神经形态计算读书报告

1. 研究背景

神经形态计算是人工智能技术与神经科学所交融的一个领域，这里包含了机器学习、人工智能、硬件开发、新材料以及神经科学的新发现。“神经形态”这个词在最初被提出时指的是一类特制的设备或芯片，在它们的结构中包含了模拟电路，并且能够模仿生物神经行为[1]。时至今日，神经形态计算已包括了软硬件协同、材料科学、神经科学等多种不同领域，而且同人工智能技术有着紧密的结合，其中最吸引人的便是对脑的研究。

随着人工智能技术的发展，传统计算机在人脸识别、语言处理以及逻辑推理等方面取得了极大的进步，但是在效率上仍然无法与以人脑为代表的生物神经网络相比。尽管脑的工作机理目前尚未完全摸清，但神经系统所具有的巨大潜力仍吸引了相当多的科学家投身其中。这其中既包括了希望探明人脑工作原理与智能原理的神经科学家和认知科学家，也包括了希望通过对生物神经网络的理解来改进计算体系结构的计算机科学家。

在神经科学方面，根据已有的对神经系统的了解，神经科学家们建立了一系列用于模拟神经元行为的神经元模型，探明了神经元工作过程中的多种电化学过程。但是仅对单个神经元的行为进行研究是不够的，生物神经系统中不仅有大量的神经元，神经元之间还存在着复杂的突触网络，生物神经系统的各种功能是由特定区域中所有神经元之间交互的结果。

另外，神经系统的结构并不固定。人类的大脑中具有数十亿的神经元细胞，在整个生命过程中，一个人可能会失去约20%的神经细胞，但是大脑仍然能保持同样的功能。这说明了神经系统的结构并不是一成不变的，其结构具有可塑性。因此，在神经元模型之外，神经科学家还提出了多种的神经网络模型来研究神经系统的拓扑结构，例如Hopfield神经网络、Hohonen自组织网络等等

随着对生物神经系统理解的加深，计算机科学在也在试图从生物神经系统中寻找灵感。1958年，心理学家及计算机科学家Rosenblatt提出了感知机的概念[2]。感知机仿照神经元设计，将一组输入分别乘以权重最后求和，将最后的总和与事先设定好的阈值进行比较，如果超过阈值，那么感知机被“激活”，输出置为有效，反之输出则为无效。

单层感知机对于线性问题具有分类效果，但是对于非线性的分类问题难以处理。因此，人们考虑将多个感知机按一定的拓扑结构进行组合，从而增加分类系统的维度，于是形成了多个感知机以网络结构的形式实现多个输出的多层感知机。这也是人工神经网络的最初原型。

在人工神经网络刚开始发展时，多层感知机中的权重值主要是依赖人工方式调整。Rosenblatt在提出感知机时一并提出了一个直观的“学习”方法：将输出结果与实际值相比较，如果输出偏低则调高权重值，反之亦然。直到20世纪80年代，反向传播算法开始出现，从而使得多层感知机具备了自行从数据中计算误差并修正权重的能力，形成了较为成熟的模型自动训练机制。自此之后，各种浅层模型开始出现，例如支持向量机、最大熵方法等。

2006年，Geoffrey Hinton发表文章指出，具有多层隐层的神经网络可以更好地对数据提取特征，同时利用无监督学习实现的逐层初始化方法可以克服原本训练深度神经网络的瓶颈[3]，这标志着深度学习概念的开启。深度学习所使用的逐层分级处理的方法也有仿生学角度的支持：1981年获得诺贝尔医学奖的Hubel-Wiesel模型中发现了动物的视觉系统分级处理信息，每一级的神经元会对上一级的信息进行进一步的抽象与迭代，最终得到大脑中的感知概念。

深度学习概念提出后，世界范围内很多大学与研究机构都对此进行了研究，并提出了更多的人工神经网络模型，如卷积神经网络CNN、递归神经网络RNN、二值神经网络BNN等。随着深度神经网络模型的日益复杂化，如何对神经网络模型进行快速有效的处理成为了一个重要课题。另一方面，随着对人脑研究的深入，我们也需要构建高效的、规模足够大的神经系统模拟器，以探究人脑的工作机理，甚至更进一步，探索智能所具有的基本特性乃至真正利用计算机实现智能。这两方面问题的逐渐交汇，构成了今天的神经形态计算。

有效地支持人工神经网络模型的训练与运行，和构建高效的大规模神经系统模拟器并不是两个独立的问题。人工神经网络的提出与发展都受到生物神经系统结构的启发，各种模型也或多或少地带有一定的仿生学特点，这代表两者之间存在相似的问题；而如果存在一个可以高效地对神经系统进行模拟的软硬件系统，那么通过调整神经元模型、网络结构和输入的刺激，这样的系统也应该能同样地支持人工神经网络的运行。于是这两个问题就可以归结到，什么样的系统才是最适合进行神经系统的计算。

为了解决这一问题，首先我们需要了解生物神经网络和人工神经网络的结构与特点，接下来的两节将分别对两者进行介绍。

1. 生物神经网络
2. 总述

尽管有关意识和脑的理论可以追溯到古代文明，但目前认为关于神经系统的基本生物学知识主要开始于19世纪下半叶。最初的研究手段是利用染色和显微镜的解剖学方法，经由染色方法的改进，生物学家们成功地揭示了神经细胞的结构与连接。

此后电生理学开始发展，生物学家开始探究神经细胞中的电活动。常见的方法有两个：胞内记录和胞外记录。胞内记录方法利用电极记录细胞内电信号的变化情况，通常使用动物的脑组织，由于需要将电极插入胞体内部，这种方法会损害实验使用的组织部分。胞外记录方法观测的对象是细胞外电信号的变化情况，例如突触间的电脉冲。与胞内记录方法相比，这种方法对组织的损害较小，可以用于活体动物。

对脑组织功能的研究则使用更先进的研究手段，例如核磁共振成像（MRI）、脑电图（EEG）、可以将细胞膜电位转化成光学信号的电压敏感染料（VSD）、正电子成像术（PET）等等。这些技术不仅可以用于观测大脑的内部结构，还可以对特定的功能过程进行记录。这些新技术帮助人们发现了许多未曾发现的现象并探明了许多认知过程的机理，进一步推动了神经科学和认知科学的发展。

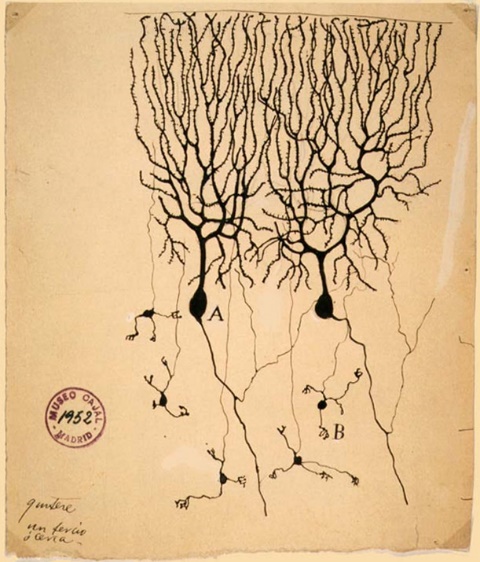
当代神经科学家已经理解了神经信号、突触传递以及信息存储和加工的神经编码等基本概念。研究人员致力于在基本的分子及细胞水平的加工与单个神经元及神经环路的活动之间建立联系。尽管关于生物事件决定行为的准确方式的理解仅仅处于起始阶段，但目前的研究已经积累了大量的神经元水平的生物学知识。本节接下来的部分会分别简述神经系统中主要成分的结构及其功能。

1. 神经元

神经元是基本的信号处理单位。他们依靠各自的外形、功能、位置和神经系统内的连接方式来彼此区分。神经元接收信息，遵循某些相当简单的规律做出“决定”，然后改变自己的活动水平，最后将信息传递给其他神经元。这些功能与神经元的形态或结构的特异性密切相关。

20世纪初，Ramon y Cajal通过对神经系统的显微观察提出神经系统由单个神经元构成的理论，并最终被普遍接受。Ramon y Cajal观察到，尽管神经元彼此之间紧密相邻，但被很小的缝隙分隔。通过这些观察，他确定了神经元的两个主要原理：

