LSTM、GRU与神经图灵机：详解深度学习最热门的循环神经网络

2017-07-08 机器之心

选自MachineLearningMastery

作者：Jason Brownlee

机器之心编译

参与：熊猫

> *循环神经网络是当前深度学习热潮中最重要和最核心的技术之一。近日，Jason Brownlee 通过一篇长文对循环神经网络进行了系统的介绍。机器之心对本文进行了编译介绍。*

循环神经网络（RNN/recurrent neural network）是一类人工神经网络，其可以通过为网络添加额外的权重来在网络图（network graph）中创建循环，以便维持一个内部状态。

为神经网络添加状态的好处是它们将能在序列预测问题中明确地学习和利用背景信息（context），这类问题包括带有顺序或时间组件的问题。

在这篇文章中，你将踏上了解用于深度学习的循环神经网络的旅程。

在读完这篇文章后，你将了解：

- 用于深度学习的顶级循环神经网络的工作方式，其中包括 LSTM、GRU 和 NTM。
- 顶级 RNN 与人工神经网络中更广泛的循环（recurrence）研究的相关性。
- RNN 研究如何在一系列高难度问题上实现了当前最佳的表现。

注意，我们并不会覆盖每一种可能的循环神经网络，而是会重点关注几种用于深度学习的循环神经网络（LSTM、GRU 和 NTM）以及用于理解它们的背景。

那就让我们开始吧！

概述

我们首先会设置循环神经网络领域的场景；然后，我们将深入了解用于深度学习的 LSTM、GRU 和 NTM；之后我们会花点时间介绍一些与用于深度学习的 RNN 相关的高级主题。

- 循环神经网络
  - 完全循环网络（Fully Recurrent Networks）
  - 递归神经网络（Recursive Neural Networks）
  - 神经历史压缩器（Neural History Compressor）

- 长短期记忆网络（LSTM）
- 门控循环单元（GRU）神经网络
- 神经图灵机（NTM）

循环神经网络

首先让我们设置场景。

人们普遍认为循环（recurrence）是给网络拓扑结构提供一个记忆（memory）。

一种更好的看法是训练集包含一种样本——其带有一组用于循环训练样本的输入。这是「传统的惯例」，比如传统的多层感知器

> X(i) -> y(i)

但是该训练样本得到了来自之前的样本的一组输入的补充。这是「非传统的」，比如循环神经网络

> [X(i-1), X(i)] -> y(i)

和所有的前馈网络范式一样，问题的关键是如何将输入层连接到输出层（包括反馈激活），然后训练该结构使其收敛。

现在，让我们来看看几种不同的循环神经网络，首先从非常简单的概念开始

完全循环网络

多层感知器的分类结构得到了保留，但该架构中的每个元素与其它每个元素之间都有一个加权的连接，并且还有一个与其自身的反馈连接。

并不是所有这些连接都会被训练，而且其误差导数的极端非线性意味着传统的反向传播无法起效，因此只能使用通过时间的反向传播（Backpropagation Through Time）方法或随机梯度下降（SGD）。

- 另外，可参阅 Bill Willson 的张量积网络（Tensor Product Networks）：http://www.cse.unsw.edu.au/~billw/cs9444/tensor-stuff/tensor-intro-04.html

递归神经网络

循环神经网络是递归网络的线性架构变体。

递归（recursion）可以促进分层特征空间中的分支，而且其所得到的网络架构可以在训练进行中模拟它。

其训练是通过子梯度方法（sub-gradient methods）使用随机梯度实现的。

R. Socher 等人 2011 年的论文《Parsing Natural Scenes and Natural Language with Recursive Neural Networks》中使用 R 语言对其进行了详细的描述，参阅：http://machinelearning.wustl.edu/mlpapers/paper_files/ICML2011Socher_125.pdf

神经历史压缩器

在 1991 年，Schmidhuber 首先报告了一种非常深度的学习器，其可以通过一种 RNN 层次的无监督预训练来在数百个神经层上执行功劳分配（credit assignment）。

每个 RNN 都是无监督训练的，可以预测下一个输入。然后只有产生错误的输入会被前馈，将新信息传送到该层次结构中的下一个 RNN，然后以更慢的、自组织的时间尺度进行处理。

