Paraphrasing Complex Network: Network Compression via Factor Transfer

复杂网络的释义: 通过因子转移进行网络压缩

Jangho Kim
金浩
Seoul National University
首尔国立大学
Seoul, Korea
韩国首尔
kjh91@snu.ac.kr
SeongUk Park
成旭 · 朴
Seoul National University
首尔国立大学
Seoul, Korea
韩国首尔
swpark0703@snu.ac.kr
Nojun Kwak
俊 · 郭
Seoul National University
首尔国立大学
Seoul, Korea
韩国首尔
nojunk@snu.ac.kr

Abstract

摘要

Many researchers have sought ways of model compression to reduce the size of a deep neural network (DNN) with minimal performance degradation in order to use DNNs in embedded systems. Among the model compression methods, a method called knowledge transfer is to train a student network with a stronger teacher network. In this paper, we propose a novel knowledge transfer method which uses convolutional operations to paraphrase teacher's knowledge and to translate it for the student. This is done by two convolutional modules, which are called a paraphraser and a translator. The paraphraser is trained in an unsupervised manner to extract the teacher factors which are defined as paraphrased information of the teacher network. The translator located at the student network extracts the student factors and helps to translate the teacher factors by mimicking them. We observed that our student network trained with the proposed factor transfer method outperforms the ones trained with conventional knowledge transfer methods.

许多研究人员一直在寻求模型压缩的方法，以在尽量减少性能下降的情况下减小深度神经网络 (DNN) 的大小，以便在嵌入式系统中使用 DNN。在模型压缩方法中，一种称为知识转移的方法是用一个更强的教师网络来训练学生网络。本文提出了一种新颖的知识转移方法，该方法使用卷积操作来改写教师的知识并将其翻译给学生。这是通过两个卷积模块完成的，分别称为改写器和翻译器。改写器以无监督的方式训练，以提取教师因素，这些因素被定义为教师网络的改写信息。位于学生网络的翻译器提取学生因素，并通过模仿教师因素来帮助翻译教师因素。我们观察到，使用所提出的因素转移方法训练的学生网络优于使用传统知识转移方法训练的网络。

1 Introduction

1 引言

In recent years, deep neural nets (DNNs) have shown their remarkable capabilities in various parts of computer vision and pattern recognition tasks such as image classification, object detection, localization

and segmentation. Although many researchers have studied DNNs for their application in various fields, high-performance DNNs generally require a vast amount of computational power and storage, which makes them difficult to be used in embedded systems that have limited resources. Given the size of the equipment we use, tremendous GPU computations are not generally available in real world applications.

近年来，深度神经网络 (DNN) 在计算机视觉和模式识别任务的各个方面表现出了显著的能力，例如图像分类、目标检测、定位和分割。尽管许多研究人员已经研究了 DNN 在各个领域的应用，但高性能 DNN 通常需要大量的计算能力和存储，这使得它们难以在资源有限的嵌入式系统中使用。考虑到我们使用的设备的大小，巨大的 GPU 计算在现实世界的应用中通常不可用。

To deal with this problem, many researchers studied DNN structures to make DNNs smaller and more efficient to be applicable for embedded systems. These studies can be roughly classified into four categories: 1) network pruning, 2) network quantization, 3) building efficient small networks, and 4) knowledge transfer. First, network pruning is a way to reduce network complexity by pruning the redundant and non-informative weights in a pretrained model [26, 17, 7]. Second, network quantization compresses a pretrained model by reducing the number of bits used to represent the weight parameters of the pretrained model [20, 27]. Third, Iandola et al. [13] and Howard et al. [11] proposed efficient small network models which fit into the restricted resources. Finally, knowledge transfer (KT) method is to transfer large model's information to a smaller network [22, 30, 10].

为了解决这个问题，许多研究者研究了深度神经网络 (DNN) 结构，以使 DNN 更小、更高效，从而适用于嵌入式系统。这些研究大致可以分为四类:1) 网络剪枝，2) 网络量化，3) 构建高效的小型网络，以及 4) 知识迁移。首先，网络剪枝是一种通过剪除预训练模型中冗余和无信息的权重来减少网络复杂性的方式 [26, 17, 7]。其次，网络量化通过减少表示预训练模型权重参数所使用的位数来压缩预训练模型 [20, 27]。第三，Iandola 等人 [13] 和 Howard 等人 [11] 提出了适合于有限资源的高效小型网络模型。最后，知识迁移 (KT) 方法是将大模型的信息转移到较小的网络中 [22, 30, 10]。

Among the four approaches, in this paper, we focus on the last method, knowledge transfer. Previous studies such as attention transfer (AT) [30] and knowledge distillation (KD) [10] have achieved meaningful results in the field of knowledge transfer, where their loss function can be collectively summarized as the difference between the attention maps or softened distributions of the teacher and the student networks. These methods directly transfer the teacher network's softened distribution [10] or its attention map [30] to the student network, inducing the student to mimic the teacher.

在这四种方法中，本文重点关注最后一种方法，即知识迁移。先前的研究，如注意力迁移 (AT) [30] 和知识蒸馏 (KD) [10] 在知识迁移领域取得了有意义的成果，其损失函数可以统称为教师网络和学生网络的注意力图或软化分布之间的差异。这些方法直接将教师网络的软化分布 [10] 或其注意力图 [30] 转移到学生网络，促使学生模仿教师。

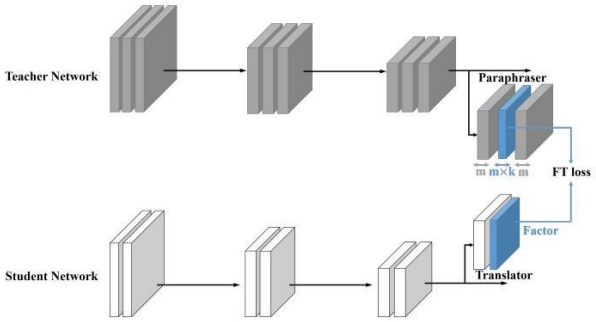


Figure 1: Overview of the factor transfer. In the teacher network, feature maps are transformed to the ' teacher factors' by a paraphraser. The number of feature maps of a teacher network(m)are resized to the number of feature maps of teacher factors ($m \times k$) by a paraphrase rate k . The feature maps of the student network are also transformed to the 'student factors' with the same dimension as that of the teacher factor using a translator. The factor transfer (FT) loss is used to minimize the difference between the teacher and the student factors in the training of the translator that generates student factors. Factors are drawn in blue. Note that before the FT, the paraphraser is already trained unsupervisedly by a reconstruction loss.

图 1: 因子转移概述。在教师网络中，特征图通过一个改述器转化为 "教师因子"。教师网络的特征图数量 (m) 被调整为教师因子的特征图数量 ($m \times k$) ，调整比例为 k 。学生网络的特征图也通过翻译器转化为与教师因子相同维度的 "学生因子"。因子转移 (FT) 损失用于最小化在生成学生因子的翻译器训练中教师因子与学生因子之间的差异。因子以蓝色表示。请注意，在 FT 之前，改述器已经通过重构损失进行无监督训练。

While these methods provide fairly good performance improvements, directly transferring the teacher's outputs overlooks the inherent differences between the teacher network and the student network, such as the network structure, the number of channels, and initial conditions. Therefore, we need to re-interpret the output of the teacher network to resolve these differences. For example, from the perspective of a teacher and a student, we came up with a question that simply providing the teacher's knowledge directly without any explanation can be somewhat insufficient for teaching the student. In other words, when teaching a child, the teacher should not use his/her own term because the child cannot understand it. On the other hand, if the teacher translates his/her terms into simpler ones, the child will much more easily understand.

