Chapter 1

A JOURNEY FROM MACHINE LEARNING TO DEEP LEARNING

CONTENTS

"We know the past but cannot control it. We control the future but cannot know it." by Claude Shannon [1]

1.1 A brief history on Artificial Intelligence

Artificial Intelligence was kicked off in the 1950's, and its development can be divided into three phrases. In the first stage, it mainly focused on the logical mechanism via fuzzy logic. In light of the vast amount of data available from the internet and contemporary clinical studies, AI has moved to the stream of statistical and machine learning in the previous two decades. Getting into the third stage, in just the past few years, we get into the paradigm of deep learning.

The main challenge faced by AI study is to teach a computer how to resolve an apparently trivial task to humans but cannot be tackled by simply using preassigned algorithm or routine logical instructions; for

[1] Shannon, C. E. (1959). Coding theorems for a discrete source with a fidelity criterion. IRE Nat. Conv. Rec, 4(142-163), 1.

instance, distinguishing a cat from a dog is obvious to a human, but writing an algorithm to do this task, so that all aspects are taken into account at once, would be very complicated. Particularly, the most recent proposal on the use of machine learning is to handle problems similar to this; indeed, with the availability of hand-held electronic devices, such as smart phones and smart watches, collecting huge amounts of data on human behavior is far easier nowadays, and this can help to train the machines to learn to mimic us on how to solve different matter. In the primitive models in statistical learning, most of them are only composed with a few layers of complexity, and therefore they lack the ability to pick up the more subtle latent information embedded deeply in the ocean of data. Facing at this bottleneck, to overcome this, with the latest advance in computational power and the availability of labeled data, scholars turn to strengthening the network approach which leads us to the most recently popular topic - Deep Learning.

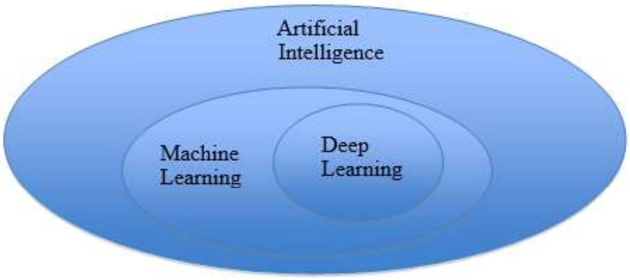

Figure 1.1.1: Relations among AI, Machine Learning and Deep Learning

First Wave (1950-1975): Mechanical logical reasoning

In 1950, Alan Turing proposed the influential yet controversial Turing test in his paper *Computing Machinery and Intelligence* [1]. In the test, one of the two humans serves as an examiner and communicates with the second human and a computer through text messages, where the last two are kept away out of the sight of the examiner. The computer is considered to possess artificial intelligence if the examiner is unable to distinguish the responses between the human and the computer.

In 1951, Marvin Minsky built the first Stochastic Neural Analog Reinforcement Calculator (SNARC). The machine, essentially a neural network consisting of 40 neurons, enables human to first simulate the transmission of neural signals. To honor his contribution, Minsky received the Turing Award, the most prestigious prize in computer science, in 1969.

In 1955, Allen Newell, Herbert Simon, and Cliff Shaw [2] wrote a compute program called the **Logic Theorist** to mimic the problem-solving skills of humans. This program successfully proved 38 out of 52 theorems from *Principia Mathematica* by Whitehead and Russel (1910).

For the formal origin of AI, the first workshop on Artificial Intelligence held in Dartmouth in the summer of 1956 is commonly regarded as the date of birth of AI, and it was attended by the representative scholars in information science and intelligence such as John McCarthy, Marvin Minsky, and Claude Shannon. The workshop covered topics including neural networks, natural language processing, abstraction, and creativity. After this series of talks, scientists and engineers have been constantly dreaming of a hypothetical machine that can exhibit behavior at least as skillful and flexible as humans do, can reason, and can possess the human soul and mind; researchers often refer to this collective wisdom and research program as General (Strong) Artificial Intelligence.

