基于层间知识蒸馏的神经机器翻译

金畅，段仁翀，肖妮妮，段湘煜*
（苏州大学 自然语言处理实验室，江苏 苏州 215006）

摘要

神经机器翻译（NMT）通常采用多层神经网络模型结构，随着网络层数的加深，所得到的特征也越来越抽象，但是在现有的神经机器翻译模型中，高层的抽象信息仅在预测分布时被利用。为了更好地利用这些信息，本文提出了层间知识蒸馏，目的在于将高层网络的抽象知识迁移到低层网络，使低层网络能够捕捉更加有用的信息，从而提升整个模型的翻译质量。区别于传统教师模型和学生模型的知识蒸馏，层间知识蒸馏实现的是同一个模型内部不同层之间的知识迁移。通过在中文-英语、英语-罗马尼亚语、德语-英语三个数据集上的实验，结果证明层间蒸馏方法能够有效提升翻译性能，分别在中-英、英-罗、德-英上提升1.19，0.72，1.35的BLEU值，同时也证明有效地利用高层信息能够提高神经网络模型的翻译质量。

关键词： 知识蒸馏；神经网络；神经机器翻译；信息传递

Inter-layer Knowledge Distillation for Neural Machine Translation

Jin Chang, Duan Renchong, Xiao Nini, Duan Xiangyu
（Natural Language Processing Laboratory,Soochow University,Suzhou,Jiangsu 215006）

Abstract

Neural Machine Translation (NMT) usually adopts a multilayer neural network model structure, and as the number of network layers deepens, the features obtained become more and more abstract, but in existing neural machine translation models, the high-level abstract information is only utilized in predicting the distribution. To make better use of such information, this paper proposes Inter-layer Knowledge Distillation, which aims to transfer the abstract knowledge from the higher layer networks to the lower layer networks, so that the lower layer networks can capture more useful information and thus improve the translation quality of the whole model. Unlike the traditional knowledge distillation of teacher model and student model,Inter-layer Knowledge Distillation achieves knowledge migration between different layers within the same model. Through experiments on three datasets of Chinese-English, English-Romanian, and German-English, the results demonstrate that the inter-layer distillation method can

*indicates corresponding author.

effectively improve the translation performance by enhancing the BLEU values of 1.19, 0.72, and 1.35 on Chinese-English, English-Romanian, and German-English, respectively, and also demonstrate that the effective use of high-level information can improve the translation quality of neural network models.

Keywords: knowledge distillation , neural network , neural machine translation , information transfer

1 引言

目前，神经机器翻译（Neural Machine Translation，NMT）因为其优秀的性能和端到端的便捷性，在大多数领域已经取代了统计机器翻译(Brown et al., 1993; Koehn et al., 2003)，成为机器翻译领域的首选。神经机器翻译在近些年获得了迅速的发展(Sutskever et al., 2014; Bahdanau et al., 2014; Vaswani et al., 2017)，其中由Vaswani等(2017)研究者在2017年提出的Transformer模型成为了神经机器翻译的主流模型，区别于循环神经网络RNN（Recurrent Neural Network）(Cho et al., 2014)，长短期记忆网络LSTM（Long Short-Term Memory）(2014)，卷积神经网络CNN（Convolutional Neural Network）(Gehring et al., 2017)模型，Transformer对序列的建模过程完全基于注意力机制，并且能够并行化训练获得更好的性能。

Transformer采用编码器-解码器的框架。其编码器和解码器都由N层相同的网络层堆叠而成，对于给定的源端句子，编码端会按从低层到高层逐层生成语义向量表示，而解码端会根据编码端生成的语义向量和翻译历史来生成当前的译文，最终得到整个目标端句子。句子经过编码器或者解码器的多层网络后，其对应的语义信息会更加抽象，但是在现有的神经机器翻译系统中，高层抽象信息仅在预测分布时被利用。为了更好地利用这些信息，本文提出了层间知识蒸馏的方法，通过利用高层网络的抽象信息来辅导低层网络，使低层网络能够更加准确地捕获句子的信息，从而提升整个模型的性能，提高翻译质量。

