软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2023,34(4):1837−1849 [doi: 10.13328/j.cnki.jos.006413]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

跨模态信息融合的端到端语音翻译*

刘宇宸[1,2], 宗成庆[1,2]

[1](模式识别国家重点实验室 (中国科学院自动化研究所), 北京 100190)

[2](中国科学院大学 人工智能学院, 北京 100049)

通信作者: 宗成庆, E-mail: cqzong@nlpr.ia.ac.cn

摘　要: 语音翻译旨在将一种语言的语音翻译成另一种语言的语音或文本. 相比于级联式翻译系统, 端到端的语音翻译方法具有时间延迟低、错误累积少和存储空间小等优势, 因此越来越多地受到研究者们的关注. 但是, 端到端的语音翻译方法不仅需要处理较长的语音序列, 提取其中的声学信息, 而且需要学习源语言语音和目标语言文本之间的对齐关系, 从而导致建模困难, 且性能欠佳. 提出一种跨模态信息融合的端到端的语音翻译方法, 该方法将文本机器翻译与语音翻译模型深度结合, 针对语音序列长度与文本序列长度不一致的问题, 通过过滤声学表示中的冗余信息, 使过滤后的声学状态序列长度与对应的文本序列尽可能一致; 针对对齐关系难学习的问题, 采用基于参数共享的方法将文本机器翻译模型嵌入到语音翻译模型中, 并通过多任务训练方法学习源语言语音与目标语言文本之间的对齐关系. 在公开的语音翻译数据集上进行的实验表明, 所提方法可以显著提升语音翻译的性能.

关键词: 语音翻译; 神经机器翻译; 端到端模型; 多模态学习

中图法分类号: TP391

中文引用格式: 刘宇宸, 宗成庆. 跨模态信息融合的端到端语音翻译. 软件学报, 2023, 34(4): 1837–1849. http://www.jos.org.cn/1000-9825/6413.htm

英文引用格式: Liu YC, Zong CQ. End-to-end Speech Translation by Integrating Cross-modal Information. Ruan Jian Xue Bao/Journal of Software, 2023, 34(4): 1837–1849 (in Chinese). http://www.jos.org.cn/1000-9825/6413.htm

End-to-end Speech Translation by Integrating Cross-modal Information

LIU Yu-Chen[1,2], ZONG Cheng-Qing[1,2]

[1](National Laboratory of Pattern Recognition (Institute of Automation, Chinese Academy of Sciences), Beijing 100190, China)

[2](School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Speech translation aims to translate the speech in one language into the speech or text in another language. Compared with the pipeline system, the end-to-end speech translation model has the advantages of low latency, less error propagation, and small storage, so it has attracted much attention. However, the end-to-end model not only requires to process the long speech sequence and extract the acoustic information, but also needs to learn the alignment relationship between the source speech and the target text, leading to modeling difficulty with poor performance. This study proposes an end-to-end speech translation model with cross-modal information fusion, which deeply combines text-based machine translation model with speech translation model. For the length inconsistency between the speech and the text, a redundancy filter is proposed to remove the redundant acoustic information, making the length of filtered acoustic representation consistent with the corresponding text. For learning the alignment relationship, the parameter sharing method is applied to embed the whole machine translation model into the speech translation model with multi-task training. Experimental results on public speech translation data sets show that the proposed method can significantly improve the model performance.

Key words: speech translation; neural machine translation; end-to-end model; multi-modal learning

* 基金项目: 国家自然科学基金重点项目 (U1836221)
收稿时间: 2020-12-29; 修改时间: 2021-03-13, 2021-05-21; 采用时间: 2021-06-30; jos 在线出版时间: 2022-07-15
CNKI 网络首发时间: 2022-11-15

语音翻译旨在将一种自然语言 (一般称为源语言) 的语音翻译成另一种自然语言 (一般称为目标语言) 的语音或文本[1]. 语音翻译是一项集语音识别、口语翻译和语音合成 3 种技术为一体的综合技术, 可以广泛应用于会议演讲、商业会谈、跨境电商、出国旅游等各个领域. 随着全球化活动的不断拓展和日益频繁, 语音翻译已经成为当前移动互联时代最受瞩目的影响人类生活的重大技术之一[2].

传统的语音翻译系统通常由语音识别、机器翻译和语音合成等多个模块级联组成[1]. 给定语音输入, 语音识别模块首先将其转录为源语言文本, 之后机器翻译模块将转录文本翻译为目标语言文本, 最终由语音合成模块输出目标语音. 尽管这种系统已经取得了很大的进展, 是目前商业系统采用的主流范式, 但是这种方法存在一些明显的局限性. 一方面, 语音翻译系统中的不同模块是分别独立训练的, 容易出现数据分布不一致和错误累积问题, 导致翻译性能严重下降; 另一方面, 系统需要串行完成转录、翻译和合成等多个过程, 面临着计算复杂度高、存储空间大和延迟时间长等缺陷. 因此, 语音翻译系统亟待一种新的实现范式.

