

Deep Learning with Random Neural Networks

Abstract—This paper introduces techniques for Deep Learning in conjunction with spiked random neural networks that closely resemble the stochastic behaviour of biological neurons in mammalian brains. The paper introduces clusters of such random neural networks and obtains the characteristics of their collective behaviour. Combining this model with previous work on extreme learning machines, we develop multilayer architectures which structure Deep Learning Architectures a a “front end” of one or two layers of random neural networks, followed by an extreme learning machine. The approach is evaluated on a standard – and large – visual character recognition database, showing that the proposed approach can attain and exceed the performance of techniques that were previously reported in the literature.

摘要—本文介绍了深度学习技术以及与随机神经网络相似的尖峰随机神经网络，该网络非常类似于哺乳动物大脑中生物神经元的随机行为。本文介绍了这种随机神经网络的集群，并获得了其集体行为的特征。将此模型与以前在极限学习机上的工作相结合，我们开发了多层体系结构，该结构将深度学习体系结构构造为一到两层随机神经网络的“前端”，然后是极限学习机。在标准的大型视觉字符识别数据库上对该方法进行了评估，表明该方法可以达到并超过文献中先前报道的技术性能。

I. INTRODUCTION

In recent years, deep learning with regular and tightlycoupled arrays of sigmoidal neurons has come to the forefront as a possible way to overcome the limitations of neural networks when they are applied to real-world challenges [1], [2]., while many engineering applications require significant advances in learning from big data [3], [4], [5], [6].

Tightly coupled clusters of natural neuronal cells communicate with each other in multiple ways, both through spiking [7], via soma-type interactions with other cells [8], through neuromodulators [9], and with the help of important structures such as glial cells [10] which are known to exercise complex functions in the hippocampus and cerebellum that contribute to synaptic transmission and modulate synaptic function. This complexity of natural neural information processing and learning [11] goes well beyond the models traditionally exploited in machine learning [12], and goes significantly beyond the capabilities of sigmoid-based neural models.

The Random Neural Network (RNN) [13] is a spiking “integrate and fire” model where an arbitrarily large set of cells interact with each other via excitatory and inhibitory spikes which modify each cell's action potential in continuous time, and mathematically described by a system of differential equations known as the Chapman-Kolmogorov equations [14]. It was originally developed to mimic the behaviour of biological neurons [15]. However subsequently it was exploited for numerous applications that exploit the recurrent structure of the network and of its learning algorithm [16], including combinatorial optimisation [17], several examples of image and video processing [18], [19], [20], [21], [22], [23], and routing [24], [25], [26] in cyber-physical systems and computer networks. All these applications exploit the recurrent structure of the network.

The computational power of the RNN originates in the fact that in steady-state the network is characterised by the joint probability distribution of the activation state of each cell, which is equal to the product of the marginal probabilities of the activation states. This property, known as “product form” in the probability literature [14] makes the RNN particularly amenable to exact yet simple, fast (and easily parallisable) computational algorithms.

近年来，使用规则且紧密耦合的S形神经元阵列进行深度学习已成为最前沿的方法，可以克服将神经网络应用于实际挑战时的局限性[1]，[2]。工程应用程序需要从大数据学习中取得重大进步[3]，[4]，[5]，[6]。

紧密耦合的天然神经元细胞簇以多种方式相互通信，既可以通过尖峰[7]，通过与其他细胞的体细胞相互作用[8]，通过神经调节剂[9]，也可以通过重要结构（例如神经胶质）已知在海马和小脑中行使复杂功能的细胞[10]，这些功能有助于突触传递并调节突触功能。自然神经信息处理和学习的复杂性[11]远远超出了机器学习中传统开发的模型[12]的范围，并且大大超出了基于S形神经模型的功能。

随机神经网络（RNN）[13]是一个尖峰的“整合并发射”模型，其中任意大的细胞组通过兴奋性和抑制性尖峰相互作用，这些尖峰在连续时间内改变了每个细胞的动作电位，并通过数学方式描述为微分方程组称为Chapman-Kolmogorov方程[14]。它最初是为了模仿生物神经元的行为而开发的[15]。然而，随后它被用于利用网络的递归结构及其学习算法的众多应用[16]，包括组合优化[17]，图像和视频处理的几个示例[18]，[19]，[20]，[21]，[22]，[23]以及网络物理系统和计算机网络中的路由[24]，[25]，[26]。所有这些应用程序都利用网络的循环结构。

RNN的计算能力源于以下事实：在稳定状态下，网络的特征在于每个单元的激活状态的联合概率分布，该概率分布等于激活状态的边际概率的乘积。这种特性在概率文献[14]中被称为“乘积形式”，这使得RNN特别适合于精确而又简单，快速（且易于并行化）的计算算法。

II. THE MATHEMATICAL MODEL

We consider the Random Neural Network Model developed in [27], [28], composed of M neurons or cells, each of which receives excitatory (positive) and inhibitory (negative) spike trains from external sources which may be sensory sources or cells. These arrivals occur according to independent Poisson processes of rates λ_m^+ for the excitatory spike train, and λ_m^- for the inhibitory spike train, respectively, to cell $m \in \{1, \dots, M\}$.

In this model, each neuron is represented at time $t \geq 0$ by its internal state $k_m(t)$ which is a non-negative integer. If $k_m(t) > 0$, then the arrival of a negative spike to neuron m at time t results in the reduction of the internal state by one unit: $k_m(t^+) = k_m(t) - 1$. The arrival of a negative spike to a cell has no effect if $k_m(t) = 0$. On the other hand, the arrival of an excitatory spike always increases the neuron's internal state by $+1$.