本文的主要贡献可以总结为三个方面：1)提出层间知识蒸馏的方法来更好地利用高层神经网络的信息；2)经过层间知识蒸馏后得到的伪学生模型，解码端只有一层网络层，但是其性能能够接近甚至超越解码端拥有六层网络的基准模型，可以大大压缩模型的参数。3)针对层间知识蒸馏设计了三种简单的蒸馏方式，在保证模型参数不变，训练时间基本一致的条件下，显著地提升了神经机器翻译模型的性能。

本文的第2节将介绍多层网络信息传递的神经机器翻译的相关工作和知识蒸馏的相关研究；第3节将介绍层间知识蒸馏的相关细节；第4节将介绍具体的实验过程和实验结果；第5节对实验结果进行分析；第6节总结全文。

2 相关工作

本节将介绍多层网络信息传递的神经机器翻译的相关工作和知识蒸馏的相关研究。针对多层网络信息传递的研究工作：Dou等人(Dou et al., 2020)意识到Transformer在进行解码时只利用了编码器和解码器最后一层的信息，而忽略了低层网络的信息，为了提高各层网络信息的利用率，他们提出了多种层间隐状态融合的方法来缓解信息在深层网络传递过程中的退化问题。He等人(He et al., 2018)发现Transformer的解码器在计算编码器解码器注意力时始终只利用到了编码器最后一层的信息，而解码器的低层对于编码端高层的语义向量表示注意力不够集中，会损害模型的翻译质量，为此他们提出编码端和解码端在相对应的层之间进行注意力的计算。

神经网络模型随着网络的加深，随之倍增的参数量也使得模型的训练更加困难，在实际的应用中也会严重受到限制。而由Hinton等人(Hinton et al., 2015)提出的知识蒸馏方法则能够实现模型的压缩同时保证性能。知识蒸馏能够将复杂的教师模型的知识迁移到简单的学生模型中，一般来说，教师模型具有更强的性能和表现，而学生模型参数量更少，更容易部署。通过知识蒸馏，希望学生模型的表现可以逼近或是超过教师模型，从而实现用浅层网络达到深层网络类似的效果。

Hinton等人(2015)采用教师模型的预测分布作为监督，使学生模型尽量去靠近教师模型

的分布。Jiao等人(Jiao et al., 2019)认为注意力矩阵可以捕获丰富的语法信息，这对于自然语言理解至关重要，因此，他们提出将注意力矩阵作为知识来进行迁移，希望学生模型能更加关注教师模型所关注的部分。Sun等人(Sun et al., 2019)认为仅利用教师模型最后一层的预测分布是不够的，中间网络的特征同样有利用价值。Zhang等人(Zhang et al., 2019)提出了自蒸馏，目的是将网络高层知识迁移到低层网络，使得低层网络能够捕获更加有用的信息，从而提高模型的性能，相比于传统的知识蒸馏，训练效率更高。而本文的层间知识蒸馏方法与Zhang(2019)类似，直接利用网络层间的信息进行模型内部的知识迁移，区别主要有两点：1）应用领域的不同，Zhang等人(2019) 采用的是ResNet模型，所作的任务是图像分类，而我们是基于Transfomer模型的翻译任务，2）我们进一步提出了参数冻结的技术，防止模型内部知识在蒸馏的过程中相互干扰，从而影响翻译的性能。

3 层间知识蒸馏

在本节中我们将介绍有关层间知识蒸馏的细节，我们把模型分成两个部分，一个是伪教师模型，另一个是伪学生模型。伪教师模型由六层编码层和六层解码层组成，而伪学生模型由六层编码层和前M层解码层组成（M=1,2,3,4,5）。为了实现伪教师模型到伪学生模型知识的迁移，我们比较了三种不同的蒸馏方式，1) 层间隐状态的知识蒸馏HKD（Hidden Knowledge Distillation）。2)层间注意力矩阵的知识蒸馏AKD（Attention Knowledge Distillation）。3)层间概率分布的知识蒸馏PKD（Probability Knowledge Distillation）。在训练的过程中，伪教师模型通过正确的答案来进行训练，伪学生模型通过伪教师模型的蒸馏来进行训练。

