时序组合的半监督学习

Samuli Laine Timo Aila

摘 要：在本文中，我们提出了一种简单有效的方法，用于在半监督环境中训练深度神经网络，其中只有一小部分训练数据被标记。我们引入自我集成，它使用不同时期网络训练的输出形成未知标签的共识预测，并且最重要的是，这是基于不同的正则化和输入增强条件下。对于未知的标签，这种组合式的预测可能比训练中网络最近的输出更加准确，因此可以用作训练目标。使用我们的方法，我们为两个标准的半监督学习基准设置了新的记录，将SVHN中的（非增加的）分类错误率从18.44％降低到7.05％（500个标签）和CIFAR-10中从18.63％降低到16.55％ （4000个标签），通过启用标准扩充，进一步达到5.12％和12.16％。通过使用来自Tiny Images数据集的随机图像作为训练期间未标记的额外输入，我们还获得了CIFAR-100分类准确度的明显改善。最后，我们展示了对不正确标签的良好耐受性。

# 介绍

人们早就知道，多个神经网络的集合通常比单个网络产生更好的预测。这种效果也被直接的应用在通过dropout,dropconnect或随机深度正则化方法等方法训练单个网络。在swapout网络中。训练总是专注于网络的特定子集，因此整个网络可以被视为这种训练的子网络的隐式组合。 我们通过在训练期间使用单个网络的输出在不同的训练时期和不同的正则化以及输入增强条件下形成集合预测来扩展这个想法。我们的训练仍然操作在单一网络上，但在不同的时期进行的预测对应于大量独立的子网络，因为dropout正则化。

该集合预测可以用于半监督学习，其中仅标记一小部分训练数据。如果我们将组合预测与正在训练的网络的当前输出进行比较，则组合预测可能更接近未标记输入的正确标签。因此，以这种方式推断的标签可以用作未标记输入的训练目标。我们的方法在很大程度上依赖于dropout正则化和多功能输入增强。实际上，如果没有这两种情况，那么对于未标记的训练数据所推断出的任何标签都存在信心的理由要少得多。

我们介绍了两种方法来实现自我组合：Π-模型和时序组合。这两种方法都在半监督学习中超过了先前最先进的结果。我们进一步观察到，自我组合也提高了完全标记情况下的分类准确性，并提供了对不正确标签的容忍度。

最近引入的Sajjadi等人的变换/稳定性损失基于与我们的工作相同的原理，并且Π-模型可以被视为它的特例。Π模型也可以看作是Rasmus等人对梯形网络的e-模型（一种先前提出的用于半监督学习的网络架构）的简化。我们的时序组合方法与Reed等人的自举方法有关，都是针对于训练中的噪音。