虽然这些方法提供了相当不错的性能提升，但直接转移教师的输出忽视了教师网络和学生网络之间固有的差异，例如网络结构、通道数量和初始条件。因此，我们需要重新解释教师网络的输出，以解决这些差异。例如，从教师和学生的角度来看，我们提出了一个问题，即简单地直接提供教师的知识而没有任何解释，对于教学学生来说可能是不够的。换句话说，在教导孩子时，教师不应使用他/她自己的术语，因为孩子无法理解。另一方面，如果教师将自己的术语翻译为更简单的术语，孩子将更容易理解。

In this respect, we sought ways for the teacher network to deliver more understandable information to the student network, so that the student comprehends that information more easily. To address this problem, we propose a novel knowledge transferring method that leads both the student and teacher networks to make transportable features, which we call 'factors' in this paper. Contrary to the conventional methods, our method is not simply to compare the output values of the network directly, but to train neural networks that can extract good factors and to match these factors. The neural network that extracts factors from a teacher network is called a paraphraser, while the one that extracts factors from a student network is called a translator. We trained the paraphraser in an unsupervised way, expecting it to extract knowledges different from what can be obtained with supervised loss term. At the student side, we trained the student network with the translator to assimilate the factors extracted from the paraphraser. The overview of our proposed method is provided in Figure 1 . With various experiments, we succeeded in training the student network to perform better than the ones with the same architecture trained by the conventional knowledge transfer methods.

在这方面，我们寻求教师网络以更易理解的信息传递给学生网络，以便学生能够更轻松地理解这些信息。为了解决这个问题，我们提出了一种新颖的知识传递方法，使得学生和教师网络都能够生成可迁移的特征，我们在本文中称之为 “因子”。与传统方法相反，我们的方法并不是简单地直接比较网络的输出值，而是训练能够提取良好因子的神经网络，并匹配这些因子。提取教师网络因子的神经网络称为释义器，而提取学生网络因子的神经网络称为翻译器。我们以无监督的方式训练释义器，期望它提取与监督损失项所能获得的知识不同的知识。在学生端，我们使用翻译器训练学生网络，以同化从释义器提取的因子。我们提出的方法概述见图 1。通过各种实验，我们成功地训练学生网络，使其表现优于采用传统知识转移方法训练的相同架构的网络。

Our contributions can be summarized as follows:

我们的贡献可以总结如下:

- We propose a usage of a paraphraser as a means of extracting meaningful features (factors) in an unsupervised manner.
- 我们提出使用释义器作为无监督方式提取有意义特征 (因子) 的手段。
- We propose a convolutional translator in the student side that learns the factors of the teacher network.
- 我们在学生端提出了一种卷积翻译器，用于学习教师网络的因子。
- We experimentally show that our approach effectively enhances the performance of the student network.
- 我们通过实验表明，我们的方法有效地提升了学生网络的性能。

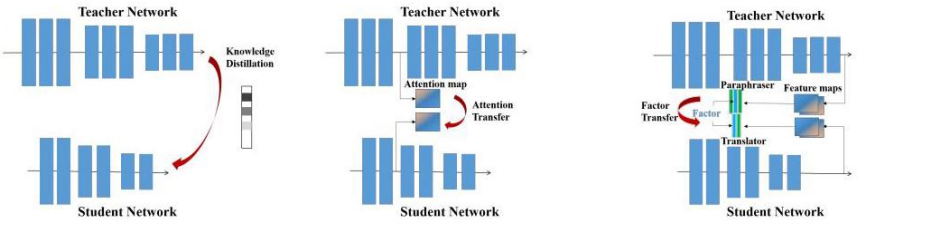
2 Related Works

2 相关工作

A wide variety of methods have been studied to use conventional networks more efficiently. In network pruning and quantization approaches, Srinivas et al. [26] proposed a data-free pruning method to remove

redundant neurons. Han et al. [7] removed the redundant connection and then used Huffman coding to quantize the weights. Gupta et al. [6] reduced float point operation and memory usage by using the 16 bit fixed-point representation. There are also many studies that directly train convolutional neural networks (CNN) using binary weights [3, 4, 20]. However, the network pruning methods require many iterations to converge and the pruning threshold is manually set according to the targeted amount of degradation in accuracy. Furthermore, the accuracies of binary weights are very poor, especially in large CNNs. There are many ways to directly design efficient small networks such as SqueezeNet [13], Mobile-Net [11] and Condense-Net [12], which showed a vast amount of reduction in the number of parameters compared to the original network sacrificing some accuracies. Also, there are methods of designing a network using a reinforcement learning algorithm such as MetaQNN [2] and Neural Architecture Search [33]. Using the reinforcement learning algorithm, the network itself searches for an efficient structure without human assistance. However, they only focused on performance without considering the number of parameters. In addition, it takes a lot of GPU memories and time to learn.

已经研究了多种方法以更有效地使用传统网络。在网络剪枝和量化方法中，Srinivas 等人 [26] 提出了一个无数据剪枝方法来去除冗余神经元。Han 等人 [7] 去除了冗余连接，然后使用霍夫曼编码对权重进行量化。Gupta 等人 [6] 通过使用 16 位定点表示法减少了浮点运算和内存使用。还有许多研究直接使用二进制权重训练卷积神经网络 (CNN) [3, 4, 20]。然而，网络剪枝方法需要多次迭代才能收敛，并且剪枝阈值是根据目标精度下降量手动设置的。此外，二进制权重的准确性非常差，尤其是在大型 CNN 中。还有许多直接设计高效小型网络的方法，如 SqueezeNet [13]、Mobile-Net [11] 和 Condense-Net [12]，这些方法在参数数量上与原始网络相比显示出了巨大的减少，尽管牺牲了一些准确性。此外，还有使用强化学习算法设计网络的方法，如 MetaQNN [2] 和神经架构搜索 [33]。通过使用强化学习算法，网络本身在没有人工帮助的情况下搜索高效的结构。然而，它们仅关注性能，而没有考虑参数的数量。此外，学习需要大量的 GPU 内存和时间。



(a) Knowledge Distillation (b) Attention Transfer (c) Factor Transfer (Proposed)

Figure 2: The structure of (a) KD [10], (b) AT [30] and (c) the proposed method FT. Unlike KD and AT, our method does not directly compare the softened distribution (KD) or the attention map (AT) which is defined as the sum of feature maps of the teacher and the student networks. Instead, we extract factors from both the teacher and the student, whose difference is tried to be minimized.

图 2: (a) KD [10]、(b) AT [30] 和 (c) 提出的 FT 方法的结构。与 KD 和 AT 不同，我们的方法并不直接比较软化分布 (KD) 或注意力图 (AT)，后者被定义为教师网络和学生网络特征图的总和。相反，我们从教师和学生中提取因素，试图最小化它们之间的差异。

Another method is the 'knowledge transfer'. This is a method of training a student network with a stronger teacher network. Knowledge distillation (KD) [10] is the early work of knowledge transfer for deep neural networks. The main idea of KD is to shift knowledge from a teacher network to a student network by leaning the class distribution via softened softmax. The student network can capture not only the information provided by the true labels, but also the information from the teacher. Yim et al. [28] defined the flow of solution procedure (FSP) matrix calculated by Gram matrix of feature maps from two layers in order to transfer knowledge. In FitNet [22], they designed the student network to be thinner and deeper than the teacher network, and provided hints from the teacher network for improving performance of the student network by learning intermediate representations of the teacher network. FitNet attempts to mimic the intermediate activation map directly from the teacher network. However, it can be problematic since there are significant capacity differences between the teacher and the student. Attention transfer (AT) [30], in contrast to FitNet, trains a less deep student network such that it mimics the attention maps of the teacher network which are summations of the activation maps along the channel dimension. Therefore, an attention map for a layer is of its the spatial dimensions. Figure 2 visually shows the difference of KD [10], AT [30] and the proposed method, factor transfer (FT). Unlike other methods, our method does not directly compare the teacher and student networks' softend distribution, or attention maps.

