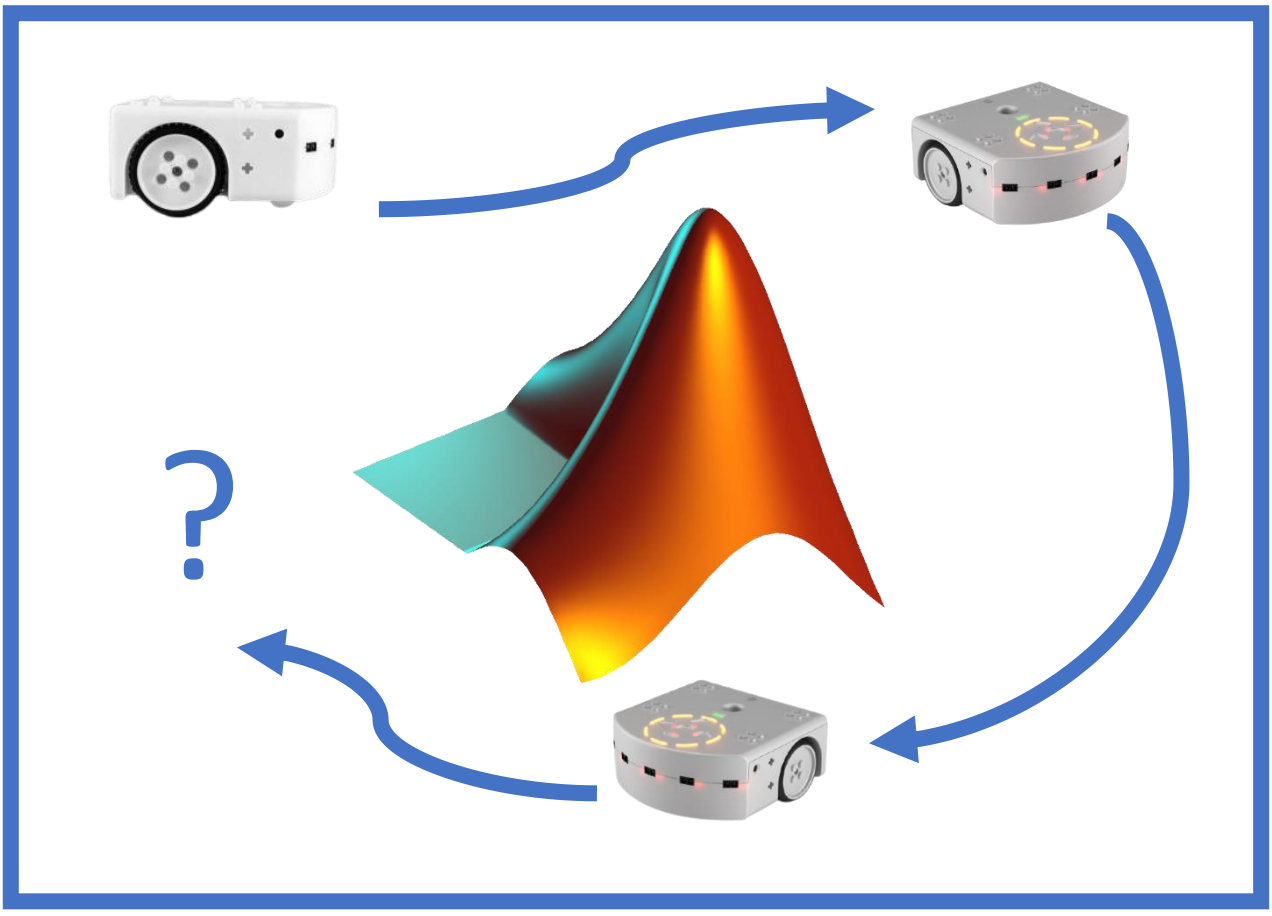
STAFFORDSHIRE UNIVERSITY

Artificial Intelligence and Machine Learning

Machine Learning – Neural Networks Coursework 1



Timothée A.Milhomme --- Artificial Intelligence and Machine Learning --- Academic Year 2020-2021

Abstract

This report is produced for the project of first semester of Master 1 at Staffordshire University in the module Artificial Intelligence and Machine Learning and is supervised by C GOONAWARDANE. In the introductory part we will try to define the key principles of the scope of this module study. Then, we will expose the two chosen solution to solve this project. The first solution is a hidden layer customized LSTM neural network and the second is a bidirectional LSTM neural network.

The Certifications

During the module we had the chance to develop our coding skill a lot through MATLAB certifications. This was interesting as we had a lot of exercise and practice and to play with the different form of Machine Learning. We had to complete 5 certifications:

- Machine Learning Onramp
- Deep Learning Onramp
- Image Processing Onramp
- Machine Learning with MATLAB
- Deep Learning with MATLAB