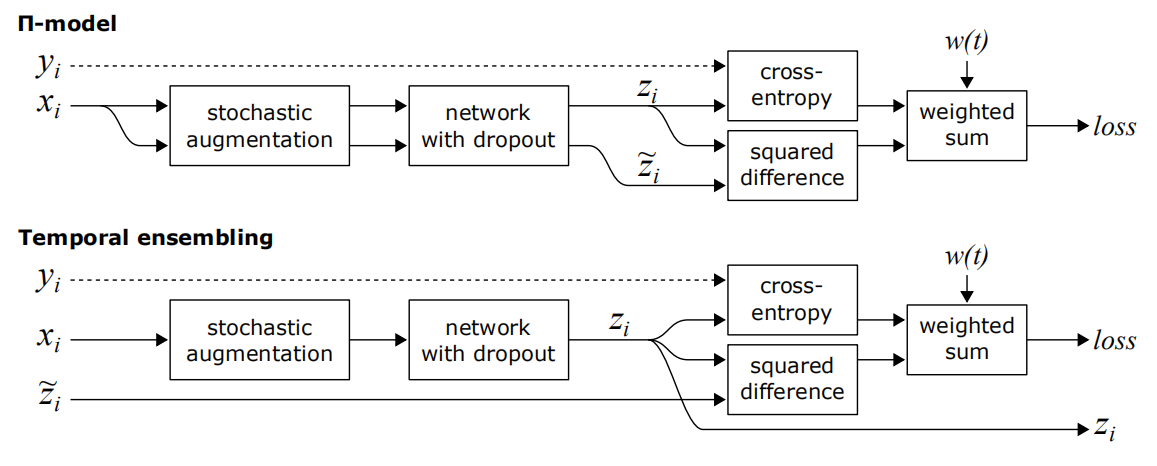
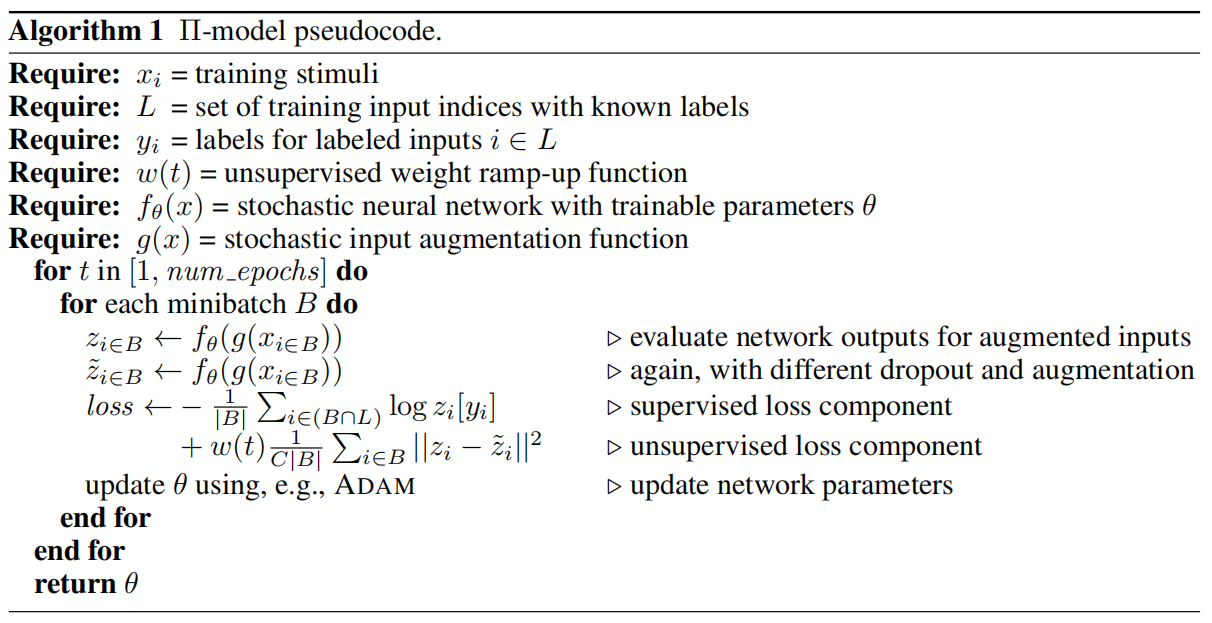


Figure 1：我们方法的训练结构：上面：Π-模型，下面：时序组合方法。标签yi仅适用于带标签的输入，并且仅针对那些评估相关的交叉熵损失组件。



# 训练中的自组合

我们在训练期间提出了两种自我组合的实现方式。第一个是Π-模型，在两个相同输入激活，不同dropout条件下，激励同一网络的输出。第二个是时序组合，通过考虑多个先前训练时期的网络预测来简化和扩展这一点。

我们将在传统图像分类网络的背景下描述我们的方法。训练数据由总共N个输入组成，其中M被标记。可用于所有训练数据的输入激活表示为xi，其中i∈{1......N}。设L包含标记输入的索引，| L | = M。对于每个i∈L，我们有一个已知的正确标签yi∈{1......C}，其中C是不同类的数量。

## Π-模型

Π模型的结构如图1（上图），算法的伪代码在算法1中。在训练期间，我们对训练的输入进行两次评估，得到预测向量zi和z〜i。我们的损失函数由两部分组成。第一个组成部分是标准交叉熵损失，仅针对标记输入进行评估。第二部分将评价所有的输入，通过使用均方误差对相同的训练输入的不同预测进行惩罚。在预测向量zi和z~i之间。为了组合有监督和无监督的损失项，我们通过时间依赖的加权函数w（t）来对后者进行规范化。通过比较整个输出向量zi和z~i，我们有效地要求两个评估之间的“黑暗知识”（Hinton等）接近，与要求只有最终分类保持不变相比，这是一个更强烈的要求，这是传统训练中发生的情况。

重要的是要注意，由于dropout正则化，训练期间的网络输出是随机变量。因此，在相同网络权重θ下对相同输入xi的两次评估产生不同的结果。此外，高斯噪声和增强诸如随机平移被评估两次，导致额外的变化。这些效果的组合解释了预测向量zi和zi之间的差异。考虑到原始输入xi是相同的，这种差异可以被视为分类中的误差，因此最小化它是合理的目标。