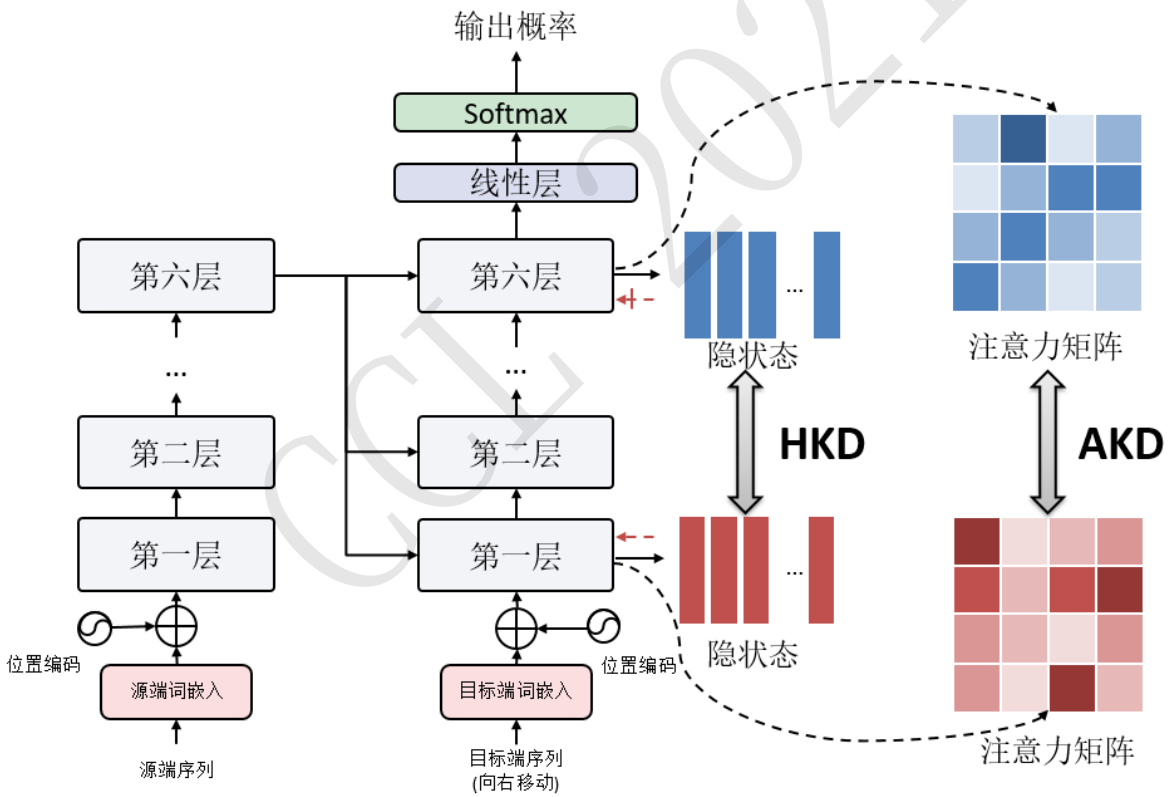


图 1: HKD和AKD蒸馏方式模型图

3.1 HKD

为了让伪学生模型去学习伪教师模型的知识，我们首先通过蒸馏层间的隐状态来完成，如图1所示。受Sun等人(2019)的启发，我们计算伪学生模型和伪教师模型经过归一化后的隐状态

之间的均方差损失来作为额外的损失函数，HKD损失函数公式如下：

$$L_{HKD} = \sum_{i=1}^{N} \left\| \frac{h_{i,j}^{s}}{\left\| h_{i,j}^{s} \right\|^{2}} - \frac{h_{i}^{t}}{\| h_{i}^{t} \|^{2}} \right\|^{2} \quad (1 <= j <= M-1) \tag{1}$$

其中M表示解码器的层数，$h_{i,j}^{s}$表示解码器第j层的隐状态，h_{i}^{t}表示解码器最后一层的隐状态，N表示训练样本的个数，上标s和t来区分伪学生模型和伪教师模型。HKD方式蒸馏总的优化目标函数为：

$$L = L_{CE} + \alpha L_{HKD} \tag{2}$$

$$L_{CE} = -\sum_{t=1}^{T} log\left(P\left(y_t \left| y_{<t}; X\right.\right)\right) \tag{3}$$

L_{CE}为标准的机器翻译优化目标函数，X表示源端句子，T为目标端句子的长度，$P\left(y_t \left| y_{<t}; X\right.\right)$表示的是目标端第t个词的概率。$\alpha$是超参数，用来控制两个损失函数的比例。