1. 连接的特异性：神经元间彼此不相通，是相互独立的；但神经元间的连接并不是随意的，这是神经元传递信息的特异性通路。
2. 功能性两极分化：神经元的一部分专门负责接收信息，另一部分专门负责将信息传递给其他神经元或肌肉。



图：Cajal绘制的神经元结构图[4]

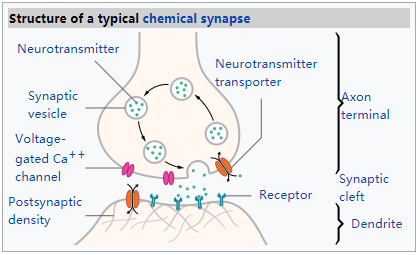
神经元的组成成分之一是细胞体。与其他细胞一样，细胞体中包含有维持神经元新陈代谢的细胞器。细胞体是神经元中电化学活动的处理中心，这些电化学活动会导致神经元活动水平的改变，产生传递给其他神经元的电信号或化学信号。因此细胞体可以说是是神经元的中枢处理单元。

除细胞体外，神经元的另一个组成成分是延伸至胞体外的特异性突起，树突和轴突。神经元的两种突起反映了功能的两极分化。树突通常是大树样的突起，接收来自其他神经元的传入信息。另一类突起是远离胞体的轴突，该结构代表着神经元的输出端：电信号传送至轴突末梢，轴突末梢通过释放传递信号的化学物质神经递质实现信息的交流。

人脑中共有约1011个神经元，这些神经元有多种类别，在体积、形状上各不相同。不同种类的神经元有着不同的行为模式、形态学相似的神经元倾向于集中于神经系统的某一特定区域，并具有相似的功能。

1. 突触

前面提到，神经元通过树突接收来自其他神经元的传入信息，具体接收部位是被称为突触的结构。因此对于信息流而言，树突是位于突触后的结构，通常又被称为突触后；同时，对于信息流而言轴突是位于突触前的结构，通常又被称为突触前。只有针对某一个突触时，才会提及突触前和突触后，大多数的神经元既是突触前又是突触后：当它们的轴突与其他神经元建立连接时，它们是突触前；而当其他神经元与它的树突建立连接时，它又是突触后。



图：突触结构

在人脑中有超过1015个突触，平均每个神经元通过突触与大约10000个神经元互相连接。这些突触大多是通过化学信号传递信息，少部分突触直接利用电信号传递信息。在利用化学信号传递的情景中，突触前神经元的轴突末端接收到电信号，释放出包含神经递质的囊泡，神经递质与突触后神经元的树突相结合，激活特定的离子通道，进而改变突触后神经元的状态；直接利用电信号传递的情境中，两个神经元通过细胞膜上的孔道相连，直接传递电信号的变化。

突触主要分为两种：激活和抑制。激活型的突触会打开突触后神经元的钠离子通道，正离子的流入会使细胞膜电位去极化，令突触后神经元更容易发生激活。抑制型的突触会激活氯离子通道，负离子的流入加强了膜电位的极化，令突触后神经元更难发生激活。突触活动所涉及的电化学过程可以用基于电导的方程进行量化描述。

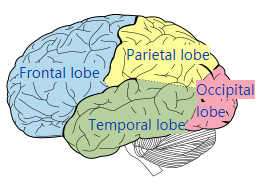
突触结构并不是一成不变的。神经系统在发育过程中会调节突触连接的强度、增加新突触或者去除掉不需要的突触。突触结构的这种变化体现了神经系统的可塑性，神经系统的可塑性在智能中具有重要意义。

1. 脑结构

尽管脑的确切机理仍然没有探明，神经科学家仍然探明了大脑的整体结构，并从脑结构的层面解释了许多认知与行为现象。脑结构中最重要的部分有三个：脑干，小脑，大脑皮质。

脑干位于大脑结构最下层，与脊髓相连。脑干的体积较小，但其中包括了大量的运动与感觉神经核团，控制呼吸乃至意识水平如睡眠和觉醒。小脑覆盖于脑干结构上部，包含了大量的输入输出神经纤维，负责整合有关身体和运动指令的信息并调整运动，使其变得流畅而协调，在维持姿势、行走以及协调运动过程中都至关重要。

大脑皮质是脑结构中最大的部分，分为两个对称的半球，每个半球都是由分层的神经元组成。人类和其他高等哺乳动物的皮质有很多褶皱，其中凹陷的部分称为沟，突起的部分称为回。皮质褶皱缩小了容纳脑结构所需要的空间，根据测定，人类的皮质总面积大约为2200-2400cm2，但是因为褶皱的存在，大约有三分之二的部分被折叠到了沟裂中。褶皱还使得神经元之间形成了非常紧密的三维联系，加快了神经传导的速度。

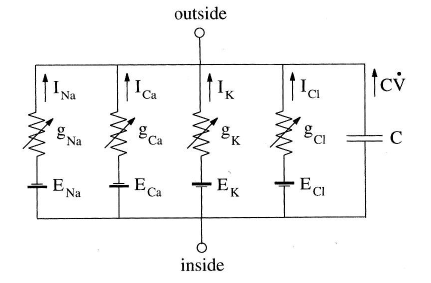


图：大脑皮质分叶

对于智能的研究来说，大脑皮质是最重要的研究对象。按照解剖学分区方式，大脑的两个半球主要可以分为四个脑叶：额叶、顶叶、颞叶和枕叶。这些区域担负着不同的功能，每个叶在解剖结构上与其他叶有着明显标志进行区分。大脑皮质的各个叶在神经加工的过程中发挥着多方面的作用，虽然主要的功能系统一般能够定位在某个脑叶中，但是也有很多系统是跨脑叶的。大体上来说，额叶主要负责运动的准备与执行，顶叶主要负责躯体感觉，枕叶负责视觉加工，颞叶负责听觉加工，另外还有位于感觉运动交互区域的联合皮质，主要负责高级心理加工。

1. 神经元的数学模型

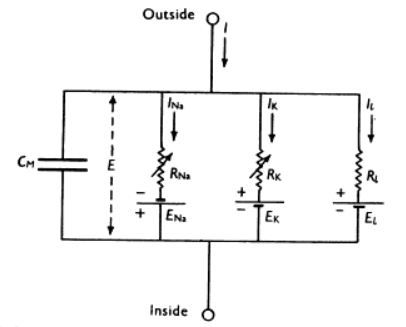
神经元的活动是在细胞膜附近一系列离子运动的结果，主要参与的离子有四种：钠离子（Na+）、钾离子（K+）、钙离子（Ca2+）和氯离子（Cl-）。这四种离子在细胞膜内外的浓度不同，受浓度差和细胞内外电位差的驱动而移动。浓度差和电位差向不同的方向驱动离子，当两者达到平衡时，跨越细胞膜的离子净流量为零。不同的离子有着不同的平衡点，此时该离子在细胞膜内外形成的电位差称为该离子的平衡电位，可以用Nernst方程来量化分析特定离子的平衡电位。根据神经元的电生理活动特点，就可以建立描述神经元活动的数学模型。



图：细胞膜的等效电路图

1）Hodgkin-Huxley模型

HH模型是一组非线性微分方程，描述神经元细胞膜的电生理活动，由两位研究者Alan Lloyd Hodgkin与Andrew Huxley于1952年提出[5]，获得了1963年诺贝尔生理学奖。

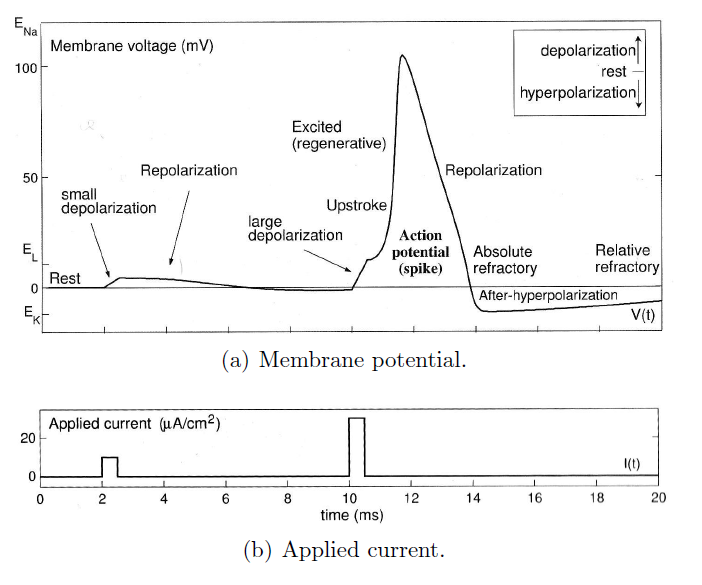


图：Hodgkin-Huxley（HH）模型电路图[引用]