在我们的实现中，无监督损失加权函数w（t）在前80个训练期间沿着高斯曲线从零开始斜坡上升。有关此参数和其他培训参数的详细信息，请参阅附录A。开始时，总的损失和学习梯度受到标记数据部分支配，我们还发现无监督损失组成部分是足够慢的，因此网络很容易陷入局部解并且这种情况下无法获得有意义的分类数据。

我们的方法有点类似于Rasmus等人的梯形网络的e-模型,但在概念上更简单。在Π模型中，比较直接在网络输出上完成，在softmax激活之后，两个分支之间没有辅助映射，例如梯形网络体系结构中的学习去噪函数。此外，我们不是像在n模型中那样有一个“干净”和一个“损坏”分支，而是对两个分支的输入应用相同的增量和噪声。

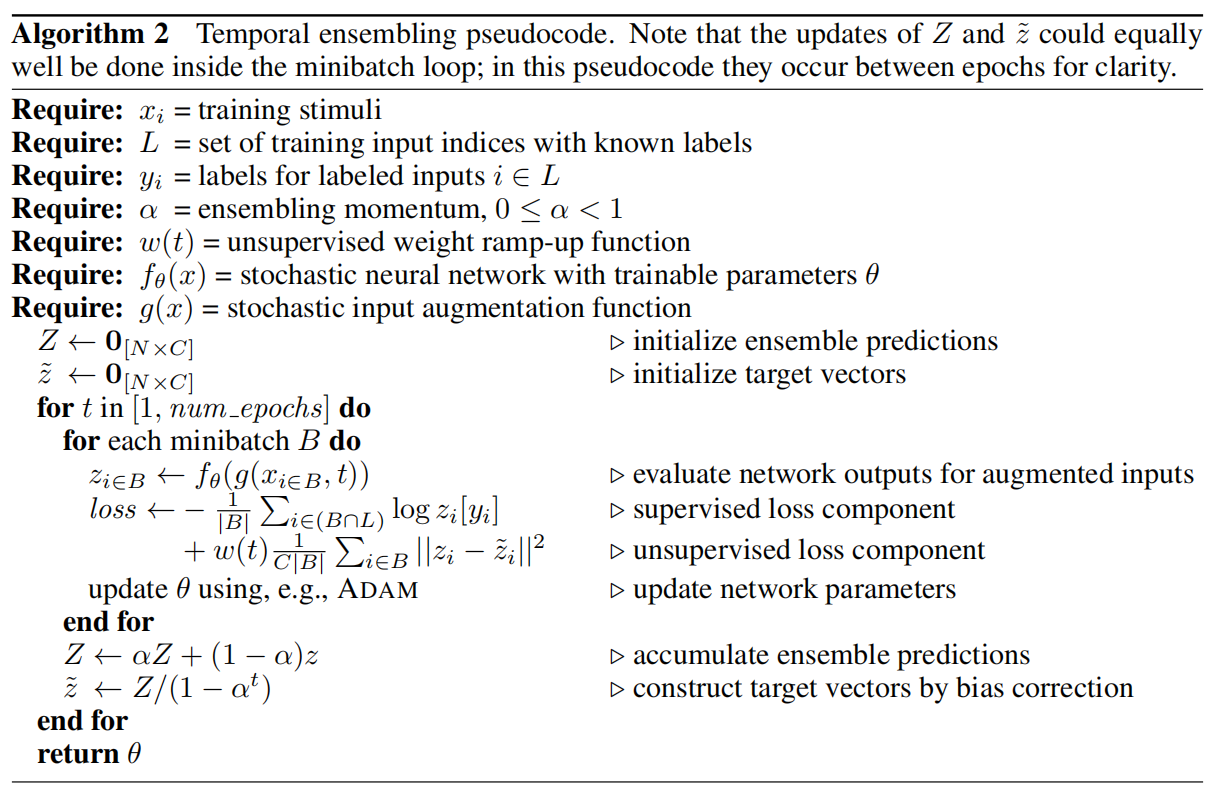
如第3节所示，Π-模型与良好的卷积网络结构相结合，在分类精度方面提供了优于现有技术的显着改进。

## 时序组合

分析Π模型的工作原理，我们同样可以在两个不同的阶段将两个分支的评估分开：首先对训练集进行一次分类而不更新权重θ，然后在不同的增强和dropout下用相同的输入进行网络训练，使用刚刚获得的预测作为无监督损失分量的目标。由于以这种方式获得的训练目标是基于对网络的单一评估，因此可以预期它们是嘈杂的。时序组合通过将多个先前网络评估的预测聚合到一个预测中来缓解这种情况。 它还允许我们在训练期间仅对网络进行一次评估，比Π模型获得大约2倍的加速。