事实表明不会有信息丢失，只是会有压缩。该 RNN stack 是数据的一个「深度生成模型（deep generative model）。这些数据可以根据其压缩形式重建。

- 参阅 J. Schmidhuber 等人 2014 年的论文《Deep Learning in Neural Networks: An Overview》：http://www2.econ.iastate.edu/tesfatsi/DeepLearningInNeuralNetworksOverview.JSchmidhuber2015.pdf

当误差被反向传播通过大型结构时，非线性导数的极限（extremity）的计算会增长，会使功劳分配（credit assignment）困难甚至是不可能，使得反向传播失败。

长短期记忆网络

使用传统的通过时间的反向传播（BPTT）或实时循环学习（RTTL/Real Time Recurrent Learning），在时间中反向流动的误差信号往往会爆炸（explode）或消失（vanish）

反向传播误差的时间演化指数式地依赖于权重的大小。权重爆炸可能会导致权重振荡，而权重消失则可能导致学习弥合时间滞后并耗费过多时间或根本不工作。

- LSTM 是一种全新的循环网络架构，可用一种合适的基于梯度的学习算法进行训练。
- LSTM 是为克服误差反向流动问题（error back-flow problem）而设计的。它可以学习桥接超过 1000 步的时间间隔。
- 在有噪声的、不可压缩的输入序列存在，而没有短时间滞后能力的损失时，这是真实的。

通过一种有效的基于梯度的算法，误差反向流动问题可以克服，因为该算法让网络架构可以强迫常量（因此不会有爆炸或消失）误差流过特殊单元的内部状态。这些单元可以减少「输入权重冲突（Input Weight Conflict）」和「输出权重冲突（Output Weight Conflict）」的影响。

输入权重冲突：如果输入是非零的，则同样的输入权重必须被同时用于存储特定的输入并忽略其它输入，那么这就将会经常收到有冲突的权重更新信号。

这些信号将会试图使该权重参与存储输入并保护该输入。这种冲突会使学习变得困难，并且需要一个对背景更敏感的机制来通过输入权重控制「写入操作（write operations）」。

输出权重冲突：只要一个单元的输出是非零的，那么这个单元的输出连接的权重就将吸引在序列处理过程中生成的有冲突的权重更新信号。

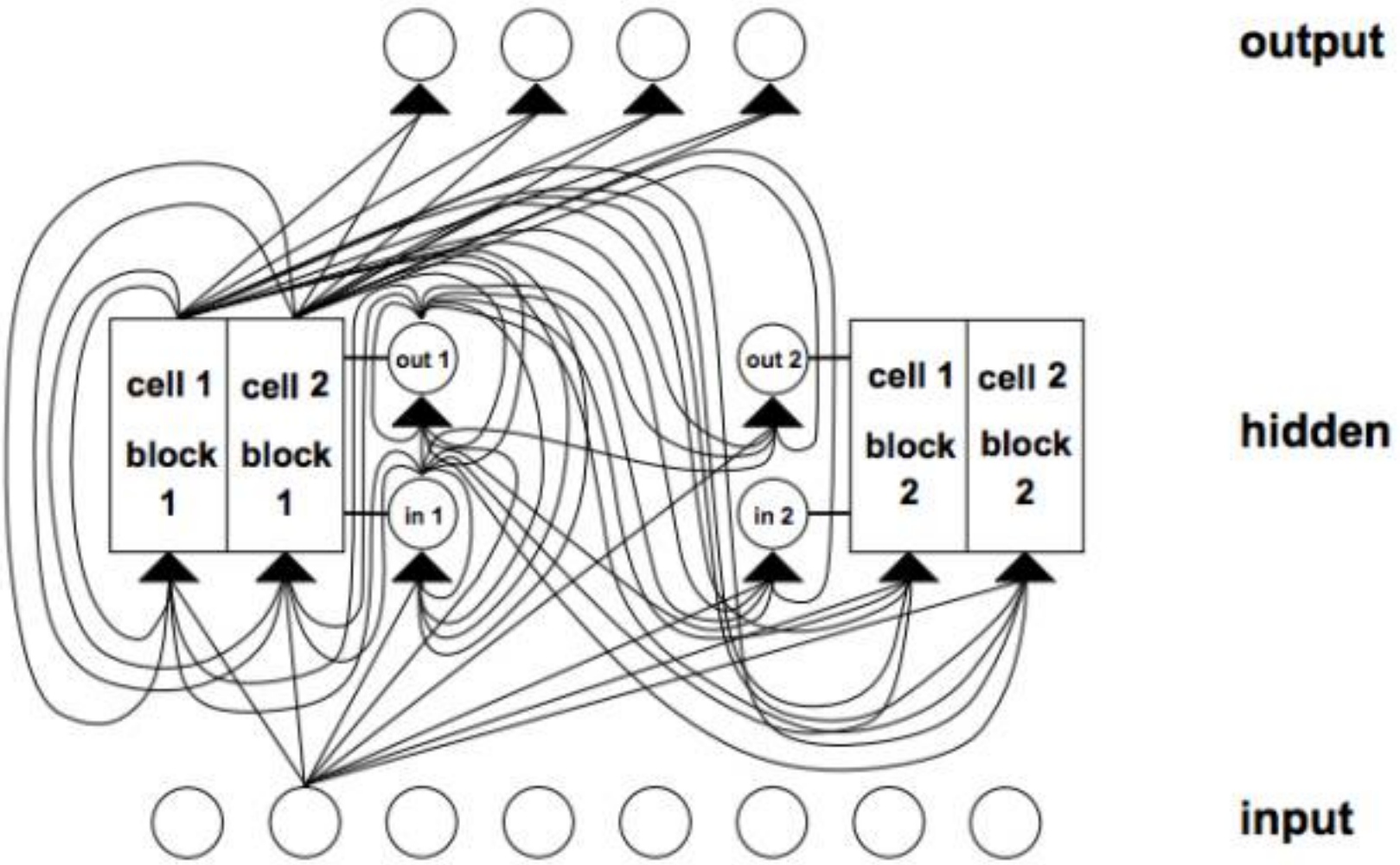
这些信号将试图使正在输出的权重参与进来，获取存在在处理单元中信息，并且在不同的时间保护后续的单元免受正被馈送的单元的输出的干扰。