Amazed by its unlimited potentials, AI had started to flourish. During this time, some contemporaries optimistically foresaw that a machine completely driven by AI would come to birth in 20 years time. In 1963, the MIT initialized the Project on Mathematics and Computation (Project MAC), with Minsky and McCarthy joining at a later time, in which they promoted a series of research topics on image and speech recognitions. From 1964 to 1966, Joseph Weizenbaum built the world's first natural language processing computer program; meanwhile, on the other side of the globe, Waseda University in Japan announced the invention of the first biped walking robot.

However, the hunger of scientists had yet to be satisfied. Criticism on AI began to rise starting in the 1970's; indeed, the rapidly growing demand on computational power could not be fulfilled at that time. In addition, the variety and complexity of demanding problems in image and natural language processings had created severe hurdles given the contemporary technological conditions. Reaching this bottleneck, the public awareness and grant funding started to rapidly decline in the mid of 1970's, AI development then fell into decay.

Second Wave (1980-1987): Rise and fall of the expert system

Stepping into the 80's, the breakthroughs in expert systems and artificial neural networks drew the public's attention back to AI. Expert systems can be dated back to the 60's, being introduced in a project led by Edward Feigenbaum [4], who was advocated as the "father of expert systems". An expert system is a computer program that simulates the judgment and behavior of a human with expertise collected before in a particular field under a set of prescribed rules. In the 70's, researchers at Stanford University invented a system called MYCIN, which diagnosed a person's blood, to identify bacteria causing infection such as bacteremia and meningitis so as to recommend the appropriate dose of antibiotics, based on around 600 manually assigned rules. In the 80's, the Carnegie Mellon University invented an expert system called XCON (eXpert CONfiger) [5] for the Digital Equipment Corporation, which could automatically select the combinations of computer components on behalf of a customer's needs; that XCON had helped the corporation save over 40 million US dollars annually at that time.

With the success of expert systems, the purpose of developing AI started to deviate from its original goal of obtaining general intelligence, instead, the interest now is to develop more tailor made system to solve target practical problems in specific areas. In 1982, John Hopfield proposed a new network model which was later called the Hopfield network, a kind of recurrent artificial neural network [6], which incorporates the mechanism of associative memory (the ability to learn and remember the relationship between unrelated items). In 1986, David Rumelhart, Geoffrey Hinton and Ronald Williams jointly published the paper *Learning representation by back-propagating errors* [7] in which they proved empirically the method of backward propagation can help train a multi-layer neural network such that it can learn the appropriate inherent representations of an arbitrary mapping of input to output.

During this new wave of passion for AI, Japan's Ministry of International Trade and Industry initialized a project of building a "fifth-generation computer" in 1982 [8]. It aimed to create a machine with supercomputer-like performance through large scale simultaneous sychronized calculations, in order to provide a platform for future developments in AI. However, after spending over 50 billion Japanese yen in 10 years time, the project could still not meet the planned target. In the late 80's, negative impressions on AI started to grow in the industry, as it failed to meet the expectations of the tremendous investments that had been made, AI had once again faded out of people's mind.

Third Wave (2011-now): Deep learning

After the previous two waves, researchers had given up their idealistic thoughts and AI had emerged a solution to solving practical concrete problems rather than a general approach and on a case-by-case basis. The extensive use of mathematics has opened up the interdisciplinary collaboration between AI researchers and scholars from other disciplines; and new sophisticated models and more effective algorithms are subsequently developed, for instance, statistical learning theory, the support vector machine and probabilistic graphical model, and now more prevalent deep neural networks, to name a few.