我们时序组合方法的结构如图1（下面）所示，算法伪代码在算法2中。与Π模型的主要区别在于，网络和增强仅对每个时期的每个输入进行一次评估。并且无监督损失部分的目标向量z〜i是基于先前的网络评估而不是网络的第二评估。

在每个训练期之后，网络输出zi被积累到一个输出集Zi,并通过Zi←αZi+（1-α）zi更新Zi，其中α是控制集合到达训练历史快慢的动量项。由于dropout正则化和随机增强，因此Z包含来自先前训练时期的网络集合f的输出的加权平均值，其中最近的时期具有比较远时期更大的权重。为了生成训练目标z~，我们需要通过除以因子（1-αt）来校正Z中的启动偏差。类似的偏差校正已经用于例如Adam和仅平均批量标准化。在第一个训练时期，Z和z~为零，因为没有来自先前时期的数据可用。出于这个原因，我们在第一个训练时期指定无监督重量上升函数w（t）也为零。



与Π模型相比，时序组合的好处是双重的。首先，训练更快，因为每个时期的每个输入仅评估一次网络。其次，可以预期训练目标z~比使用Π模型的噪声更小。如第3节所示，在相同数量的训练时期中，我们确实获得了比使用Π模型更好的结果。与Π模型相比的缺点是需要跨时期存储辅助数据，以及新的超参数α。虽然当数据集包含大量项目和类别时矩阵Z可能相当大，但其元素相对不频繁地被访问。因此，它可以存储在例如存储器映射文件中。

时间集合的一个有趣的额外可能性是从网络预测zi收集除平均值之外的其他统计数据。例如，通过跟踪网络输出的第二个原始时刻，我们可以估计每个输出分量zij的方差。这使得有可能以原则的方式推断网络输出的不确定性。基于这些信息，我们可以例如在无监督损失项中对更确定的预测与更不确定的预测放置更多的权重。但是，我们将这些途径的探索作为未来的工作。

# 结果

我们的网络结构在表5中给出，测试设置和所有训练参数在附录A中详述。我们在两个图像分类任务CIFAR-10和SVHN中测试Π模型和时序组合，并使用不同的随机种子报告10次运行的平均值和标准差。

虽然很少明确说明，但我们认为我们的比较方法不使用输入增强，即仅限于丢失和其他形式的置换不变噪声。因此，除非另有明确说明，否则我们会报告错误率而不进行扩充。鉴于算法从增强中提取益处的能力也是一个重要属性，我们也使用一组标准增强来报告分类准确性。在纯监督训练中，增强CIFAR-10数据集的事实标准方法包括水平翻转和随机翻译，而SVHN仅限于随机翻译。通过使用这些相同的增强，我们也可以与最佳的完全监督结果进行比较。毕竟，完全监督的结果应该表明可获得的准确性的上限。