上图描述了HH模型对应的电路图，其中C代表脂质双层的电容；，，分别代表钠离子通道，钾离子通道与漏电通道的电阻；，，代表由于膜内外离子浓度差导致的电位差，分别是漏电平衡电位，钠离子平衡电位，钾离子平衡电位。以V代表细胞膜电位，那么HH模型的完整方程式如下：

其中：

上式中，C代表细胞膜电容，是一个常量；I是来自细胞外界的电流，如来自突触的电流，或来自插入电极的电流；，，是各通道的电导，是对应通道电阻的倒数。



图：HH模型中膜电位与输入脉冲的关系[6]

上图是一个典型HH的膜电位变化图，共施加了两个脉冲电流刺激。当较小的脉冲到达时，钠离子通道的电导增加，流入胞体的钠离子使膜电位升高，在图中产生了一个峰值。但是由于膜电位没有超过阈值，无法达到能够产生脉冲的激活状态，膜电位很快回落到复位电位上。如果输入的脉冲足够大的话，那么膜电位将会越过阈值，最终产生一个神经脉冲。在脉冲产生后，钠离子通道会受到抑制，同时钾离子的外流会将膜电位拉至低于复位电位的水平，在这段时间内神经元无法产生新的脉冲，称为不应期。在不应期结束后，膜电位恢复到复位电位。

2）整合放电（Integrate-and-fire）模型

Hodgkin-Huxley模型在生物学上非常精确，但是这个模型对于模拟来说太过复杂了。为了降低模拟的运算复杂程度，科学家提出了行为级的神经元模型。行为级模型不追求参数在生物学上的准确含义，而是在模型中体现出神经元的关键特性，从而使得模拟所需要的计算资源降低。一个行为级的神经元模型通常必须要包括三个关键特征：

* 当膜电位超过预先定义好的阈值时能够产生脉冲
* 当神经元激活并发射神经脉冲后，能够将膜电位复位到一个初始值
* 模型应具有一个不应期，防止连续地产生脉冲

在各种行为级模型中，Leaky Integerate-and-fire（LIF）模型是最为著名的一种[7, 8]。LIF神经元的建模有多个变种，其中最简单的一种，称为基于电流的LIF模型，其公式描述为

其中V为膜电位，为膜电容，为漏电电导，为漏电平衡电压，I为输入电流。从公式中可以看出，这是一个经典RC电路。当膜电位V升至阈值，就会触发一个输出脉冲，发送给它相连的下游神经元，此后膜电位快速复位至。此后膜电位有一段不应期，对外界新输入的脉冲没有响应。LIF模型采用基于一个阈值参数的脉冲触发取代了HH模型中基于电导的复杂变化，提高了运算效率，其代价是牺牲了较大的生物精确性。LIF模型存在一个较为严重的缺点，就是因为过于简化，真实神经元中的一些脉冲发射模式无法用LIF模型进行还原。

3）Izhikevich模型

另一个重要的行为级模型是Izhikevich模型，该模型在2003年由Eugene M. Izhikevich从非线性动力系统的角度提出[9]，其模型结果可以呈现多种生物神经元的放电行为，运算复杂度则接近于LIF，其公式描述为：

其中U代表了膜电位恢复能力，反映了如K+活跃和Na+抑制等因素的影响。变量a是U的时间尺度，减小a的值可以使膜电位恢复变慢；变量b是膜电位对U的敏感程度，增大b可以使U得变化对V影响更大；变量c是产生脉冲后的复位电位；d是产生脉冲后U恢复时的偏移量。

可以看到，Izhikevich模型中各个参数并不具有明确的生物学含义。但是相对于HH模型，Izhikevich模型在计算上极大地降低了复杂度，以1ms的模拟尺度来衡量，Izhikevich模型仅需要进行13次浮点运算来模拟1ms的模型运行，而HH模型则需要1200次浮点运算。另外，Izhikevich模型比LIF模型更加灵活，通过调整a，b，c，d四个参数，可以模拟出多种神经元的放电行为。

4）非脉冲神经元模型

HH模型、LIF模型和Izhikevich模型都是脉冲神经元模型。脉冲神经元模型的输出是一系列的神经脉冲，通过脉冲产生的时间来传递信息。除了脉冲神经元之外，还有着非脉冲神经元模型。这一类模型将输入的动作值根据神经元的活动水平转化为一个实数值输出，输出值范围通常在0-1之间。非脉冲神经元模型的一个典型的例子是多层感知机网络。

非脉冲神经元模型的输出通常被视为神经元的脉冲平均率。一般认为脉冲神经元模型在生物学上更加接近现实情况。脉冲神经元模型的输出可以比较容易地编码为脉冲平均率的模式，而非脉冲神经元模型的输出中不含有时间信息，所以无法转换成脉冲的模式。

1. 神经系统信息编码

无论是使用脉冲模型还是非脉冲模型，一个必须解决的问题是如何编码神经系统的信息，特别是在实际的脑或神经系统中信息如何编码。虽然这个问题对于神经系统建模至关重要，但仍旧没有得到很好的解答。

目前常用的编码方式有两种：脉冲平均率编码和时间编码。脉冲平均率编码用脉冲的频率来表达神经元的输出信息，好处在于易于通过实验测量。但是在对人类视觉信号处理过程的研究中发现，如果使用脉冲平均率编码的话，处理速度无法达到人类实际的反应速度。这个发现意味着人脑中并不仅仅使用脉冲平均率编码。另一方面，其他研究也表明在神经活动中时间确实是一个重要的信息[10]，因此在编码时需要将脉冲发生的时间纳入考虑中，这就是时间编码。

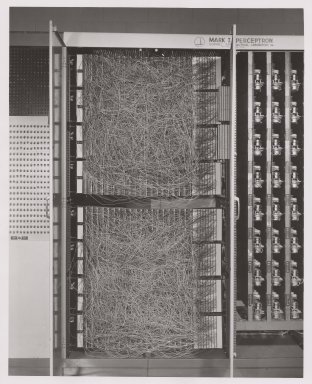
除这两种常用编码方式外，神经科学研究还发现了神经信息的群体编码，即通过神经元群而不是单个神经元的状态来表达信息。由于神经系统的复杂性，单个神经元的活动容易受到影响，所以群体编码有利于提高信息编码的可靠性。

1. 人工神经网络
2. 总述

自人工智能概念提出时，人工神经网络的思想就出现了。人工神经网络是经典人工智能的路线之一：既然人脑智能是由神经网络产生的，那就通过人工方式来构造神经网络，进而产生智能。这一流派强调智能活动是由大量简单的单元通过复杂的连接后并行运行的结果，因此又被称为“连接主义”。

人工神经网络的思想可以追溯到1943年神经生理学家Warren McCulloch和数理逻辑学家Walter Pitts提出的神经元数学模型。McCulloch和Pitts提出的这一模型基于阈值逻辑，奠定了对于人工神经网络研究的基础。此后对于人工神经网络的研究分为了两种道路，一种关注脑活动的生物过程并试图从结构上进行模拟，另一种则关注如何将神经网络应用于智能。在1956年达特茅斯会议提出“人工智能”概念一年前，美国西部计算机联合大会在洛杉矶召开。此次会议上展开了“学习机讨论会”，参会者中有两位参加了第二年的达特茅斯会议，他们是Oliver Selfridge和Alan Newell，两人分别代表了两派观点。会议的主持人Walter Pitts总结时说“（一派人）企图模拟神经系统，而Newell则企图模拟心智，……，但殊途同归”。

1958年，Rosenblatt创造了感知机。感知机最初在IBM 704上使用软件实现，之后Rosenblatt设计了第一台原型机“Mark 1 Perceptron”。Mark 1 Perceptron为图像识别任务设计，内部包含400个光电管的阵列，权重使用电位计编码，学习过程中通过电机实现权重的更新。感知机在刚出现时引起了人工智能学界的热烈讨论，但很快人们发现存在很多感知机无法有效分类的特征。1969年Minsky和Papert的研究指出了当时神经网络研究存在的两个关键问题：其一是基本的感知机模型无法处理异或电路，第二个是当时的计算能力并不能有效支持大规模的神经网络。这一发现导致了神经网络领域研究的停滞。