这些冲突并不特定于长期滞后（long-term lags），并且也可以同样影响到短期滞后（short-term lags）。值得注意的是，随着滞后的增长，存储的信息必须被保护起来免受干扰，尤其是在学习的高级阶段。

网络架构：不同类型的单元都可能传递关于网络当前状态的有用信息。比如说，一个输入门（输出门）可能会使用来自其它记忆单元（memory cell）的输入来决定是否存储（读取）其记忆单元中的特定信息。

记忆单元包含门（gate）。门特定于它们调解的连接。输入门是为了纠正输入权重冲突，而输出门是为了消除输出权重冲突。

门：具体来说，为了缓解输入和输出权重冲突以及干扰，我们引入了一个乘法输入门单元来保护存储的记忆内容免受不相关输入的干扰，还引入了一个乘法输出门单元来保护其它单元免受存储中当前不相关记忆内容的干扰。



LSTM 架构示例。这个 LSTM 网络带有 8 个输入单元、4 个输出单元和 2 个大小为 2 的记忆单元模块。in1 是指输入门，out1 是指输出门，cell1 = block1 是指 block 1 的第一个记忆单元。来自 1997 年的《Long Short-Term Memory》

因为处理元素的多样性和反馈连接的，LSTM 中的连接比多层感知器的连接复杂。

记忆单元模块：记忆单元共享同一个输入门和同一个输出门，构成一种名叫记忆单元模块（memory cell block）的结构。

记忆单元模块有利于信息存储；就像传统的神经网络一样，在单个单元内编码一个分布式输入可不是一件容易的事情。一个大小为 1 的记忆单元模块就是一个简单的记忆单元。

学习（Learning）：一种考虑了由输入和输出门导致的修改过的、乘法动态的实时循环学习（RTRL/Real Time Recurrent Learning）的变体被用于确保通过记忆单元误差的内部状态反向传播到达「记忆单元网络输入（memory cell net inputs）」的非衰减误差（non-decaying error）不会在时间中被进一步反向传播。

猜测（Guessing）：这种随机方法可以超越许多时间滞后算法。事实已经说明，之前的工作中所使用的许多长时间滞后任务可以通过简单的随机权重猜测得到比提出的算法更快的解决。

- 参见 S. Hochreiter 和 J. Schmidhuber《Long-Short Term Memory》： http://dl.acm.org/citation.cfm?id=1246450