我们考虑在[27]，[28]中开发的由 M 个神经元或细胞组成的随机神经网络模型，它们每个从外部来源（可能是感觉来源或外部来源）接受兴奋（正）和抑制（负）尖峰（或叫脉冲）序列。这些到达是根据独立的泊松过程发生的，速率分别为兴奋尖峰序列 λ_m^+ 和抑制尖峰序列的 λ_m^- ，到 $m \in \{1, \dots, M\}$ 。

在该模型中，每个神经元在时间 $t \geq 0$ 时都以其内部状态 $k_m(t)$ 表示，该状态为非负整数。如果 $k_m(t) > 0$ ，则在时间 t 负尖峰到达神经元 m 导致内部状态减少一个单位： $k_m(t^+) = k_m(t) - 1$ 。如果 $k_m(t) = 0$ ，负尖峰到达单元将不起作用。另一方面，兴奋性尖峰的到来总是使神经元的内部状态增加 $+1$ 。

If $k_m(t) > 0$, then the neuron m is said to be “excited”, and it may “fire” a spike with probability $r_m \Delta t$ in the interval $[t, t + \Delta t]$, where $r_m > 0$ is its “firing rate”, so that r_m^{-1} may be viewed as the average firing delay of the excited m – th neuron.

Neurons in this model can interact in the following manner at time $t \geq 0$. If neuron i is excited, i.e. $k_i(t) > 0$, then when i fires its internal state drops by 1 and we have $k_i(t+) = k_i(t) - 1$, and:

如果 $k_m(t) > 0$, 则神经元 m 被称为被“兴奋”, 并且它可能在区间 $[t, t + \Delta t]$ 以概率 $r_m \Delta t$ “火” (激发) 一个尖峰 (脉冲), 其中 $r_m > 0$ 叫做他的“激发率”, 所以 r_m^{-1} 可以被看作是第 m 个兴奋的神经元的平均激发延时。

在这个模型中的神经元可以在时刻 $t \geq 0$ 时和下列方式相互作用。如果神经元 i 被兴奋, 即 $k_i(t) > 0$, 那么当 i 激发时, 它的内部状态下降1, 所以我们可以得到 $k_i(t+) = k_i(t) - 1$, 并且

- It can either send a positive or excitatory spike to neuron j with probability $p^+(i, j)$ resulting in $k_i(t^+) = k_i(t) - 1$ and $k_j(t^+) = k_j(t) + 1$,
- Or it may send a negative or inhibitory spike to neuron j with probability $p^-(i, j)$ so that $k_i(t^+) = k_i(t) + 1$ and $k_j(t^+) = k_j(t) - 1$ if $k_j(t) > 0$, else $k_j(t^+) = 0$ if $k_j(t) = 0$
- Or neuron i can “trigger” neuron j with probability $p(i, j)$ so that $k_i(t^+) = k_i(t) - 1$ and $k_j(t^+) = k_j(t) - 1$ if $k_j(t) > 0$.
- When neuron i triggers neuron j , both $k_i(t^+) = k_i(t) - 1$ and $k_j(t^+) = k_j(t) - 1$, and one of two things may happen. Either:
 - 以概率 $p^+(i, j)$ 发送一个正的或兴奋尖峰到神经元 j , 导致 $k_i(t^+) = k_i(t) - 1$ 和 $k_j(t^+) = k_j(t) + 1$,
 - 或者可能以概率 $p^-(i, j)$ 发送一个负的或抑制尖峰到神经元 j , 所以如果 $k_j(t) > 0$, 则 $k_i(t^+) = k_i(t) + 1$ 和 $k_j(t^+) = k_j(t) - 1$, 否则如果 $k_j(t) = 0$, 则 $k_j(t^+) = 0$
 - 或者神经元以概率 $p(i, j)$ 激励神经元 j , 所以如果 $k_j(t) > 0$, 则 $k_i(t^+) = k_i(t) - 1$ 和 $k_j(t^+) = k_j(t) - 1$
 - 当神经元 i 激励神经元 j 时, 有 $k_i(t^+) = k_i(t) - 1$ 和 $k_j(t^+) = k_j(t) - 1$, 并且, 两种情况中会有一种发生。分别是:
 - (A) With probability $Q(j, m)$ we have $k_m(t^+) = k_m(t) + 1$ so that i and j together have incremented the state of m . Thus we see that a trigger allows two neurons i and j to increase the excitation level of a third neuron m by +1, while i and j are both depleted by -1.
 - (B) Or with probability $\pi(j, m)$ the trigger moves on to the neuron m and then with probability $Q(m, l)$ the sequence (A) or (B) is repeated.
- 注意 $\sum_{j=1}^M [p(i, j) + p^-(i, j) + p^+(i, j)] = 1 - d_i$ where d_i is the probability that when the neuron i fires, the corresponding spike or trigger is lost or it leaves the network. Also, $1 = \sum_{m=1}^M [Q(j, m) + \pi(j, m)]$.
- 注意 $\sum_{j=1}^M [p(i, j) + p^-(i, j) + p^+(i, j)] = 1 - d_i$, 其中 d_i 是当神经元 i 激发时的概率, 相应的峰值或触发器丢失或离开网络。同样的,

Since cells in different layers of mammalian brain also communicate through simultaneous firing patterns of densely packet somas, the RNN was extended in [29], [28] using a branch of the theory of stochastic networks called G-Networks [30]. In the sequel we will exploit this structure for deep learning.

由于哺乳动物大脑不同层中的细胞也通过密集包体的同时发射模式进行通信，因此RNN在[29]，[28]中使用了称为G-Networks的随机网络理论的一个分支[30]进行了扩展。接下来，我们将利用这种结构进行深度学习。