# 共享相关任务表征，一文读懂深度神经网络多任务学习

2017-06-23 机器之心

选自sebastianruder.com

**作者：Sebastian Ruder**

**机器之心编译**

**参与：Jane W、黄小天**

> *近日，自然语言处理方向博士生、AYLIEN 研究科学家 Sebastian Ruder 在其同名博客上发表了一篇长文，从多任务学习 MTL 的背景、现状、动机、方法、机制、实践等方面，全面而详实地对深度神经网络多任务学习（Multi-Task Learning in Deep Neural Networks）进行了深度介绍。机器之心对该文进行了编译，原文链接请见文末。*

## 目录

## 介绍

在机器学习（ML）中，通常的关注点是对特定度量进行优化，度量有很多种，例如特定基准或商业 KPI 的分数。为了做到这一点，我们通常训练一个模型或模型组合来执行目标任务。然后，我们微调这些模型，直到模型的结果不能继续优化。虽然通常可以通过这种方式使模型达到可接受的性能，但是由于我们的关注点集中在单个任务上，我们忽略了可能帮助优化度量指标的其它信息。具体来说，这些信息来自相关任务的训练信号。通过共享相关任务之间的表征，可以使我们的模型更好地概括原始任务。这种方法被称为多任务学习（MTL），这正是本文的主题。

MTL 有很多形式：联合学习（joint learning）、自主学习（learning to learn）和带有辅助任务的学习（learning with auxiliary task）等都可以指 MTL。一般来说，优化多个损失函数就等同于进行多任务学习（与单任务学习相反）。这些情况有助于你明确地思考如何在 MTL 方面做尝试并从中获得启发。

即使只优化一个损失函数（如在典型情况下），也有可能借助辅助任务来改善原任务模型。Rich Caruana [1] 简要总结了 MTL 的目标：「MTL 通过利用包含在相关任务训练信号中的特定领域的信息来改进泛化能力」。

在本文中，我将尝试概括一下多任务学习的现状，特别是当涉及到具有深度神经网络的 MTL 时。我将首先从不同的角度阐述 MTL 的动机。然后，我将介绍 MTL 在深度学习中最常用的两种方法。随后，我将描述 MTL 的机

制，并阐述为什么 MTL 在实践中效果良好。在研究更先进的基于神经网络的 MTL 方法之前，我将通过讨论 MTL 的文献来提供一些背景。然后，我将介绍一些最近提出的更强大的深度神经网络 MTL 方法。最后，我将讨论常用的辅助任务类型，并讨论什么是一个好的 MTL 辅助任务。

## 动机

多任务学习的动机有不同的方式：从生物学的角度，多任务学习可以看作是受到人类学习的启发。对于学习新任务，我们经常应用通过学习相关任务获得的知识。例如，宝宝首先学会识别面部，然后可以应用这些知识来识别其它对象。

从教学的角度，我们经常通过学习任务来获得必要的技能，以便掌握更复杂的技术。学习武术（比如柔道）的恰当方式也适用于学习编程。

以流行文化为例，一个例子是《空手道少年（1984）》（感谢 Margaret Mitchell 与 Adrian Benton 提供灵感）。在电影中，老师宫城先生教导了空手道孩子看起来是无关紧要的任务，如打地板和给车打蜡。事后看来，这些可以让他掌握与学习空手道相关的宝贵技巧。

最后，从机器学习的角度：我们可以将多任务学习看作归纳转移的一种形式。归纳传递可以通过引入归纳偏置（inductive bias）来帮助改进模型，这导致模型比其它模型更喜欢某些假设。例如，一种常见形式的归纳偏置是 L1 正则化，这导致偏好稀疏解。在 MTL 模型下，归纳偏置由辅助任务提供，这导致模型更喜欢假设而不是解释多个任务。正如我们将在下面看到的，这通常会导致更好的一般化解决方案。

## 两种深度学习 MTL 方法

到目前为止，我们只研究了 MTL 的理论动机。为了使 MTL 的思想更具体化，现在我们来看一下在深度神经网络中执行多任务学习的两种最常用的方法。在深度学习中，多任务学习通常通过隐藏层的 Hard 或 Soft 参数共享来完成。

### Hard 参数共享

共享 Hard 参数是神经网络 MTL 最常用的方法，可以追溯到 [2]。在实际应用中，通常通过在所有任务之间共享隐藏层，同时保留几个特定任务的输出层来实现。

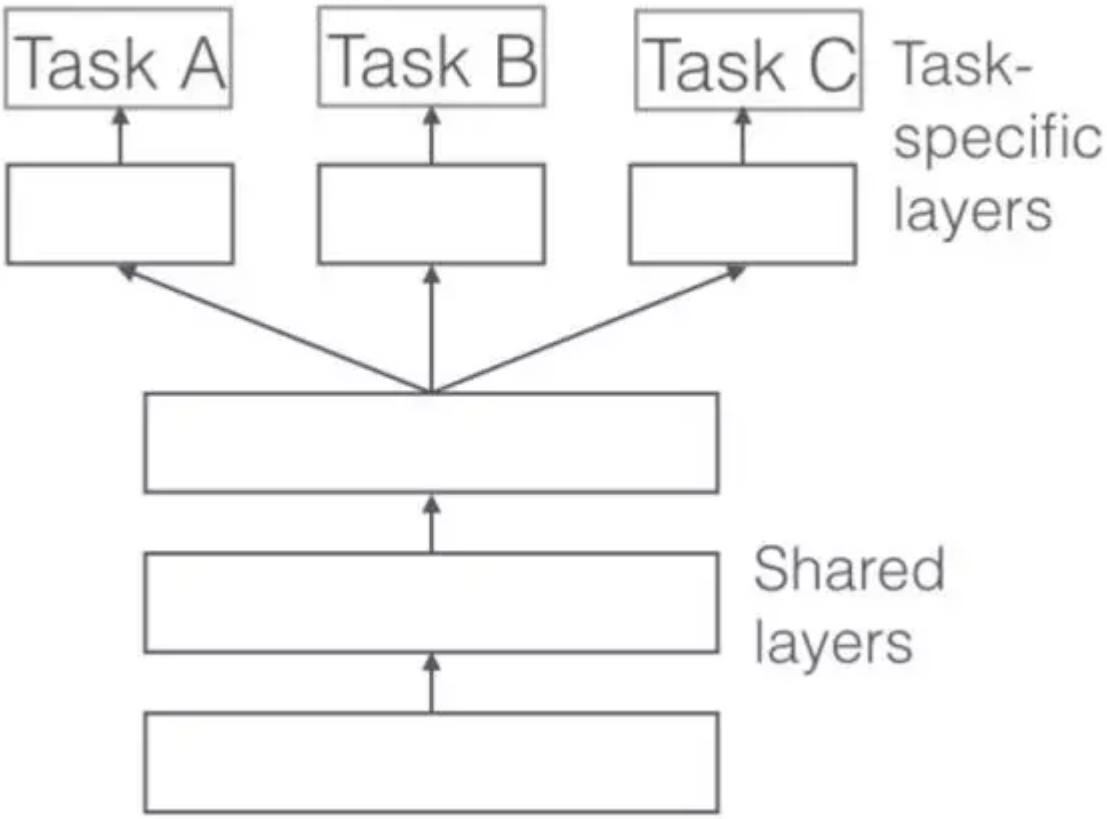


*图 1：深度神经网络多任务学习的 Hard 参数共享*

共享 Hard 参数大大降低了过拟合的风险。实际上，[3] 表明过拟合共享参数的风险为 O(N)——其中 N 是任务数——小于过拟合特定任务参数，即输出层。这很直观：我们同时学习的工作越多，我们的模型找到一个含有所有任务的表征就越困难，而过拟合我们原始任务的可能性就越小。