LSTM 循环神经网络最有意思的应用出现在语言处理领域。更全面的描述可参阅 Gers 的论文：

- F. Gers 和 J. Schmidhuber 2001 年的论文《LSTM Recurrent Networks Learn Simple Context Free and Context Sensitive Languages》： ftp://ftp.idsia.ch/pub/juergen/L-IEEE.pdf
- F. Gers 2001 年的博士论文《Long Short-Term Memory in Recurrent Neural Networks》： http://www.felixgers.de/papers/phd.pdf

LSTM 的局限性

LSTM 有效的截断版本无法轻松解决类似于「强延迟的异或（strongly delayed XOR）」这样的问题。

每个记忆单元模块都需要一个输入门和一个输出门。并不一定需要其它循环方法。

在记忆单元内穿过「常量误差传送带（Constant Error Carrousels）」的常量误差流可以得到与传统的前馈架构（会一次性获得整个输入串）一样的效果。

和其它前馈方法一样，LSTM 也有「regency」概念上的缺陷。如果需要精密的计数时间步骤，那么可能就需要额外的计数机制。

LSTM 的优点

该算法桥接长时间滞后的能力来自其架构的记忆单元中的常量误差反向传播。

LSTM 可以近似有噪声的问题域、分布式表征和连续值。

LSTM 可以很好地泛化其所考虑的问题域。这是很重要的，因为有的任务无法用已有的循环网络解决。

在问题域上对网络参数进行微调看起来是不必要的。

在每个权重和时间步的更新复杂度方面，LSTM 基本上就等于 BPTT。

LSTM 很强大，在机器翻译等领域实现了当前最佳的结果。

门控循环单元神经网络

门控循环单元神经网络已经在序列和时间数据上得到了成功的应用。

最适合语音识别、自然语言处理和机器翻译。与 LSTM 一起，它们在长序列问题领域表现优良。

门控（gating）被认为是在 LSTM 主题中，涉及到一个门控网络生成信号来控制当前输入和之前记忆发生作用的方式，以更新当前的激活，从而更新当前的网络状态。

门本身是自我加权的，会在整个学习阶段中根据一个算法有选择性地更新。

门网络会增加计算复杂度，从而会增加参数化（parameterization），进而引入额外的计算成本。

LSTM RNN 架构将简单 RNN 的计算用作内部记忆单元（状态）的中间候选项。门控循环单元（GRU）RNN 将 LSTM RNN 模型的门控信号减少到了 2 个。这两个门分别被称为更新门（update gate）和重置门（reset gate）。

GRU（和 LSTM）RNN 的门控机制和在参数化方面的简单 RNN 一样。对应这些门的权重也使用 BPTT 随机梯度下降来更新，因为其要试图最小化成本函数。

每个参数更新都将涉及到与整体网络的状态相关的信息。这可能会有不利影响。

门控的概念可使用三种新变体的门控机制来探索和扩展。

这三种门控变体为：GRU1（其中仅使用之前的隐藏状态和偏置来计算每个门——、GRU2（其中仅使用之前的隐藏状态来计算每个门—）和 GRU3（其中仅使用偏置来计算每个门）。我们可以观察到参数显著减少，其中 GRU3 的参数数量最小。

这三种变体和 GRU RNN 在手写数字的 MNIST 数据库和 IMDB 电影评论数据集上进行了基准测试。

从 MNIST 数据集生成了 2 个序列长度，而从 IMDB 数据集生成了 1 个序列长度。

这些门的主要驱动信号似乎是（循环）状态，因为其包含关于其它信号的基本信息。

随机梯度下降的使用隐含地携带了有关网络状态的信息。这可以解释仅在门信号中使用偏置的相对成功，因为其自适应更新携带了有关网络状态的信息。

门控变体可使用有限的拓扑结构评估来探索门控机制。

更多信息请参阅：

- R. Dey 和 F. M. Salem 2017 年的论文《Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks》：https://arxiv.org/ftp/arxiv/papers/1701/1701.05923.pdf
- J. Chung 等人 2014 年的论文《Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling》：https://pdfs.semanticscholar.org/2d9e/3f53fcdb548b0b3c4d4efb197f164fe0c381.pdf

