0.1 History of AI

The following sections present a broad overview of the history of Artificial Intelligence (AI) without specifying or detailing too much on particular topics of this theme. The main objective is to present some context by presenting important articles in order for the reader to be able to have a notion of the progress that has been made over the past decades, the hardships encountered and how important AI is in our lives.

Philosophy

On October 1950, in his article *Computing Machinery and Intelligence*, Alan Turing questioned: ”Can machines think?” [40]. At the time, the question was too meaningless to answer since not only the theory but also the technology available weren't developed enough. Nonetheless, Turing still predicted that in the future there would be computers that could, effectively, display human-like intelligence and discernment under the conditions proposed on the aforementioned article.

Relevant events to the birth of AI

The breakthroughs of AI are predominant, and its importance in our everyday life is undeniable, but the theory behind it has several early roots. The interest in the area grew immensely with, for example, all the Turing's theoretical research, the proposal of the first mathematical Artificial Neuron model in 1943 by Warren McCulloch and Walter Pitts (based of binary inputs and output) [26] and in 1949 Donald Hebb revolutionized the way the artificial neurons were treated by proposing what is known as the Hebb's rule[1]. Taking into consideration the latter two, but specially Hebb's proposals, Belmont Farley and Westley Clark implemented in 1954 one of the first successful Artificial Neural Networks (ANN), also called Perceptron, composed of two layers of 128 artificial neurons with weighted inputs [11]. Over the span of approximately ten years, multiple researches were performed attempting to computerize the human brain. However, only in 1956, during the *Dartmouth Summer Research Project on Artificial Intelligence* [25], was the term ”Artificial Intelligence” firstly proposed by John McCarthy *et al.*, beginning what is now considered to be the birth of AI [48].

[1] ”When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.” [14], meaning that when two neurons fire together their relation is strengthened.

The fading of general interest

The succeeding two decades following the Dartmouth conference were filled with important developments, with special emphasis in the works published in 1958 by Frank Rosenblatt (generalized the Farley and Clark training to multilayer networks rather than only two) [32], the 1959 General Problem Solver implemented by Allen Newel *et al.* (a program intended to work as a universal problem solver that was capable of solving exercises such as the Towers of Hanoi[2]) [28] and the ELIZA a natural language processing tool program developed by Joseph Weizenbaum between 1964 and 1966 [42]. Unfortunately, part of the interest and development around AI met an unforeseen fade after criticisms about the exaggerated public funding [13] and the Marvin Minksy and Seymour Papert 1969 book *The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain* [27] that reported on the problems of the Perceptron network. The overall sentiments regarding this topic of research was of doubt and fear of no progress, mainly due to the spending and two issues raised by Minsky and Papert: the ANN couldn't solve linear inseparable problems[3] and there were limitations due to a lack of sufficient computing power to handle the processing of multi-layer large networks.

A better approach

Minksy and Papert raised important questions, but it shouldn't have discouraged other researchers from further trying, since they failed to acknowledge alternative approaches that had already solved those exact problems. As previously stated, the model proposed by McCulloch and Pitts, later improved by the Farley-Clark implementation and, finally, Rosenblatt, couldn't handle linearly inseparable classes. A possible solution for cases like this started being studied in the 1960s [18, 33] and, although it didn't produce relevant results, in 1965 Alexey Ivakhnenko and Valentin Lapa [16] were, indeed, successful in implementing what is nowadays considered to be the first deep learning network of its kind [35]. In 1971 Ivakhnenko also published an article describing a deep learning network with 8 layers that was already able to create hierarchical internal representations [17].

The years progressed, in 1979 Kunihiko Fukushima introduced the first Convolutional Neural Network (CNN) in a structural sense, due to its similarity to the architecture of modern ones of this category. Ten years later, Yann LeCun *et al.* applied for the first time a revolutionizing training algorithm called Backprop-

[2] *The Towers of Hanoi* is a game with 3 stacks of increasingly smaller disks. The goal is to stack them one at a time, so that they are arranged in a decreasing radius manner.

[3] That is, if two sets X and Y in $\mathbb{R}^d$ can't be divided by a hyperplane such that the elements of X and Y stay on opposing sides, then we're dealing with linear inseparable classes [10]

agation to a CNN [20], creating what is now a pillar for most of the modern competition winning networks in computer vision [35] and employing the term "convolution" for the first known time [23]. He also introduced the MNIST (**M**odified **N**ational **I**nstitute of **S**tandards and **T**echnology) dataset, a collection of handwritten digits [22], that to this day is still one of the most famous benchmarks in Machine Learning. Backpropagation can be traced back many decades, but the modern version was first described by Seppo Linnainmaa (1970) [24], implemented for the first time by Stuart Dreyfus (1973) [9] and, finally in 1986, David Rummelhart *et al.* popularized it in the Neural Network's (NN) domain by demonstrating the growing usefulness of internal representations [34].