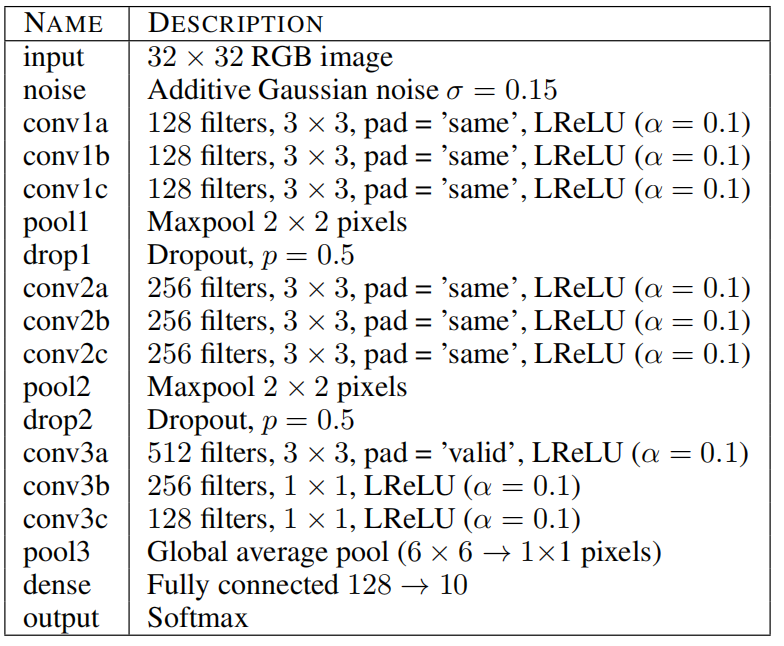


Table 5:所有测试中使用的网络结构

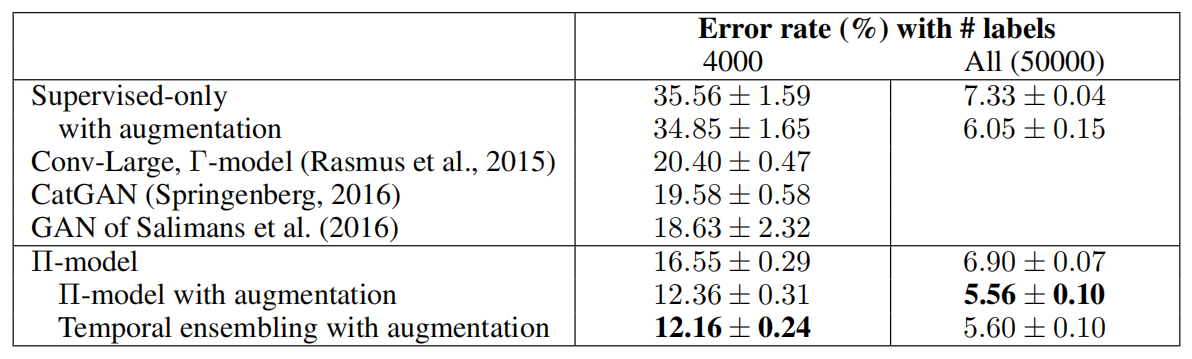


Table 1:CIFAR-10结果有4000个标签，平均10次运行（所有标签运行4次）。

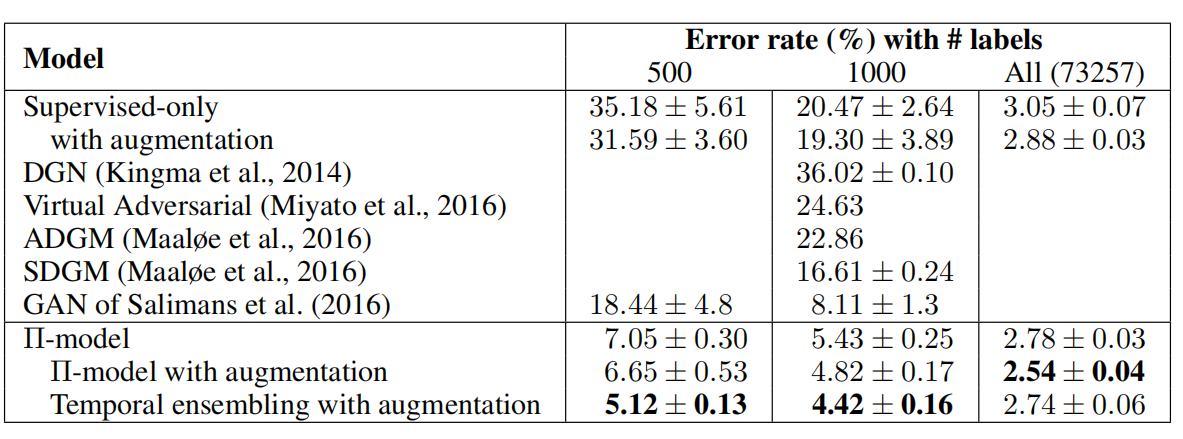


Table 2:SVHN结果为500和1000个标签，平均10次运行（所有标签运行4次）。

## CIFAR-10

CIFAR-10是一个由10个类别的32×32像素RGB图像组成的数据集。表1显示了与非增强Π模型的早期方法相比，具有4000个标签（每类400个）的分类错误率降低2.1个百分点。

启用标准增强集进一步将错误率降低4.2个百分点至12.36％。时序组合略微好一点在12.16％，而训练速度则快两倍。这个小的改进掩盖了一个微妙的事实，即随机水平翻转需要在时间集合中为每个时期独立完成，而Π模型可以每对评估一次随机化，根据我们的测量，它比独立翻转好大约0.5个百分点。