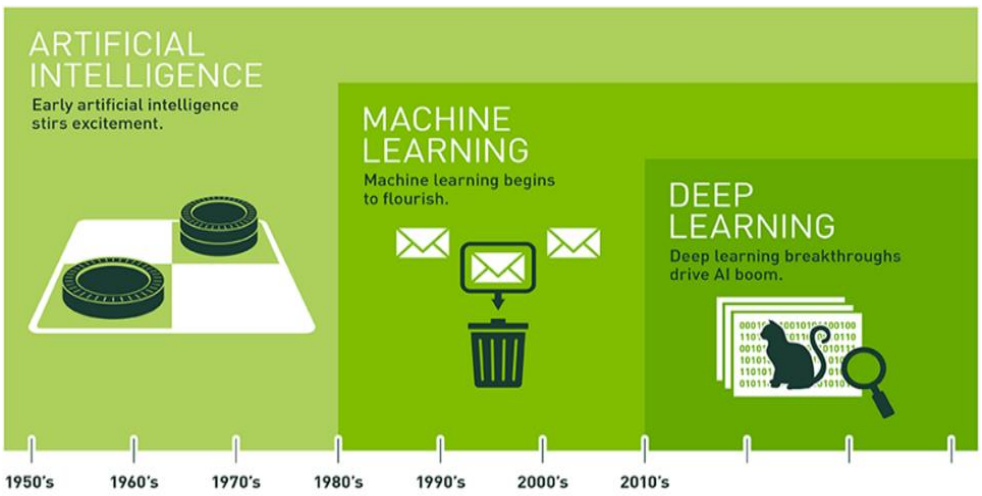
All the proof of the certification will be shown in the Annex.

Course Completion Certificate

Introduction

Artificial Intelligence

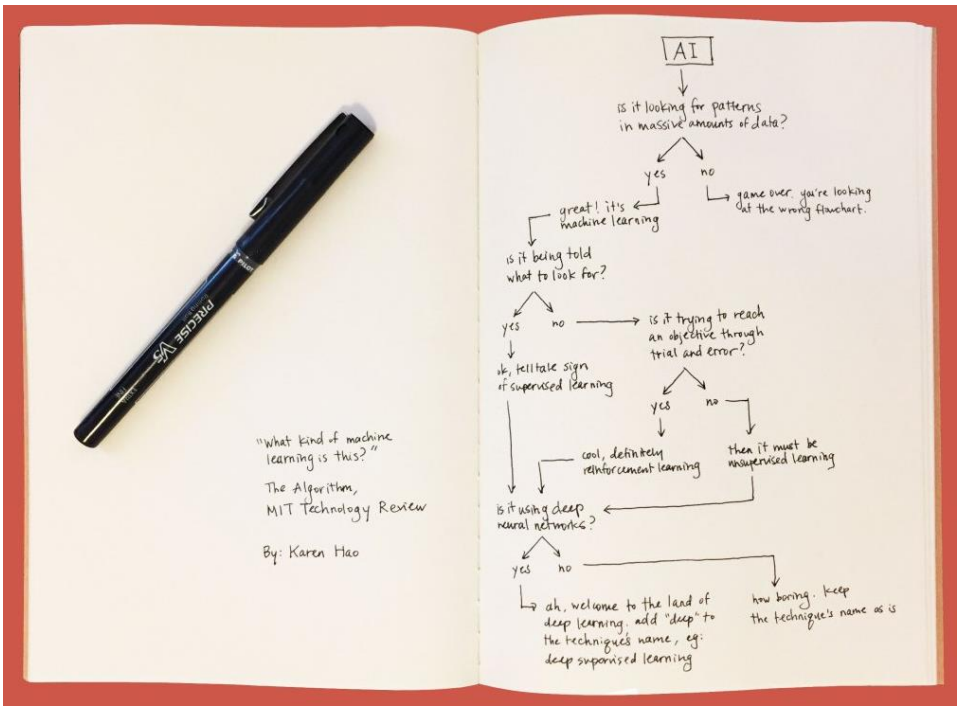
Artificial Intelligence (AI) was born in the Dartmouth Conference in 1956. A handful of computer scientists met and start to dream about the idea. They imagined a concept where we could use the growing power of computers to create a machine possessing the same characteristics as a human. The concept was created and reflected in cinema with C3PO and Terminator. AI is more a general description than a precise science, there is different fields in AI that we will see. This term is widely used to describe every system that can express any level of decision making or recognition or any form of autonomy from human direct control.



Machine Learning

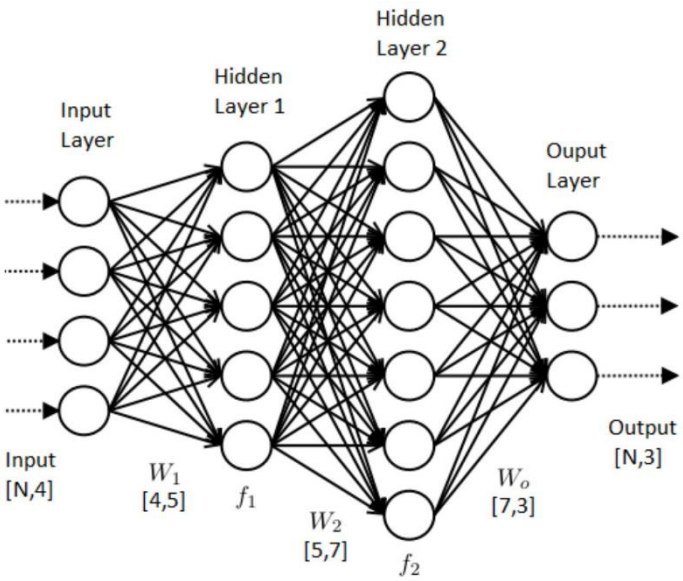
When we talk about Machine Learning, the definition is more practical and less philosophical. The basic concept is to use algorithms to parse data, learn from it and then make a determination or a prediction of a given element. This process involves decision tree and logic programming (AND OR NOR…) and some complex model such as Bayesian networks. Machine Learning is a style algorithm, looking for patterns into massive amount of data. There is a lot of different application in our daily life, in social media, online marketing or finance. The applications are endless as we can automate a lot of fastidious process. The is three main types of machine learning:

- Supervised learning: Here the algorithm is fed with be labelled data, and the outputs to create the model.
- Unsupervised learning: Here the algorithm is fed with unlabelled data and is not told the expected output. The model will be created by discovering what it's looking for on its own by clustering for example.
- Reinforced learning: Here the algorithm also is fed with unlabelled data, but it will be told when it's right or wrong in its decision we call that trial and errors. It will then create the model considering each errors and success.