Stepping into the 21$^{\text{st}}$ century, the rapid globalization and development of the internet have significantly boosted the volume of available digital information. On the other hand, the computing capability of Graphics Processing Unit (GPU), first appearing in the 1990's and then gaining popularity over the next two decades, has been proliferating; for instance, the calculation speed of a NVIDIA[2][3] Tesla V100 GPU[4] has exceeded 10 trillion FLOPs (floating-point operations per second), surpassing the world's fastest supercomputer in 2001.

With the rapid development of effective big data collection and computing technology, AI has achieved major breakthroughs. The multi-layer neural network AlexNet[5] invented by researchers at the University of Toronto won the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC). AlexNet outdid, to a large extent, the first runner-up in the challenge, where its algorithm was based on convolutional neural networks in machine learning. Henceforth, deep learning based on multi-layer neural networks has been applied to various areas. For instance, with an advance in deep reinforcement learning, AlphaGo recently developed by Google has defeated several Go world champions [10]. All of these have captured public at-

[2] Originally, the founders first thought of "NV" standing for "Next Version", then they added "invidia" referring to the Latin word envy.

[3] We here again would like to express our gratitude to NVIDIA for supporting the joint-institute with CUHK.

[4] The product NVIDIA Tesla V100 GPU can be found in `https://www.nvidia.com/en-gb/data-center/tesla-v100/`.

[5] AlexNet is the name of a convolutional neural network (CNN), designed by Alex Krizhevsky in collaboration with Ilya Sutskever and Geoffrey Hinton, who was Krizhevsky's Ph.D. advisor.

tention on the potentials of deep learning and brought back the revenge of AI.

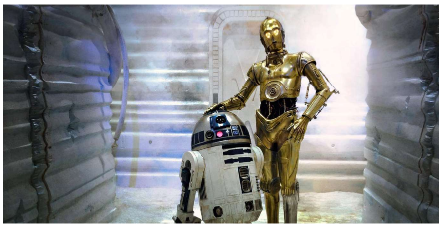

Figure 1.1.2: The robot “R2-D2” in the *Star Wars* series.

Figure 1.1.3: T-800 “Model 101” in the movie *The Terminator*.

Figure 1.1.4: Which of these robots would you prefer to see?

1.2 Machine Learning - Learn from data

Machine learning, also known as statistical machine learning, is devoted to building statistical models, based on data, for analysis and making predictions. Instead of executing manually assigned commands, machine learning solves problems by utilizing the inherent insight and structure within the input data. The main purpose of machine learning is to generalize, that is, to learn the rules from any hidden patterns embadded in the collected data, and then apply this recently acquired “laws” to new scenarios for making decisions or predictions.

Machine learning originated from the early stages of artificial intelligence, and it evolved gradually and brought in new inspirations into different sub-branches of pattern recognition and computer learning theories. It is an interdisciplinary subject that involves statistics, linear algebra, optimization and numerical analysis. According to variations in purposes and methodologies, machine learning is classified into supervised learning, unsupervised learning and reinforcement learning. Assume that

1. there are N samples in dataset, and
2. $\boldsymbol{x}_m$ represents the out-sample feature vector, *i.e.* not in the training dataset, for $m > N$.

1.2.1 Supervised Learning

In supervised learning, models are trained using labeled data. Each datapoint in the dataset consists of a feature vector (input) and their respective labels (output). Common learning scheme is called classification if the output variable is discrete-valued, and regression if it is continuous.

We start the learning procedure by choosing a suitable model. Common supervised learning models include logistic regression, generalized linear models, classification and regression trees, support vector machine

(SVM), K-nearest neighbors (KNN), naive Bayes classifiers, and many common Deep Neural Networks. The model is tested by comparing the predicted values against the actual labels, so that the model can be adjusted accordingly. The training process is repeated until sufficient accuracy is obtained. The learning is supervised by the feedback obtained from the values of the actual labels; Once the training is finished, new data can be input into the model for predictions.

