

## Chapter 2

# A Brief Review on Deep Learning and Types of Implementation for Deep Learning

**Uthra Kunathur Thikshaja**

*Kyungpook National University, South Korea*

**Anand Paul**

*Kyungpook National University, South Korea*

### ABSTRACT

*In recent years, there's been a resurgence in the field of Artificial Intelligence and deep learning is gaining a lot of attention. Deep learning is a branch of machine learning based on a set of algorithms that can be used to model high-level abstractions in data by using multiple processing layers with complex structures, or otherwise composed of multiple non-linear transformations. Estimation of depth in a Neural Network (NN) or Artificial Neural Network (ANN) is an integral as well as complicated process. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. This chapter describes the motivations for deep architecture, problem with large networks, the need for deep architecture and new implementation techniques for deep learning. At the end, there is also an algorithm to implement the deep architecture using the recursive nature of functions and transforming them to get the desired output.*

### INTRODUCTION

The increase in demand for organizing the data and analyzing them is mainly due to the abundance in the raw data generated by social network users. Not all the data generated are linear and hence the single perceptron layer network or the linear classifier as it is popularly known, cannot be used for data classification. No hidden layers are required when we have a linearly separable data. In most other cases,

DOI: 10.4018/978-1-7998-0414-7.ch002

one hidden layer is enough for a majority of problems. In few problems, two hidden layers are used for full generality in multilayer perceptrons. But lots of random initializations or other methods for global optimization are required. Local minima with two hidden layers can have extreme blades or spikes even when the number of weights is much smaller than the number of training cases (Panchal, 2011). Deep learning is a new AI trend that uses multi-layer perceptron network. Multilayer sensor containing multiple hidden layers is a deep learning structure. Deep learning architecture is a good way to extract feature, it can be used for specific issues of classification, regression, information retrieval, speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics.

## **Motivation for Deep Architecture**

There are three main reasons for us to have a deep network. The first one is that insufficient depth can hurt, i.e. when the depth is two, the number of nodes in the flow graph (used for representing deep architecture) may grow very large. Theoretical studies (Hastad's theorems) have shown that the order of complexity in  $O(n)$  for depth  $d$  and  $O(2^n)$  for depth  $d-1$ .

The next reason is that, the human brain itself has a deep architecture. The visual cortex is well-studied and shows a sequence of areas each of which contains a representation of the input, and signals flow from one to the next. Each level of this feature hierarchy represents the input at a different level of abstraction, with more abstract features further up in the hierarchy, defined in terms of the lower-level ones.

The last motivation for a deep architecture is that cognitive processes are deep. Humans organize their ideas and concepts hierarchically. Humans first learn simpler concepts and then compose them to build more abstract ones. Engineers break up solutions into multiple levels of abstraction and processing.

## **Problems With Large Networks**

Increasing the depth decreases the complexity. However, there are also problems associated (Vasilev, 2015). when the number of hidden layers is increased (depth). The first one is the Vanishing Gradients. This is a problem when the transfer of data from output layer through the hidden layers to the input layer becomes more and more difficult as the number of hidden layers is increased. This concept is called back propagation. Another important concern is Over-fitting. It is the phenomenon of fitting the training data too closely. The training data may be fitted really close and good, but these will fail badly in real cases. Scientists came up with several architectures to overcome the above mentioned problems. The upcoming sections describe some of the most common deep learning methods in use.

## **Deep Learning Methods**

### **Autoencoders**

An autoencoder is typically a feedforward neural network which aims to learn a compressed, distributed representation (encoding) of a dataset. Conceptually, the network is trained to "recreate" the input, i.e. the input and the target data are the same. In other words, we are trying to output the same thing that was provided as the input, but compressed in some way.

9 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/a-brief-review-on-deep-learning-and-types-of-implementation-for-deep-learning/237861](http://www.igi-global.com/chapter/a-brief-review-on-deep-learning-and-types-of-implementation-for-deep-learning/237861)

## Related Content

---

### Analysis of Student Study of Virtual Learning Using Machine Learning Techniques

Neha Singhand Umesh Chandra Jaiswal (2022). *International Journal of Software Science and Computational Intelligence* (pp. 1-21).

[www.irma-international.org/article/analysis-of-student-study-of-virtual-learning-using-machine-learning-techniques/309995](http://www.irma-international.org/article/analysis-of-student-study-of-virtual-learning-using-machine-learning-techniques/309995)

### Contour Gradient Optimization

Zhou Wu, Shi Chengand Yuhui Shi (2020). *Handbook of Research on Advancements of Swarm Intelligence Algorithms for Solving Real-World Problems* (pp. 247-276).

[www.irma-international.org/chapter/contour-gradient-optimization/253428](http://www.irma-international.org/chapter/contour-gradient-optimization/253428)

### Petri Nets and Discrete Events Systems

Juan L. G. Guiraoand Fernando L. Pelayo (2011). *International Journal of Software Science and Computational Intelligence* (pp. 13-22).

[www.irma-international.org/article/petri-nets-discrete-events-systems/60746](http://www.irma-international.org/article/petri-nets-discrete-events-systems/60746)

### Nature-Inspired-Based PTS for PAPR Reduction in OFDM Systems

Mohamed Mounir, Mohamed Bakry El Mashadeand Gurjot Singh Gaba (2020). *Nature-Inspired Computing Applications in Advanced Communication Networks* (pp. 57-88).

[www.irma-international.org/chapter/nature-inspired-based-pts-for-papr-reduction-in-ofdm-systems/240953](http://www.irma-international.org/chapter/nature-inspired-based-pts-for-papr-reduction-in-ofdm-systems/240953)

### A Generic Framework for Feature Representations in Image Categorization Tasks

Adam Csapo, Barna Resko, Morten Lind, Peter Baranyiand Domonkos Tikk (2012). *Software and Intelligent Sciences: New Transdisciplinary Findings* (pp. 491-512).

[www.irma-international.org/chapter/generic-framework-feature-representations-image/65147](http://www.irma-international.org/chapter/generic-framework-feature-representations-image/65147)