另一种方法是“知识迁移”。这是一种用更强的教师网络训练学生网络的方法。知识蒸馏 (KD)[10] 是深度神经网络知识迁移的早期工作。KD 的主要思想是通过软化的 softmax 将知识从教师网络转移到学生网络。学生网络不仅可以捕获真实标签提供的信息，还可以捕获来自教师的信息。Yim 等人 [28] 定义了通过两个层的特征图的 Gram 矩阵计算的解决方案过程流 (FSP) 矩阵，以便进行知识转移。在 FitNet [22] 中，他们设计了一个比教师网络更薄更深的学生网络，并提供了来自教师网络的提示，以通过学习教师网络的中间表示来提高学生网络的性能。FitNet 尝试直接模仿教师网络的中间激活图。然而，由于教师和学生之间存在显著的容量差异，这可能会成为问题。与 FitNet 相反，注意力迁移 (AT)[30] 训练一个较浅的学生网络，使其模仿教师网络的注意力图，这些图是沿通道维度的激活图的总和。因此，某一层的注意力图是其空间维度。图 2 直观地展示了 KD [10]、AT [30] 和所提出的方法因素迁移 (FT) 之间的差异。与其他方法不同，我们的方法并不直接比较教师和学生网络的软化分布或注意力图。

As shown in Figure 1, our paraphraser is similar to the convolutional autoencoder [18] in that it is trained in an unsupervised manner using the reconstruction loss and convolution layers. Hinton et al. 9 proved that autoencoders produce compact representations of images that contain enough information for reconstructing the original images. In [16], a stacked autoencoder on the MNIST dataset achieved great results with a greedy layer-wise approach. Many studies show that autoencoder models can learn meaningful, abstract features and thus achieve better classification results in high-dimensional data, such as images [19, 24]. The architecture of our paraphraser is different from convolutional autoencoders in that convolution layers do not downsample the spatial dimension of an input since the paraphraser uses sufficiently downsampled feature maps of a teacher network as the input.

如图 1 所示，我们的释义器类似于卷积自编码器 [18]，因为它是通过重构损失和卷积层以无监督的方式进行训练的。Hinton 等人 [9] 证明，自编码器能够生成紧凑的图像表示，这些表示包含足够的信息以重构原始图像。在 [16] 中，基于 MNIST 数据集的堆叠自编码器采用贪婪逐层的方法取得了良好的结果。许多研究表明，自编码器模型能够学习有意义的抽象特征，从而在高维数据 (如图像) 中实现更好的分类结果 [19, 24]。我们的释义器的架构与卷积自编码器不同，因为卷积层不会对输入的空间维度进行下采样，因为释义器使用的是经过充分下采样的教师网络特征图作为输入。

3 Proposed Method

3 提出的方法

It is said that if one fully understands a thing, he/she should be able to explain it by himself/herself. Correspondingly, if the student network can be trained to replicate the extracted information, this implies that the student network is well informed of that knowledge. In this section, we define the output of paraphraser's middle layer, as 'teacher factors' of the teacher network, and for the student network, we use the translator made up of several convolution layers to generate 'student factors' which are trained to replicate the 'teacher factors' as shown in Figure 1 with these modules, our knowledge transfer process consists of the following two main steps: 1) In the first step, the paraphraser is trained by a reconstruction loss. Then, teacher factors are extracted from the teacher network by a paraphraser. 2) In the second step, these teacher factors are transferred to the student factors such that the student network learns from them.

有人说，如果一个人完全理解某件事，他/她应该能够自己解释它。因此，如果学生网络能够被训练来复制提取的信息，这意味着学生网络对该知识有充分的了解。在本节中，我们将释义器中间层的输出定义为教师网络的“教师因子”，而对于学生网络，我们使用由多个卷积层组成的翻译器来生成“学生因子”，这些因子经过训练以复制“教师因子”。如图 1 所示，我们的知识转移过程包括以下两个主要步骤:1) 在第一步中，通过重构损失训练释义器。然后，通过释义器从教师网络中提取教师因子。2) 在第二步中，这些教师因子被转移到学生因子，使得学生网络能够从中学习。

3.1 Teacher Factor Extraction with Paraphraser

3.1 使用释义器提取教师因子

ResNet architectures [8] have stacked residual blocks and in [30] they call each stack of residual blocks as a 'group'. In this paper, we will also denote each stacked convolutional layers as a 'group'. Yosinski et al. 29 verified lower layer features are more general and higher layer features have a greater specificity. Since the teacher network and the student network are focusing on the same task, we extracted factors from the feature maps of the last group as clearly can be seen in Figure 1 because the last layer of a trained network must contain enough information for the task.

ResNet 架构 [8] 具有堆叠的残差块，在 [30] 中，他们将每一组残差块称为“组”。在本文中，我们也将每一组堆叠的卷积层称为“组”。Yosinski 等人 [29] 验证了较低层特征更具通用性，而较高层特征具有更大的特异性。由于教师网络和学生网络专注于相同的任务，我们从最后一组的特征图中提取因素，如图 1 所示，因为经过训练的网络的最后一层必须包含足够的任务信息。

In order to extract the factor from the teacher network, we train the paraphraser in an unsupervised way by assigning the reconstruction loss between the input feature maps x and the output feature maps $P(x)$ of the paraphraser. The unsupervised training act on the factor to be more meaningful, extracting different kind of knowledge from what can be obtained with supervised cross-entropy loss function. This approach can also be found in EBGAN [32], which uses an autoencoder as discriminator to give the generator different kind of knowledge from binary output.

为了从教师网络中提取因素，我们通过在输入特征图 x 和重构器的输出特征图 $P(x)$ 之间分配重构损失，以无监督的方式训练重构器。无监督训练使得提取的因素更具意义，从中提取不同类型的知识，而这些知识是通过监督交叉熵损失函数无法获得的。这种方法也可以在 EBGAN [32] 中找到，该方法使用自编码器作为判别器，为生成器提供不同类型的知识，而不是二元输出。

The paraphraser uses several convolution layers to produce the teacher factor F_T which is further processed by a number of transposed convolution layers in the training phase. Most of the convolutional autoencoders are designed to downsample the spatial dimension in order to increase the receptive field. On the contrary, the paraphraser maintains the spatial dimension while adjusting the number of factor channels because it uses the feature maps of the last group which has a sufficiently reduced spatial dimension. If the teacher network produces m feature maps, we resize the number of factor channels as $m \times k$. We refer to hyperparameter k as a paraphrase rate.

重构器使用多个卷积层来生成教师因素 F_T ，在训练阶段通过多个转置卷积层进一步处理。大多数卷积自编码器旨在下采样空间维度，以增加感受野。相反，重构器在调整因素通道数量的同时保持空间维度，因为它使用了最后一组的特征图，该特征图具有足够减少的空间维度。如果教师网络生成 m 特征图，我们将因素通道的数量调整为 $m \times k$ 。我们将超参数 k 称为重构率。

To extract the teacher factors, an adequately trained paraphraser is needed. The reconstruction loss function used for training the paraphraser is quite simple as

为了提取教师因素，需要一个经过充分训练的改写器。用于训练改写器的重构损失函数相当简单，因为

$$\mathcal{L}_{\text{rec}} = \| x - P(x) \|^2, \tag{1}$$

where the paraphraser network $P(\cdot)$ takes x as an input. After training the paraphraser, it can extract the task specific features (teacher factors) as can be seen in the supplementary material.