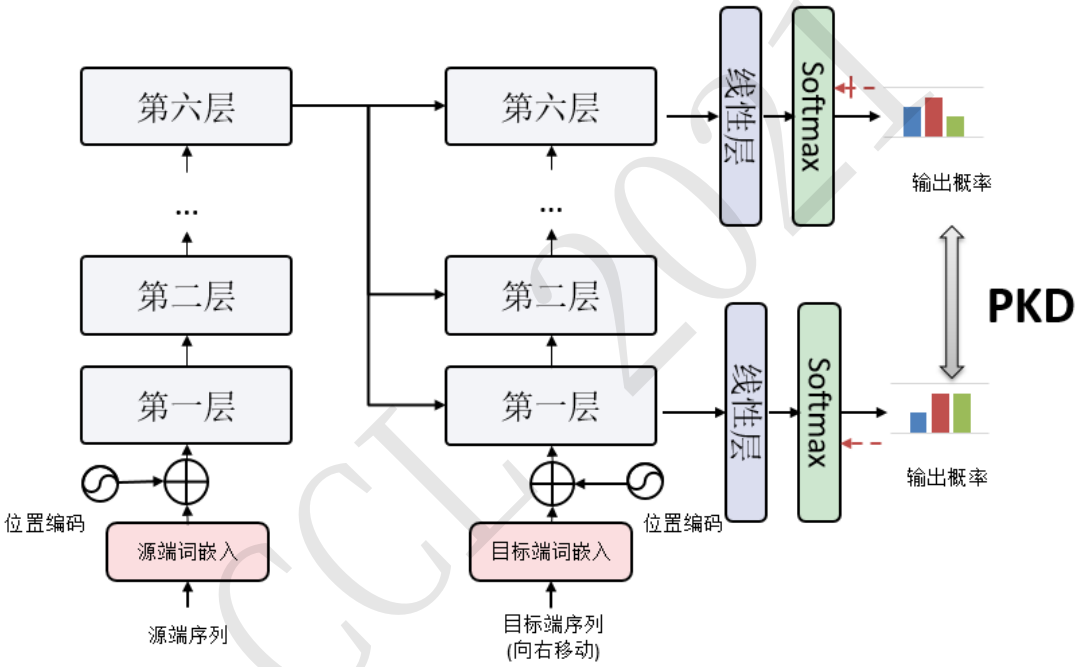


图 2: PKD蒸馏方式模型图

3.2 AKD

受到Jiao等人(2019)的启发，因为注意力矩阵携带着大量的语法信息，所以我们希望能将伪教师模型的这种知识迁移到伪学生模型中，让伪学生模型更多的关注伪教师模型关注的部分，如图1所示。Transformer中采用的是多头注意力机制，多头注意力机制的具体公式如下所示：

$$MultiHead\left(Q,K,V\right) = concat\left(head_1,...,head_n\right)W^{O} \tag{4}$$

$$head_i\left(Q,K,V\right) = Attention\left(Q\,W_i^{Q}, K\,W_i^{K}, V\,W_i^{V}\right) \tag{5}$$

$$Attention\left(Q,K,V\right) = softmax\left(\frac{Q\,K^{T}}{\sqrt{d_k}}\right)V \tag{6}$$

其中$W^O \in R^{hd_v \times d_{model}}$，$W^Q \in R^{d_{model} \times d_k}$，$W^k \in R^{d_{model} \times d_k}$，$W^V \in R^{d_{model} \times d_v}$表示的是线性层的权重，head表示多头注意力的个数。d_k表示K的维度，相当于一个缩放的操作。AKD损失函数公式为：

$$L_{AKD} = MSE\left(\sum_{i=1}^{h} A_{i,j}^{s}, \sum_{i=1}^{h} A_{i}^{t}\right) \ (1 <= j <= M-1) \tag{7}$$