图：Mark 1 Percptron

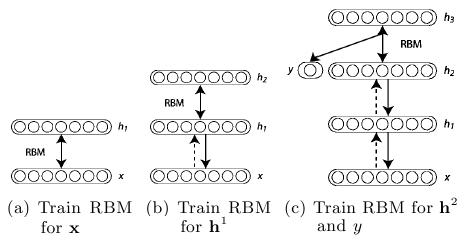
1975年，Werbo提出了后向传播算法应用于神经网络训练中[11]。后向传播算法在一系列数据处理完成后计算每个处理单元对总误差的贡献，据此调整每个处理单元的权重。后向传播算法不仅很好地解决异或问题，而且加快了多层神经网络的训练过程。这使得神经网络模型有能力向增加更多层数的方向发展，极大地增强了整个神经网络的处理能力。然而Werbo提出这一想法时正处于人工智能发展的低潮期，直到1986年，后向传播的训练方法才因为David Rumelhart、Geoffrey Hinton和Ronald Williams发表的文章而被广泛接受。

后向传播算法被应用于神经网络的训练后，神经网络的研究迎来了一次新的发展，神经网络被应用于多种领域，如文字识别、图像识别等等。神经网络在多种应用中取得了喜人的成果，但是在语音识别领域遇到了难题。在图像识别中，输入的数据是一张张图片，图片之间相互独立；而在语音识别中，输入的数据是一个语音流，这种输入难以处理为批处理的数据。一方面，从一段语音中分离出单词是一件不容易做到的事；另一方面，语言的理解依赖于上下文环境而不仅仅是单词的语义。为了处理这类输入，特别是处理需要上下文环境的输入，需要对神经元引入“记忆”。一个办法是将网络的输出作为输入提供给神经元，使得网络形成递归的结构，这就是递归神经网络（Recurrent Neural Network, RNN）。

RNN在语音识别中取得了较好的效果，但是也暴露出了难以训练的问题。Yoshua Bengio在1993年发表文章总结了在RNN训练中遇到的问题[12]，指出虽然在RNN中能够引入长间隔的依赖关系，但是短间隔的依赖关系对参数训练的影响仍然占有主要部分。这是由于后向传播算法中的梯度消失所导致的。

在后向传播式的训练中，需要将误差值逐层传播来修正神经元的权重参数，通常使用误差函数的梯度值作为权重的修正标准。然而随着层数的增加，梯度值会呈指数变化，这就导致在传播层数较多时，梯度值可能会变得非常小以至于无法有效地修正神经元参数。这种梯度值迅速减小的现象称为梯度消失。

由于无法有效地解决梯度消失问题，神经网络的研究在90年代再一次陷入低潮。1997年Schmidhuber提出了长短期记忆（Long Short Term Memory）的神经网络模型[13]，可以将长时间步之前的事件影响有效传递下去。但是这仍无法改变当时神经网络的普遍印象。在90年代中期，学界普遍对神经网络失去了信心，转而研究支持向量机等人工智能模型，直到2006年深度学习概念被提出后才有所转变。



图：深度置信网络的训练过程

2006年，Geoffrey Hinton、Simon Osindero和Yee-Whye The发表文章，提出深度学习概念[3]，在神经网络研究上取得了一次巨大突破，使得人们对神经网络重燃信心。这篇文章指出在设置合适的初始权重的情况下，深层的神经网络也可以快速而有效地进行训练。神经网络的初始值可以利用有限玻尔兹曼机（Restricted Boltzmann Machine, RBM）来产生：首先在训练数据集上训练一个RBM，这个RBM输出的数据将作为第二层的训练数据集使用，通过反复训练新的RBM来最终形成一个置信网络，当分类结果令人满意后，加入一个小的隐层数据集来对权重进行进一步的精细调整。这种训练方式结合了监督学习和无监督学习，因而又被称为半监督学习。使用这种方法训练的深度置信网络在MNIST文字识别中取得了非常好的结果，远远超出仅有数层的其他神经网络。

在深度学习概念提出后，神经网络应用变得越来越广泛，各种人工智能应用也在不断涌现出来。从大体上来看，人工神经网络的发展过程就是一个不断加深网络层数，并解决随着层数加深而浮现出的网络训练问题的过程。神经网络研究的两次低潮与两次复兴都与网络训练方法密切相关，可见网络训练对于神经网络的关键性。

1. 卷积神经网络（Convolutional Neural Network, CNN）

1）卷积神经网络简介

卷积神经网络是一类深度的前向人工神经网络，在视觉相关的应用中有很好的效果。相比于其他的图像分类算法，CNN不需要对图像进行太多的预处理。这意味着CNN可以学习传统算法中需要手工编程的特征，这种不需要预备知识和人力参与的特点是CNN的一个重要优点。

在20世纪50到60年代，神经生理学家Hubel和Wesel发现猫和猴子的视觉皮层中的神经元只对视野中一小块区域有响应，这块区域称为该神经元的感受野。Hubel和Wesel发现相邻的视神经细胞之间具有相似且部分重叠的感受野；在整个视觉皮层中，感受野的位置与大小随神经元所处的位置系统地变化，并最终组成视野区域的完整映射[14]。此后的研究发现视觉皮层中有两类细胞，其中一类细胞感受野较小，当有物体位于感受野正中央时细胞活跃程度最大，称为简单细胞；另一类细胞感受野范围较大，只要有物体进去感受野这类细胞就会活跃，但是对物体在感受野中所处的位置并不敏感，这类细胞称为复杂细胞。两类细胞的特性导致的结果是，简单细胞可以用于精确定位，复杂细胞可以用于快速筛选。

受Hubel和Wesel的工作启发，Kunihiko Fukushima在80年代提出了新感知机（Neocognitron）的模型[15]。这是一个层次化的多层人工神经网络模型，主要用于手写文字识别和其他的模式识别任务，并对此后卷积神经网络的提出有所启发。新感知机参考了视觉皮层中简单细胞与复杂细胞的分类，在网络中使用了两种不同的神经元模型：一类称为S-cell，用于提取局部特征；另一类称为C-cell，用于将局部特征进行变形与整合。这种提取局部特征进行整合从而得到全局特征的方法启发了后来的LeNet和SIFT等模型。

1998年，LeCun提出了LeNet-5[16]，这是一个7层的卷积神经网络模型，也是大多数现代卷积神经网络模型的雏形。LeNet-5对数字判断的准确性较高，曾被美国数家银行用来识别支票上的手写数字，成为了卷积神经网络应用中的先驱。LeNet-5的输入图像为32x32大小，大于MNIST数据库中最大的字母，这样做是希望笔画断点或角点等潜在的明显特征能够出现在感受野的中心。如果希望进一步提高输入图像的分辨率，LeNet-5需要增加更多的卷积层，对当时的计算能力来说这很困难。

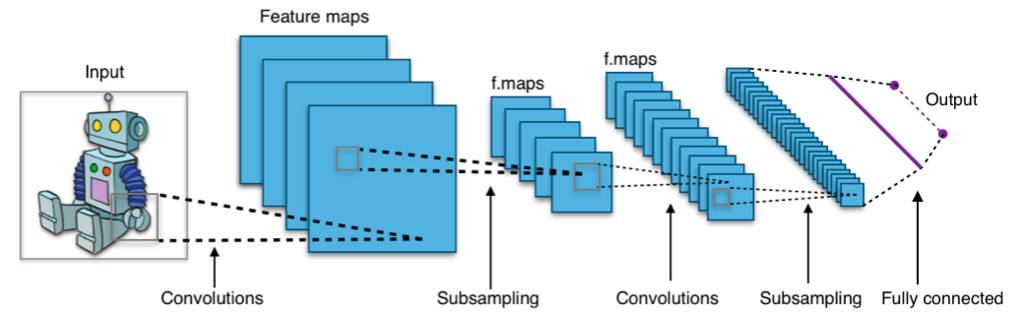
2005年，Dave Steinkrau等人提出将GPGPU应用于机器学习中[17]，此后许多研究者提出了利用GPGPU来加快卷积神经网络训练过程的方法，这其中就包括了Hinton等人提出的深度学习方法。在GPGPU获得广泛应用后，卷积神经网络也在很多应用领域中取得了优秀的成果，例如谷歌公司的AlphaGo使用卷积神经网络来学习围棋棋谱。