**Soft 参数共享**

另一方面，在共享 Soft 参数时，每个任务都有自己的参数和模型。模型参数之间的距离是正则化的，以便鼓励参数相似化。例如使用 L2 距离进行正则化 [4]，而 [5] 使用迹范数（trace norm）。

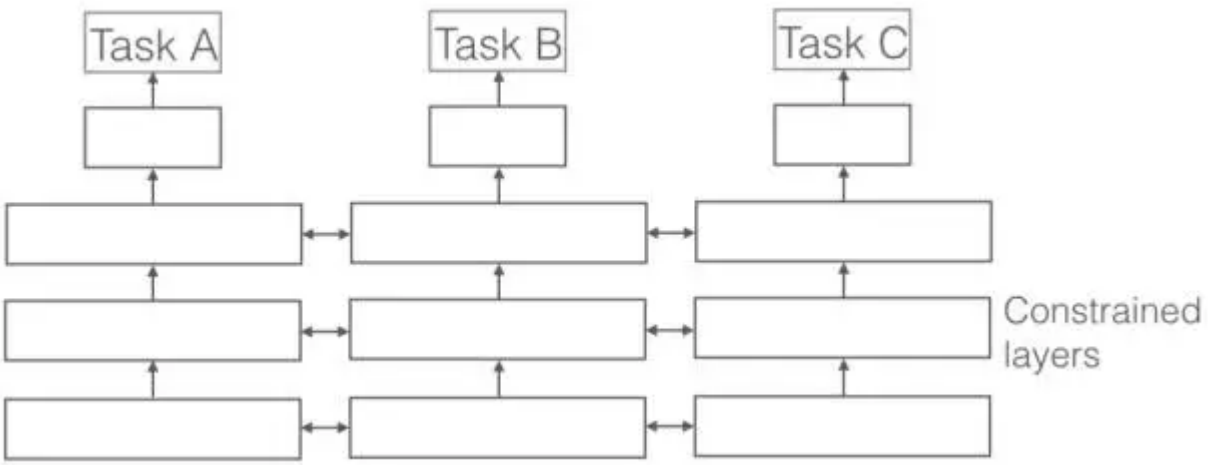

*图 2：深度神经网络多任务学习的 Soft 参数共享*

约束深度神经网络 Soft 参数共享的思想受到了 MTL 正则化技术的极大启发，这种思想已经用于其它模型开发，我们将在下面讨论。

## 为什么 MTL 有效?

即使多任务学习获得的归纳偏置看起来是可信的，为了更好地了解 MTL，我们仍需要阐述它所基于的机制。其中大部分最初都是由 Caruana（1998）提出的。对于所有例子，假设我们有两个相关的任务 A 和 B，它们依赖于一个共同的隐藏层表征 F。

### 隐式数据增加

MTL 有效地增加了我们用于训练模型的样本大小。由于所有任务不同程度地存在噪声，当在某些任务 A 上训练模型时，我们的目标是为任务 A 学习一个很好的表征，理想情况下，这个表征能忽略与数据相关的噪声并具有良好的泛化性。由于不同的任务具有不同的噪声模式，所以同时学习两个任务的模型能够学习更一般的表征。只学习任务 A 有可能过拟合任务 A，而联合地学习 A 和 B 使模型能够通过平均噪声模式获得更好的表征。

### 注意力机制

如果一个任务非常嘈杂或数据量有限并且高维，模型可能难以区分相关与不相关的特征。MTL 可以帮助模型将注意力集中在重要的特征上，因为其它任务将为这些特征的相关性或不相关性提供额外的证据。

### 窃听（eavesdroping）

某特征 G 很容易被任务 B 学习，但是难以被另一个任务 A 学习。这可能是因为 A 以更复杂的方式与特征进行交互，或者因为其它特征阻碍了模型学习 G 的能力。通过 MTL，我们可以允许模型「窃听」，即通过任务 B 学习 G。最简单的方法是通过提示（hint）[6]，即直接训练模型来预测最重要的特征。

### 表征偏置

MTL 任务偏好其它任务也偏好的表征，这造成模型偏差。这将有助于模型在将来泛化到新任务，因为在足够数量的训练任务上表现很好的假设空间也将很好地用于学习具有相同环境的新任务 [7]。

### 正则化

最后，MTL 通过引入归纳偏置作为正则化项。因此，它降低了过拟合的风险以及模型的 Rademacher 复杂度（即适合随机噪声的能力）。

## 非神经模型中的 MTL

为了更好地了解深度神经网络中的 MTL，我们将研究关于 MTL 在线性模型、核函数方法和贝叶斯算法方面的论文。特别地，我们将讨论一直以来在多任务学习的历史中普遍存在的两个主要思想：通过范数正则化制造各任务间的稀疏性；对任务间的关系进行建模。

请注意，许多 MTL 的论文具有同构性假设：它们假设所有任务与单个输出相关，例如，多类 MNIST 数据集通常被转换为 10 个二进制分类任务。最近的方法更加接近实际，对问题进行异构性假设，即每个任务对应于一组唯一的输出。

### 块稀疏（block-sparse）正则化

为了更好地与下面的方法衔接，我们首先介绍一些符号。我们有 T 个任务。对于每个任务 t，我们有一个模型 m_t，其参数 a_t 的维度是 d。我们可以将参数作为列向量写出 a_t=[a_{1,t} ... a_{d,t}]^⊤。我们现在逐列地将这些列向量 a_1, ... ,a_T 进行堆叠，形成矩阵 A∈ℝ^{d×T}。A 的第 i 行包含与每个任务的模型的第 i 个特征对应的参数 a_{i,.}，而 A 的第 j 列包含对应于第 j 个模型的参数 a_{.,j}。

许多现有的方法对模型的参数做了稀疏性假设。例如，假设所有模型共享一小组特征 [8]。对我们任务的参数矩阵 A 来说，这意味着除了几行之外，所有数据都是 0，对应于所有任务只共同使用几个特性。为了实现这一点，他们将 L1 范数推广到 MTL。回想一下，L1 范数是对参数之和的约束，这迫使除几个参数之外的所有参数都为 0。这也被称为 lasso（最小绝对收缩与选择算子）。

