Notes on Residual Networks

Artur Back de Luca

February 11, 2019

Introduction

This is the compilation of my personal notes on Residual Networks. They cover introductory aspects such as the success of deep learning models, the intrinsic challenge of learning that prevented such models to reach popularity in earlier decades and its solutions; the degradation problem and finally the structure and explanation of residual networks.

1 Success of deep learning

The success of deep learning models resides on the ability to derive complex features hierarchies from low-level inputs. This is achieved by substantially coupling non-linear operations throughout layers which whose parameters, in turn, rearrange, gradually adapting to a target function. Features of an input can be progressively exploited as the number of layers in a Neural Network increase, establishing network depth as a determinant factor of success in models, especially in intricate tasks such as image recognition.

Figure 1: Compositional representation of features throughout layers: from pixels to gradients and edges. Source: Goodfellow et al. (2016, pg. 6)

Research outlines from He et al. (2015, pg. 2) and Bengio (2009, pg. 6) indicate performance improvement upon an increase in network depth assisted by techniques which will be later here commented such as max-pooling, batch normalization, and initialization methods[1]. Yet, as the reader may conjecture, the absolute number of layers in a network is not the main issue, rather the relative size to how many layers are necessary to effectively represent the target function. As Bengio (2009, pg. 9) describes:

> "More precisely, functions that can be compactly represented by a depth k architecture might require an exponential number of computational elements to be represented by a depth $k-1$ architecture. Since the number of computational elements one can afford depends on the number of training examples available to tune or select them, the consequences are not just computational but also statistical: poor generalization may be expected when using an insufficiently deep architecture for representing some functions."

In this statement, Bengio (2009) drives an analogy of deep learning architectures and logic circuits, supported by the work of Hastad (1986) and Yao (1985) which imply that logic architecture limited in-depth presents an exponential number of components. As an illustration, consider the calculation of the parity function, defined as:

$$f : \{0,1\}^n \rightarrow \{0,1\} \; s.t. \; f(x) = (\sum_{i=1}^{n} x_i) \, mod \, 2$$

[1] More unmentioned techniques are applied such as dynamic learning rates, dropout and appropriate activation functions, such as the ReLu.

The work developed by Yao (1985) suggests that the number of logical components for a depth-limited architecture of 2 layers is of exponential order of 2^{n-1}, in oposition of depth unlimited architectures which in turn can derive less complex orders, such as the balanced tree structure, of $O(N \log N)$ Bengio and LeCun (2007).

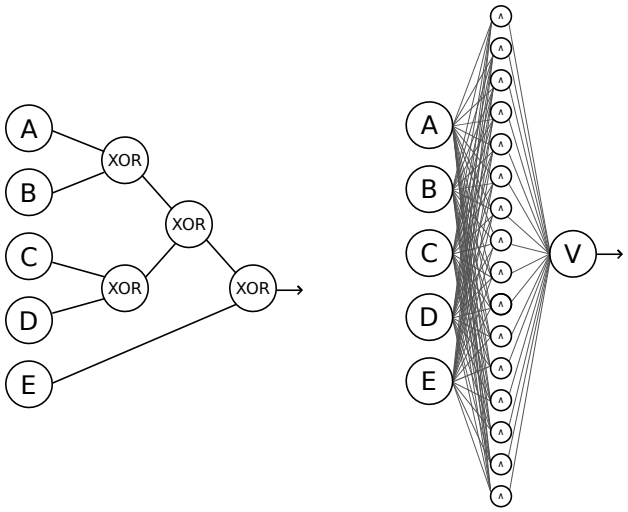

Figure 2: Distinct architectures to represent the parity function. A balanced tree structure with 3 layers and complexity of $O(N \log N)$ (*left).* A DNF (Disjunctive normal form) structure, with complexity of $O(2^N)$ (*right*).

So, ideally, a function represented by a model may have a minimum number of layers which are necessary to properly generalize its behavior. As one may assume, complex tasks, in this case, specified as functions, would require deeper networks. And supporting evidence from recent breakthroughs in Machine Learning are associated with the recent capability of stacking an increasing amount of layers, which will be further discussed.

Next topics:

2 - Why deep learning was not popularly employed
3 - Techniques to improve the increasing amount of layers in a network
4 - The degradation problem
5 - Residual Networks
Conclusion

References

Bengio, Y. (2009). Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127.

Bengio, Y. and LeCun, Y. (2007). Scaling Learning Algorithms towards AI. page 41.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Hastad, J. (1986). Almost Optimal Lower Bounds for Small Depth Circuits. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, STOC '86, pages 6–20, New York, NY, USA. ACM. event-place: Berkeley, California, USA.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep Residual Learning for Image Recognition. *arXiv:1512.03385*.

Yao, A. C. (1985). Separating the polynomial-time hierarchy by oracles. In *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, pages 1–10.