改写器网络 $P(\cdot)$ 将 x 作为输入。在训练完改写器后，它可以提取特定任务的特征 (教师因素)，如补充材料中所示。

3.2 Factor Transfer with Translator

3.2 使用翻译器进行因素转移

Once the teacher network has extracted the factors which are the paraphrased teacher's knowledge, the student network should be able to absorb and digest them on its own way. In this paper, we name this procedure as 'Factor Transfer'. As depicted in Figure 1, while training the student network, we inserted the translator right after the last group of student convolutional layers.

一旦教师网络提取了作为改写教师知识的因素，学生网络应该能够以自己的方式吸收和消化这些因素。在本文中，我们将此过程称为“因素转移”。如图 1 所示，在训练学生网络时，我们在最后一组学生卷积层之后插入了翻译器。

The translator is trained jointly with the student network so that the student network can learn the paraphrased information from the teacher network. Here, the translator plays a role of a buffer that relieves the student network from the burden of directly learning the output of the teacher network by rephrasing the feature map of the student network.

翻译器与学生网络共同训练，以便学生网络能够从教师网络学习改写的信息。在这里，翻译器充当一个缓冲区，减轻学生网络直接学习教师网络输出的负担，通过重新表述学生网络的特征图。

The student network is trained with the translator using the sum of two loss terms, i.e. the classification loss and the factor transfer loss:

学生网络与翻译器一起训练，使用两个损失项的总和，即分类损失和因素转移损失:

$$\mathcal{L}_{\text{student}} = \mathcal{L}_{cls} + \beta \mathcal{L}_{FT}, \tag{2}$$

$$\mathcal{L}_{cls} = \mathcal{C}\left(S\left(I_x\right), y\right) \tag{3}$$

$$\mathcal{L}_{FT} = \left\| \frac{F_T}{\left\|F_T\right\|_2} - \frac{F_S}{\left\|F_S\right\|_2} \right\|_p. \tag{4}$$

With (4), the student's translator is trained to output the student factors that mimic the teacher factors. Here, F_T and F_S denote the teacher and the student factors, respectively. We set the dimension of F_S to be the same as that of F_T . We also apply an l_2 normalization on the factors as [30]. In this paper, the performances using l_1 loss ($p = 1$) is reported, but the performance difference between $l_1\,(p = 1)$ and $l_2\,(p = 2)$ losses is minor (See the supplementary material), so we consistently used l_1 loss for all experiments.

根据 (4)，学生的翻译器被训练为输出模仿教师因素的学生因素。在这里，F_T 和 F_S 分别表示教师和学生因素。我们将 F_S 的维度设置为与 F_T 相同。我们还对因素应用了 l_2 归一化，如 [30] 所示。在本文中，报告了使用 l_1 损失 ($p = 1$) 的性能，但 $l_1\,(p = 1)$ 和 $l_2\,(p = 2)$ 损失之间的性能差异很小 (见补充材料)，因此我们在所有实验中一致使用 l_1 损失。

In addition to the factor transfer loss (4), the conventional classification loss (3) is also used to train student network as in (2). Here, β is a weight parameter and $C\left(S\left(I_x\right), y\right)$ denotes the cross entropy between ground-truth label y and the softmax output $S\left(I_x\right)$ of the student network for an input image I_x , a commonly used term for classification tasks.

除了因子转移损失 (4) 外，常规分类损失 (3) 也用于训练学生网络，如 (2) 所示。这里，β 是一个权重参数，$C\left(S\left(I_x\right), y\right)$ 表示真实标签 y 与学生网络对输入图像 I_x 的 softmax 输出 $S\left(I_x\right)$ 之间的交叉熵，这是分类任务中常用的术语。

The translator takes the output features of the student network, and with (2), it sends the gradient back to the student networks, which lets the student network absorb and digest the teacher's knowledge in its own way. Note that unlike the training of the teacher paraphraser, the student network and its translator are trained simultaneously in an end-to-end manner.

翻译器获取学生网络的输出特征，并通过 (2) 将梯度反馈给学生网络，这使得学生网络能够以自己的方式吸收和消化教师的知识。请注意，与教师释义器的训练不同，学生网络及其翻译器是以端到端的方式同时训练的。

4 Experiments

4 实验

In this section, we evaluate the proposed FT method on several datasets. First, we verify the effectiveness of FT through the experiments with CIFAR-10 [14] and CIFAR-100 [15] datasets, both of which are the basic image classification datasets, because many works that tried to solve the knowledge transfer problem used CIFAR in their base experiments [22, 30]. Then, we evaluate our method on ImageNet LSVRC 2015 [23] dataset. Finally, we applied our method to object detection with PASCAL VOC 2007 [5] dataset.

在本节中，我们在多个数据集上评估所提出的 FT 方法。首先，我们通过在 CIFAR-10 [14] 和 CIFAR-100 [15] 数据集上的实验验证 FT 的有效性，这两个数据集都是基本的图像分类数据集，因为许多尝试解决知识转移问题的工作在其基础实验中使用了 CIFAR [22, 30]。然后，我们在 ImageNet LSVRC 2015 [23] 数据集上评估我们的方法。最后，我们将我们的方法应用于 PASCAL VOC 2007 [5] 数据集的目标检测。

To verify our method, we compare the proposed FT with several knowledge transfer methods such as KD [10] and AT [30]. There are several important hyperparameters that need to be consistent. For KD, we fix the temperature for softened softmax to 4 as in [10], and for β of AT, we set it to 10^3 following [30]. In the whole experiments, AT used multiple group losses. Alike AT, β of FT is set to 10^3 in ImageNet and PASCAL VOC 2007. However, we set it to 5×10^2 in CIFAR-10 and CIFAR-100 because a large β hinders the convergence.

为了验证我们的方法，我们将提出的 FT 与几种知识迁移方法进行比较，如 KD [10] 和 AT [30]。有几个重要的超参数需要保持一致。对于 KD，我们将软化 softmax 的温度固定为 4，如 [10] 所示，对于 AT 的 β ，我们将其设置为 10^3 ，遵循 [30]。在整个实验中，AT 使用了多个组损失。与 AT 类似，FT

的 β 在 ImageNet 和 PASCAL VOC 2007 中设置为 10^3 。然而，我们在 CIFAR-10 和 CIFAR-100 中将其设置为 5×10^2 ，因为较大的 β 会阻碍收敛。

We conduct experiments for different k values from 0.5 to 4 . To show the effectiveness of the proposed paraphraser architecture, we also used two convolutional autoencoders as paraphrasers because the autoencoder is well known for extracting good features which contain compressed information for reconstruction. One is an undercomplete convolutional autoencoder (CAE), the other is an overcomplete regularized autoencoder (RAE) which imposes l_1 penalty on factors to learn the size of factors needed by itself [1]. Details of these autoencoders and overall implementations of experiments are explained in the supplementary material.

我们对不同的 k 值进行了从 0.5 到 4 的实验。为了展示所提出的释义器架构的有效性，我们还使用了两个卷积自编码器作为释义器，因为自编码器以提取包含压缩信息的良好特征而闻名，这些特征用于重建。其中一个是欠完备卷积自编码器 (CAE)，另一个是过完备正则化自编码器 (RAE)，它对因子施加 l_1 惩罚，以学习自身所需因子的大小 [1]。这些自编码器的详细信息以及实验的整体实现将在补充材料中解释。

In some experiments, we also tested KD in combination with AT or FT because KD transfers output knowledge while AT and FT delivers knowledge from intermediate blocks and these two different methods can be combined into one (KD+AT or KD+FT).