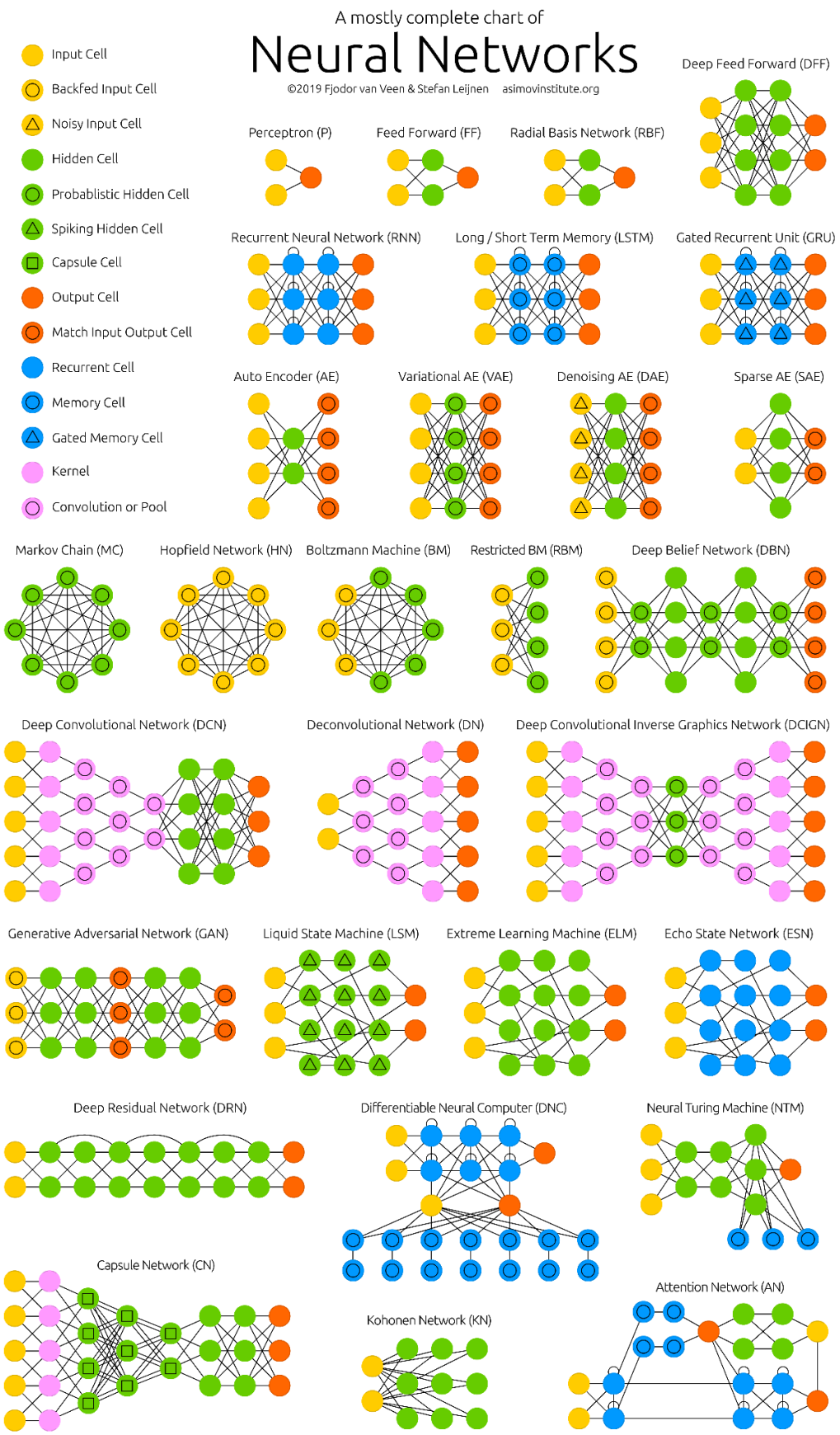


Deep Learning

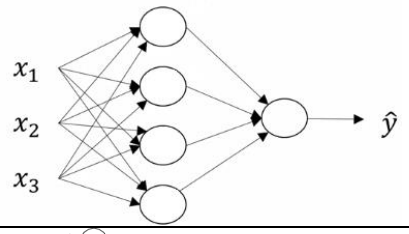
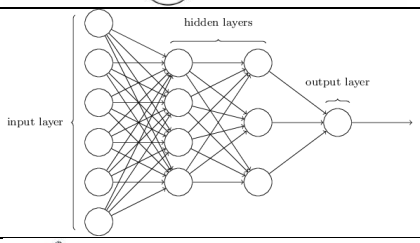
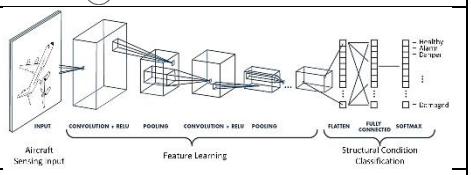
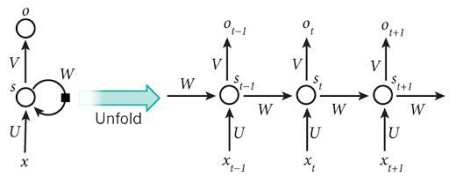
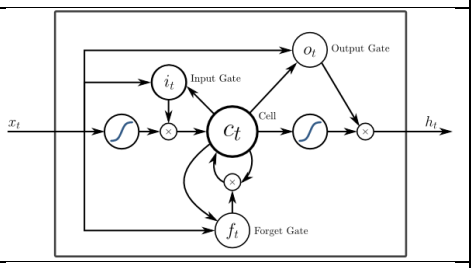
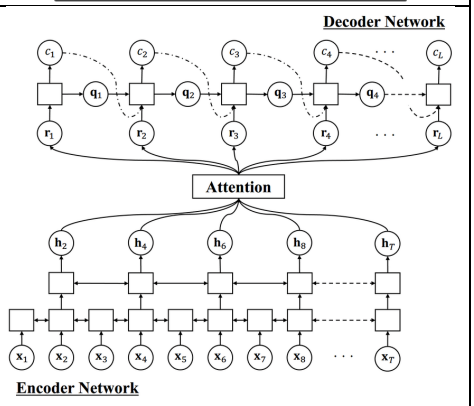
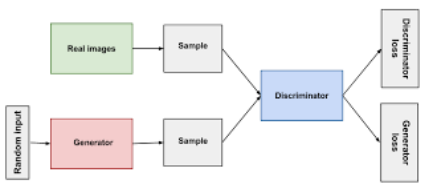
Finally, we talk about Deep Learning as the present and future of AI. To build an Intelligence, scientist took as a model the only thing possible of creating some, the human brain. They reproduce Neural Networks: this is a system composed of several layers, the first level is composed of the inputs. Nowadays with big data technologies we can feed algorithm huge amount of data to this layer. Then there is a given number of hidden