在单任务中，L1 范数根据相应任务 t 的参数向量 a_t 被计算，在 MTL 中，我们在任务参数矩阵 A 中计算它。为了做到这一点，我们首先对每行 a_i 计算出包含与第 i 个特征相对应的参数的 Lq 范数，其产生向量 b=[‖a_1‖_q ...‖a_d‖_q]∈ℝ^d。然后，我们计算该向量的 L1 范数，这迫使 b（即 A 矩阵的行）中除少数元素（entry）外，所有元素都为 0。

可以看到，根据我们希望对每一行的约束，我们可以使用不同的 Lq。一般来说，我们将这些混合范数约束称为 L1/Lq 范数，也被称为块稀疏正则化，因为它们导致 A 的整行被设置为 0。[9] 使用 L1/L∞ 正则化，而 Argyriou 等人（2007）使用混合的 L1/L2 范数。后者也被称为组合 lasso（group lasso），最早由 [10] 提出。

Argyriou 等人（2007）也表明，优化非凸组合 lasso 的问题可以通过惩罚 A 的迹范数（trace norm）来转化成凸问题，这迫使 A 变成低秩（low-rank），从而约束列参数向量 a_{.,1}，...，a_{.,t} 在低维子空间中。[11] 进一步

使用组合 lasso 在多任务学习中建立上限。

尽管这种块稀疏正则化直观上似乎是可信的，但它非常依赖于任务间共享特征的程度。[12] 显示，如果特征不重叠太多，则 Ll/Lq 正则化可能实际上会比元素一般（element-wise）的 L1 正则化更差。

因此，[13] 通过提出一种组合了块稀疏和元素一般的稀疏（element-wise sparse）正则化的方法来改进块稀疏模型。他们将任务参数矩阵 A 分解为两个矩阵 B 和 S，其中 A=B+S。然后使用 L1/L∞ 正则化强制 B 为块稀疏，而使用 lasso 使 S 成为元素一般的稀疏。最近，[14] 提出了组合稀疏正则化的分布式版本。

**学习任务的关系**

尽管组合稀疏约束迫使我们的模型仅考虑几个特征，但这些特征大部分用于所有任务。所有之前的方法都基于假设：多任务学习的任务是紧密相关的。但是，不可能每个任务都与所有任务紧密相关。在这些情况下，与无关任务共享信息可能会伤害模型的性能，这种现象称为负迁移（negative transfer）。

与稀疏性不同，我们希望利用先验信息，指出相关任务和不相关任务。在这种情况下，一个能迫使任务聚类的约束可能更适合。[15] 建议通过惩罚任务列向量 a_{.,1}，...，a_{.,t} 的范数与它们具有以下约束形式的方差来强加聚类约束：

$$\Omega = \|\bar{a}\|^2 + \frac{\lambda}{T}\sum_{t=1}^{T}\|a_{.,t} - \bar{a}\|^2$$

其中

$$\bar{a} = (\sum_{t=1}^{T} a_{.,t})/T$$

为参数向量的均值。这个惩罚项强制将任务参数向量 a_{.,1}，...，a_{.,t} 向由 λ 控制的均值聚类。他们将此约束应用于核函数方法，但这同样适用于线性模型。

[16] 也提出了对于 SVM 的类似约束。这个约束受到贝叶斯方法的启发，并试图使所有模型接近平均模型。由于损失函数的平衡制约，使每个 SVM 模型的间隔（margin）扩大并产生类似于平均模型的结果。

[17] 在聚类的数量 C 已知的假设下，通过形式化对 A 的约束，使聚类正则化更加明确。然后他们将惩罚项分解为 3 个独立的范数：

- 衡量列参数向量平均大小的全局惩罚项：

$$\Omega_{mean}(A) = \|\bar{a}\|^2.$$

- 衡量类间距离的类间方差（between-cluster variance）：

$$\Omega_{between}(A) = \sum_{c=1}^{C} T_c \|\bar{a}_c - \bar{a}\|^2$$

其中 T_c 是第 c 个类中任务的数量，$\bar{a}_c$是第 c 个类中任务参数向量的均值向量。

- 衡量类内数据紧密度的类内方差（within-cluster variance）：

$$\Omega_{within} = \sum_{c=1}^{C} \sum_{t \in J(c)} \|a_{\cdot,t} - \bar{a}_c\|$$

其中 J(c) 是第 c 个类中任务的集合。

最终的约束形式是这 3 个范数的加权和：

$$\Omega(A) = \lambda_1 \Omega_{mean}(A) + \lambda_2 \Omega_{between}(A) + \lambda_3 \Omega_{within}(A).$$

由于此约束假设聚类是已知的，所以它们引入了上述惩罚项的凸松弛（convex relaxation），这使算法允许同时学习聚类。

在另一种情况下，任务可能不在类结构中，但具有其它的结构。[18] 使用扩展组合 lasso 来处理树型结构（tree structure）中的任务，而 [19] 将其应用于具有图结构（graph structure）的任务。

虽然之前对任务之间关系建模的方法使用了范数正则化，但也存在没有用到正则化的方法：[20] 第一个提出了使用 k-nn 的任务聚类算法，而 [21] 通过半监督学习从多个相关任务中学习通用结构。

其它 MTL 任务之间关系的建模使用了贝叶斯方法：

- [22] 提出了使用模型参数先验的贝叶斯神经网络方法，来鼓励任务间使用相似的参数。[23] 将高斯过程（Gaussian process/GP）应用于 MTL，该方法利用 GP 推断共享协方差矩阵的参数。由于这在计算上非常昂贵，它们采用稀疏近似方案用来贪心地选择信息量最大的样本。[24] 同样将 GP 应用于 MTL，该方法利用 GP 假设所有模型抽样于同一个普通先验分布。
- [25] 在每个任务特定的层上使用高斯分布作为先验分布。为了鼓励不同任务之间的相似性，他们建议使用任务依赖的平均值，并引入使用混合分布的任务聚类。重要的是，它们首先需要确定聚类和混合分布数量的任务特征。

基于此，[26] 使用 Dirichlet process 提取分布，并使模型能够学习任务之间的相似性以及聚类的数量。然后，算法在同一个类中的所有任务间共享相同的模型。[27] 提出了一个分层贝叶斯模型，它学习一个潜在的任务层次结构，而 [28] 对于 MTL 使用基于 GP 的正则化，并扩展以前的基于 GP 的方法，以便在更复杂的设置中增加计算的可行性。

其它方法侧重于 on-line 多任务学习设置：[29] 将一些现有的方法，如 Evgeniou 等人的方法（2005）应用到 on-line 算法。他们还提出了一种使用正则化感知机（perceptron）的 MTL 扩展方法，该算法计算任务相关性矩阵。他们使用不同形式的正则化来偏置该任务相关性矩阵，例如，任务特征向量（characteristic vector）的接近程度或跨度子空间（spanned subspace）的维数。重要的是，与之前的方法类似，它们需要首先确定构成该矩阵的任务特征。[30] 通过学习任务关系矩阵来扩展之前的方法。

[31] 假设任务形成相互分隔的组，并且每个组中的任务位于低维子空间中。在每个组内，任务共享相同的特征表征，其参数与组分配矩阵（assignment matrix）一起使用代替的最小化方案学习。然而，组之间的完全分隔可能不是理想的方式，因为任务可能分享一些有助于预测的特征。