在一些实验中，我们还测试了 KD 与 AT 或 FT 的组合，因为 KD 转移输出知识，而 AT 和 FT 则从中间块传递知识，这两种不同的方法可以结合成一个 (KD+AT 或 KD+FT)。

4.1 CIFAR-10

4.1 CIFAR-10

The CIFAR-10 dataset consists of 50 K training images and 10 K testing images with 10 classes. We conducted several experiments on CIFAR-10 with various network architectures, including ResNet [8], Wide ResNet (WRN) [31] and VGG [25]. Then, we made four conditions to test various situations. First, we used ResNet-20 and ResNet-56 which are used in CIFAR-10 experiments of [8]. This condition is for the case where the teacher and the student networks have same width (number of channels) and different depths (number of blocks). Secondly, we experimented with different types of residual networks using ResNet-20 and WRN-40-1. Thirdly, we intended to see the effect of the absence of shortcut connections that exist in Resblock on knowledge transfer by using VGG13 and WRN-46-4. Lastly, we used WRN-16-1 and WRN-16-2 to test the applicability of knowledge transfer methods for the architectures with the same depth but different widths.

CIFAR-10 数据集由 50 K 张训练图像和 10 K 张测试图像组成，共有 10 个类别。我们在 CIFAR-10 上进行了多项实验，使用了多种网络架构，包括 ResNet [8]、Wide ResNet (WRN) [31] 和 VGG [25]。然后，我们设置了四种条件以测试不同的情况。首先，我们使用了 ResNet-20 和 ResNet-56，这些网络在 [8] 的 CIFAR-10 实验中被使用。该条件适用于教师网络和学生网络具有相同宽度 (通道数) 但深度 (块数) 不同的情况。其次，我们使用 ResNet-20 和 WRN-40-1 实验了不同类型的残差网络。第三，我们希望通过使用 VGG13 和 WRN-46-4 来观察缺少存在于 Resblock 中的快捷连接对知识转移的影响。最后，我们使用 WRN-16-1 和 WRN-16-2 测试知识转移方法在具有相同深度但不同宽度的架构中的适用性。

Student	Teacher	Student	AT	KD	FT	AT+KD	FT+KD		Teacher
ResNet-20 (0.27M)	ResNet-56 (0.85M)	7.78	7.13	7.19	6.85	6.89	7.04		6.39
ResNet-20 (0.27M)	WRN-40-1 (0.56M)	7.78	7.34	7.09	6.85	7.00	6.95		6.84
VGG-13 (9.4M)	WRN-46-4 (10M)	5.99	5.54	5.71	4.84	5.30	4.65		4.44
WRN-16-1 (0.17M)	WRN-16-2 (0.69M)	8.62	8.10	7.64	7.64	7.52	7.59		6.27
Student	Teacher	$k = 0.5$	$k = 0.75$		$k = 1$	$k = 2$	$k = 4$	CAE	RAE
ResNet-20 (0.27M)	ResNet-56 (0.85M)	6.85	6.92		6.89	6.87	7.08	7.07	7.24
ResNet-20 (0.27M)	WRN-40-1 (0.56M)	7.16	7.05		7.04	6.85	7.05	7.26	7.33
VGG-13 (9.4M)	WRN-46-4 (10M)	4.84	5.09		5.04	5.01	4.98	5.85	5.53
WRN-16-1 (0.17M)	WRN-16-2 (0.69M)	7.64	7.83		7.74	7.87	7.95	8.48	8.00

学生	教师	学生	AT	KD	FT	AT+KD	FT+KD		教师
ResNet-20 (0.27M)	ResNet-56 (0.85M)	7.78	7.13	7.19	6.85	6.89	7.04		6.39
ResNet-20 (0.27M)	WRN-40-1 (0.56M)	7.78	7.34	7.09	6.85	7.00	6.95		6.84
VGG-13 (9.4M)	WRN-46-4 (10M)	5.99	5.54	5.71	4.84	5.30	4.65		4.44
WRN-16-1 (0.17M)	WRN-16-2 (0.69M)	8.62	8.10	7.64	7.64	7.52	7.59		6.27
学生	教师	$k = 0.5$	$k = 0.75$		$k = 1$	$k = 2$	$k = 4$	CAE	RAE
ResNet-20 (0.27M)	ResNet-56 (0.85M)	6.85	6.92		6.89	6.87	7.08	7.07	7.24
ResNet-20 (0.27M)	WRN-40-1 (0.56M)	7.16	7.05		7.04	6.85	7.05	7.26	7.33
VGG-13 (9.4M)	WRN-46-4 (10M)	4.84	5.09		5.04	5.01	4.98	5.85	5.53
WRN-16-1 (0.17M)	WRN-16-2 (0.69M)	7.64	7.83		7.74	7.87	7.95	8.48	8.00

Table 1: Mean classification error (%) on CIFAR-10 dataset (5 runs). All the numbers are the results of our implementation. AT and KD are implemented according to [30].

表 1:CIFAR-10 数据集上的平均分类错误率 (%)(5 次运行)。所有数字均为我们实现的结果。AT 和 KD 的实现依据 [30]。

Student	Teacher	Student	AT	F-ActT	KD	AT+KD	Teacher	FT ($k = 0.5$)	Teacher
WRN-16-1 (0.17M)	WRN-40-1 (0.56M)	8.77	8.25	8.62	8.39	8.01	6.58	8.12	6.55
WRN-16-2 (0.69M)	WRN-40-2 (2.2M)	6.31	5.85	6.24	6.08	5.71	5.23	5.51	5.09

学生	教师	学生	AT	F-ActT	KD	AT+KD	教师	FT ($k = 0.5$)	教师
WRN-16-1 (0.17M)	WRN-40-1 (0.56M)	8.77	8.25	8.62	8.39	8.01	6.58	8.12	6.55
WRN-16-2 (0.69M)	WRN-40-2 (2.2M)	6.31	5.85	6.24	6.08	5.71	5.23	5.51	5.09

Table 2: Median classification error (%) on CIFAR-10 dataset (5 runs). The first 6 columns are from Table 1 of [30], while the last two columns are from our implementation.

表 2:CIFAR-10 数据集上的中位分类错误率 (%)(5 次运行)。前 6 列来自 [30] 的表 1，而最后两列来自我们的实现。

In the first experiment, we wanted to show that our algorithm is applicable to various networks. Result of FT and other knowledge transfer algorithms can be found in Table 1 In the table, 'Student' column provides the performance of student network trained from scratch. The 'Teacher' column provides the performance of the pretrained teacher network. The numbers in the parentheses are the sizes of network parameters in Millions. The performances of AT and KD are better than those of 'Student' trained from scratch and the two show better or worse performances than the other depending on the type of network used. For FT, we chose the best performance among the different k values shown in the bottom rows in the table. The proposed FT shows better performances than AT and KD consistently, regardless of the type of network used.

在第一次实验中，我们希望展示我们的算法适用于各种网络。FT 和其他知识转移算法的结果可以在表 1 中找到。在表中，“学生”列提供了从头开始训练的学生网络的性能。“教师”列提供了预训练教师网络的性能。括号中的数字是网络参数的大小 (以百万为单位)。AT 和 KD 的性能优于从头开始训练的“学生”，而且这两者的表现根据所使用的网络类型而有所不同。对于 FT，我们选择了表底部显示的不同 k 值中的最佳性能。所提出的 FT 无论在何种类型的网络中，均表现出比 AT 和 KD 更好的性能。

In the cases of hybrid knowledge transfer methods such as AT+KD and FT+KD, we could get interesting result that AT and KD make some sort of synergy, because for all the cases, AT+KD performed better than standalone AT or KD. It sometimes performed even better than FT, but FT model trained together with KD loses its power in some cases.