2）卷积神经网络的结构

卷积神经网络在结构上和传统的多层感知机比较接近，可以认为是多层感知机的一个生物学变种。相比多层感知机，卷积神经网络有以下独特的特征：

* 三维神经元：CNN中的神经元有三个维度，width，height和depth，这里depth并不是指神经网络的深度，而是神经元的属性。
* 局部连接：按视觉皮层中神经元细胞的感受野概念，CNN中神经元只和前一层的一小部分相连接。
* 共享权值：CNN中相同深度维度上的神经元会共享同一组权值，共享权值的这些神经元会响应同种类的特征，这些神经元称为一个深度切片（depth slice）。这种共享权值的方式使得特征的发现不依赖于特征在视野中的出现位置，保证了特征的平移不变性。

卷积神经网络通常包括五个层，分别是提取特征运算的卷积层、进行数据抽样的池化层、进行分类操作的全连接层以及控制输出的激活层。



图：典型的卷积神经网络结构

卷积层是CNN的核心组成部分，通常实现的是一个类似于滤波的操作。卷积层的中的神经元相当于过滤器（filter），每一个神经元是一个大小的卷积核，在神经元的感受野上进行滑窗操作，在每个窗口内将卷积核与输入的特征图进行卷积操作。卷积相当于对个数据点进行逐元素的相乘，然后进行累加：

卷积层使用三个超参数来控制输出的规格：深度、步幅和补零。

* 深度参数控制每层中连接到相同输入区域的神经元数量，这些神经元对输入数据中的不同特征进行学习。例如，第一个卷积层把原始数据作为输入，深度参数为2，其中一个神经元对边缘的方向进行学习，而另一个神经元对输入的颜色进行学习。
* 步幅参数控制神经元在空间维度上的分布，即每次滑窗操作时窗口移动的距离。步幅为1时，每次滑窗移动一个像素，这会导致神经元的两次处理之间存在很大的重叠部分，输出的特征图尺寸也比较大。如果将不复增大到2，那么重叠部分会减少，输出的特征图尺寸也会相应减小。
* 在一些情况下，为输入数据的边界进行补零可以带来方便。补零的数量就是第三个超参数。通过控制补零的数量，可以直接对输出的空间尺寸进行控制。

池化层对卷积层的输出进行一个下采样，以降低特征图的空间尺寸，对特征进行进一步的抽象。池化的基本思想是特征的精确位置并不是特别重要，特征与特征之间大致的相对位置关系更重要。通常的池化操作包括最大值池化、平均值池化、L2范数池化等。以常用的最大值池化为例，通常将特征图划分为多个大小的矩形区域，并选择每个子区域中的最大值输出。下图是一个使用2x2过滤器、步幅为2的最大值池化过程示意：



图：最大值池化示意

可以看到，在池化后特征图的尺寸从4x4减小到了2x2，由于对每一个深度切片分别进行池化，输出的特征图在深度尺寸上没有改变，仅在空间尺寸上缩小了。通常会周期性地在连续的卷积层后插入池化层，来控制参数的数量以及网络的计算量，同时也控制了过匹配的发生。

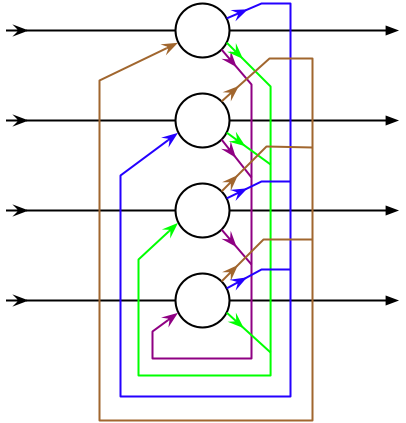
全连接层负责进行高级推理。顾名思义，全连接层的神经元与前一层的所有输出相连接，这和一般的神经网络结构相同，可以认为全连接层神经元的感受野就是整个视野。在全连接层，经过前面的卷积层和池化层抽取出的特征得到了进一步的整合，从而由局部特征提升为全局特征。

激活层决定神经元能否将输出值继续向后传递。和生物神经网络的原理相似，在输出值超过一定的阈值时，我们选择将神经元激活，使得其输出可以向后传递。CNN中常用的激活函数包括纠正线性函数ReLU、双曲正切函数tanh、S形函数sigmoid等。激活前的输出值将在激活函数上进行乘法叠加后再行输出，同时激活函数的导数将用于反向传播的训练过程中。

1. 递归神经网络（Recurrent Neural Network, RNN）

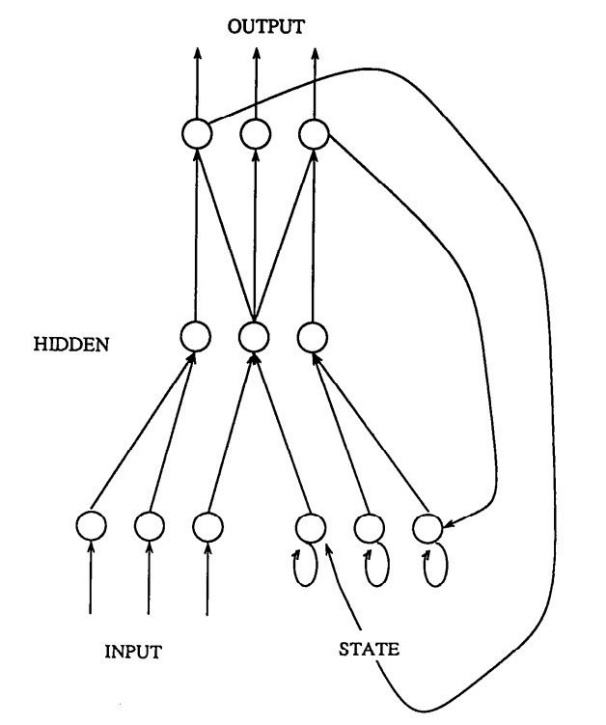
1）递归神经网络简介

递归神经网络可以追溯至20世纪80年代。1982年，John Hopfield提出了Hopfield网络[18]，这是率先在网络结构中使用递归连接形式的神经网络模型之一。Hopfield网络中，所有的连接都是双向的，同时要求输入必须是固定的，因此Hopfield网络并不是一个典型的RNN网络。



图：四个节点的Hopfield网络模型

1990年，Jeff Elman提出了三层的Elman网络，这一模型受到了Michael Irwin Jordan所提出模型的启发，Jordan所提出的网络模型如下图：



图：Jordan网络结构

在Jordan网络中，输出层将把结果反馈回到能反映上下文状态的状态单元，同时状态单元也会将输出反馈给自己。

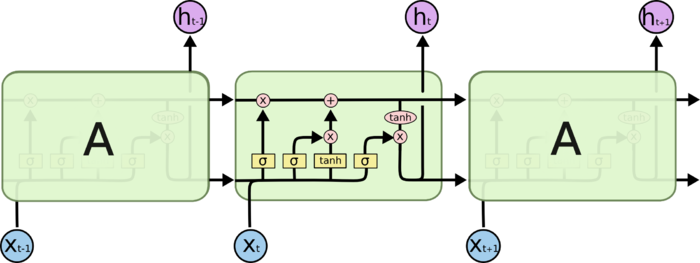
1992年，Schmidhuber发现了神经网络训练中的梯度消失问题，并在1997年提出了长短期记忆（Long Short Term Memory, LSTM）模型[13]。LSTM能够比较好的解决梯度消失问题，并且能更好地处理长时期依赖关系，在现在仍然有广泛的应用。在LSTM之后，还有数种RNN模型被提出来，如双向递归神经网络（Bidirectional Recurrent Neural Network, BRNN）、神经抽象金字塔（Neural Abastraction Pyramid, NAP）、门控循环单元（Gated Recurrent Unit, GRU）等。最近几年还有研究者尝试将RNN与CNN组合起来。

RNN所引入的递归结构让处理单元可以结合上下文信息对特征进行处理，同时递归的结构相当于用比较少的参数增加了神经网络的深度。RNN的递归结构让网络有了一定的记忆能力，因此常用于输入是一个流的情况，例如自然语言处理、语音识别、机器翻译等任务。