[32] 通过假设存在少量潜在的基础任务，反过来允许来自不同组的两个任务重叠。然后，他们将每个实际任务 t 的参数向量 a_t 建模为下面的线性组合：a_t=Ls_t，其中 L∈ℝ^{k×d} 是包含 k 个潜在任务的参数向量的矩阵，而 s_t∈ℝ^k 是包含线性组合系数的向量。此外，它们约束潜在任务中的线性组合为稀疏；约束在稀疏模式下两个任务之间的重叠然后控制它们之间的共享数量。最后，[33] 学习一个小的共享假设池，然后将每个任务映射到一个假设。

## 最近 MTL 的深度学习研究

虽然许多最近的深度学习方法已经将多任务学习（无论是显式使用或隐式使用）作为其模型的一部分（例子将在下一节中介绍），但是它们都采用了我们之前介绍的两种方法，Hard 和 Soft 参数共享。相比之下，只有少数论

文研究了如何在深度神经网络中开发更优的 MTL 算法。

### 深度关系网络

在用于计算机视觉的 MTL 中，通常的方法是共享卷积层，同时学习特定任务的全连接层。[34] 通过提出深度关系网络（Deep Relationship Network）来改进这些模型。除了图 3 中可以看到的共享和特定任务层的结构之外，他们在全连接层上使用矩阵先验（matrix priors），这样可以让模型学习任务之间的关系，类似于一些之前看过的贝叶斯模型。然而，这种方法仍然依赖于预定义的共享结构，这可能对于已经充分研究的计算机视觉问题是可行的，但是其证明对于新任务来说容易出错。

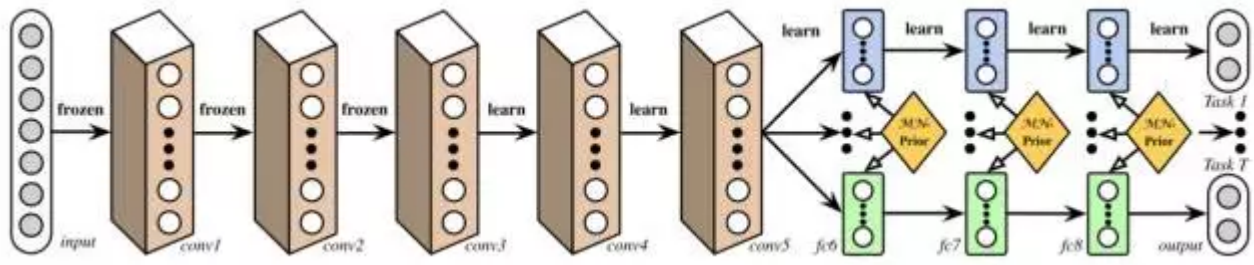


*图 3：具有共享卷积和特定任务的全连接层与矩阵先验的深度关系网络（Long 和 Wang，2015）。*

### 全自适应特征共享

从另一个极端，[35] 提出了一个从窄网络（thin network）开始的自下而上的方法，并在训练过程中使用一个促进类似任务分组的标准，贪婪地动态拓宽网络。动态创建分支的拓宽过程可以在图 4 中看到。但是，贪婪方法可能无法发现全局最优的模型，而将每个分支正好分配给一个任务不允许模型学习更复杂任务交互。

*图 4：完全自适应特征共享的拓宽过程（Lu 等人，2016）。*

### 十字绣网络

[36] 从两个独立的模型架构开始，如共享 Soft 参数一样。然后，他们使用称为十字绣（cross stitch）的单位，以允许模型通过学习前面层的输出的线性组合来确定如何使特定任务的网络利用其它任务的知识。图 5 为模型架

构，其中它们仅在池化（pooling）和全连接层之后使用十字绣单位。

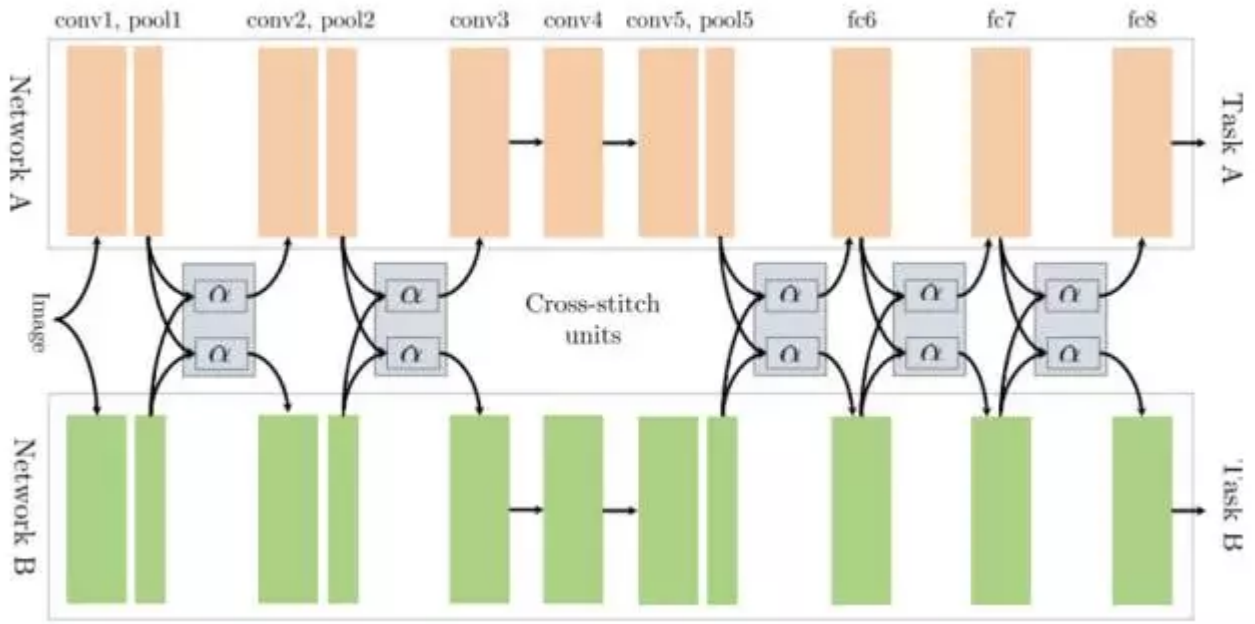


*图 5：两个任务的十字绣网络（Misra 等人，2016）。*

**低监督**

相比之下，在自然语言处理（NLP）中，最近的工作侧重于为多任务学习找到更好的任务层次：[37] 显示当低级任务用作辅助任务时应该在低层（lower layer）监督，如通常用于预处理的 NLP 任务（如词性标注/part-of-speech tagging 和命名实体识别/named entity recognition）。