1. **Dataset**: A collection of labeled examples $\{\boldsymbol{x}_n, y_n\}_{n=1}^N$, where
   (a) $\boldsymbol{x}_n$ (Input): A D-dimensional feature vector, *i.e.*
   $$\boldsymbol{x}_n = (x_n^{(1)}, x_n^{(2)}, \cdots, x_n^{(D)}).$$
   (b) y_n (Label): Anything, *e.g.* an element belonging to a finite set of classes $\{1, \cdots, C\}$, a real number, a vector, a matrix, a tree, or a graph.
2. **Goal**: Produce a model that allows "correctly" guessing the label y_m from the new feature vector $\boldsymbol{x}_m$.

Example 1.2.1. Spam Detection: Suppose that we have 10,000 email messages, each label with "spam" or "not_spam". However, these email messages cannot be directly used in the model, these labels and passages in the emails are not numbers! Hence, each email message has to be converted into a feature vector. One common way is called **bag of words**: Let say the bag (dictionary) contains 20,000 alphabetically sorted words, then

1. the first feature has a value of 1 if the email message contains the word "a"; 0 otherwise;
2. the second feature has a value of 1 if the email message contains the word "aaron"; 0 otherwise;
3. $\vdots$
4. the $20,000^{th}$ feature has a value of 1 if the email message contains the word "zulu"; 0 otherwise.

$$x_n^{(1)} = \begin{cases} 1, & \text{if the } n^{th} \text{ message contains "a"} \\ 0, & \text{otherwise} \end{cases} \quad \cdots \quad x_n^{(20{,}000)} = \begin{cases} 1, & \text{if the } n^{th} \text{ message contains "zulu"} \\ 0, & \text{otherwise} \end{cases}.$$

Similarly, the output labels have to be converted into numbers. For example,

$$y_n = \mathbb{1}\{\text{the } n^{th} \text{ message is spam}\} = \begin{cases} 1, & \text{if the } n^{th} \text{ message is spam} \\ 0, & \text{otherwise} \end{cases}.$$

where $\mathbb{1}\{\cdot\}$ is the indicator function. This example will be further discussed in support vector machine, random forest, naive bayes classifier, and CIBer.

1.2.2 Unsupervised Learning

In practice, labeled data are not always available, this leads us to the setting of unsupervised learning. Only available information is the feature vectors of datapoints, these are analyzed in order to find out the hidden patterns (inner structures) or clusters (organizations) within the data source. Typical approaches of unsupervised learning include principal component analysis, recommended systems, K-means clustering, dimension reduction, and feature extraction, etc.

The performance of traditional unsupervised learning in feature extraction for any complex data structure may not be too appealing; alternatively, deep learning has proven its strong unsupervised learning abilities, especially in the field of computer vision, or when there are some natural ordering metric, or algebraic structures among feature variables of datapoints; particularly, it is through the Convolutional layers in Convolutional Neural Network (CNN) and the feedback mechansim via backpropagation in the Deep NN (latter) part of CNN. Recently, some reasearch also suggests semi-supervised learning, falling in between supervised and unsupervised learning. It makes use of unlabeled data together with a small amount of labeled data, striking a balance between the learning performance and the costs in obtaining labeled data.