2）递归神经网络结构

递归神经网络有多种变种，这里选择两个应用比较广泛的变种进行介绍：LSTM和GRU。

LSTM网络模型由Schmidhuberh和Hochreiter在1997年共同提出[13]。LSTM相对于传统RNN最大的改进是引入了细胞状态，使得神经元可以长期地记忆信息。LSTM网络中神经元包括了四个部分：细胞状态、输入门、输出门和遗忘门。



图：LSTM示意图

细胞状态是图中上方贯穿的水平线，神经元通过细胞状态传递信息，三个门用来控制细胞状态的改变。在每个处理周期开始时，神经元首先根据输入来处理上一个周期的输出状态，决定需要遗忘哪些信息，这是遗忘门所完成的工作；接下来输入门根据权重决定输入中的哪些部分会被更新到细胞状态中，；最后输出门对细胞状态进行过滤，决定哪些细胞状态可以输出出去，最终产生新的输出。

GRU是LSTM的一个变体。GRU将LSTM的三个门缩减为两个门：复位门和更新门。复位门决定如何把新的输入与之前的细胞状态相组合，更新门决定保留多少之前的状态。GRU删除了输出门，对神经元的输出不进行筛选。GRU与LSTM在运行效果上没有明显的优劣。

1. 脉冲神经网络（Spiking Neural Network, SNN）

相比于其他人工神经网络，脉冲神经网络引入了时间的概念。网络中的神经元并不是根据固定的循环周期给出输出，而是以类似生物神经系统中的模式，在神经元的状态超过某一阈值后才向之后的神经元传递信息。这种工作模式让脉冲神经网络更接近生物神经网络。

脉冲神经网络使用的神经元模型往往直接来源于神经科学所提出的模型，如HH模型、LIF模型和Izhikevich模型。由于使用了更接近真实生物神经元的模型，理论上来说脉冲神经网络不仅可以像其他人工神经网络一样完成信息处理的任务，还可以用来模拟生物神经系统，验证生物学假说是否成立。

但是在实际工程中，脉冲神经网络还无法达到理论上的高度。脉冲神经网络目前在生物学模拟方面的有效性已经获得了证明，然而在真实世界应用上仍然存在不足，一个重要的原因是多层SNN没有特别有效的训练方法。对于单层SNN，目前脉冲神经元的主要学习算法包括ReSuMe[19]、SPAN[20]等，已经取得了较大的进展。但是单层的SNN无法解决异或等重要问题，为此有必要增大SNN的层数。在多层SNN中，由于广泛使用的学习机制脉冲时间依赖可塑性（Spike-Timing Dependent Plasticity, STDP）难以进行后向传播以及神经脉冲所具有的的不连续性，多层SNN的监督学习算法难以构造。

为了解决这一问题，一些研究者提出将其他人工神经网络转化为SNN，从而绕开SNN的训练难题。这些工作目前局限于前馈神经网络，而且构建的网络规模仍然较小。例如目前尚未有人尝试将SNN应用于图像物体识别测试集ImageNet，这可能是因为ImageNet需要庞大的深度神经网络，对此转化之后的SNN难以有效地运行在现有的计算平台上。这也说明了在训练之外，SNN还存在着缺乏有效运算平台的局面。

1. 神经形态计算研究现状
2. 神经网络研究中需要解决的问题

在生物神经网络的模拟方面，几个必须要考虑的问题是：如何权衡精度和处理效率、以何种形式存储信息、通信如何实现。

在精度和处理效率的权衡上，主要的考虑因素是神经元模型的选择。在大规模系统中，效率通常比模型精度更加重要，因此倾向于选择运算上较为简单的行为模型，如前文提及的LIF模型和Izhikevich模型；当需要对模型行为的假说进行验证时，模型精度就显得更为重要，这时就会选用HH模型等更加生物精确的模型，或者使用假说提出的模型。

存储方面，模拟生物神经网络需要存储的信息包括：控制神经元模型的常量参数、突触连接的权重、供突触进行学习的历史信息等。由于神经网络中突触连接的数量非常之大，所以权重所占用的存储空间是最大的。对于只有一个节点的模拟系统而言，这部分信息可以统一保存在一处。但是在大规模系统中，就必须考虑所有的处理节点如何有效地对突触连接的信息进行访问，如何放置突触权重不仅仅关系到处理节点的运行效率，而且与整个系统的互联结构密不可分。

由于神经系统中突触的巨大数量，在对生物神经网络进行模拟时突触将会占用相当多的处理资源，如何构建突触连接是模拟系统中至关重要的问题。尽管电路连接的信号传导速度要远远快于生物神经系统中神经纤维的速度，但是单纯靠电路连接能够形成的突触数量远远低于生物神经系统，而且可塑性受到很大限制。

人工神经网络和生物神经系统的模拟一样面临着精度与效率的权衡、网络结构如何组织等问题，除此之外人工神经网络的研究中还面临着两个重要的问题，一是训练和运行时间，二是系统的能效性。

从人工神经网络的发展历史中可以看出，训练方法在整个发展历史中扮演了重要的地位。近年来面向大规模图像数据集的模型其权重个数已经达到了百万量级，不仅加剧了训练算法的计算量，同时也对训练完成后模型的响应时间提出了更高的要求。以目前常用的GPU和CPU混合训练方式来看，对于大型的神经网络，训练就需要花费数周的时间。如何提高神经网络模型的训练和计算速度，是当前神经网络研究中的重要课题。

随着神经网络应用范围越来越广，移动设备以及嵌入式设备对于神经网络应用的需求也开始浮现出来。在这类设备中，系统的计算资源和能源供应都受到了很大的限制。特别是能源方面，这类设备必须要考虑续航问题，而且通常不会有大体积的散热设备，所以需要严格控制计算的能耗。因此能效性是当前神经网络研究的另一重要课题。根据Horowitz教授在2014年ISSCC会议上的报告[21]，对定点数进行乘法操作时能耗与数据宽度呈二次关系，进行加法操作的能耗则呈线性关系，同时移动数据消耗的能量与数据距离处理器的距离呈现出指数增长的趋势。因此人工神经网络系统中如何组织数据存储，以及如何平衡数据位宽与模型精度都对提高整个系统能效性有重要的意义。

1. 神经形态计算有关研究的分类

神经形态计算可以概括为“受神经系统启发的非冯诺依曼计算系统”[22]，类似于神经系统，具有“神经元”和“突触”这两种基本组成单元，在组成单元之间具有简单的通信并且具有学习能力，同时还有共处一地的存储和计算。

需要注意的是，神经形态计算还并不能完全取代冯诺依曼结构。冯诺依曼结构有着图灵机理论等坚固的数学理论基础，保证了这一结构能够完成通用计算任务，而神经形态计算目前还没有提出类似的通用理论。因此，神经形态计算是一种利用非冯诺依曼结构对于传统计算模式的补强，对神经形态计算的研究是为了应对涌现出的新问题而对传统计算做出的改革。

从计算机科学的角度来看，对神经形态计算的研究可以分为三类：

第一类研究被神经科学所启发，主要目的是构建能够高精确度地模拟大规模生物神经系统结构的软硬件系统，从而更好地理解大脑的结构以及工作过程。尽管这类研究的一个重要目的是为神经科学的发展提供支持，但是其中的许多项目开拓出了与传统的超级计算机集群不同的体系结构。这类研究可以概括为神经模拟系统的构建。

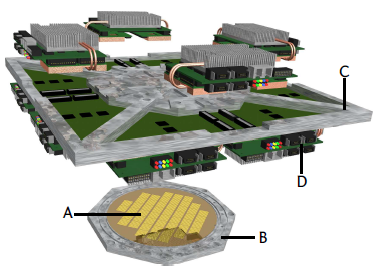
第二类研究从现有的深度学习网络结构出发，针对特定类型的神经网络以及特定的训练算法开发专用的硬件来加速深度学习中的学习或训练过程。这类研究可以概括为神经网络加速器的开发。

第三类研究受神经系统结构启发，参考神经系统的结构来开发新形态的硬件系统。这类系统参考生物的神经系统来组织自身结构，并利用数字电路或模拟电路来实现神经系统中的组成成分，如神经元、突触和灰质等。这类研究与第一类研究不同，因为其目的并不是为了模拟神经系统，而是希望从神经科学中得到启发来构建低功耗高性能的新型体系结构；同时，这类研究也不同于第二类，因为这类研究并没有针对特定的一种或一类神经网络模型去进行设计。这类研究可以概括为新型神经形态硬件的开发。