**联合多任务模型**

基于这一发现，[38] 预定义了由几个 NLP 任务组成的层次结构，图 6 为联合多任务学习模型。

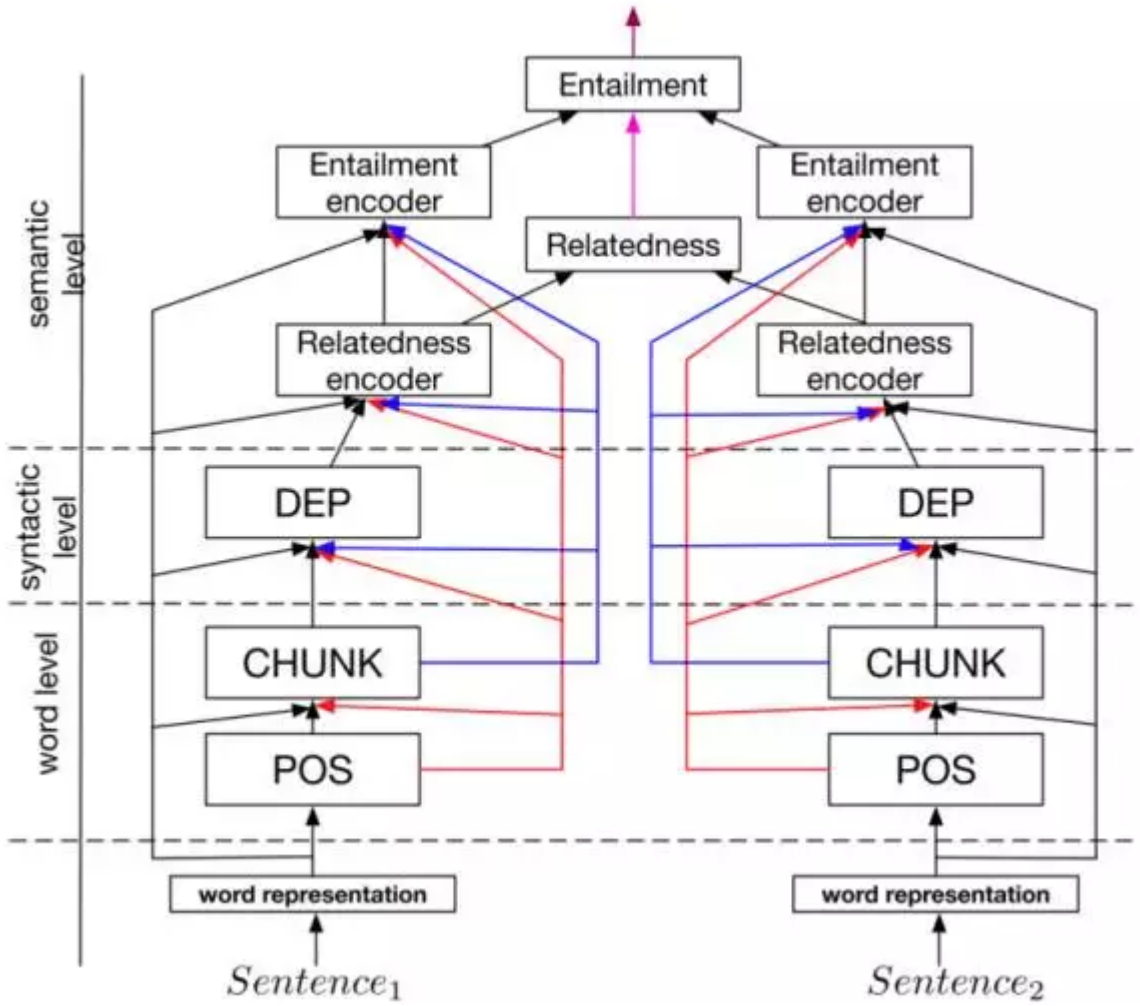


*图 6：联合多任务模型（Hashimoto 等人，2016）。*

**加权损失与不确定性**

与学习共享的结构不同，[39] 通过考虑每个任务的不确定性应用正交方法（orthogonal approach）。然后，他们通过基于最大化任务决定的不确定性的高斯似然估计，求导多任务损失函数，并以此来调整成本函数中的每个任务的相对权重。每一像素的深度回归（per-pixel depth regression）、语义和实例分割的架构可以在图 7 中看到。

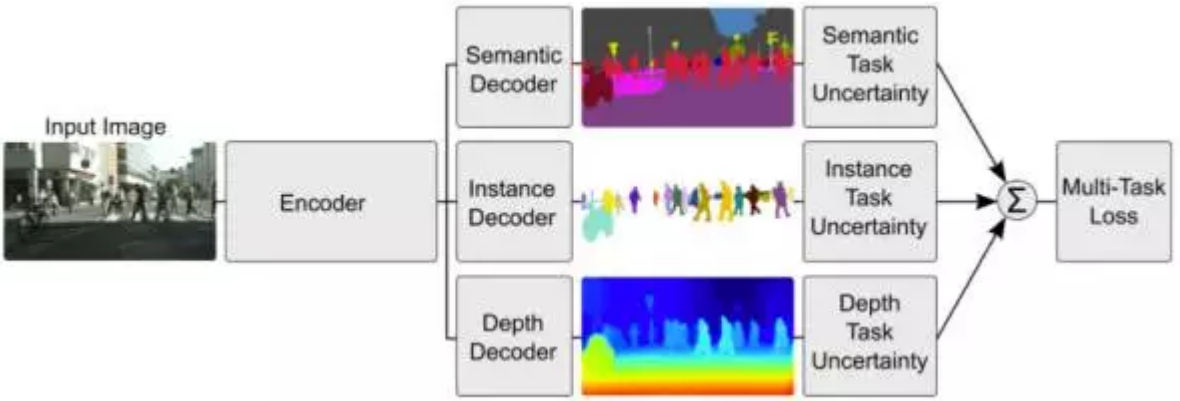

*图 7：用于多任务学习的基于不确定性的损失函数加权（Kendall 等人，2017）。*

**MTL 的张量因子分解**

最近的研究旨在将现有方法泛化到深度学习的 MTL：[40] 概括了一些之前讨论的矩阵因子分解的方法，通过使用张量因子分解将每层的模型参数分为共享和特定任务参数。

**水闸网络**

最后，我们提出了水闸网络（Sluice Network）[41]，一种泛化基于深度学习的 MTL 方法（比如 Hard 参数共享和十字绣网络、块稀疏正则化方法以及最近的任务层次结构的 NLP 方法）的模型，。图 8 为该模型，该模型可以学习哪些层和子空间应该共享，以及网络在哪层学到了输入序列的最佳表征。