端到端的语音翻译方法在理论上可以缓解传统级联式系统面临的缺陷, 因此逐渐受到研究者们的关注[3−5]. 该方法通过直接建立源语言语音与目标语言文本之间的映射关系, 端到端地实现从源语言语音到目标语言文本的翻译过程. 近年来, 基于注意力机制的编-解码神经网络模型得到了迅速发展, 并已成为语音识别和机器翻译等任务的主流方法[6,7]. 有研究者对这种方法进行了拓展, 他们将语音识别和机器翻译组合在一个单一的模型中, 构建了端到端的语音到文本的翻译模型[3,4]. 但是, 直接实现源语言语音到目标语言文本的建模过程充满了困难和挑战, 因为这种模型需要解决两个复杂的问题: (1) 不同于文本机器翻译, 语音翻译需要处理一个长度远长于对应文本的语音序列, 并学习其中的声学信息; (2) 同时也不同于语音识别, 语音翻译需要学习源语言语音和目标语言文本之间非单调性的对齐关系, 特别是对于差异较大的两种语言之间的翻译, 涉及长距离的调序问题.

为此, 本文提出了一种融合跨模态信息的端到端的语音翻译方法, 该方法能够将语音翻译模型与文本机器翻译模型深度结合. 针对上述第 1 个问题, 本文通过信息过滤器过滤声学表示中的冗余信息, 使得声学状态序列的长度与对应文本序列尽可能一致; 针对上述第 2 个问题, 本文采用参数共享策略将文本机器翻译模型嵌入到语音翻译模型中, 并提出模态迁移的方法对语音和文本模态的表示进行约束. 具体地, 本文将语音翻译的编码器拆分成 3 个模块, 分别是声学特征提取器、冗余信息过滤器和语义特征编码器. 其中声学特征提取器用于生成源语言语音的声学状态序列, 该模块联合连接时序分类 (connectionist temporal classification, CTC) [8]损失函数进行训练, 可作为独立的语音识别模型. 由于声学状态序列的长度远长于对应的文本序列, 阻碍了源端和目标端对齐关系的学习, 因此, 本文提出了冗余信息过滤器. 该模块基于 CTC 输出的概率进行判断, 当非空白标签的预测概率超过触发阈值时, 对应时刻的隐层状态才会被抽取, 该过程可以显著地缩短声学状态序列的长度. 之后语义特征编码器对过滤后的声学状态序列进一步编码, 以提取其中的语义信息. 为了辅助语音翻译模型学习输入序列和输出序列之间的对齐关系, 本文通过参数共享策略将整个文本机器翻译模型嵌入到语音翻译模型中, 并使用多任务学习的方法同时训练语音翻译任务和文本机器翻译任务. 本文在 3 个公开的语音翻译数据集上进行了实验, 结果表明, 所提方法能够显著提升语音翻译模型的性能, 同时兼具模块化和灵活性的特点, 可以方便地使用外部的语音识别数据或文本机器翻译数据进行训练, 极大地提升了模型的可拓展性.

本文第 1 节综述语音翻译研究的相关工作. 第 2 节介绍语音翻译任务的定义和方法所涉及的注意力机制及 CTC 损失函数. 第 3 节描述本文所提出的模型和多任务训练方法. 第 4 节描述实验设置, 对实验结果进行对比分析. 最后一节给出论文的结语和对未来研究的展望.

1 相关工作

传统的语音翻译系统通常由独立训练的语音识别模块和机器翻译模块级联组成. 虽然这种系统构建简单, 容易部署, 但是不可避免地面临着错误累积的问题. 研究者们多关注于如何将两个模型紧密结合以缓解语音识别错误对机器翻译的影响. 例如, Saleem 等人[9]、Sperber 等人[10]和 Zhang 等人[11]提出了将包含多个语音识别输出的词格网络输入到机器翻译模型的方法. Tsvetkov 等人[12]和 Cheng 等人[13]将模拟的语音识别错误引入到机器翻译语料中训练, 得到对噪声鲁棒的机器翻译模型. Kano 等人[14]和 Anastasopouslos 等人[15]提出了可微分的级联式翻

译系统, 以达到联合优化语音识别和机器翻译模型的目的.