在 AT+KD 和 FT+KD 等混合知识转移方法的情况下，我们得到了有趣的结果，即 AT 和 KD 形成了一种协同效应，因为在所有情况下，AT+KD 的表现优于单独的 AT 或 KD。有时它的表现甚至优于 FT，但与 KD 一起训练的 FT 模型在某些情况下失去了其效力。

As stated before in section 3.1, to check if having a paraphraser per group in FT is beneficial, we trained a ResNet-20 as student network with paraphrasers and translators combined in group1, group2 and group3, using the ResNet-56 as teacher network with $k = 0.75$. The classification error was 7.01% , which is 0.06% higher than that from the single FT loss for the last group. This indicates that the combined FT loss does not improve the performance thus we have used the single FT loss throughout the paper. In terms of paraphrasing the information of the teacher network, the paraphraser which maintains the spatial dimension outperformed autoencoders based methods which use CAE or

如第 3.1 节所述，为了检查在 FT 中每组是否有一个释义器是有益的，我们训练了一个 ResNet-20 作为学生网络，结合了组 1、组 2 和组 3 的释义器和翻译器，使用 ResNet-56 作为教师网络，带有 $k = 0.75$ 。分类错误为 7.01% ，这比最后一组的单一 FT 损失高出 0.06% 。这表明组合 FT 损失并没有提高性能，因此我们在整篇论文中使用了单一 FT 损失。在对教师网络信息进行释义方面，保持空间维度的释义器的表现优于使用 CAE 或其他方法的自编码器。RAE.

As a second experiment, we compared FT with transferring FitNets-style hints which use full activation maps as in [30]. Table 2 shows the results which verify that using the paraphrased information is more beneficial than directly using the full activation maps (full feature maps). In the table, FT gives better accuracy improvement than full-activation transfer (F-ActT). Note that we trained a teacher network from scratch for factor transfer (the last column) with the same experimental environment of [30] because there is no pretrained model of the teacher networks.

作为第二个实验，我们将 FT 与转移 FitNets 风格的提示进行了比较，这些提示使用与 [30] 中相同的完整激活图。表 2 显示的结果验证了使用改写的信息比直接使用完整激活图 (完整特征图) 更有益。在表中，FT 的准确性提升优于完整激活转移 (F-ActT)。请注意，我们从头开始训练了一个教师网络用于因子转移 (最后一列)，实验环境与 [30] 相同，因为没有教师网络的预训练模型。

4.2 CIFAR-100

4.2 CIFAR-100

For further analysis, we wanted to apply our algorithm to more difficult tasks to prove generality of the proposed FT by adopting CIFAR-100 dataset. CIFAR-100 dataset contains the same number of images as CIFAR-10 dataset, 50 K (train) and 10 K (test), but has 100 classes, containing only 500 images per classes. Since the training dataset is more complicated, we thought the number of blocks (depth) in the network has much more impact on the classification performance because deeper and stronger networks will better learn the boundaries between classes. Thus, the experiments on CIFAR-100 were designed to observe the changes depending on the depths of networks. The teacher network was fixed as ResNet-110, and the two networks ResNet-20 and ResNet-56, that have the same width (number of channels) but different depth (number of blocks) with the teacher, were used as student networks. As can be seen in Table 3, we got an impressive result that the student network ResNet-56 trained with FT even outperforms the teacher network. The student ResNet-20 did not work that well but it also outperformed other knowledge transfer methods.

为了进一步分析，我们希望将我们的算法应用于更困难的任务，以证明所提出的 FT 的普遍性，因此采用了 CIFAR-100 数据集。CIFAR-100 数据集包含与 CIFAR-10 数据集相同数量的图像，50 K (训练) 和 10 K (测试)，但有 100 个类别，每个类别仅包含 500 张图像。由于训练数据集更复杂，我们认为网络中的块数 (深度) 对分类性能的影响更大，因为更深更强的网络能够更好地学习类别之间的边界。因此，CIFAR-100 的实验设计旨在观察网络深度变化带来的影响。教师网络固定为 ResNet-110，而两个网络 ResNet-20 和 ResNet-56 具有与教师相同的宽度 (通道数) 但不同的深度 (块数)，被用作学生网络。如表 3 所示，我们得到了一个令人印象深刻的结果，即使用 FT 训练的学生网络 ResNet-56 甚至超越了教师网络。学生网络 ResNet-20 的表现不如预期，但它也优于其他知识转移方法。

Student	Teacher	Student	AT	KD	FT	AT+KD	FT+KD	Teacher
ResNet-56 (0.85M)	ResNet-110 (1.73M)	28.04	27.28	27.96	25.62	28.01	26.93	26.91
ResNet-20 (0.27M)	ResNet-110 (1.73M)	31.24	31.04	33.14	29.08	34.78	32.19	26.91
Student	Teacher	$k = 0.5$	$k = 0.75$		$k = 1$	$k = 2$ $k = 4$	CAE	RAE
ResNet-56 (0.85M)	ResNet-110 (1.73M)	25.62	25.78		25.85	25.6325.87	26.41	26.29
ResNet-20 (0.27M)	ResNet-110 (1.73M)	29.20	29.25		29.28	29.1929.08	29.84	30.11

学生	教师	学生	AT	KD	FT	AT+KD	FT+KD	教师
ResNet-56 (0.85M)	ResNet-110 (1.73M)	28.04	27.28	27.96	25.62	28.01	26.93	26.91
ResNet-20 (0.27M)	ResNet-110 (1.73M)	31.24	31.04	33.14	29.08	34.78	32.19	26.91
学生	教师	$k = 0.5$	$k = 0.75$		$k = 1$	$k = 2$ $k = 4$	CAE	RAE
ResNet-56 (0.85M)	ResNet-110 (1.73M)	25.62	25.78		25.85	25.6325.87	26.41	26.29
ResNet-20 (0.27M)	ResNet-110 (1.73M)	29.20	29.25		29.28	29.1929.08	29.84	30.11

Table 3: Mean classification error (%) on CIFAR-100 dataset (5 runs). All the numbers are from our implementation.

表 3:CIFAR-100 数据集上的平均分类错误率 (%)(5 次运行)。所有数字均来自我们的实现。

Paraphraser	Translator	CIFAR-10	CIFAR-100	Number of layers in Paraphraser	CIFAR-10	CIFAR-100
Yes	No	6.18	27.61	1 Layer [0.07M]	6.09	27.07
No	Yes	6.12	27.39	2 Layers [0.22M]	5.99	27.03
Yes	Yes	5.71	26.91	3 Layers [0.26M]	5.71	26.91
Student (WRN-40-1[0.6M])		7.02	28.81	Teacher (WRN-40-2[2.2M])	4.96	24.10

改写器	翻译器	CIFAR-10	CIFAR-100	Paraphraser 中的层数	CIFAR-10	CIFAR-100
是	否	6.18	27.61	1 层 [0.07M]	6.09	27.07
否	是	6.12	27.39	2 层 [0.22M]	5.99	27.03
是	是	5.71	26.91	3 层 [0.26M]	5.71	26.91
学生 (WRN-40-1[0.6M])		7.02	28.81	教师 (WRN-40-2[2.2M])	4.96	24.10

Table 4: Left: Ablation study with and without the paraphraser ($k = 0.5$) and the Translator. (Mean classification error (%) of 5 runs). Right: Effect of number of layers in the paraphraser.