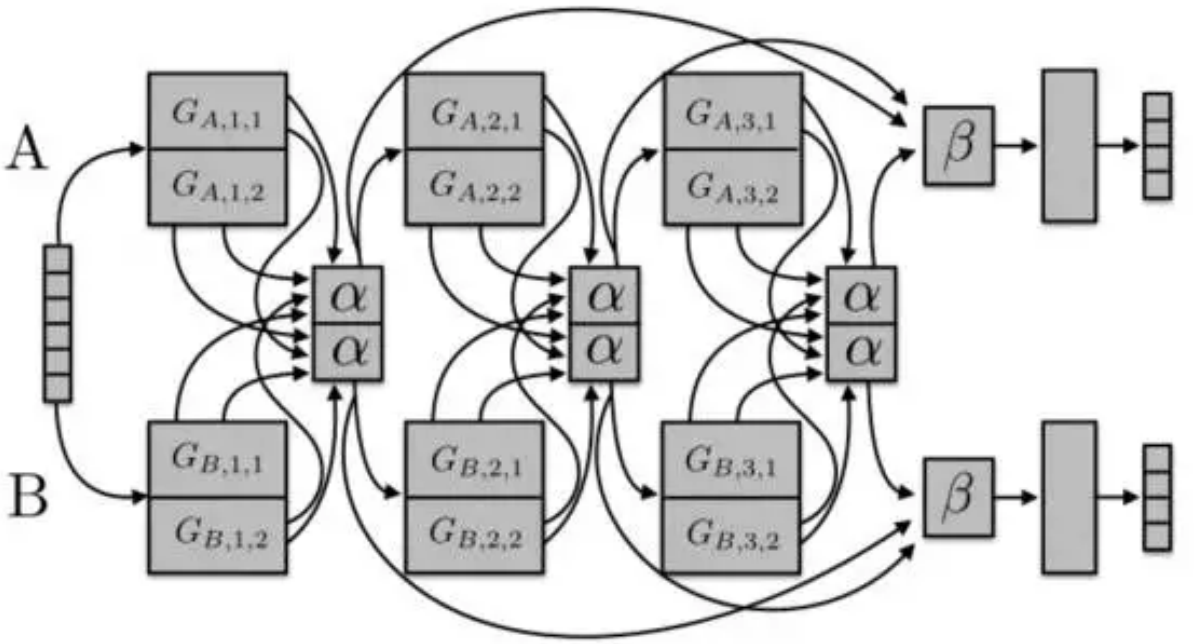


*图 8：两个任务的水闸网络（Ruder 等人，2017）。*

**模型应该共享什么？**

在对这些最近的方法进行研究之后，我们现在对深度 MTL 模型中怎样共享进行简要总结并得出结论。传统的大多数 MTL 方法都侧重于从相同分布中抽样任务（Baxter，1997）。虽然这种情况有利于共享，但并不总是如此。为了开发强大的 MTL 模型，我们必须能够处理无关或只松散相关的任务。

虽然 MTL 早期的深度学习研究已预先指定了为每个任务分配哪些层，但是这一策略并没有衡量反而严重偏差了 MTL 结构。在技术发明 20 年之后，Hard 参数共享（由 Caruana（1997）最初提出）仍然是标准。虽然在许多情况下很有用，但如果任务不紧密相关或需要不同层次的推理，Hard 参数共享就会快速失效。因此，最近的方

法正在考虑「学习」分享什么，并且这个方法通常优于 Hard 参数共享。此外，赋予我们的模型学习任务层次结构的能力是有帮助的，特别是在需要不同粒度的情况下。

正如最初提到的，一旦我们优化了多个损失函数，我们就会做 MTL。与限制模型将所有任务的信息压缩到相同的参数空间中不同，基于 MTL 的优势，我们上面讨论的 MTL 的先进算法是有用的，并使模型能够了解任务之间是如何交互的。

## 辅助任务

当我们希望一次获得多个任务的预测时，MTL 是非常适合的。这种情况在金融或经济预测中较为常见，我们可能希望预测多种相关指标的值；或者在生物信息学中同时预测多种疾病的症状。

然而在大多数情况下，我们只关心一项任务的效果。在本节中，我们将介绍在利用多任务学习时如何找到合适的辅助任务。

### 相关任务

经典的方法是使用相关任务作为 MTL 的辅助任务。为了了解相关任务是什么，我们将介绍一些典型的例子。Caruana（1997）使用多任务学习预测自动驾汽车的转向方向，并将预测道路的不同特征作为辅助任务；[42] 使用多任务学习进行面部特征点检测，并将头姿势估计和面部属性推断作为辅助任务；[43] 共同学习查询分类和网页搜索；[44] 共同预测图像中对象的类和坐标；最后，[45] 共同预测文本到语音的音素持续时间和频率分布（frequency profile）。

### 对抗性

通常，相关任务的标签数据不可用。然而，在某些情况下，我们想要实现的任务与可用的任务结果相反。可以使用对抗损失函数来利用这些数据，该损失函数不会使用梯度反转层（gradient reversal layer）来最小化训练误差，相反会最大化训练误差。这个设置发现最近在域适应方面取得了成功 [46]。在这种情况下的对抗任务是预测输入的域；通过反转对抗任务的梯度，使对抗任务的损失函数得到最大化，由于它迫使模型学习不能区分域的表征，这将有利于主任务。

### 提示

如前所述，MTL 可用于学习使用原任务不容易学习的特征。实现这一点的有效方法是使用提示（hint），即将特征预测为辅助任务。在自然语言处理方面，最近的应用这个方法的例子是 [47]，他们将预测一个输入句是否包含

一个正或负的情感词作为情感分析（sentiment analysis）的辅助任务，同时，[48] 预测语句中是否存在名字作为名字错误检测的辅助任务。

**注意力机制**

类似地，辅助任务可用于将注意力集中在网络可能通常忽略的部分图像上。例如，对于学习驾驶（Caruana，1997），单任务模型通常可能忽略车道标记，因为它们仅构成图像的一小部分，而且并不总在图中出现。然后，将车道标记预测作为辅助任务，强制模型学习它们的表征；这个信息也可以用于主任务。类似地，对于面部识别，人们可能会学习将预测面部特征点的位置作为辅助任务，因为它们通常是独特的。

**量化平滑**

对于许多任务，训练目标是量化的，即可用的标签是离散的，但是连续数值可能更合理。在许多情况下，需要人为评估收集的数据，例如预测疾病（低/中/高）或情感分析（积极/中性/消极）的风险。使用降低量化的辅助任务可能有助于这些情况，由于目标更平滑，它们可能更容易学习。

**预测输入**

在某些情况下，使用某些特征作为输入是不切实际的，因为它们对预测所需的目标无益。但是，它们仍然可以指导学习任务。在这些情况下，这些特征可以用作输出而不是输入。[49] 提出了这种方法适用的几种情况。

**用未来预测现在**

在许多情况下，某些特征只能在预测之后才能使用。例如，对于自动驾驶汽车，车辆通过后可以进行更准确的障碍物测量和车道标记。Caruana（1997）也给出了肺炎预测的例子，预测后的结果能够提供额外的医学试验结果。对于这些例子，附加数据不能用作特征，因为它不会在建模时作为输入使用。然而，它可以用作辅助任务，以便在训练期间向模型传入额外的信息。

**表征学习**

MTL 中辅助任务的目标是使模型能够学习对主任务有共享或有帮助的表征。迄今为止所讨论的所有辅助任务都是隐式的：它们与主任务密切相关，以便帮助模型学习有用的表征。更显式的建模是可能的，例如通过采用已知的任务使模型能够学习可迁移的表征。Cheng 等人（2015）和 [50] 采用语言模型的目标作为辅助任务。类似地，自编码器的目标也可以用于辅助任务。