layers, in each layer there are some neurons. These neurons are small unit of decision interconnected to each other. Each neuron attributes a weight to the decision and send it to the next layer. The last layer displays the output, which is the most probable decision, the one with the biggest weight. Deep learning as a huge potential as the application can be very broad the very efficient the downside is that this technology is complex as well expensive and time consuming.



There are hundreds of different neural networks, each created for a specific reason. We can see here an overview of many Neural Networks created by The Asimov institute.



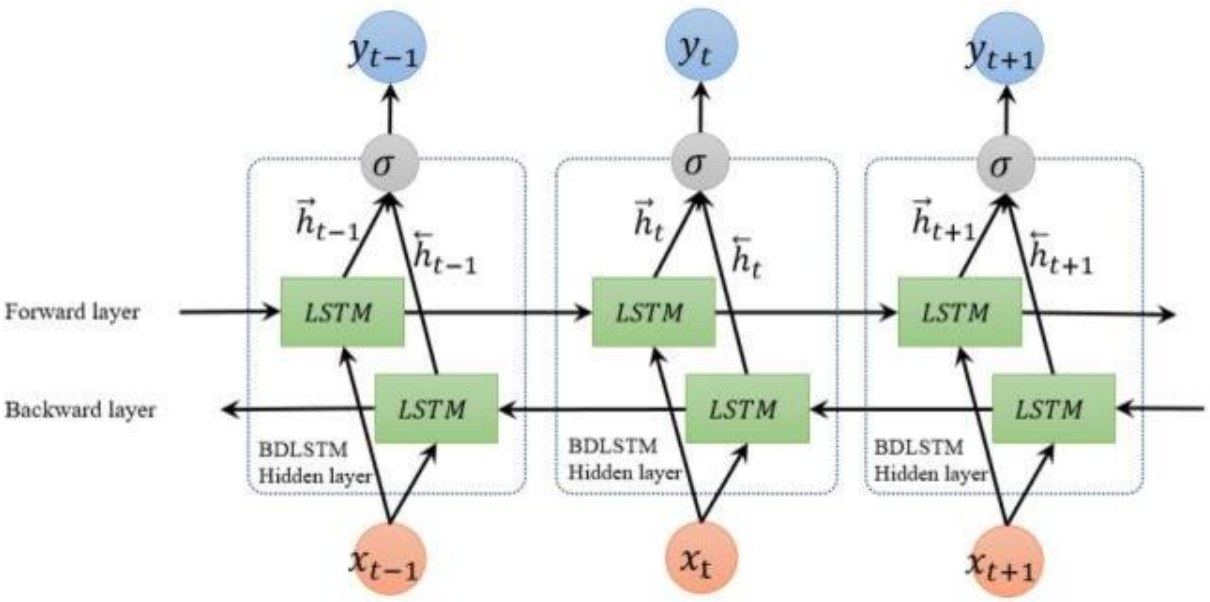
Instead of investigating each network one by one we will classify all the main families:

Name	Description	Application	Representation
Shallow Neural Networks (Collaborative Filtering)	Single hidden layer of the perceptron.	Recommendation algorithm: Netflix, YouTube, Amazon...	
Multilayer Perceptron (Deep Neural Networks)	Multiple hidden layers of perceptron. Base on Universal Approximation theorem and optimized with back propagation	Speech recognition Image recognition	
Convolutional Neural Network (CNN)	Convolution at each pixel to recognize a pattern	Google Translator Google Lens	
Recurrent Neural Network (RNN)	The output of some hidden layer is fed to the input of other layer to aggregate and move forward for the next epoch	Handwriting recognition Speech recognition	
Long Short-Term Memory (LSTM)	Form of RNN solving vanishing gradient problem. Prevent the loss function to go to 0 by preventing activation function and stored value unmutated	Google speech recognition and Translate	
Attention-based Networks	Focus on the important information to avoid treatment of the background coupled with RNN and CNN	Google and Facebook use this in their image recognition process	
Generative Adversarial Network (GAN)	Use reinforce learning to put some noise in the creation of the model to make a model more resistant to noise. Most advance form of AI	Conditional GAN (CGAN), Laplacian Pyramid GAN (LAPGAN), Super Resolution GAN (SRGAN)	

Issue in Data Classification

Data classification is the process of taking a large amount of input data and automatically store them into classes predefined (Supervised learning) or not (Clustering). One of the big issues with data classification is the length of the process. Most of the time a huge amount of data is necessary to obtain a good solution, therefore, it requires a lot of computation. To solve the problem there is two main options the first one is to use exceptional computing power to make it faster. This is often expensive; a little trick is to run the computing on the GPU (Graphics Processing Unit) instead of the classical CPU (Central Processing Unit) as the calculation process is more adapted on the GPU. Also, the GPU are nowadays extremely powerful thanks to the gaming industry.

The second option is making the calculation faster. For this a deep knowledge of the Neural Networks and their specific application is required for the optimal solution time. A second big issue is the noise in the data. To solve this better harvesting data methods are required, or some pre-processing can be run to clean the data.



The chosen models

The type of data for the project is sequences. The most efficient way to treat sequence data in 2020 is with a LSTM structure, we will use then create two very different models using the LSTM base. This does not mean the two model will be identical. They will be quite some differences. The first model will be a LSTM architecture where we will modify the internal structure by changing the size of the hidden layers. The second will be a Bidirectional LSTM (meaning propagation goes forward and backward).

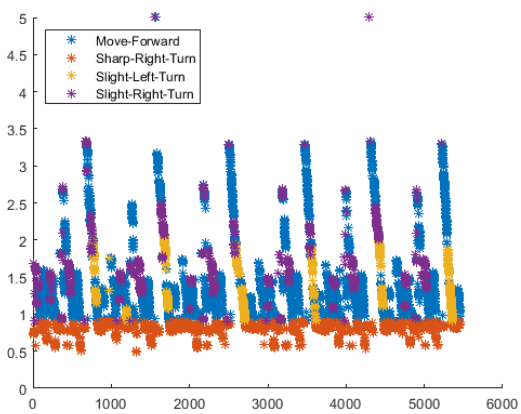
Classification of range data for the navigation of a robot

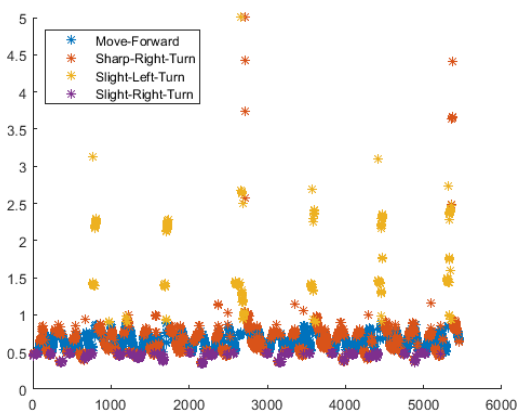
The data given in this project were sequences of doubles matching a label we can plot them using the following code:

```
% Visualize the given Data
plotSensors(sensor1,label);
plotSensors(sensor2,label);

% Concatenate and transpose data
bothSensor = [sensor1 sensor2]';
labelRow  = label';
summary(labelRow)

function plotSensors(X,Y)

    figure
    cat = categories(Y);

    for i = 1:length(cat)
        label = cat(i);
        idx = find(Y == label);
        hold on
        plot(idx,X(idx),'*')
    end

    hold off
    legend(cat,'Location','northwest')

end
```

The output of this code is:

Move-Forward	**Sharp-Right-Turn**	**Slight-Left-Turn**	**Slight-Right-Turn**
2205	2097	328	826





Considering the values, we can design the first model: The simple LSTM. The method basic method is quite trivial and was learnt in the Deep Learning with MATLAB tutorial. First, we separate the training and testing sets. I decided to add also a validation set this is not mandatory and has pros and con. The inconvenient is when we work in limited dataset, we have less data to be more accurate. The pro is that it allows us to know if we are overfitting the model or not. I considered this last point important, so I decided to use validation steps.

```
% Create training,testing and validation sets
n = 1000;
m = 500;

XTrain = bothSensor(:,1:end-(n+m));
YTrain = labelRow(1:end-(n+m));


XValidation = bothSensor(:,end-(n+m)+1:end-n);
YValidation = labelRow(end-(n+m)+1:end-n);


XTest = bothSensor(:,end-n+1:end);
YTest = labelRow(end-n+1:end);
```

The longest part is to find the good coefficients for each option of the model and each layer. Each training takes between 15 and 50 minutes so it's really time consuming to try a lot of model. Therefore, I created a little function testing automatically all the possible outputs and let my computer turn non-stop for 60h.