其中h为注意力头的个数，M表示解码器的层数，$A_{i,j}^{s}$为解码器第j层第i个头的编码器解码器注意力矩阵，A_{i}^{t}为解码器最后一层第i个头的编码器解码器注意力矩阵，上标s和t来区分伪学生模型和伪教师模型，$MSE(.)$表示均方差损失函数。AKD方式蒸馏总的优化目标函数为：

$$L = L_{CE} + \alpha L_{AKD} \tag{8}$$

3.3 PKD

本小节我们将介绍概率分布知识蒸馏PKD。如图2所示，与伪教师模型类似，对于伪学生模型我们同样为其添加了一层线性层作为输出层，以得到整个词表中每个词的概率。伪教师模型和伪学生模型通过线性层和softmax层分别得到其预测分布，PKD通过拉近其预测分布来实现伪教师模型到伪学生模型的知识迁移。在实验中我们共享了伪教师模型和伪学生模型输出层的参数。PKD损失函数公式如下：

$$L_{PKD} = KL\left(q_j^s, q^t\right) \ (1 <= j <= M-1) \tag{9}$$

$$q^i = softmax\left(h^i W\right) \tag{10}$$

其中h^i表示伪学生模型或者伪教师模型的隐状态，W为输出层的权重，$KL(.)$表示Kullback-Leibler 散度，用来衡量两个概率分布的差异。M表示解码器的层数，q_j^s表示解码器第j层的预测分布，q^t表示解码器最后一层的预测分布，上标s和t来区分伪学生模型和伪教师模型。PKD方式蒸馏总的优化目标函数为：

$$L = L_{CE} + \alpha L_{PKD} \tag{11}$$

3.4 参数冻结

为了使得深层网络的知识蒸馏到浅层网络，本文采用了三种不同的蒸馏方式，但是由于伪学生模型和伪教师模型处于同一个网络中，网络高层和网络低层的信息是相互关联，相互影响的。这就有可能使得在进行层间知识蒸馏时，低层网络的知识流向高层网络，从而破坏高层网络更加抽象的知识，影响翻译质量。基于这个原因，我们使用了参数冻结的技术。如图2虚线箭头所示，当网络计算完梯度，进行反向传播的时候，我们冻结掉伪教师模型的参数，仅仅让伪学生模型去更新参数。为了验证使用参数冻结方法的有效性，我们对是否使用参数冻结进行了对比实验，实验结果见5.1节消融分析。本文除了消融实验之外，其余实验都默认使用了参数冻结的技术。

4 实验

4.1 数据集

本文在中-英，英-罗和德-英翻译任务上进行实验来验证我们方法的有效性。德-英翻译任务中使用的数据集选自IWSLT'14（2014 International Workshop on Spoken Language Translation，IWSLT'14），包含平行语料17万句，利用MOSES[0]的处理脚本进行分词[1]和过滤句子长度超过175的句子，得到16万的平行数据，测试集取自IWSLT14.TED.dev2010、IWSLT14.TED.tst2010、IWSLT14.TED.dev2012、IWSLT14.TED.tst2011和IWSLT14.TED.tst2012共6750句。并且对英语和德语语料分别进行了字节对编码(Sennrich et al., 2015)（Byte Pair Encoding，BPE）获得各自的词表，英语词表大小为6628，德语词表为8844。

英-罗翻译任务中使用的数据集来自WMT'16（2016 Third Conference on Machine Translation，WMT'16 ），包含训练语料61万句，我们采用newsdev2016和newstest2016分别作为验

[0]http://www.statmt.org/moses/

[1]https://github.com/moses-smt/mosesdecoder/scripts/tokenizer/tokenizer.perl

证集和测试集。英语语料和罗马尼亚语料使用联合字节对编码处理，获得大小为3.5万的联合词表。

中-英翻译任务中使用的数据集来自语言数据联盟（Linguistic Data Consortium, LDC），其中训练集包括125万句，我们使用NIST06（1664句）作为验证集，使用NIST02、NIST03、NIST04、NIST05、NIST08（分别包含平行句对878、919、1788、1082、1357句）作为测试集。分别对中英文语料进行字节对编码处理，其中中文词表为4.2万，英文词表为3.1万。