表 4: 左侧: 有无释义器 ($k = 0.5$) 和翻译器的消融研究。(5 次运行的平均分类错误率 (%))。右侧: 释义器中层数的影响。

Additionally, in line with the experimental result in [30], we also got consistent result that KD suffers from the gap of depths between the teacher and the student, and the accuracy is even worse compared to the student network in the case of training ResNet-20. For this dataset, the hybrid methods (AT+KD and FT+KD) was worse than the standalone AT or FT. This also indicates that KD is not suitable for a situation where the depth difference between the teacher and the student networks is large.

此外，与 [30] 中的实验结果一致，我们也得到了相同的结果，即知识蒸馏 (KD) 受到教师网络与学生网络深度差距的影响，在训练 ResNet-20 的情况下，准确率甚至比学生网络更差。对于该数据集，混合方法 (AT+KD 和 FT+KD) 表现不如独立的 AT 或 FT。这也表明，KD 不适合教师网络与学生网络之间深度差距较大的情况。

4.3 Ablation Study

4.3 消融研究

In the introduction, we have described that the teacher network provides more paraphrased information to the student network via factors, and described a need for a translator to act as a buffer to better understand factors in the student network. To further analyze the role of factor, we performed an ablation experiment on the presence or absence of a paraphraser and a translator. The result is shown in Table 4 The student network and the teacher network are selected with different number of output channels. One can adjust the number of student and teacher factors by adjusting the paraphrase rate k of the paraphraser. As described above, since the role of the paraphraser (making F_T with unsupervised training loss) and the translator (trained jointly with student network to ease the learning of Factor Transfer) are not the same, we can confirm that the synergy of two modules maximizes the performance of the student network. Also, we report the performance of different number of layers in the paraphraser. As the number of layers increases, the performance also increases.

在引言中，我们描述了教师网络通过因子向学生网络提供更多的释义信息，并描述了需要一个翻译器作为缓冲，以更好地理解学生网络中的因子。为了进一步分析因子的作用，我们进行了一个关于释义器和翻译器存在与否的消融实验。结果如表 4 所示。学生网络和教师网络选择了不同数量的输出通道。可以通过调整释义器的释义率 k 来调整学生和教师因子的数量。如上所述，由于释义器 (使用无监督训练损失进行 F_T) 和翻译器 (与学生网络共同训练以促进因子转移的学习) 的作用不同，我们可以确认这两个模块的协同作用最大化了学生网络的性能。此外，我们报告了释义器中不同层数的性能。随着层数的增加，性能也随之提高。

4.4 ImageNet

4.4 ImageNet

The ImageNet dataset is a image classification dataset which consists of 1.2M training images and 50 K validation images with 1,000 classes. We conducted large scale experiments on the ImageNet LSVRC 2015 in order to show our potential availability to transfer even more complex and detailed informations. We chose ResNet-18 as a student network and ResNet-34 as a teacher network same as in [30] and validated the performance based on top-1 and top-5 error rates as shown in Table 5

ImageNet 数据集是一个图像分类数据集，包含 1.2M 张训练图像和 50 K 张验证图像，共有 1,000 个类别。我们在 ImageNet LSVRC 2015 上进行了大规模实验，以展示我们转移更复杂和详细信息的潜力。我们选择 ResNet-18 作为学生网络，ResNet-34 作为教师网络，和 [30] 中相同，并根据表 5 中显示的 top-1 和 top-5 错误率验证了性能。

As can be seen in Table 5, FT consistently outperforms the other methods. The KD, again, suffers from the depth difference problem, as already confirmed in the result of other experiments. It shows just adding the FT loss helps to lower about 1.34% of student network' s (ResNet-18) Top-1 error on ImageNet.

如表 5 所示，FT 始终优于其他方法。KD 再次受到深度差异问题的影响，这在其他实验的结果中已经得到确认。它表明，仅仅添加 FT 损失有助于降低学生网络 (ResNet-18) 在 ImageNet 上的 Top-1 错误率约 1.34% 。

4.5 Object Detection

4.5 目标检测

In this experiment, we wanted to verify the generality of FT, and decided to apply it on detection task, other than classifications. We used Faster-RCNN pipeline [21] with PASCAL VOC 2007 dataset [5]

在本实验中，我们希望验证 FT 的普适性，并决定将其应用于检测任务，而非分类。我们使用了 Faster-RCNN 流水线 [21] 和 PASCAL VOC 2007 数据集 [5]。

Method	Network	Top-1	Top-5
Student	Resnet-18	29.91	10.68
KD	Resnet-18	33.83	12.55
AT	Resnet-18	29.36	10.23
FT ($k = 0.5$)	Resnet-18	28.57	9.71
Teacher	Resnet-34	26.73	8.57

方法	网络	Top-1	前五名
学生	Resnet-18	29.91	10.68
知识蒸馏	Resnet-18	33.83	12.55
对抗训练	Resnet-18	29.36	10.23
微调 ($k = 0.5$)	Resnet-18	28.57	9.71
教师	Resnet-34	26.73	8.57

Method	mAP
Student(VGG-16)	69.5
FT(VGG-16, $k = 0.5$)	70.3
Teacher(ResNet-101)	75.0

方法	平均精度均值
学生 (VGG-16)	69.5
FT(VGG-16, $k = 0.5$)	70.3
教师 (ResNet-101)	75.0

Table 6: Mean average precision on PASCAL VOC 2007 test dataset.

表 6:PASCAL VOC 2007 测试数据集上的平均精度。

Table 5: Top-1 and Top-5 classification error (%) on Ima-geNet dataset. All the numbers are from our implementation.

表 5:ImageNet 数据集上的 Top-1 和 Top-5 分类错误率 (%)。所有数字均来自我们的实现。

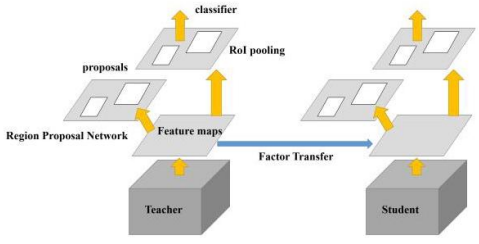


Figure 3: Factor transfer applied to Faster-RCNN framework

图 3: 应用于 Faster-RCNN 框架的因子转移。

for object detection. We used PASCAL VOC 2007 trainval as training data and PASCAL VOC 2007 test as testing data. Instead of using our own ImageNet FT pretrained model as a backbone network for detection, we tried to apply our method for transferring knowledges about object detection. Here, we set a hypothesis that since the factors are extracted in an unsupervised manner, the factors not only can connote the core knowledge of classification, but also can convey other types of representations.

用于目标检测。我们使用 PASCAL VOC 2007 trainval 作为训练数据，PASCAL VOC 2007 test 作为测试数据。我们没有使用自己训练的 ImageNet FT 预训练模型作为检测的主干网络，而是尝试应用我们的方法来转移关于目标检测的知识。在这里，我们设定一个假设: 由于这些因素是以无监督的方式提取的，这些因素不仅可以隐含分类的核心知识，还可以传达其他类型的表征。

In the Faster-RCNN, the shared convolution layers contain knowledges of both classification and localization, so we applied factor transfer to the last layer of shared convolution layer. Figure 3 shows where we applied FT in the Faster-RCNN framework. We set VGG-16 as a student network and ResNet-101 as a teacher network. Both networks are fine-tuned at PASCAL VOC 2007 dataset with ImageNet pretrained model. For FT, we used ImageNet pretrained VGG-16 model and fixed the layers before conv3 layer during training phase. Then, by the factor transfer, the gradient caused by the $\mathcal{L}_{FT}$ loss back-propagates to the student network passing by the student translator.

