community Edition

https://www.deepnetts.com/blog/deep-netts-community-edition

Code examples:
https://github.com/deepnetts/deepnetts-communityedition/tree/community-visrec/deepnetts-examples

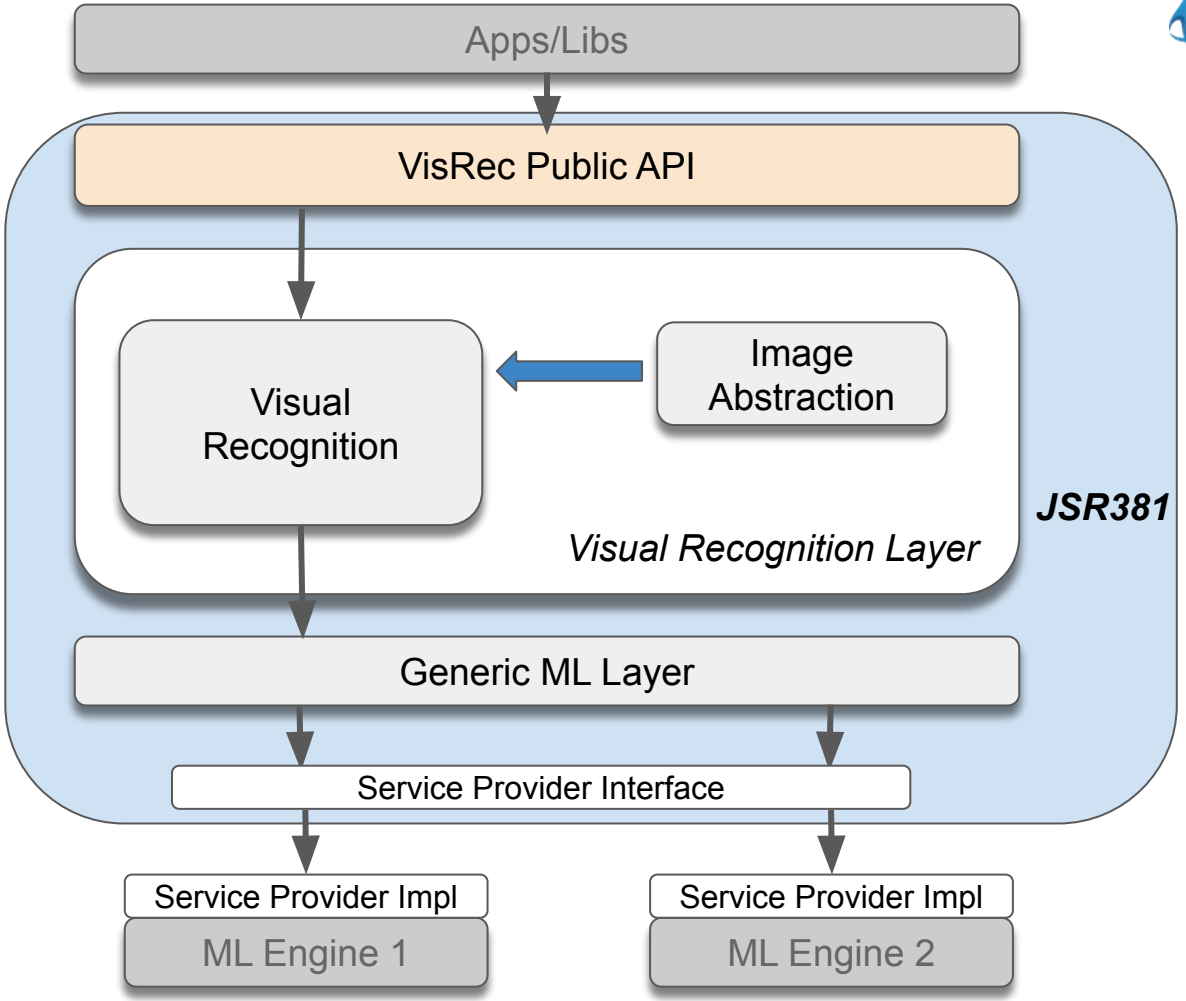
VisRec API, official Java Standard JSR381

Reference Implementation Based on Deep Netts

https://github.com/JavaVisRec

https://github.com/JavaVisRec/visrec-api/wiki/Getting-Started-Guide

DEEP NETTS



JSR 381 ARCHITECTURE

DEEP NETTS

Java Code Example in VisRec API

```
LOG.info("Loading images...");
imageSet.loadLabels(new File(labelsFile));
imageSet.loadImages(new File(trainingFile));
ImageSet[] imageSets = imageSet.split(0.60, 0.40);

// Configuration to train the model
ImageClassifier<BufferedImage> classifier = NeuralNetImageClassifier.builder()
                                                .inputClass(BufferedImage.class)
                                                .imageHeight(imageHeight)
                                                .imageWidth(imageWidth)
                                                .labelsFile(new File(labelsFile))
                                                .trainingFile(new File(trainingFile))
                                                .networkArchitecture(new File("mnist1.json"))
                                                .exportModel(Paths.get("mnist.dnet"))
                                                .maxError(0.09f)
                                                .maxEpochs(1000)
                                                .learningRate(0.01f)
                                                .build();
```

Next Steps

Download and Try Deep Netts
https://deepnetts.com/download

Only for participants of this session:
Build a pilot project and write blog post about it
To Get Free Development License

Oracle Groundbreaker Ambassador

The **Oracle Groundbreaker Ambassadors** award recognizes modern developers for their contribution to the developer community.

Experts in Modern Development

- Cloud
- Microservices and Containers
- Java, JavaScript/Node.js, PHP, Python
- DevOps
- Continuous Delivery
- Open Source Technologies
- SQL/NoSQL Databases
- Machine Learning, AI, Chatbots

Oracle Groundbreakers

Nominate a Groundbreaker Ambassador

gbanomination.oracle.com

Learn more at developer.oracle.com/ambassador

@groundbreakers