4.2 实验参数

实验方法的实现是基于开源的fairseq[2](Ott et al., 2019)框架，使用Transformer作为我们的基准系统，其结构包括六层编码器和六层解码器。对于中-英数据集我们设置每个batch中最多包括7500个词，学习率和dropout分别为0.0005 和0.3,前馈神经网络的维度为2048，注意力头的个数为8。对于德-英数据集每个batch中最多包括8192个词，学习率设置为0.0005，dropout为0.1，前馈神经网络的维度为1024，注意力头的个数为4。对于英-罗数据集我们同样设置每个batch中的最大词数为8192，学习率设置为0.0003，dropout为0.3，前馈神经网络维度和注意力头个数分别为2048和8。所有的实验均使用了标签平滑(Szegedy et al., 2016)且值为0.1，全采用Adam优化器(Kingma and Ba, 2014)和逆平方根学习率衰减，其中优化器参数为$\beta_1 = 0.9$，$\beta_2 = 0.98$，$\varepsilon = 10^{-9}$。在解码时采用束搜索(Wiseman and Rush, 2016)（beam search）的解码方式，除了英-罗的搜索宽度为4之外，中-英和英-德宽度设置都为5。其他实验参数与Vaswani等人(2017)相同。

对于模型评估，在所有的翻译任务中，均采用不区分大小写的机器双语互译评估(Papineni et al., 2002)（BLEU）得分来评估模型的翻译质量。对于每个翻译任务，均保留最后十个模型进行平均，并使用multi-bleu.perl[3]脚本进行评测。

4.3 实验结果

表1列出了本文提出的三种不同的蒸馏方式在各个数据集上的实验结果，均在原有的基准模型上取得了一定的提升。其中PKD的效果最为明显，分别在中-英，英-罗，德-英翻译任务上提升了1.19，0.72，1.35个BLEU值，这表明利用高层网络的预测分布有利于模型翻译性能的提升。对于HKD蒸馏方式，在英-罗，德-英翻译任务上相比基准模型也分别获得了0.65，0.69个BLEU的提升，表明利用高层网络的隐状态信息同样有利于改善模型的翻译性能。AKD蒸馏方式也均在基准模型的基础上取得了一定的提升，说明利用高层网络的注意力矩阵对神经网络模型也是有利的。实验默认采用解码器第一层作为伪学生模型来进行层间知识蒸馏，同时我们也探索了不同低层作为伪学生模型进行层间知识蒸馏时对于翻译性能的影响，详细实验结果见表2。

实验系统	中-英						英-罗	德-英
	NIST02	NIST03	NIST04	NIST05	NIST08	AVG		
基准系统	47.29	46.57	47.52	46.56	37.79	45.15	34.53	34.14
HKD	47.27	46.77	47.77	47.62	38.17	45.52	35.18	34.83
AKD	47.59	46.17	47.47	47.46	37.72	45.28	34.89	34.33
PKD	47.99	47.95	48.46	48.28	39.03	**46.34**	**35.25**	**35.49**

表 1: 各模型在各个数据集上的翻译性能

表2列出了Transformer不同低层作为伪学生模型来进行层间知识蒸馏的实验结果。实验选用的是德-英数据集，均采用PKD方式来进行蒸馏。实验证明不管利用哪一层作为伪学生模型，都有利于提升整体模型的翻译性能，特别是当伪学生模型和伪教师模型层间间隔较大时，性能最好。从表2可以看出，基准模型在验证集和测试集上的得分分别为35.76和34.14，而选取第一层网络做为伪学生模型进行层间知识蒸馏后验证集和测试集的得分为36.91和35.49，BLEU的得分分别提升了1.15和1.35，表明层间知识蒸馏确实有利于模型翻译性能的提升。

[2]https://github.com/pytorch/fairseq
[3]https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

伪学生模型	无	第一层	第二层	第三层	第四层	第五层
验证集	35.76	**36.91**	36.83	36.67	36.54	36.31
测试集	34.14	**35.49**	35.14	35.04	34.97	34.77

表 2: 伪学生模型的选取对德英翻译效果的影响

我们比较了伪学生模型和基准模型的解码效果，实验结果如表3所示。实验采用了效果最好的PKD蒸馏方式，并且在各个数据集上都进行了比较。观察表3，可以发现经过层间知识蒸馏后得到的伪学生模型解码的效果非常好，基本接近甚至超越基准模型的表现。值得一提的是在德-英数据集上，使用伪学生模型解码可以达到34.82的BLEU值，超过基准模型0.68分，并且伪学生模型解码端仅仅包括一层网络层，而基准模型解码端包括六层网络层，可以大大压缩模型参数。

实验系统	中-英						英-罗	德-英
	NIST02	NIST03	NIST04	NIST05	NIST08	AVG		
基准系统	47.29	46.57	47.52	46.56	37.79	45.15	**34.53**	34.14
伪学生模型	47.21	46.72	48.15	47.32	36.83	**45.25**	34.11	**34.82**