1. 三种分类的研究现状
2. 神经模拟系统的构建

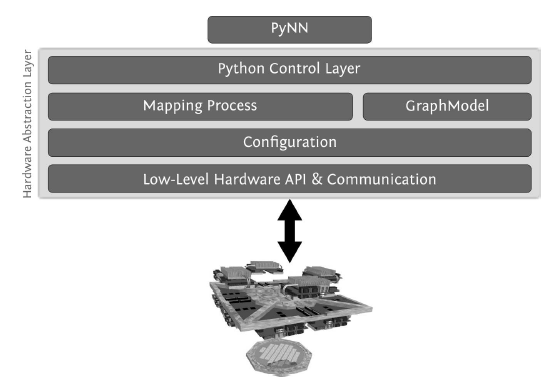
在神经模拟系统构建方面，一个突出的成果是欧盟的BrainScaleS计划。欧盟在2011年启动了BrainScaleS计划，共有来自10个欧洲国家的19个研究机构参加，其目标是理解和模拟脑信息处理中多个时空尺度的功能和相互作用。BrainScaleS在2015年结束，此后BrainScaleS的成果进入了欧盟2013年启动的“人类脑计划”（Human Brain Project, HBP）中神经形态计算部分并继续延续下去。

BrainScaleS是2005年欧盟FACETS（Fast Analog Computing with Emergent Transient States）项目的延续，FACETS项目由海德堡大学牵头研制基于模拟混合信号的神经形态芯片。BrainScaleS将FACETS中产生的HICANN（High Input Count Analog Neural Network）芯片作为基本硬件组成，在未切割的晶圆上构建整个硬件系统[24]。



图：BrainScaleS中使用的晶圆级硬件系统：晶圆（A）上包括了HICANN单元、晶圆内通信结构、支架（B）、顶层框架（C）以及晶圆间/晶圆-主机通信模块(D)[24]

BrainScaleS中使用PyNN来描述脉冲神经元模型。PyNN是一款基于Python实现，独立于模拟器的神经元模型描述语言，提供了细胞模型模板并支持国际标准单位制，其底层的模拟器选择多样，支持的模拟器包括NEURON、NEST、Brian等常见的神经元模拟器。为了支持PyNN，BrainScaleS建立了一套完整的软件框架来将PyNN中的元素映射到底层的硬件实现上。

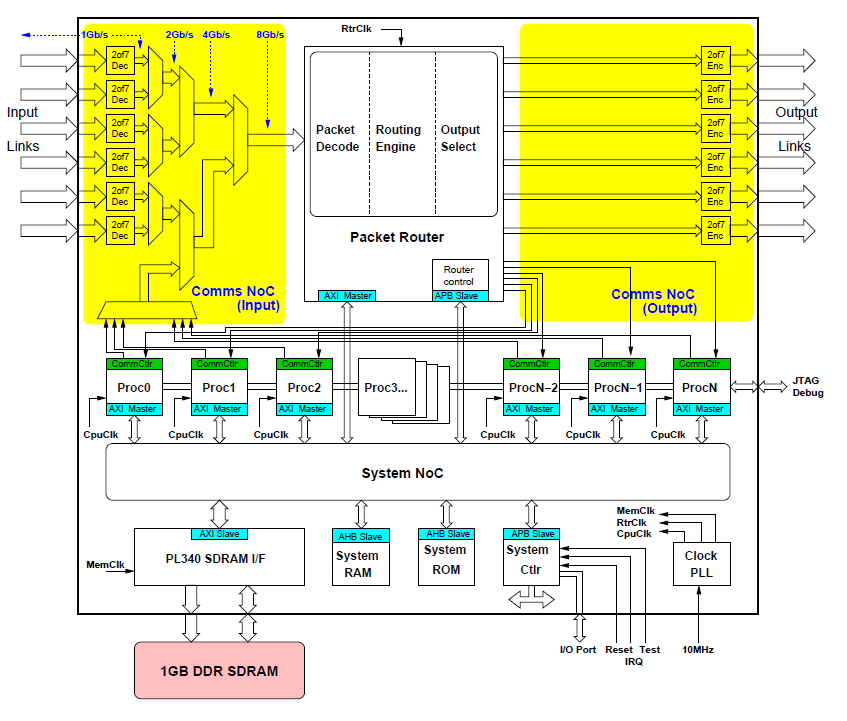


图：BrainScaleS的软件结构

BrainScaleS的系统中包括了5000万个突触和约23万个可并行工作的神经元电路。2016年3月，HBP宣布将BrainScaleS系统通过互联网对外开放使用，以支持神经微回路模拟以及在机器学习和认知计算中应用类脑原理的相关研究。

神经模拟系统中的另一个代表性成果是曼彻斯特大学开发的SpiNNaker系统，该系统于2016年与BrainScaleS一同通过互联网对外开放使用。与BrainScaleS中使用专门设计的模拟电路来模拟神经元不同，SpiNNaker中以通用处理器作为基本构成单元，完全使用软件来对神经元进行模拟，这是SpiNNaker的突出特点。在全规模系统中，SpiNNaker共使用了1036800个处理器，峰值速度可以达到每秒228兆Dhrystone指令，而能量消耗可以控制在90kW以内。

SpiNNaker项目的目标是制造通用的神经形态硬件，并能够满足大规模模拟系统在神经元状态处理、存储、扩展性、能效性、容错性等多方面的挑战。SpiNNaker系统的主要功能是实时地模拟大规模脉冲神经网络，同时也能够对非脉冲神经元模型提供支持，例如SpiNNaker可以模拟多层感知机结构。



图：SpiNNaker芯片结构

SpiNNaker的芯片结构如上图，每个芯片中最多包含20个ARM968处理器作为运算单元，即图中的Proc0到ProcN，其中一个处理器被选为监视核，担任系统管理任务。在监视核以外，每个核上运行对一组神经元的模拟程序，这一组神经元互相之间联系紧密，称为一个神经柱。监视用核的选择是通过竞争决定，以此来保证监视核能够正常工作。

每个ARM968处理器可以实时模拟至多1000个神经元。每个处理器核与两个紧耦合存储（Tightly Coupled Memory, TCM）直接相连，其中一个用于存储指令，大小32KB，从地址0x000000000开始编址；另一个TCM大小为64KB，从地址0x00400000开始编址，用于神经元模型相关的数据。除去TCM之外，每个SpiNNaker芯片与一个片外的SDRAM相连，这个SDRAM大小为1GB，用于保存突触相关信息。每个突触表项大小为4字节，包括突触延迟、突触后神经元的ID以及突触权重。

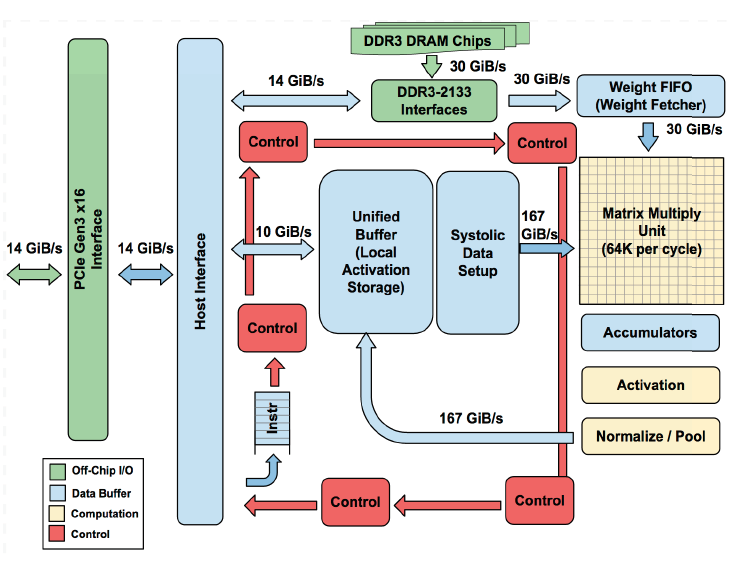
SpiNNaker的处理器核通过包交换网络来传递神经脉冲。在芯片内部，处理核之间通过片上网络进行信息交互；芯片间的交互通过片上路由器进行转发，即图中的Packet Router。每个SpiNNaker芯片与六个其他芯片相互连接，形成一个六边形的蜂窝结构。