神经图灵机

神经图灵机通过将神经网络与外部记忆资源耦合而对该神经网络的能力进行了延展——它们可以通过注意（attention）过程与外部记忆资源交互。参阅机器之心文章《神经图灵机深度讲解：从图灵机基本概念到可微分神经计算机》。

这种组合的系统类似于图灵机或冯·诺依曼结构，但它是端到端可微分的，使得其可以有效地使用梯度下降进行训练。

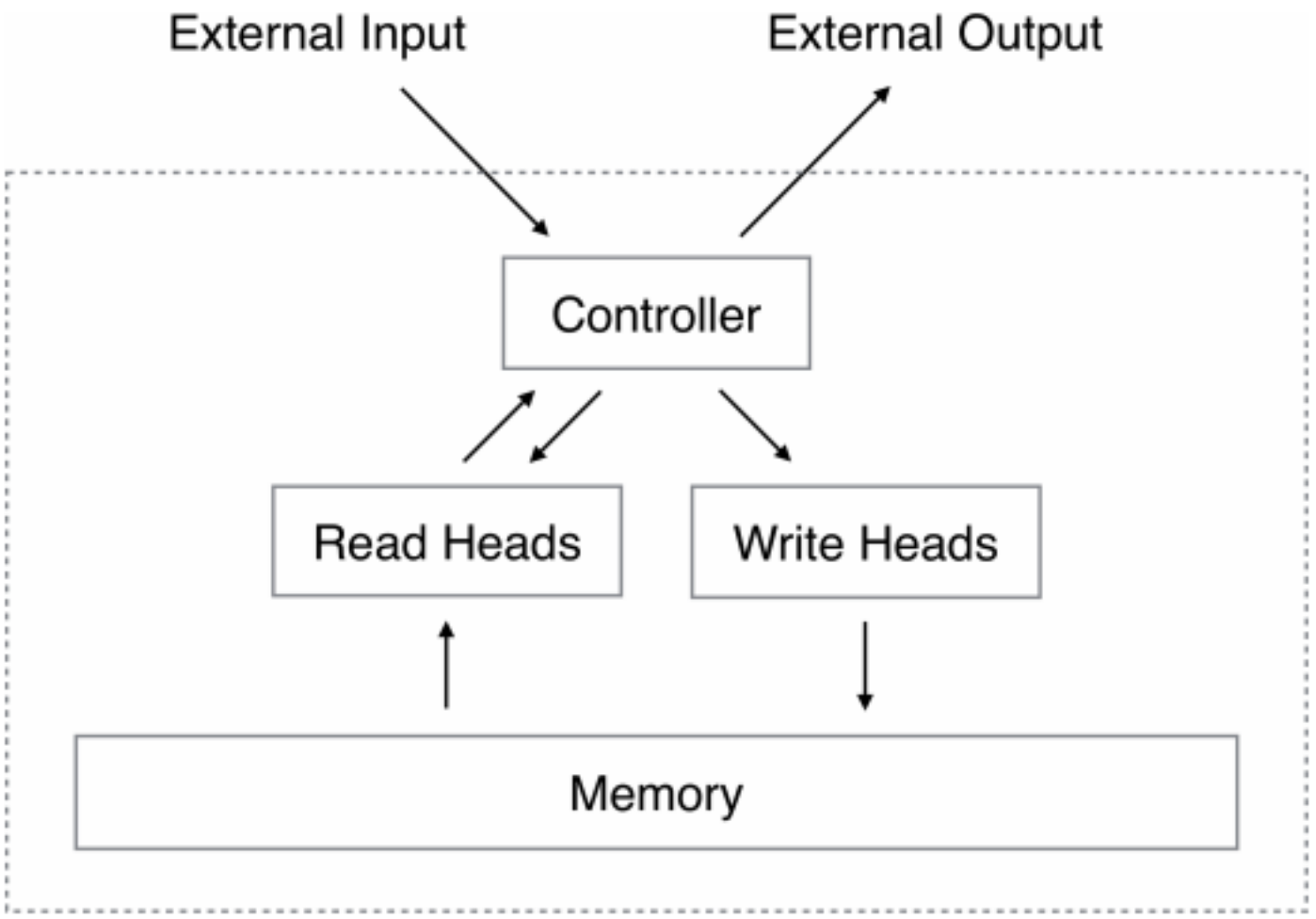
初步的结果表明神经图灵机可以根据输入和输出样本推理得到基本的算法，比如复制、排序和联想回忆（associative recall）。