与Sajjadi等人同等比较有点困难，因为他们仅为一些相当极端的增强集（转移，翻转，旋转，拉伸和剪切）上的部分max pooling提供了结果，这引入了网络内部的随机局部拉伸，并且已知可以大大改善分类结果。对于有4000个标签的受监督培训，他们引用的错误率仅为13.60％，而我们相应的基线为34.85％。这种差距表明了多功能增强和部分max pooling带来的巨大好处。实际上，它们的基线结果已经优于之前的半监督结果。通过启用半监督学习，他们的分类错误率下降了17％（从13.60％降至11.29％），而我们看到相对下降幅度大得多65％（从34.85％降至12.16％）。

## SVHN

街景房屋号码（SVHN）数据集由真实世界房屋号码的32×32像素RGB图像组成，任务是对最中心的数字进行分类。在SVHN中，我们选择只使用Salimans等人的官方73257培训实例。即使有了这个选择，我们对所有标签的错误率仅为3.05％，无需增加。

表2将我们的方法与先前的技术水平进行了比较。使用最常用的1000个标签，我们观察到2.7个百分点的改善，在没有增强的情况下从8.11％到5.43％，并且通过标准增强进一步增加到4.42％。

我们还调查了500个标签的表现，在没有增强的情况下我们得到的错误率不到Salimans等人的一半，标准偏差也明显较低。当启用增强时，时序组合进一步将错误率降低到5.12％。在该测试中，Π模型和时序组合之间的差异在1.5个百分点处非常显着。

SVHN Sajjadi等人提供没有增强的结果，但需要注意的是它们使用部分max pooling，这是一种类似于增强的技术，因为它在网络中引入了随机的局部拉伸。在仅受监督的培训中，它导致2.28％的极高错误率，而我们相应的基线为3.05％（或翻译时为2.88％）。鉴于在单独的实验中，当使用额外数据时，我们的网络与未增强的SVHN的最佳公布结果匹配（Lee等人（2015）的1.69％），这个差距非常令人惊讶，并且我们得出结论：局部的max pooling可以大大的增强数据集，远远超过简单翻译可以实现的范围。与Sajjadi等报道使用的732标记数据，错误率为6.03％相比，我们的时序组合技术可以获得500和1000标签的更好的错误率（分别为5.12％和4.42％）。

## CIFAR-100和小图像

CIFAR-100数据集由来自一百个类的32×32像素RGB图像组成。我们不知道此数据集中以前的半监督结果，并为我们的实验选择了10000个标签。表3显示了没有和有增强的错误率分别为43.43％和38.65％。与仅使用标记输入的监督学习相比，这些对应于7.8和5.9个百分点的改进。

我们使用来自Tiny Images数据集的未标记的额外数据进行了另外两项测试（Torralba等，2008）：一个随机选择的500k额外图像，大多数不对应于任何CIFAR-100类别，另一个限制集合为237k 与CIFAR-100数据集中对应的类别中的图像（有关详细信息，请参阅附录A）。结果显示在表4中。随机选择的未标记的额外图像的添加将错误率提高了2.7个百分点（从26.30％到23.63％），表明了从随机自然图像中学习的理想能力。与Π模型相比，时间集合从额外数据中获益更多。有趣的是，将额外数据限制在CIFAR-100中存在的类别并未进一步提高分类准确性。这表明为了通过添加额外数据作为未标记输入来训练更好的分类器，将额外数据大致与实际输入相同的空间就足够了。在我们的例子中我们的假设甚至可以使用适当制作的合成数据作为未标记的输入来获得改进的分类器。

为了保持训练时间可以容忍，我们在这些测试中将未标记输入的数量限制为每个时期50k，即在我们使用来自CIFAR-100的所有50k标记输入和来自Tiny Images的50k附加未标记输入训练的每个时期。在500k或237k额外输入的每个时期随机选择50k未标记的输入。在时间集合中，在每个时期之后，我们仅更新对应于该时期上使用的输入的Z行。