**哪些辅助任务是有帮助的?**

在本节中，我们讨论了可用于 MTL 的不同辅助任务，即使我们只关心一个任务。然而，我们仍然不知道什么辅助任务在实际中是有用的。寻找辅助任务主要是基于一种假设，即认为辅助任务与主任务有某种相关性，并且有助于预测主任务。

然而，我们仍然不知道什么时候两个任务应该被认为是相似或相关的。Caruana（1997）定义如果两个任务使用相同的特征作判断，那么这两个任务是相似的。Baxter（2000）认为理论上相关的任务共享一个共同的最优假设类，即具有相同的归纳偏置。[50] 提出，如果两个任务的数据可以使用一个从一组分布变换 F 得到的固定概率分布生成，那么两个任务是 F-相关的。虽然这允许对不同传感器收集的相同分类问题的数据的任务进行推理，例如用不同角度和照明条件的相机得到的数据进行对象识别，这不适用于处理不同问题的任务。Xue 等人（2007）讨论，如果两个任务的分类边界即参数向量接近，则两个任务是相似的。

在理解任务相关性方面，尽管有这些早期的理论进展，但实践中还没有太多进展。任务相似度不是二进制的，而是在一个频谱范围内。MTL 中，更多的相似任务有更大的作用，而较少的相似任务相反。允许我们的模型学习如何分享每个任务，可能使我们能够暂时避开理论的缺失，并更好利用即使是松散相关的任务。然而，我们还需要制定一个有关任务相似性的原则概念，以便了解我们应该选择哪些任务。

最近的工作 [52] 发现了标签满足紧凑且均匀分布的辅助任务，这适用于 NLP 中的序列标签问题，并且已经在实验中（Ruder 等人，2017）得到证实。此外已经发现，主任务更有可能快速达到高峰平稳（plateau），而辅助任务不容易达到高峰平稳 [53]。

然而，这些实验迄今在范围上受到限制，最近的发现仅提供了加深对神经网络中多任务学习理解的启发式线索。

## 结论

在本篇概述中，我们回顾了多任务学习的发展历程，以及最近的深度学习 MTL 的研究。虽然对 MTL 的应用更加频繁，但是有 20 年历史的 Hard 参数共享模式仍然普遍存在于神经网络 MTL 中。然而，最新的基于让模型学习共享参数的方法的进展让我们看到了希望。同时，我们对任务的理解仍然有限（如，它们的相似性、关系、层次结构和 MTL 的用处），我们需要更多地了解它们，以便更好地了解 MTL 在深度神经网络方面的泛化能力。

### 参考文献

*1. Caruana, R. (1998). Multitask Learning. Autonomous Agents and Multi-Agent Systems, 27(1), 95–133. https://doi.org/10.1016/j.csl.2009.08.003*

*2. Caruana, R. "Multitask learning: A knowledge-based source of inductive bias." Proceedings of the Tenth International Conference on Machine Learning. 1993.*

*3. Baxter, J. (1997). A Bayesian/information theoretic model of learning to learn via multiple task sampling. Machine Learning, 28, 7–39. Retrieved from http://link.springer.com/article/10.1023/A:1007327622663*

*4. Duong, L., Cohn, T., Bird, S., & Cook, P. (2015). Low Resource Dependency Parsing: Cross-lingual Parameter Sharing in a Neural Network Parser. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), 845–850.*

*5. Yang, Y., & Hospedales, T. M. (2017). Trace Norm Regularised Deep Multi-Task Learning. In Workshop track - ICLR 2017. Retrieved from http://arxiv.org/abs/1606.04038*

*6. Abu-Mostafa, Y. S. (1990). Learning from hints in neural networks. Journal of Complexity, 6(2), 192–198. https://doi.org/10.1016/0885-064X(90)90006-Y*

*7. Baxter, J. (2000). A Model of Inductive Bias Learning. Journal of Artificial Intelligence Research, 12, 149–198.*

*8. Argyriou, A., & Pontil, M. (2007). Multi-Task Feature Learning. In Advances in Neural Information Processing Systems. http://doi.org/10.1007/s10994-007-5040-8*

*9. C.Zhang and J.Huang. Model selection consistency of the lasso selection in high-dimensional linear regression. Annals of Statistics, 36:1567–1594, 2008*

*10. Yuan, Ming, and Yi Lin. "Model selection and estimation in regression with grouped variables." Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68.1 (2006): 49-67.*

*11. Lounici, K., Pontil, M., Tsybakov, A. B., & van de Geer, S. (2009). Taking Advantage of Sparsity in Multi-Task Learning. Stat, (1). Retrieved from http://arxiv.org/pdf/0903.1468*

*12. Negahban, S., & Wainwright, M. J. (2008). Joint support recovery under high-dimensional scaling : Benefits and perils of ℓ1,∞ℓ1,∞-regularization. Advances in Neural Information Processing Systems, 1161–1168.*

*13. Jalali, A., Ravikumar, P., Sanghavi, S., & Ruan, C. (2010). A Dirty Model for Multi-task Learning. Advances in Neural Information Processing Systems. Retrieved from https://papers.nips.cc/paper/4125-a-dirty-model-for-multi-task-learning.pdf*

*14. Liu, S., Pan, S. J., & Ho, Q. (2016). Distributed Multi-task Relationship Learning. In Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (AISTATS) (pp. 751–760). Retrieved from http://arxiv.org/abs/1612.04022*

*15. Evgeniou, T., Micchelli, C., & Pontil, M. (2005). Learning multiple tasks with kernel methods. Journal of Machine Learning Research, 6, 615–637. Retrieved from http://discovery.ucl.ac.uk/13423/*

*16. Evgeniou, T., & Pontil, M. (2004). Regularized multi-task learning. International Conference on Knowledge Discovery and Data Mining, 109. https://doi.org/10.1145/1014052.1014067*

*17. Jacob, L., Vert, J., Bach, F. R., & Vert, J. (2009). Clustered Multi-Task Learning: A Convex Formulation. Advances in Neural Information Processing Systems 21, 745–752. Retrieved from http://eprints.pascal-network.org/archive/00004705/%5Cnhttp://papers.nips.cc/paper/3499-clustered-multi-task-learning-a-convex-formulation.pdf*

*18. Kim, S., & Xing, E. P. (2010). Tree-Guided Group Lasso for Multi-Task Regression with Structured Sparsity. 27th International Conference on Machine Learning, 1–14. https://doi.org/10.1214/12-AOAS549*

*19. Chen, X., Kim, S., Lin, Q., Carbonell, J. G., & Xing, E. P. (2010). Graph-Structured Multi-task Regression and an Efficient Optimization Method for General Fused Lasso, 1–21. https://doi.org/10.1146/annurev.arplant.56.032604.144204*

*20. Thrun, S., & O』Sullivan, J. (1996). Discovering Structure in Multiple Learning Tasks: The TC Algorithm. Proceedings of the Thirteenth International Conference on Machine Learning, 28(1), 5–5. Retrieved from http://scholar.google.com/scholar?cluster=9560540185077238 32&hl=en*