1. **Dataset**: A collection of unlabeled examples $\{\boldsymbol{x}_n\}_{n=1}^N$.
2. **Goal**: Produce a model that transforms the feature vector $\boldsymbol{x}_n$ into the real-valued output y_n or a vector output $\boldsymbol{y}_n$. For example, in the following cases, the model returns:
   (a) **Clustering**: The identity of the cluster for each group of feature vectors in the dataset, *i.e.*
   $$y_n \in \{1, \cdots, C\},$$
   where C is the total number of clusters. K-means clustering is one of such method in clustering K different subgroups, where K is a hyperparameter; see Section 9.1.
   (b) **Dimension Reduction**: A new feature matrix $\boldsymbol{Y} \in \mathbb{R}^{N_Y \times D_Y}$ that has a smaller dimension than the input feature matrix $\boldsymbol{X} \in \mathbb{R}^{N \times D}$. Principal component analysis (PCA) reduces the dimension of the input feature matrix through looking for the dominant eigenvalues of covariance matrix of feature vectors; see Section 3.6.
   (c) **Outlier Detection**: A real-valued number y_ℓ that indicates how $\boldsymbol{x}_\ell$ is different from a "typical" examples in the dataset $\{\boldsymbol{x}_n\}_{n=1}^N$. The Mahalanobis distance [9] D_n for the independent and identically distributed (iid) datum $\boldsymbol{x}_n$ is such that
   $$D_n^2 = (\boldsymbol{x}_n - \bar{\boldsymbol{x}}_N)^\top \boldsymbol{S}^{-1}(\boldsymbol{x}_n - \bar{\boldsymbol{x}}_N) \sim \chi_D^2, \qquad n = 1, 2, \cdots, N$$
   approximately, where $\bar{\boldsymbol{x}}_N$ and $\boldsymbol{S}$ are respectively the mean and the covariance matrix of $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N$, and χ_D^2 is the chi-squared distribution with D degrees of freedom; especially if N is large enough, these D_n^2's, for $n = 1, \cdots, N$, are also approximately independent of each other.

1.2.3 Reinforcement Learning

In reinforcement learning, an agent provides guidance on how to act in response to the current, mostly abstracted, environment in order to maximize the expected benefit (pay-off or bonus obtained). This concept is originated from the behavioral science in psychology, *i.e.* the study on how living organisms after a great number of attempts can gradually form expectations on stimuli under a series of rewards and penalties in the course of their interactions with the (usually uncertain) environment, and by then they exhibit habitual behaviours that can lead to a maximum return. Reinforcement learning solves a particular kind of problem where decision making is sequential, and the goal is long-term, such as gaming, robotics, resource management, or logistics.

Basic reinforcement learning includes the following components:

1. **Environment**: The world in which the agent organsim operates, quite often, it is a simulated version of the real physical world with abstracted features being modelled.

2. **State**: Current situation of the agent, *e.g.*, the distribution of black and white stones in a *Go* board, the stock prices dynamics in a stock market.
3. **Action**: A set of possible moves the agent can take or react.
4. **Reward**: Feedback from the environment, which measures the success (bonus) or failure (penalty) of the agent's action taken. Penalty can be seen as a negative reward, while bonus takes a positive value.

Unlike supervised or unsupervised learnings, reinforcement learning requires neither correctly labeled input/output pairs in advance nor explicit future corrections of sub-optimal actions taken now; to the latter point, the agent only knows all the immediate, but not long-term implication on the overall ultimate pay-off, rewards by taking each of the possible actions at the current situation; Hence, the agent cannot foresee which action would yield the best reward in the long-term; for instance, in the wizard chess game of the *"Harry Potter" - Sorcerer's Stone*, Ron Weasley sacrificed himself in that current situation, but it opened an opportunity in later stage for Harry Potter to checkmate the White King. Reinforcement Learning emphasizes on instant (feedback) planning, and balances between the exploration of the unknown future and the use of the up-to-date skill and knowledge. The learning process is a continuous accumulation of strategic experience learnt from interacting with the changing environment. AlphaGo, which defeated the world Go champion Lee Sedol in 2016, was trained by (deep) reinforcement learning; see Silver et al. (2016).