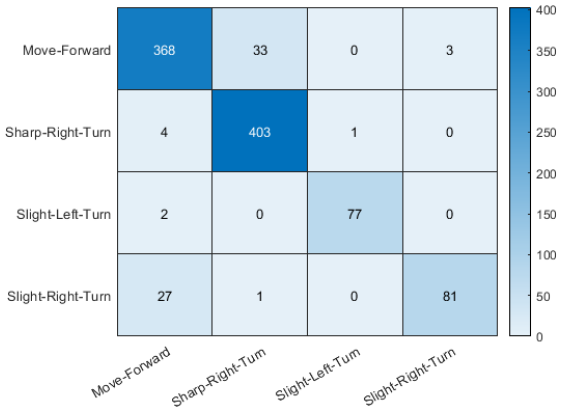
```
%Train the model and evaluate in a double for loop for different epoch and learning rate
% Set v in format:starting epoch: range of epoch: final epoch
% Set W in format:starting learning rate:range learning rate:final learning rate
    v = 1:499:500
    w = 0.001:0.001:0.001
    testAllModels(v,w,XTrain,YTrain,XTest,YTest,bilstmLayer)
```

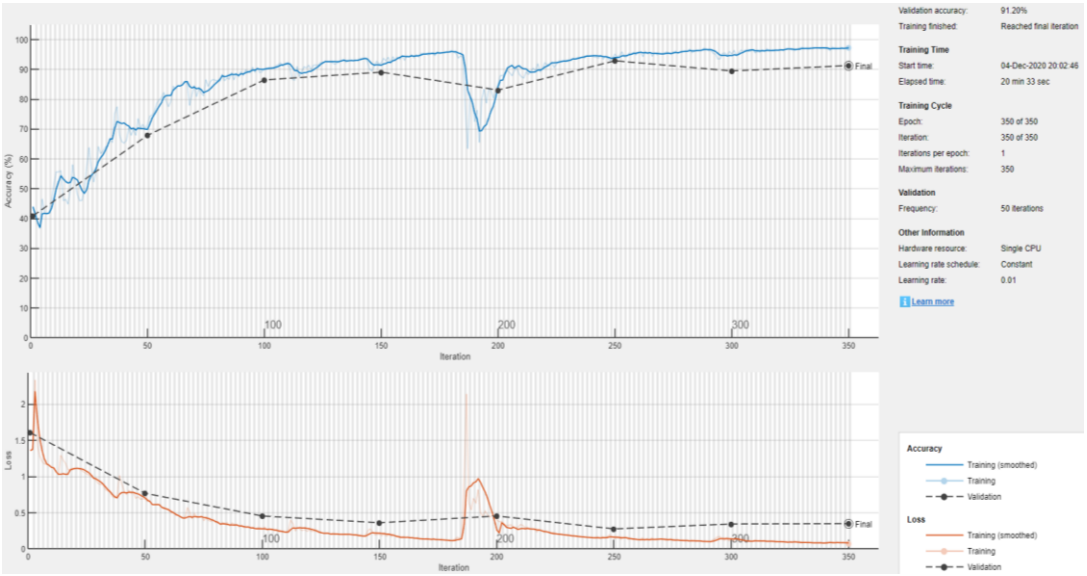
```
function testAllModels(v,w,XTrain,YTrain,XTest,YTest,bilstmLayer)
fileID = fopen('Result','w');

    for epoch = v
        for rate = w

            %Set the options of the model
            options = trainingOptions('adam','Plots',...
                'training-progress',"MaxEpochs",epoch,"InitialLearnRate",rate);

            %Train the model
            net = trainNetwork(XTrain,YTrain,bilstmLayer,options);

            %Evaluate Model
            prediction = classify(net,XTest);
            accuracy = nnz(prediction == YTest)/numel(YTest);

            %Create a txt document with the accuracy
            accDisp = ['Accuracy_' num2str(epoch,'%.2f')...
                '_' num2str(rate,'%.3f') '.doc'];
            fprintf(fileID, accDisp);
            fprintf(fileID,'%.2f\n',accuracy);
```

I concluded that the best coefficient for my first model were:

```
% Create the layers of the Bidirectional LSTM model
numFeatures = 2;
numHiddenUnits1 = 125;
numHiddenUnits2 = 100;
numClasses = 4;
lstmLayer = [
    sequenceInputLayer(numFeatures)
    lstmLayer(numHiddenUnits1,'OutputMode','sequence')
    dropoutLayer(0.2)
    lstmLayer(numHiddenUnits2,'OutputMode','sequence')
    dropoutLayer(0.2)
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];
%Set the options of the model
options = trainingOptions(...
    'adam', ...
    'MaxEpochs',350, ...
    'InitialLearnRate',0.01, ...
    'Plots','training-progress',...
    'ValidationData',{XValidation,YValidation}, ...
    'ValidationFrequency',50);
```

It gave and excellent accuracy of 92.9% a well as an excellent validation accuracy of 91.2% above this the validation accuracy would mean overfitting but in this scenario it's perfect. I'm very please with this result!