不同于传统的级联式翻译系统, 端到端的语音翻译模型具有延迟低、错误累积少和存储小等潜在优势. Zong 等人在 1999 年就提出了采用端到端的方式直接实现语音到文本翻译的设想[16], 但限于当时的条件, 那仅是一种设想. 近年来, 随着深度学习技术的快速发展, 端到端的语音翻译方法逐渐成为可能. Duong 等人[17]和 Bérard 等人[18]使用基于注意力机制的编码器-解码器在不使用任何中间文本表示的情况下实现了端到端的语音到文本的翻译. 随后这种方法越来越多地受到学术界和产业界的关注. 但是, 这种方法面临着数据资源严重匮乏的问题. 为此, 研究者们相继提出了预训练和多任务学习等方法以引入额外数据和其他任务, 从而提升翻译模型的性能[3−5]. 此外, 知识蒸馏[19]、子模块预训练[20]、交互式解码[21]、课程学习[22]、元学习[23]等一系列方法被陆续提出. 然而现有工作无法在统一的模型中有效利用语音和文本两种不同模态的表示, 导致文本翻译模型学习到的语义知识难以融合到语音翻译模型中. 与现有工作相比, 本文从源端特征编码的角度出发, 通过缩小语音表示和文本表示之间的模态差异性, 实现不同模态特征层面的迁移, 进而实现不同模态对应模型的深度结合, 最终借助语音识别和文本翻译模型辅助提升语音翻译模型的译文质量.

2 研究背景

2.1 任务定义

语音翻译任务旨在将源语言语音翻译到目标语言文本. 语音翻译的训练数据通常由包含源语言语音, 源语言转录文本和目标语言文本的三元组构成, 可以形式化的表示为 $\mathcal{D}_{\mathrm{ST}}=\{(\boldsymbol{s},\boldsymbol{x},\boldsymbol{y})\}$. 其中 $\boldsymbol{s}=[s_1,s_2,\ldots,s_{T_s}]$ 是从语音信号中提取到的语音特征序列, $\boldsymbol{x}=[x_1,x_2,\ldots,x_{T_x}]$ 是源语言转录文本序列, $\boldsymbol{y}=[y_1,y_2,\ldots,y_{T_y}]$ 是目标语言翻译文本序列, T_s,T_x 和 T_y 分别是语音特征序列, 源语言转录文本和目标语言翻译文本序列的长度, 其中 $T_s\gg T_x$. 本文的目标是学习一个端到端的语音翻译模型, 在给定源语言语音特征序列 $\boldsymbol{s}$ 的情况下直接生成目标语言翻译文本序列 $\boldsymbol{y}$. 在训练过程中可以借助源语言转录文本 $\boldsymbol{x}$, 也可以使用外部的语音识别领域训练数据 $\mathcal{D}_{\mathrm{ASR}}=\{(\boldsymbol{s}',\boldsymbol{x}')\}$ 和文本机器翻译领域训练数据 $\mathcal{D}_{\mathrm{MT}}=\{(\boldsymbol{x}'',\boldsymbol{y}'')\}$.

2.2 Transformer 模型

本文以 Transformer 模型作为语音识别, 文本机器翻译以及端到端的语音翻译模型的主体框架. 该模型由 Vaswani 等人[24]于 2017 年提出, 现已成为主流的文本机器翻译模型, 在语音识别等任务上也达到了当前的最优效果. Transformer 模型遵循编码器-解码器框架, 其中编码器首先将输入序列映射到连续空间表示, 解码器依据该连续空间表示从左到右依次解码生成目标序列. 编码器有两个子网络, 分别是自注意力网络和前馈网络; 解码器则由自注意力网络, 跨注意力网络和前馈网络组成. 每个子网络之间使用了残差连接和层级正则化. Transformer 模型使用注意力机制对序列进行编码, 其计算公式如下:

$$Attn(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{1}$$

其中, $\boldsymbol{Q}$, $\boldsymbol{K}$ 和 $\boldsymbol{V}$ 分别表示查询, 键, 值, d_k 代表 $\boldsymbol{K}$ 的维度. 对于编码器, $\boldsymbol{Q}$, $\boldsymbol{K}$ 和 $\boldsymbol{V}$ 由同一隐层状态编码经过不同的线性映射得到; 对于解码器, $\boldsymbol{Q}$ 来自解码器底层的隐层状态编码, $\boldsymbol{K}$ 和 $\boldsymbol{V}$ 来自编码器生成的隐层状态编码.

前馈网络则由两个线性变换网络层组成, 并使用 ReLU 激活函数. 其计算过程如下:

$$FFN(\boldsymbol{x})=\max(0,\boldsymbol{x}\boldsymbol{W}_1+\boldsymbol{b}_1)\boldsymbol{W}_2+\boldsymbol{b}_2 \tag{2}$$

其中, $\boldsymbol{W}_1,\boldsymbol{W}_2,\boldsymbol{b}_1,\boldsymbol{b}_2$ 为模型需要学习的线性变换参数矩阵.