Figure 1.2.1: Ron Weasley (black knight) sacrificed himself to the White Queen scene

In simple words, in reality there is no answer key or preassigned agenda for the machine to learn. The agent "lives" in an environment and is capable of perceiving the state of that environment as a feature vector. The agent can execute an action in choosing each state. Different actions bring in different rewards and could also advance the machine to another state of the environment. Figure 1.2.2 shows a simple flow chart for reinforcement learning:

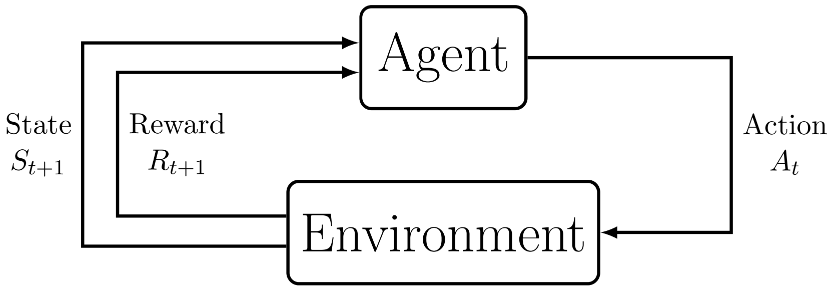

Figure 1.2.2: Progressing Reinforcement Learning Flow Chart in time t.

1. **Dataset**: Feature vector is the environment which returns the next state after an action is taken, but

the label, whether is good or bad in logn-run, is **Hidden**.

2. **Goal**: Learn a strategic policy such that the reinforcement agent decides a suitable action to maximize future expected average rewards at the current state.

As a remark, reinforcement learning (RL) is similar to the **hidden markov model** (HMM)[11], the label y_n is **Hidden**, but we can observe the feature vector $\boldsymbol{x}_n$ which relates to the label y_n. The difference between RL and HMM is that reinforcement learning does not assume the underlying probability distribution (known a *priori*) or knowledge of exact mathematical model; whereas HMM is a statistical Markovian model, the hidden states generated by the label y_n is modelled by a Markov process.

1.3 Machine Learning Algorithms

1.3.1 Parameters vs. Hyperparameters

1. **Parameters**: Variables that define the model, which aims to learn their true values by some learning algorithms. Parameters are directly estimated by the learning algorithm based on the training dataset. The goal of machine learning is to find such values of parameters, based on some dataset, that optimize the model in a certain sense.

2. **Hyperparameters**: Variables that control the learning process, *e.g.* the speed of convergence to the optimal solution and the accuracy of the estimates. Hyperparameters are not learnt by the algorithm itself from the training dataset, they have to be set a priori by the data analyst before running the algorithm; see Section 4.6.

1.3.2 Classification vs. Regression

1. **Classification**: Assigning a binary or multiclass predicted label y_n to an unlabeled observation $\boldsymbol{x}_n$.

2. **Regression**: Predicting a real-valued label y_n for an unlabeled observation $\boldsymbol{x}_n$.

1.3.3 Model-Based vs. Instance-Based Learnings

1. **Model-Based Learning**: Use the training dataset to create a model that has parameters learnt from the dataset. Most supervised learning algorithms are model-based, *e.g.* SVM, logistic regression, random forest, DNNs. Once the model is built (*i.e.* the optimal parameters are found), the training dataset can be discarded.

2. **Instance-Based Learning**: Use the dataset growing in time as the model. This learning approach is similar to the *online learning*, the model is used to predict any new instances, and then the model is updated, but in online learning, the dataset is in sequential order, while instance-based learning has no such restriction. One example of instance-based learning algorithm is the **K-Nearest Neighbors** (KNN), it looks at the K closest neighborhoods of an input in the dataset and make a label prediction through majority vote among these K neighbours.

1.3.4 Shallow vs. Deep Learnings

1. **Shallow Learning**: Learns the parameters directly from the feature vector of the training dataset. Most supervised learning algorithms are shallow: Artificial Neural Netwrok (ANN)
2. **Deep Learning**: Learns the parameters directly from the outputs of the preceding layers, usually with more than one hidden layers in between the input and the output layers: Deep Neural Network (DNN). Those preceding layers are normally used as filtering out the hidden patterns or inner organisations from the data points.