在 Faster-RCNN 中，共享卷积层包含分类和定位的知识，因此我们将因子转移应用于共享卷积层的最后一层。图 3 显示了我们在 Faster-RCNN 框架中应用 FT 的位置。我们将 VGG-16 设置为学生网络，将 ResNet-101 设置为教师网络。两个网络都在 PASCAL VOC 2007 数据集上进行了微调，使用了 ImageNet 预训练模型。对于 FT，我们使用了 ImageNet 预训练的 VGG-16 模型，并在训练阶段固定了 conv3 层之前的层。然后，通过因子转移，由 $\mathcal{L}_{FT}$ 损失引起的梯度通过学生翻译器反向传播到学生网络。

As can be seen in Table 6, we could get performance enhancement of 0.8 in mAP (mean average precision) score by training Faster-RCNN with VGG-16. As mentioned earlier, we have strong belief that the latter layer we apply the factor transfer, the higher the performance enhances. However, by the limit of VGG-type backbone network we have used, we tried but could not apply FT else that the backbone network. Experiment on the capable case where the FT can be applied to the latter layers like region proposal network (RPN) or other types of detection network will be our future work.

如表 6 所示，通过使用 VGG-16 训练 Faster-RCNN，我们在 mAP(平均精度均值) 得分上获得了 0.8 的性能提升。如前所述，我们坚信应用因子转移的后层性能提升越高。然而，由于我们使用的 VGG 类型主干网络的限制，我们尝试过但无法将 FT 应用到主干网络之外的其他层。能够将 FT 应用到后续层 (如区域提议网络 RPN 或其他类型的检测网络) 的实验将是我们未来的工作。

4.6 Discussion

4.6 讨论

In this section, we compare FitNet [22] and FT. FitNet transfers information of an intermediate layer while FT uses the last layer, and the purpose of the regressor in FitNet is somewhat different from our translator. More specifically, Romero et al. [22] argued that giving hints from deeper layer over-regularizes the student network. On the contrary, we chose the deeper layer to provide more specific information as mentioned in the paper. Also, FitNet does not use the paraphraser as well. Note that FitNet is actually a 2-stage algorithm in that they initialize the student weights with hints and then train the student network using Knowledge Distillation.

在本节中，我们比较了 FitNet [22] 和 FT。FitNet 转移的是中间层的信息，而 FT 使用的是最后一层，FitNet 中回归器的目的与我们的翻译器有些不同。更具体地说，Romero 等人 [22] 认为，从更深层提供提示会过度正则化学生网络。相反，我们选择了更深层来提供更具体的信息，如论文中所述。此外，FitNet 也没有使用释义器。请注意，FitNet 实际上是一个两阶段算法，因为它们用提示初始化学生权重，然后使用知识蒸馏训练学生网络。

5 Conclusion

5 结论

In this work, we propose the factor transfer which is a novel method for knowledge transfer. Unlike previous methods, we introduce factors which contain paraphrased information of the teacher network, extracted from the paraphraser. There are mainly two reasons that the student can understand information from the teacher network more easily by the factor transfer than other methods. One reason

is that the factors can relieve the inherent differences between the teacher and student network. The other reason is that the translator of the student can help the student network to understand teacher factors by mimicking the teacher factors. A downside of the proposed method is that the factor transfer requires the training of a paraphraser to extract factors and needs more parameters of the paraphraser and the translator. However, the convergence of the training for the paraphraser is very fast and additional parameters are not needed after training the student network. In our experiments, we showed the effectiveness of the factor transfer on various image classification datasets. Also, we verified that factor transfer can be applied to other domains than classification. We think that our method will help further researches in knowledge transfer.

在这项工作中，我们提出了因子转移，这是一种新颖的知识转移方法。与以前的方法不同，我们引入了包含教师网络释义信息的因子，这些信息是从释义器中提取的。学生通过因子转移比其他方法更容易理解教师网络信息的主要原因有两个。一方面，因子可以缓解教师和学生网络之间的固有差异。另一方面，学生的翻译器可以通过模仿教师因子来帮助学生网络理解教师因子。该方法的一个缺点是因子转移需要训练释义器以提取因子，并需要更多的释义器和翻译器参数。然而，释义器的训练收敛非常快，训练学生网络后不需要额外的参数。在我们的实验中，我们展示了因子转移在各种图像分类数据集上的有效性。此外，我们验证了因子转移可以应用于分类以外的其他领域。我们认为我们的方法将有助于知识转移的进一步研究。

Acknowledgments

致谢

This work was supported by Next-Generation Information Computing Development Program through the NRF of Korea (2017M3C4A7077582) and ICT R&D program of MSIP/IITP, Korean Government (2017-0-00306).

本研究得到了韩国国家研究基金会 (NRF) 下一代信息计算发展计划 (2017M3C4A7077582) 和韩国政府信息通信技术研发项目 (MSIP/IITP)(2017-0-00306) 的支持。

References

参考文献

[1] Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. The Journal of Machine Learning Research, 15(1):3563-3593, 2014.

[2] Bowen Baker, Otkrist Gupta, Nikhil Naik, and Ramesh Raskar. Designing neural network architectures using reinforcement learning. arXiv preprint arXiv:1611.02167, 2016.

[3] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. In Advances in neural information processing systems, pages 3123-3131, 2015.

[4] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+1 or-1. arXiv preprint arXiv:1602.02830, 2016.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/

[6] Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. Deep learning with limited numerical precision. In International Conference on Machine Learning, pages 1737-1746, 2015.

[7] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149, 2015.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770-778, 2016.

[9] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. Science, 313(5786):504–507, 2006.

[10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015.

[11] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.

[12] Gao Huang, Shichen Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Condensenet: An efficient densenet using learned group convolutions. CoRR, abs/1711.09224, 2017.

[13] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and $<$ 0.5mb model size. arXiv preprint arXiv:1602.07360, 2016.

[14] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research).

[15] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research).

[16] Hugo Larochelle, Dumitru Erhan, Aaron Courville, James Bergstra, and Yoshua Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. In Proceedings of the 24th international conference on Machine learning, pages 473-480. ACM, 2007.

[17] Vadim Lebedev and Victor Lempitsky. Fast convnets using group-wise brain damage. In Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on, pages 2554- 2564. IEEE, 2016.

[18] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In International Conference on Artificial Neural Networks, pages 52-59. Springer, 2011.

[19] Wing WY Ng, Guangjun Zeng, Jiangjun Zhang, Daniel S Yeung, and Witold Pedrycz. Dual autoencoders features for imbalance classification problem. Pattern Recognition, 60:875-889, 2016.

[20] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In European Conference on Computer Vision, pages 525-542. Springer, 2016.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91-99, 2015.

[22] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550, 2014.

[23] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 115(3):211-252, 2015.

[24] Hoo-Chang Shin, Matthew R Orton, David J Collins, Simon J Doran, and Martin O Leach. Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4 d patient data. IEEE transactions on pattern analysis and machine intelligence, 35(8):1930-1943, 2013.

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[26] Suraj Srinivas and R Venkatesh Babu. Data-free parameter pruning for deep neural networks. arXiv preprint arXiv:1507.06149, 2015.

[27] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4820-4828, 2016.

[28] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[29] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Advances in neural information processing systems, pages 3320-3328, 2014.

[30] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. arXiv preprint arXiv:1612.03928, 2016.

[31] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. arXiv preprint arXiv:1605.07146, 2016.

[32] Junbo Zhao, Michael Mathieu, and Yann LeCun. Energy-based generative adversarial network. arXiv preprint arXiv:1609.03126, 2016.

[33] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. arXiv preprint arXiv:1611.01578, 2016.