1. 神经网络加速器的开发

神经网络加速器的开发大多由实际应用需要出发，因此由企业主导的研究较多。其中具有代表性的成果有谷歌公司开发的TPU。

TPU是谷歌公司专门为TensorFlow深度学习框架所设计的一款处理器[25]。在2006年时，谷歌曾经考虑过在自己的数据中心中为人工智能应用配备GPU、FPGA或定制ASIC，但是当时的分析显示使用数据中心富余的计算能力就足够支持当时仅有少量的人工智能应用。但是到了2013年，一项调查显示，即使用户每天只使用三分钟语音搜索，也会导致支持语音搜索服务的语音识别DNN需要的运算资源加倍。这个要求对于传统的CPU结构来说太过昂贵，因此谷歌决定开发自己的定制ASIC以支持人工智能应用。

TPU被设计成通过PCIe总线与CPU相连的一个协处理器。为了简化硬件设计与调试，TPU执行的指令由宿主服务器提供，而不是由TPU自行进行取指操作。从这个角度上，TPU比起GPU更接近于浮点处理单元FPU。下图是TPU的答题组成结构。



TPU指令通过PCIe Gen3 x16总线输入到指令缓冲区中，内部模块通过256字节宽的数据通路连接。TPU的核心部件是矩阵乘法单元，其中包含了256x256个MAC（Multiply-accumulator）部件，每个MAC可以处理8位的乘加操作，乘加的结果保存在累加器中。为了减少从Unified Buffer中读写操作的次数来减少能量消耗，矩阵乘法单元使用脉动阵列的方式进行矩阵运算。

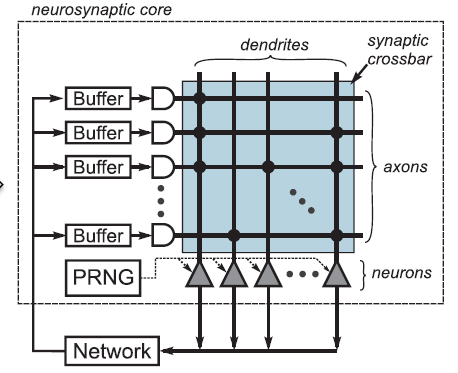
TPU的指令设计为CISC形式，具有5条关键指令，分别对应了神经网络应用中的关键操作：

* Read\_Host\_Memory将数据从宿主机内存中读取至Unified Buffer
* Read\_Weights将权重数据读取至图中的Weight FIFO，作为矩阵乘法单元的一个输入
* MatrixMultply/Convolve指示矩阵乘法单元开始进行乘法运算或卷积运算。
* Activate执行神经网络中的激活层的功能，输入为累加器中的乘法部分和，经过激活函数处理后将结果写入到Unified Buffer中，这一指令还可以用于完成池化操作。
* Write\_Host\_Memory将Unified Buffer中的数据写回到宿主机内存中。

中科院计算所

1. 新型神经形态硬件的开发

这类研究的一个代表是IBM在2014年发布的TrueNorth[26]。TrueNorth的结构受到人脑中皮质柱的结构启发。皮质柱是一小块高密度互联的神经元区域，这块区域中的神经元功能互相关联。TrueNorth模仿此结构，利用交叉开关的SRAM在一个核中实现了一小簇互相紧密连接的神经元，再将多个核互相连接起来形成TrueNorth芯片。



TrueNorth中一个核的结构大致如图中所示。图中的竖线表示神经元的树突，神经元由树突接收脉冲；横线表示神经元的轴突，由此发射脉冲；横线与竖线的交叉点代表了一个突触连接。在TrueNorth的一个这样的核中，包括了256个输入与256个输出，最多可以模拟256个神经元与256x256个突触连接。一个TrueNorth芯片中包含了4096个这样的处理核，总计可以模拟一百万神经元和256百万个突触连接，晶体管数达到54亿个。单个芯片平均放电频率为20Hz，单个神经元放电功耗可低至26pJ，功耗低至65mW，远远低于传统芯片。

TrueNorth能够取得极低的能耗，IBM也表示今后将多个TrueNorth芯片连接起来还可以提供更大的神经网络处理能力。但是也有一些对此的反对声音，例如LeCun在IBM公布TrueNorth时提出了对处理精度的质疑，他认为脉冲神经网络的输出是二值的，在处理需要更多位宽表示的数据时需要等待多个周期后脉冲的平均，这实际上拖慢了整个网络的运行效率，因此从性能功耗比的角度上讲TrueNorth并没有特别大的优势。

参考文献：

[1] Mead, Carver. "Neuromorphic electronic systems." Proceedings of the IEEE78.10: 1629–1636, 1990

[2] Rosenblatt, Frank, The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, Cornell Aeronautical Laboratory, Psychological Review, v65, No. 6, pp. 386–408.

[3] Hinton, G. E.; Osindero, S.; Teh, Y. "A fast learning algorithm for deep belief nets". Neural Computation. 18 (7): 1527–1554, 2006

[4] Ramón y Cajal, Santiago. Comparative study of the sensory areas of the human cortex. p. 85, 1899

[5] Hodgkin AL, Huxley AF "A quantitative description of membrane current and its application to conduction and excitation in nerve". The Journal of Physiology. 117 (4): 500–44, 1952

[6] Eugene M. Izhikevich. Dynamical Systems in Neuroscience: The

Geometry of Excitability and Bursting. The MIT Press, 2007.

[7] R. B. Stein. Some models of neuronal variability. Biophys., 7:37-68, 1967.

[8] Henry C. Tuckwell. Introduction to Theoretical Neurobiology. Cambridge University Press, 1988.

[9] E. M. Izhikevich. Simple model of spiking neurons. IEEE Trans. Neural Networks, 14(6):1569-1572, 2003.

[10] W. Bialek, F. Rieke, Rob R. De Ruyter Van Steveninck, and D. Warland. Reading a neural code. Science, 252:1854-1857, 1991.

[11] Werbos, P.J. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences, 1975.

[12] Yoshua Bengio, A Connectionist Approach To Speech Recognition Int. J. Patt. Recogn. Artif. Intell., 07, 647, 1993.

[13] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. Neural Comput. 9, 8 (November 1997), 1735-1780.

[14] Hubel, D. H.; Wiesel, T. N. "Receptive fields and functional architecture of monkey striate cortex". The Journal of Physiology. 195 (1): 215–243, 1968

[15] Fukushima, Kunihiko. "Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position". Biological Cybernetics. 36 (4): 193–202, 1980

[16] LeCun, Yann; Léon Bottou; Yoshua Bengio; Patrick Haffner. "Gradient-based learning applied to document recognition" (PDF). Proceedings of the IEEE. 86 (11): 2278–2324, 1998.

[17] Dave Steinkraus; Patrice Simard; Ian Buck. "Using GPUs for Machine Learning Algorithms". 12th International Conference on Document Analysis and Recognition (ICDAR 2005). pp. 1115–1119, 2005.

[18] Hopfield, J. J. Neural networks and physical systems with emergent collective computational properties. Proc. Nat. Acad. Sci. (USA) 79, 2554-2558, 1982.

[19] R. V. Florian, "The Chronotron: A neuron that learns to fire temporally precise spike patterns," PLoS One, 7(8): e40233, 2012．

[20] A. Mohemmed, S. Schliebs," SPAN: Spike pattern association neuron for learning spatio-temporal spike patterns," International Journal of Neural Systems, 22(4): 1250012, 2012．

[21] Horowitz M. Computing’s energy problem: (and what we can do about it). ISSCC invited talk, 2014.

[22] Thomas Potok, et al., Neuromorphic Computing: Architectures, Models, and Applications, DOE workshop report, 2016

[23] 欧盟人类大脑计划[EB/OL].[2016-07-25].www.humanbrainproject.eu/

[24] Brüderle, Daniel, et al. "A comprehensive workflow for general-purpose neural modeling with highly configurable neuromorphic hardware systems." Biological cybernetics 104.4-5 (2011): 263–296.

[25] Jouppi et al. "In-Datacenter Performance Analysis of a Tensor Processing Unit." In Proceedings of the 44th Annual International Symposium on Computer Architecture, 1-12. Toronto, ON, Canada: ACM. 2017.

[26] Merolla, Paul A., et al. "A million spiking-neuron integrated circuit with a scalable communication network and interface." Science 345.6197 (2014): 668–673.