1.4 How to use this book?

The main purpose of this book is to deliver the fundamental, mathematical, and statistical principles to common machine and deep learnings learners, with the aid of some examples and problems arisen in business sector through **R** and Python. Though codings involved may not be the most effective in reaching the optimal buyout, but it surely can motivate and inspire readers in constructing an efficient enough model. Unfortunately, in this book, you may not found the materials about deep thinking, in the sense of the book (see Figure 1.4.1) by professor Kawakami of design studies.

Figure 1.4.1: Kyoto University's Deep Thinking Method (Japanese) by Hiroshi Kawakami.

"Deep Thinking" by Kawakami is a book on thought that deepens one's thinking ability and cultivate his/her skill of analyzing problems and then to propose solution strategies; it is more about scientific methods and philosophical argument training, which will not be covered in the present book. Instead, we introduce various practically useful mathematical and statistical models behind a wide range of machine learners. In 1997, Garry Kimovich Kasparov, who was once a world champion chess grandmaster, lost a match to the IBM supercomputer "Deep Blue" under a limited time constraint. After 20 years, in 2017, he published a book and named it as "Deep Thinking: Where Machine Intelligence Ends and Human Creativity Begins" (see Figure 1.4.2).

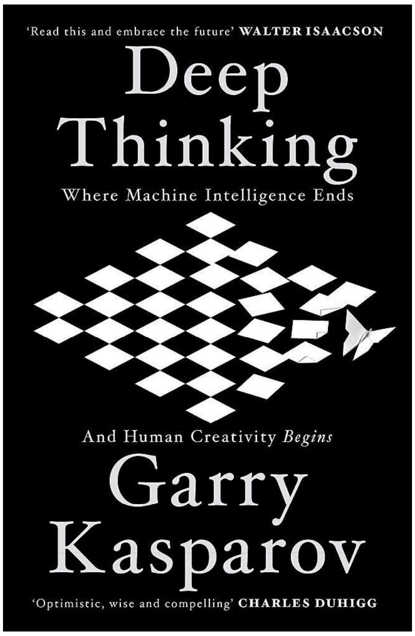

Figure 1.4.2: Deep Thinking: Where Machine Intelligence Ends and Human Creativity Begins by Garry Kasparov.

In his book, Kasparov revealed experience and strategies playing against the Deep Blue. Although there were plenty of critisms against artificial intelligence during that time, Kasparov believed that artificial intelligence could bring humans to another height, and predicted the future development of artificial intelligence. Similar to his idea, we hope that our book can bridge our readers to understand the common existing machine and deep learners, and to foster the future development of artificial intelligence.

In addition, we may not introduce any materials related to deep diving (see Figure 1.4.3), but only motivating more about self-driving/autonomous driving ("deep" driving; see Figure 1.4.4) in the due course.

Figure 1.4.3: Deep diving. Photo by http://divemagazine.co.uk/travel/7529-fresh-wrecks.

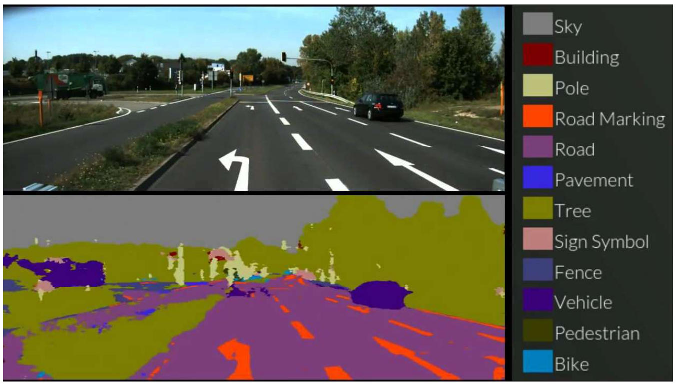

Figure 1.4.4: Deep driving. Photo by Alex Kendall https://www.youtube.com/watch?v=CxanE_W46ts.