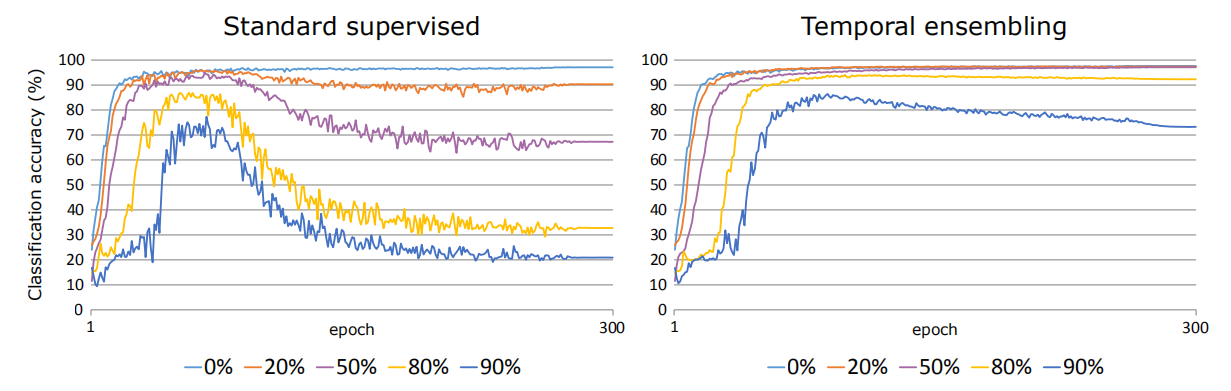


Figure 2: 当一部分标签随机化时，作为训练时期函数的正确SVHN分类的百分比。使用标准监督培训（左），即使一小部分标签给出假信号，分类准确性也会受到影响，随着随机标签部分增加到50％或更多，情况会迅速恶化。 另一方面，当半数标签是随机的时，时间集合（右）显示几乎完美的抗错误信息，即使80％的标签是随机的，也保持超过百分之九十的分类。

## 监督学习

当所有标签用于传统的监督培训时，我们的网络大致与CIFAR-10中单个模型的最新错误率相匹配，大致在6.05 ％的错误率，并且没有增强为7.33％。对于SVHN也是如此，但是最好的公布结果依赖于我们选择不使用的额外数据。

鉴于这一前提，我们的方法在使用所有标签时也会降低错误率，（表1和表2）这可能有点令人惊讶。我们认为，这表明一致性要求对许多分类任务中相当常见的模糊标签增加了一定程度的抵抗力，并且它鼓励特征对随机抽样更加不变。

## 对错误标签的容忍度

在进一步的测试中，我们研究了这样的假设：我们的方法通过在开始训练之前为训练集的特定百分比分配随机标签来增加对不正确标签的容忍度。图2显示了标准监督训练和时间集合的分类误差图。

显然，我们的方法对错误的标签提供了相当大的抵抗力，我们认为这是因为无监督的损失项鼓励网络实现的映射函数在所有输入数据点附近是平坦的，而受监督的损失项强制执行映射函数在标记的输入数据点附近具有特定值。这意味着即使是错误标记的输入也会在映射函数的形成中发挥作用，无监督损失项可以平滑映射函数，从而平滑决策边界，有效地将输入融合到相干簇中，而每个类中过多的正确标签就足够了通过监督的损失项将簇锁定到右输出向量。与经典正则化器的不同之处在于，我们仅在可能输入的流形上而不是在整个输入域上引起平滑。有关映射函数梯度重要性的进一步分析，请参阅Simard等。

# 相关工作

以前有很多关于半监督学习的工作（Zhu，2005）。在这里，我们将专注于与我们的工作最直接相关的那些。

1. model是梯形网络的一个子集，它将横向连接引入编码器 - 解码器类型的网络架构，目标是半监督学习。在n模型中，除去梯形网络中除最高横向连接之外的所有连接，并且在修剪不必要的阶段之后，剩余网络由两个平行的相同分支组成。其中一个分支采用原始训练输入，而另一个分支被赋予相同的输入，其中包含噪声。无监督损失项被计算为干净分支的（预激活）输出与损坏分支的去噪（预激活）输出之间的平方差。使用参数非线性从每个单元具有10个辅助可训练参数的损坏分支的输出计算去噪估计。我们的Π模型与e-模型的不同之处在于去除参数非线性和去噪，具有两条损坏的路径，并且比较网络的输出而不是最终层的预激活数据。
2. Sajjadi等最近为半监督学习引入了一种新的损失函数，即所谓的变换/稳定性损失，它建立在与我们工作相同的原理上。在训练期间，它们针对每个小批量运行增强和网络评估n次，然后计算不受约束的损失项作为所获得的n个网络输出之间的所有成对平方距离的总和。因此，他们的技术遵循Bachman等人的一般伪集合协议（PEA）正则化框架。此外，他们采用我们不使用的互斥特殊损失期限。我们的Π-模型可以看作是通过设置n = 2获得的变换/稳定性损失的特殊情况。具有变换/稳定性损失的训练的计算成本随着n的函数线性增加，而我们的时间集成技术的效率无论我们通过平均前一个时期的预测得到多大的有效集合，它都保持不变。