The importance of Convolutional Neural Networks

The study on Neural Networks continued and there were improvements on all types of architectures [15, 43] with special highlight to pioneering Neural Networks processed by GPUs[4] (standard NN in 2004 by [29] and CNN in 2006 by [4]). But there's a well deserved particular attention related to the developments of CNNs due to their great performance in image related tasks when compared to others networks, as proven by LeCun in his 1998 paper [22]. Some relevant examples: in 2003 the MNIST record was broken by Patrice Simard *et al.* [36], achieving an error rate of 0.4% (whereas a non-convolutional neural network by the same authors took the second place with 0.7%); three years later, the same benchmark had a new set low of 0.39% by Marc'Aurelio Ranzato *et al.* [31]; in 2009 a CNN by Yang *et al.* was the first network of this type to win an official international competition (TRECVid) [46]; a GPU implementation of a CNN [6] achieved superhuman vision performance in a competition (IJCNN 2011) in a *German Traffic Sign Recognition Benchmark* with a 0.56% error rate (0.78% for the best human performance, 1.69% for the second-best neural network contestant and 3.86% for the best non-neural method [37]). This last example conjoined with non-convolutional methods [30, 8] and the previously cited [4, 29], reinforces how fundamental GPUs were to further develop neural networks. To supplement even more the importance of CNNs and GPUs, only a year later, Alex Krizhevsky *et al.* proposed a Deep CNN trained by GPUs that was the first one to win the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), achieving an error rate of 15.3% while the second place obtained 26.2% [19].

The year of 2012 was very important for Deep Learning, CNNs and Computer Vision, due to all the attention brought to many researches on this topic after several systems of this kind won image analysis competitions ([5, 7] and the very important previously mentioned [19]), beginning what's considered to be the start

[4] Graphics Processing Unit

of the new wave, we're currently in, of interest in Artificial Intelligence, specially in the aforesaid topics [23].

0.2 Fundamentals

Convolutional Neural Networks

There are several types of Neural Networks architectures, but Convolutional Neural Networks (CNNs or Convnets) are probably the most widely implemented model overall [45, 23] with successful applications in the domains of Computer Vision [19, 38, 39, 49] or Natural Language Processing[1, 41, 44]. In the CNN category itself there are different variants, but they all abide the fundamental structure of a feedforward hierarchical multi-layer network. Feedforward because the information only flows in a singular direction without cycling [47], hierarchical because the higher complexity internal representations are learned from lower ones [21, 50] and multi-layer because it is composed of a series of stages, blocks or layers: the raw data is fed to an input layer, forwarded to a sequence of intercalating convolutional and pooling layers, transmitted to a stage of one or more fully-connected layers [21, 45, 12, 2]. The convolutional layer is designed to extract feature representations by being composed of kernels (or filter banks [21]) that compute feature maps through element-wise product, to which is applied a nonlinear activation function [12, 45]. Next is the pooling layer, that's responsible for reducing the spatial size of the input data [12] and joining identical features [21]. Finally, the fully connected layers and their core function is to perform high logic and generate semantic information [12]. Finally, the output layer

Using CNNs for Computer Vision tasks is not an arbitrary choice, but due to the fact that the network design can benefit from the intrinsic characteristics of the input data, consequently performing really well in image related applications [21, 3]. In the first place, images have an array-like structure with numerous elements, namely, each pixel has an assigned value organized in a grid-like manner [45]. In the second place, there's an inherent correlation between local groups of values, which creates distinguishable motifs [21]. Finally, the local values of images are invariant to location, that is, a certain composition should have the same value independently of the spatial location in the picture [21]. Therefore, the following key, unique features potentiate the previously stated efficient performance [3]:

1. Designed to process multidimensional arrays [21];

2. Shared weights between the same features in different locations;

3. Automatically identifies the relevant features without any human supervision, hence, small amounts of preprocessing [2, 23];

4. Local connections/receptive fields/sparse connectivity [2];

5. Pooling layers that reduces the spatial size of the input data.

The ensemble of features 2, 4 and 5 enable an invariance of the network to small shifts, distortions and rotations citations needed, while 2, 3, 4 and 5 helps to reduce the complexity of the model, and as a result training it is easier.

0.3 Related Works