表 3: 伪学生模型在各数据集上的翻译性能

5 实验分析

5.1 消融分析

为了验证使用参数冻结的有效性，我们对是否进行参数冻结进行了对比实验，实验结果如表4所示。实验采用的是PKD的蒸馏方式，并且在各个数据集上都进行了对比实验。在未进行参数冻结的层间知识蒸馏中，伪学生模型和伪教师模型的参数都未固定，所以它们之间是相互学习的过程，在这个过程中学生模型的错误知识可能会误导教师模型，从而损害翻译的质量。针对这个问题，我们在模型计算完梯度，进行反向传播的过程中冻结掉伪教师模型的参数，让其不受学生模型的影响，只允许知识从伪教师模型单向流入伪学生模型。从表4中可以看出，经过参数冻结后的层间知识蒸馏的BLEU得分都超过了未经过参数冻结的层间知识蒸馏，分别在中-英，英-罗，德-英翻译任务上领先0.41，0.51，0.88分。实验证明了在层间知识蒸馏的过程中使用参数冻结的重要性。

实验系统	中-英						英-罗	德-英
	NIST02	NIST03	NIST04	NIST05	NIST08	AVG		
基准系统	47.29	46.57	47.52	46.56	37.79	45.15	34.53	34.14
参数冻结	47.99	47.95	48.46	48.28	39.03	**46.34**	**35.25**	**35.49**
未参数冻结	48.02	46.87	48.87	47.48	38.44	45.93	34.74	34.61

表 4: 冻结参数对翻译效果的影响

5.2 模型复杂度分析

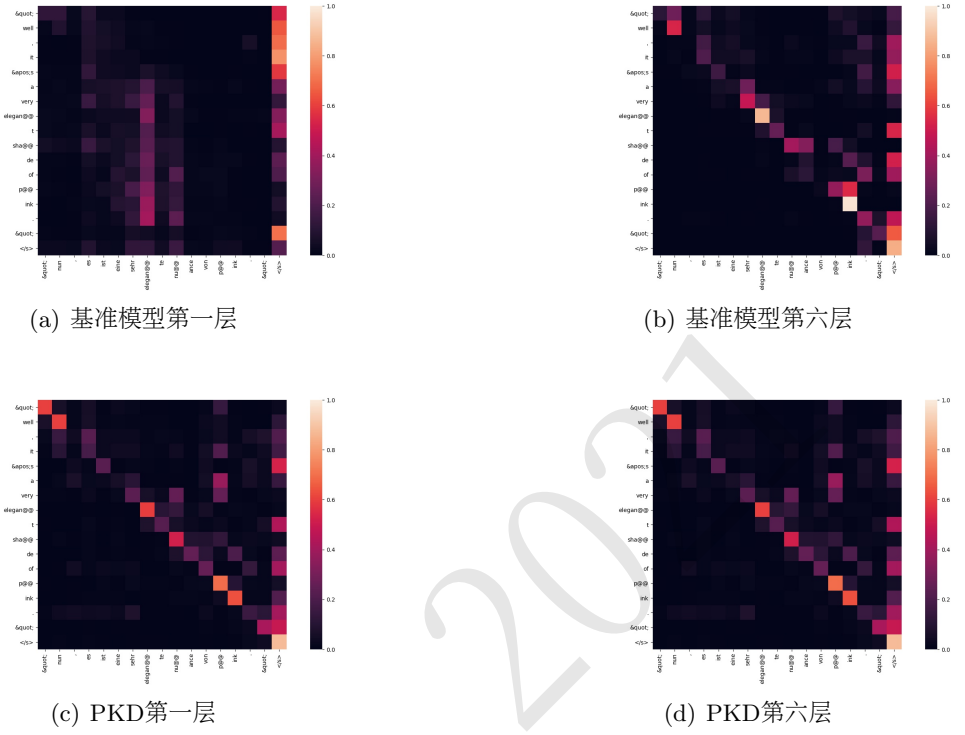
表5列出了本文采用的三种不同蒸馏方式的复杂度，实验选用的是中-英数据集，模型的训练基于4块NVIDIA Tesla V100 GPUs。如表5所示，使用HKD，AKD的蒸馏方式可以保证模型参数和训练速度基本不变，但是带来的提升不及PKD，而PKD因为要分别计算伪学生模型和伪教师模型的预测分布，所以会略微地降低训练速度。