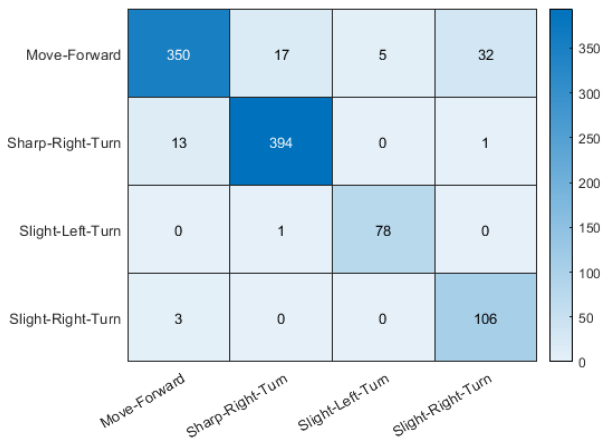


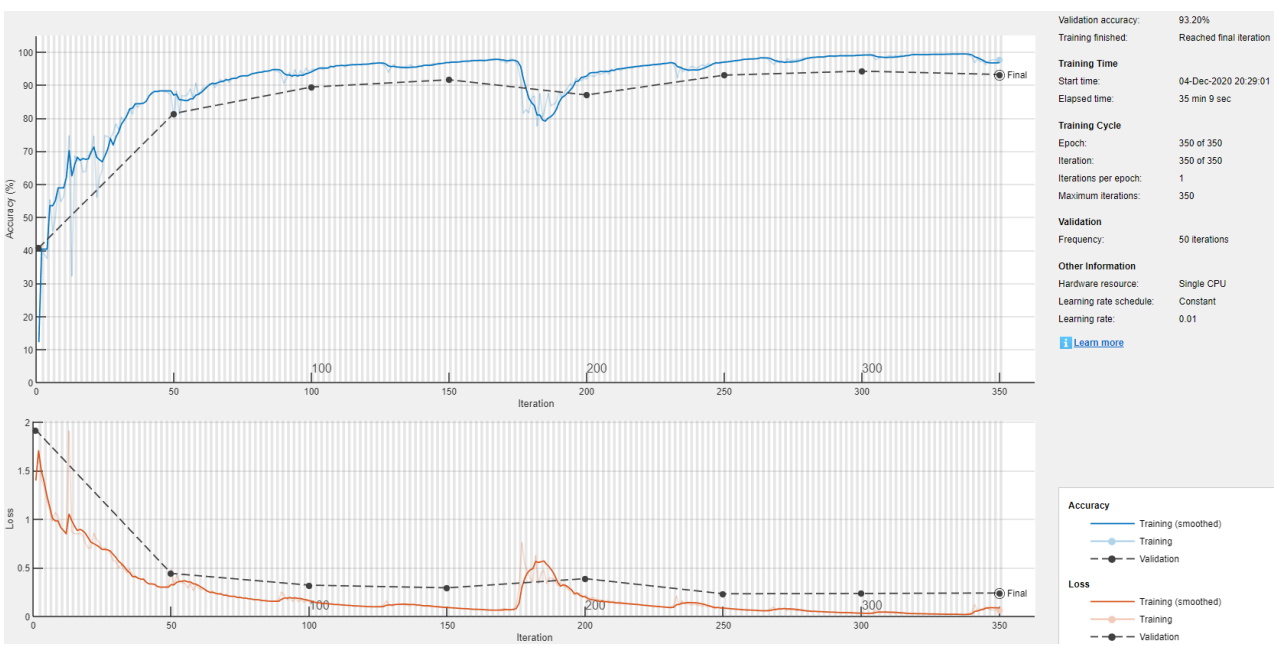


For the second model the BiLSTM is the same process but going both was so the good parameters are often the same and the process of creation of the code is basically the same. The really big difference comes in the creation of the architecture.

```
% Create the layers of the Bidirectional LSTM model
bilstmLayer = [
    sequenceInputLayer(2)
    bilstmLayer(100,'Outputmode','sequence')
    bilstmLayer(100,'Outputmode','sequence')
    fullyConnectedLayer(4)
    softmaxLayer()
    classificationLayer()];
%Set the options of the model
options = trainingOptions('adam',...
    'Plots','training-progress',...
    "MaxEpochs",350,...
    'ValidationData',{XValidation,YValidation}, ...
    'Plots','training-progress', ...
    "InitialLearnRate",0.01);
```