RNN 相比于其它机器学习方法的突出之处在于其在长时间范围内学习和执行数据的复杂转换的能力。此外，我们都知道 RNN 是图灵完备的，因此其有能力模拟任意程序，只要连接方式合适即可。

标准 RNN 的能力被扩展以简化算法任务的解决方案。这种丰富性主要是通过一个巨大的可寻址的记忆实现的，

所以通过类比于图灵的通过有线存储磁带实现的有限状态机（finite-state machine）的丰富性，其被命名为神经图灵机（NTM）。

和图灵机不同，神经图灵机是一种可微分的计算机，可以通过梯度下降训练，从而为学习程序提供了一种实用的机制。



神经图灵机架构。NTM 架构大体上如上所示。在每一个更新循环中，控制器网络接收一个来自外部环境的输入并给出一个输出作为响应。它也会通过一系列并行的读写头来读写一个记忆矩阵（memory matrix）。虚线是 NTM 回路与外部世界的分界线。来自 2014 年的《Neural Turing Machines》

关键在于，该架构的每个组件都是可微分的，使其可以直接使用梯度下降进行训练。这可以通过定义「模糊（blurry）」的读写操作来实现，其可以或多或少地与记忆中的所有元素进行交互（而非像普通的图灵机或数字计算机那样处理单个元素）。

更多信息请参阅：

- A. Graves 等人 2014 年的《Neural Turing Machines》： https://arxiv.org/pdf/1410.5401.pdf
- R. Greve 等人 2016 年的《Evolving Neural Turing Machines for Reward-based Learning》：

http://sebastianrisi.com/wp-content/uploads/greve_gecco16.pdf

NTM 实验

复制（copy）任务可以测试 NTM 是否可以存储和回调长序列的任意信息。向该网络提供一个随机二进制向量的输入序列，后面跟着一个分隔符。

该网络被训练用来复制 8 位的随机向量序列，其中序列长度是在 1 到 20 之间随机的。目标序列就仅仅是输入序列的副本而已（没有分隔符）。

重复复制任务是复制任务的扩展，即要求该网络输出被复制的序列给定的次数，然后给出一个序列终止标志。这个任务的主要目的是看 NTM 是否可以学习简单的嵌套函数。

该网络的输入是随机长度的随机二进制向量序列，后面跟着一个标量值，表示我们想要的副本的数量，其出现在一个单独的输入信道上。

联想记忆任务（associative recall tasks）涉及到组织间接产生的数据，即当一个数据项指向另一个项的时候。要创建一个项列表，使得查询其中一个项时需要该网络返回后续的项。

我们定义了一个二进制向量序列，通过分隔符对其左右进行了限制。在几个项被传播到该网络中后，通过展示随机项对该网络进行查询，看该网络是否可以产生下一个项。

动态 N-gram 任务测试的是 NTM 是否可以通过使用记忆作为可覆写的表格来快速适应新的预测分布；该网络可以使用这个表格来持续对转换统计保持计数，从而模拟传统的 N-gram 模型。

考虑在二进制序列上的所有可能的 6-gram 分布的集合。每个 6-gram 分布都可以被表达成一个有 32 个数字的表格，其基于所有可能长度的 5 个二进制历史指定了下一位（bit）为 1 的概率。通过使用当前的查找表绘制 200 个连续的位，会生成一个特定的训练序列。该网络对该序列进行观察，一次一位，然后被要求预测出下一位。

优先级排序任务测试的是 NTM 的排序能力。该网络的输入是一个随机二进制向量的序列，以及每个向量的一个标量优先级评分。该优先级是在 [-1, 1] 范围内均匀分布的。目标序列是根据它们的优先级排序后的二进制向量序列。

NTM 有 LSTM 的前馈结构作为它们的组件之一。

总结

你通过这篇文章了解了用于深度学习的循环神经网络。具体来说，你了解了：

- 用于深度学习的顶级循环神经网络的工作方式，其中包括 LSTM、GRU 和 NTM。
- 顶级 RNN 与人工神经网络中更广泛的循环（recurrence）研究的相关性。
- RNN 研究如何在一系列高难度问题上实现了当前最佳的表现。

原文链接：http://machinelearningmastery.com/recurrent-neural-network-algorithms-for-deep-learning/

阅读原文