5.3 模型注意力分析

本小节将从注意力机制的角度进行分析为何一层解码器的伪学生模型可以达到六层解码器的基准模型的性能。实验采用的是德-英数据集。我们首先可视化了基准模型第一层和第六层的编码器解码器注意力矩阵图，如图3(a)和图3(b)所示。可以发现基准模型解码端第一层相比解

实验系统	模型参数	训练速度（词/秒）
基准系统	97M	180739
HKD	97M	177671
AKD	97M	179279
PKD	97M	151875

表 5: 模型复杂度分析



(a) 基准模型第一层　(b) 基准模型第六层

(c) PKD第一层　(d) PKD第六层

图 3: 注意力矩阵图

码端第六层，目标端句子对于源端句子的注意力十分发散，无法从源端句子中选择对于生成当前词更关键的信息，所以导致翻译效果不佳。在采用PKD方式进行层间知识蒸馏后，我们同样也可视化了伪学生模型和伪教师模型的编码器解码器注意力矩阵图，从图3(c)和图3(d)中可以看出，伪学生模型的目标端句子对于源端句子注意力非常集中，和伪教师模型的注意力矩阵相差不大，这在生成当前词时有利于模型只关注更加重要的信息，提升神经网络翻译的性能。

5.4 混合蒸馏分析

我们对不同的蒸馏方式进行了组合，以探索更加高效的蒸馏方式。实验选用的是德-英数据集，并和三种蒸馏中效果最好的PKD方式进行了对比实验。如表6所示，组合PKD和HKD，PKD和AKD或者组合三种蒸馏方式一同进行蒸馏，得到的实验结果和单独使用PKD相差无几，而组合AKD和HKD进行蒸馏得到的实验结果和二者中效果较好的HKD基本一致。我们猜测由于HKD相比PKD,蒸馏的对象仅由伪学生模型和伪教师模型的隐状态变为了概率分布，而伪学生模型和伪教师模型的隐状态是经过参数共享的输出层所得到的概率分布，所以相比单独使用PKD，组合PKD和HKD并不会使伪学生模型学习到更多的知识。另外从图3(c)和图3(d)中可以看出，经过PKD之后的伪学生模型和伪教师模型的注意力矩阵十分相似，所以相比单独使用PKD，组合PKD和AKD也不会使伪学生模型捕捉到新的信息。

6 结束语

本文为了更好地利用高层神经网络的抽象信息，提出了层间知识蒸馏，目的在于把高层更加抽象的信息迁移到低层部分，使得低层网络能够捕获更多有用的特征，从而提升整个模型的

第二十届中国计算语言学大会论文集，第166页-第175页，呼和浩特，中国，2021年8月13日至15日。

实验系统	PKD	PKD+HKD	PKD+AKD	AKD+HKD	PKD+HKD+AKD
测试集	**35.49**	35.44	35.39	34.86	35.32

表 6: 混合蒸馏在德英翻译任务上的实验结果

翻译性能。我们以Transformer作为基准模型，和本文提出的三种蒸馏方式进行了对比，实验表明高层网络信息对低层网络信息的迁移可以提升神经机器翻译模型的性能。在未来的工作中，将尝试更多样性的层间知识迁移方式，并且将尝试把层间知识蒸馏应用到预训练模型领域，提升预训练模型的整体表现。

参考文献

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Peter F Brown, Stephen A Della Pietra, Vincent J Della Pietra, and Robert L Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Zi-Yi Dou, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2020. Exploiting deep representations for natural language processing. *Neurocomputing*, 386:1–7.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *International Conference on Machine Learning*, pages 1243–1252. PMLR.

Tianyu He, Xu Tan, Yingce Xia, Di He, Tao Qin, Zhibo Chen, and Tie-Yan Liu. 2018. Layer-wise coordination between encoder and decoder for neural machine translation. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, pages 7955–7965.

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

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Philipp Koehn, Franz J Och, and Daniel Marcu. 2003. Statistical phrase-based translation. Technical report, UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Sam Wiseman and Alexander M Rush. 2016. Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960*.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. 2019. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3713–3722.