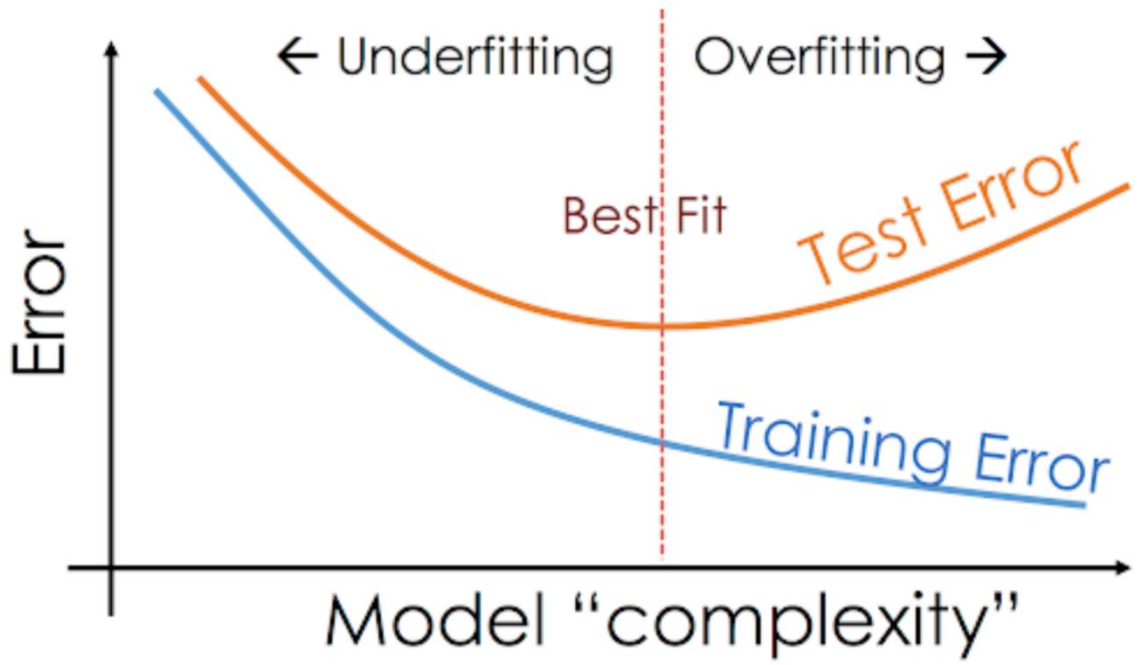
Here we obtain an almost as good overall accuracy of 92.8% and a validation accuracy of 93.2% this mean we have a slight overfitting. Which isn't the best but this is not significant enough (0.4%) to be really impactful on the solution.





Conclusion

We explored in this project the basis of artificial intelligence and its key principles. We also classified the main families of neural network while citing some the most famous ones. We also analyse what are the perfect types of model for a given problem with fixed data set. We finally explored two interesting solution of neural network to classify sequential input. The first model was a LSTM model with some customized hidden layer and the result we found are as good as it can get in the artificial intelligence domain. The second was a Bidirectional LSTM the result were not as good but still excellent. This second model allowed us to expose an important principle in neural network: overfitting. Here we see each value too many time (twice more as it goes both way) this as an impact on the time (almost twice longer) meaning the model will be too adapted to this set of variables and will be less good on every sets of data.



On a personal tone I found this project interesting as it allows us to discover in depth the basis of all neural network. It also allowed us to have some practice in tunning our own model of artificial intelligence and to explore each options and types of layers for a given dataset. I really liked this project as it gives me a little approach of how deep I can go if I want to realise my master thesis in artificial intelligence. The certifications were also a interesting part of the module that will be useful in my professional career and I'm happy to have them now.

Bibliography

Basic understanding of Artificial Intelligence and Machine Learning:

Educational Approach:

- Lectures from C GOONAWARDANE given at Staffordshire University first semester of academic year 2020-2021 in the module Artificial Intelligence and Machine Learning.
- What is machine learning? From the MIT Technology review, Karen Haoarchive, 17th November 2018:

https://www.technologyreview.com/2018/11/17/103781/what-is-machine-learning-we-drew-you-another-flowchart/

Company Approach:

- IBM Cloud Learn Hub > Machine Learning, By: IBM Cloud Education, 15th July 2020

https://www.ibm.com/cloud/learn/machine-learning

- NVIDIA courses > Getting Started with AI, By: The NVIDIA Deep Learning Institute, 2020

https://courses.nvidia.com

Business Approach:

- Investopedia, Investing > Financial Technology & Automated Investing > Machine Learning, By JAKE FRANKENFIELD, 31st Aug 2020:

https://www.investopedia.com/terms/m/machine-learning.asp

Classification of different Neural Network:

Educational Approach:

- Towards AI, Main Types of Neural Networks and its Applications — Tutorial, By: Pratik Shukla, Roberto Iriondo, 11th August 2020:

https://medium.com/towards-artificial-intelligence/main-types-of-neural-networks-and-its-applications-tutorial-734480d7ec8e

- EDUCBA, Introduction to Classification of Neural Network, 2020:

https://www.educba.com/classification-of-neural-network/

Company Approach:

- The Asimov Institute, The Neural Network Zoo By Stefan Leijnen and Fjodor van Veen, published the 12th May 2020

file:///E:/Cours%20STAFFS%204/AI%20&%20MACHINE%20LEARNING/Assigment/Assignment%201/The_Neural_Network_Zoo.pdf

Annex

MathWorks | Training Services

Course Completion Certificate

Timothee ANTHOINE-MILHOMME

has successfully completed 100% of the self-paced training course

Machine Learning Onramp

DIRECTOR, TRAINING SERVICES

16 October 2020

MathWorks | Training Services

Course Completion Certificate

Timothee ANTHOINE-MILHOMME

has successfully completed 100% of the self-paced training course

Deep Learning Onramp

DIRECTOR, TRAINING SERVICES

20 November 2020

MathWorks | Training Services

Course Completion Certificate

Timothee ANTHOINE-MILHOMME

has successfully completed 100% of the self-paced training course

Image Processing Onramp

DIRECTOR, TRAINING SERVICES

06 November 2020

MathWorks | Training Services

Course Completion Certificate

Timothee ANTHOINE-MILHOMME

has successfully completed 100% of the self-paced training course

Deep Learning with MATLAB

DIRECTOR, TRAINING SERVICES

16 November 2020

MathWorks | *Training Services*

Course Completion Certificate

Timothee ANTHOINE-MILHOMME

has successfully completed 100% of the self-paced training course

Machine Learning with MATLAB

DIRECTOR, TRAINING SERVICES

03 November 2020