*21. Ando, R. K., & Tong, Z. (2005). A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data. Journal of Machine Learning Research, 6, 1817–1853.*

*22. Heskes, T. (2000). Empirical Bayes for Learning to Learn. Proceedings of the Seventeenth International Conference on Machine Learning, 367–364.*

*23. Lawrence, N. D., & Platt, J. C. (2004). Learning to learn with the informative vector machine. Twenty-First International Conference on Machine Learning - ICML 』04, 65. https://doi.org/10.1145/1015330.1015382*

*24. Yu, K., Tresp, V., & Schwaighofer, A. (2005). Learning Gaussian processes from multiple tasks. Proceedings of the International Conference on Machine Learning (ICML), 22, 1012–1019. https://doi.org/10.1145/1102351.1102479*

*25. Bakker, B., & Heskes, T. (2003). Task Clustering and Gating for Bayesian Multitask Learning. Journal of Machine Learning Research, 1(1), 83–99. https://doi.org/10.1162/153244304322765658*

*26. Xue, Y., Liao, X., Carin, L., & Krishnapuram, B. (2007). Multi-Task Learning for Classification with Dirichlet Process Priors. Journal of Machine Learning Research, 8, 35–63.*

*27. Daumé III, H. (2009). Bayesian multitask learning with latent hierarchies, 135–142. Retrieved from http://dl.acm.org.sci-hub.io/citation.cfm?id=1795131*

*28. Zhang, Y., & Yeung, D. (2010). A Convex Formulation for Learning Task Relationships in Multi-Task Learning. Uai, 733–442.*

*29. Cavallanti, G., Cesa-Bianchi, N., & Gentile, C. (2010). Linear Algorithms for Online Multitask Classification. Journal of Machine Learning Research, 11, 2901–2934.*

*30. Saha, A., Rai, P., Daumé, H., & Venkatasubramanian, S. (2011). Online learning of multiple tasks and their relationships. Journal of Machine Learning Research, 15, 643–651. Retrieved from http://www.scopus.com/inward/record.url?eid=2-s2.0-84862275213&partnerID=tZOtx3y1*

*31. Kang, Z., Grauman, K., & Sha, F. (2011). Learning with whom to share in multi-task feature learning. Proceedings of the 28th International Conference on Machine Learning, (4), 4–5. Retrieved from http://machinelearning.wustl.edu/mlpapers/paper*files/ICML2011Kang*344.pdf*

*32. Kumar, A., & Daumé III, H. (2012). Learning Task Grouping and Overlap in Multi-task Learning. Proceedings of the 29th International Conference on Machine Learning, 1383–1390.*

*33. Crammer, K., & Mansour, Y. (2012). Learning Multiple Tasks Using Shared Hypotheses. Neural Information Processing Systems (NIPS), 1484–1492*

*34. Long, M., & Wang, J. (2015). Learning Multiple Tasks with Deep Relationship Networks. arXiv Preprint arXiv:1506.02117. Retrieved from http://arxiv.org/abs/1506.02117*

*35. Lu, Y., Kumar, A., Zhai, S., Cheng, Y., Javidi, T., & Feris, R. (2016). Fully-adaptive Feature Sharing in Multi-Task Networks with Applications in Person Attribute Classification. Retrieved from http://arxiv.org/abs/1611.05377*

*36. Misra, I., Shrivastava, A., Gupta, A., & Hebert, M. (2016). Cross-stitch Networks for Multi-task Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/CVPR.2016.433*

*37. Søgaard, A., & Goldberg, Y. (2016). Deep multi-task learning with low level tasks supervised at lower layers. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 231–235.*

*38. Hashimoto, K., Xiong, C., Tsuruoka, Y., & Socher, R. (2016). A Joint Many-Task Model: Growing a Neural Network for Multiple NLP Tasks. arXiv Preprint arXiv:1611.01587. Retrieved from http://arxiv.org/abs/1611.01587*

*39. Kendall, A., Gal, Y., & Cipolla, R. (2017). Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. Retrieved from http://arxiv.org/abs/1705.07115*

*40. Yang, Y., & Hospedales, T. (2017). Deep Multi-task Representation Learning: A Tensor Factorisation Approach. In ICLR 2017. https://doi.org/10.1002/joe.20070*

*41. Ruder, S., Bingel, J., Augenstein, I., & Søgaard, A. (2017). Sluice networks: Learning what to share between loosely related tasks. Retrieved from http://arxiv.org/abs/1705.08142*

*42. Zhang, Z., Luo, P., Loy, C. C., & Tang, X. (2014). Facial Landmark Detection by Deep Multi-task Learning. In European Conference on Computer Vision (pp. 94–108). https://doi.org/10.1007/978-3-319-10599-4_7*

*43. Liu, X., Gao, J., He, X., Deng, L., Duh, K., & Wang, Y.-Y. (2015). Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval. Naacl-2015, 912–921.*

*44. Girshick, R. (2015). Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1440–1448). https://doi.org/10.1109/iccv.2015.169*

*45. Arık, S. Ö., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., … Shoeybi, M. (2017). Deep Voice: Real-time Neural Text-to-Speech. In ICML 2017.*

*46. Ganin, Y., & Lempitsky, V. (2015). Unsupervised Domain Adaptation by Backpropagation. In Proceedings of the 32nd International Conference on Machine Learning. (Vol. 37).*

*47. Yu, J., & Jiang, J. (2016). Learning Sentence Embeddings with Auxiliary Tasks for Cross-Domain Sentiment Classification. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP2016), 236–246. Retrieved from http://www.aclweb.org/anthology/D/D16/D16-1023.pdf*

*48. Cheng, H., Fang, H., & Ostendorf, M. (2015). Open-Domain Name Error Detection using a Multi-Task RNN. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 737–746).*

*49. Caruana, R., & Sa, V. R. de. (1997). Promoting poor features to supervisors: Some inputs work better as outputs. Advances in Neural Information Processing Systems 9: Proceedings of The 1996 Conference, 9, 389. Retrieved from http://scholar.google.com/scholar?start=20&q=author:%22Rich+Caruana%22&hl=en#6*

*50. Rei, M. (2017). Semi-supervised Multitask Learning for Sequence Labeling. In ACL 2017.*

*51. Ben-David, S., & Schuller, R. (2003). Exploiting task relatedness for multiple task learning. Learning Theory and Kernel Machines, 567–580. https://doi.org/10.1007/978-3-540-45167-9_41*

*52. Alonso, H. M., & Plank, B. (2017). When is multitask learning effective? Multitask learning for semantic sequence prediction under varying data conditions. In EACL. Retrieved from http://arxiv.org/abs/1612.02251*

*53. Bingel, J., & Søgaard, A. (2017). Identifying beneficial task relations for multi-task learning in deep neural networks. In EACL. Retrieved from http://arxiv.org/abs/1702.08303*

*原文链接：http://sebastianruder.com/multi-task/index.html*