在引导聚合或装袋中，基于训练数据的子集独立地训练多个网络（Breiman，1996）。这导致整体比单个网络更稳定和准确。我们的方法可以被视为从基于单个网络的隐式集合中拉出预测，并且可变性是在不同的丢失和增强条件下评估它而不是对不同数据子集进行训练的结果。在与我们平行的工作中，Huang等人。在训练期间存储网络的多个快照，希望对应于不同的局部最小值，并将它们用作显式集合。

从部分标记数据推断新标签的一般技术通常被称为自举或自我训练，并且它首先由Yarowsky（1995）在语言分析的背景下提出。Whitney＆Sarkar（2012）分析了Yarowsky的算法，并提出了一种新的基于图形的标签传播方法。类似地，标签传播方法（Zhu＆Ghahramani，2002）通过使用合适的距离度量将相关输入与标记的训练输入进行比较来推断未标记训练数据的标签。我们的方法在两个重要方面与此不同。首先，我们从不将培训输入相互比较，而是仅依赖于未知标签保持不变，其次，我们让网络为未标记的输入生成可能的分类，而不是通过外部过程提供它们。

除了部分标记的数据之外，还在处理密集但不准确标记的数据方面付出了相当大的努力。这可以被视为半监督学习任务，其中部分培训过程是识别不可信任的标签。对于该领域的最近工作，参见例如Sukhbaatar等人。 （2014年）和Patrini等人。在嘈杂标签的背景下，Reed等人。提出了一种简单的自举方法，该方法训练分类器，其目标由前一个纪元输出和已知但可能有噪声的标签的凸组合组成。我们的时间集合通过考虑多个时期的评估而与此不同。

生成性对抗网络（GAN）最近已用于半监督学习，具有良好的结果（Maaløe等，2016; Springenberg，2016; Odena，2016; Salimans等，2016）。将生成组件纳入我们的解决方案可能是未来工作的一个有趣途径。我们还设想我们的方法可以应用于回归类型的学习任务。

附录

A 网络架构，测试设置和培训参数

表5详细介绍了我们所有测试中使用的网络架构。它受到ConvPoolCNN-C（Springenberg等人，2014）以及Salimans＆Kingma所做改进的启发。所有数据层均在He等人之后初始化。我们应用权重归一化和仅均值批量归一化，动量为0.999。我们使用泄漏的ReLU，其中α= 0.1作为非线性，并且选择使用最大池而不是跨步卷积，因为它在我们的实验中给出了始终更好的结果。

使用Adam（Kingma＆Ba）训练所有网络，最大学习率为λmax= 0.003，除了在SVHN情况下的时间集合，其中λmax= 0.001的最大学习率更好地工作。如本文所述，Adam动量参数设定为β1= 0.9且β2= 0.999。无监督损耗分量的最大值设置为wmax·M / N，其中M是标记输入的数量，N是训练输入的总数。对于Π模型运行，我们在所有运行中使用wmax = 100，除了具有Tiny Images的CIFAR-100，其中我们设置wmax = 300对于时间集合，我们在大多数运行中使用wmax = 30。对于3.5节中的损坏标签测试，我们使用wmax = 300表示0％和20％损坏，wmax = 3000表示腐败率为50％或更高。对于基本的CIFAR-100运行，我们使用wmax = 100，对于具有Tiny Images的CIFAR-100，我们使用wmax = 1000.在所有运行中，时间集合的累积衰减常数被设置为α= 0.6。

在所有运行中，我们使用高斯加速曲线exp [-5(1-T)2]在前80个时期内提高学习率λ和无监督损失分量权重w，其中T在0到1之间线性地从0前进到1。除了加速之外，我们在最后50个时期内将学习速率λ退火为零并将亚当β1退火至0.5，但是否则我们在训练期间没有衰减它们。斜降曲线类似于斜升曲线但是时间反转并且缩放常数为12.5而不是5.所有网络都训练了300个时期，小批量大小为100。

**CIFAR-10:**在完全监督学习的先前工作之后，我们使用ZCA预处理图像并使用水平翻转和随机翻译来增强数据集。翻译来自[-2,2]像素，并且独立地应用于Π模型中的两个分支。

**SVHN:**我们通过将每个输入图像偏置和缩放到零均值和单位方差来预处理输入图像。我们仅使用官方训练集中的73257项，即未使用提供的531131额外项目。除了没有使用水平翻转之外，训练设置与CIFAR-10类似

**实现：**python代码