In particular, Tesla Inc., a U.S. based company which builds electric car, uses deep learning to develop an autopilot system. This autopilot system has already been equiped in the Tesla Model 3 (see Figure 1.4.5). However, this autopilot technology can only perform several functions, including but not limited to accelerating, braking, and steering. The Tesla drivers still need to take control of the car. The U.S. National Highway Traffic Safety Administration gives a definition to a Level 5 self-driving cars:

> *"An automated driving system (ADS) on the vehicle can do all the driving in all circumstances. The human occupants are just passengers and need never be involved in driving."*[6]

While a Level 2 self-driving car is defined as:

> *"An advanced driver assistance system (ADAS) on the vehicle can itself actually control both steering and braking/accelerating simultaneously under some circumstances. The human driver must continue to pay full attention (monitor the driving environment) at all times and perform the rest of the driving task."*[6]

Therefore, the current Tesla's autopilot system only suits the Level 2 requirement. There is still a long journey for Tesla to improve its system; indeed, in March 2016, an incident was reported on Twitter that the Tesla's autopilot system mistakenly recognized the salt lines, which was caused in advance of a massive snowstorm, as the normal traffic broken white lines on the highway, see Figure 1.4.6.

Figure 1.4.5: White Tesla Model 3. Photo by `https://en.wikipedia.org/wiki/Tesla_Model_3`.

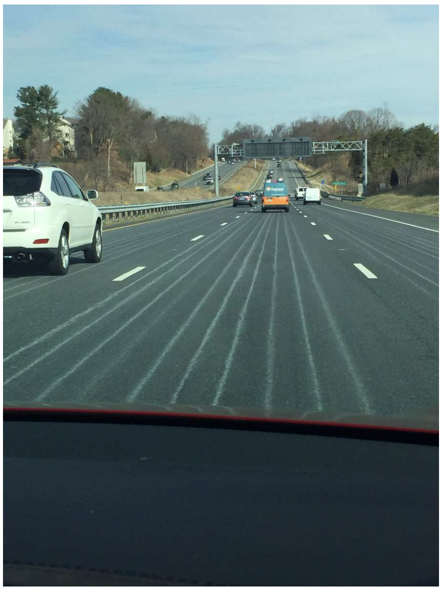

Figure 1.4.6: Original picture on Twitter: Salt lines confuse Tesla's autopilot system. Photo by `https://twitter.com/amywebb/status/841292068488118273`.

[6]Retrieved from `https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety`

BIBLIOGRAPHY

[1] Turing, A.M. (1950). Computing Machinery and Intelligence. *Mind, LIX* (236), 433-460. doi: 10.1093/mind/lix.236.433.

[2] Newell, A., and Simon, H. (1956). The logic theory machine–A complex information processing system. *IRE Transactions on information theory* 2(3), 61-79.

[3] Whitehead, A.N., and Russel, B. (1910). *Principia Mathematica*. Cambridge: University Press.

[4] Buchanan, B. G., and Feigenbaum, E. A. (1980). The stanford heuristic programming project: Goals and activities. *AI Magazine* 1(1), 25-25.

[5] Bachant, J., and Soloway, E. (1989). The engineering of XCON. *Communications of the ACM* 32(3), 311-319.

[6] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences* 79(8), 2554-2558.

[7] Rumelhart, D.E., Hinton, G.E., and Williams, R.J. (1986). Learning representations by back-propagating errors. *Nature* 323, 533-536.

[8] Shapiro, E. Y. (1983). The fifth generation project—a trip report. *Communications of the ACM* 26(9), 637-641.

[9] Mahalanobis, P. C. (1936). "On the generalised distance in statistics" (PDF). Proceedings of the National Institute of Sciences of India. 2 (1): 49–55. Retrieved 2016-09-27.

[10] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ..., and Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. nature, 529(7587), 484-489.

[11] Rabiner, L., and Juang, B. (1986). An introduction to hidden Markov models. ieee assp magazine, 3(1), 4-16.