2.3 连接时序分类 (CTC) 损失

CTC 损失是由 Graves 等人[8]提出的一种可用于处理时序可变的序列标注问题的损失函数, 在语音识别和手写体识别等领域有着广泛的应用. 传统的序列标注算法需要每一个时刻的输入与输出标签完全对齐, 而 CTC 在计算时扩展了标签集合, 允许预测序列中出现空白和重复标签. 在使用扩展标签集合对序列进行标注后, 所有能够通过映射函数转换为目标序列的预测序列都可以作为正确的预测序列, 其中映射函数可以是去除预测序列的空白标

签和重复标签的操作. 例如, 在语音识别任务中, 传统的语音识别声学模型在训练时, 需要对语音片段进行对齐预处理, 在获取到每一帧语音特征对应的发音音素后才能进行后续的训练, 而采用 CTC 损失函数的声学模型在训练时可以采用完全端到端的方式进行, 只需要给定语音特征序列和目标输出序列, 就能够使模型自动学习到输入序列和输出序列间的对齐关系.

3 提出的方法

本文提出了一种跨模态信息融合的端到端语音翻译模型, 其整体框架如图 1 所示. 该模型由声学特征提取器, 冗余信息过滤器, 语义特征编码器和解码器组成, 能够与文本机器翻译模型深度结合并进行多任务训练. 其中声学特征提取器, 语义特征编码器和解码器均基于 Transformer 模型, 由多头注意力网络和前馈网络组成.

与语音识别模型中的编码器只负责提取声学特征不同, 语音翻译模型中的编码器不仅需要编码语音中的声学特征, 还需要抽取其中的语义信息, 因此面临着更重的负担也更加难以训练. 为此, 本文将编码器拆分成 3 个模块, 分别是声学特征提取器, 冗余信息过滤器和语义特征编码器. 首先, 使用声学特征提取器提取原始语音序列中的声学信息并将其映射到对应源语言文本的隐层状态序列, 该模块结合 CTC 损失函数可单独作为语音识别模块训练; 由于上述隐层状态序列中存在着大量的冗余信息, 其长度往往是对应文本长度的数倍, 本文使用冗余信息过滤器过滤其中的冗余信息, 该过程可以显著地缩短该状态序列的长度, 使之与对应的源语言文本匹配; 继而, 语义特征编码器对过滤后的状态序列进一步编码, 获得包含语义信息的表示. 最终, 解码器基于注意力机制获取语义特征编码器得到的语义表示并输出最终的目标语言翻译结果. 本节将详细介绍每一个部分.

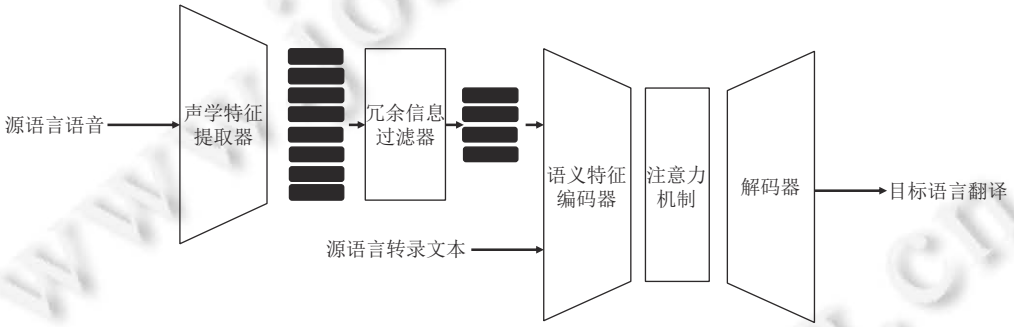

图 1 基于跨模态融合的端到端语音翻译模型框架

3.1 声学特征提取器

声学特征提取器用于提取源语言语音序列中的声学信息, 它以源语言语音序列 $\boldsymbol{s}$ 作为输入, 生成声学隐层状态序列 $\boldsymbol{h}$. 具体而言, 源语言语音序列首先经过梅尔滤波器组被转换到频域特征, 继而经过前端网络被映射到与模型同维度的隐状态空间, 最后经过多层自注意力编码器 Enc_a 得到声学特征的隐层状态序列. 上述过程可以被表示为:

$$\tilde{\boldsymbol{s}} = EncPre(\boldsymbol{s}) \tag{3}$$

$$\boldsymbol{h} = Enc_a(\tilde{\boldsymbol{s}}) \tag{4}$$

3.2 冗余信息过滤器

本文在声学特征提取器之上引入了 CTC 损失函数, 该方法可以辅助模型预测源语言转录文本并且可以加速模型的收敛速度. 在计算 CTC 损失的过程中, Softmax 函数作用于声学特征提取器生成的隐层状态序列 $\boldsymbol{h}$, 并预测出一条输出序列 $\boldsymbol{\pi} = [\pi_1, \pi_2, \ldots, \pi_{T_s}]$, 其中 $\pi_t \in \mathbb{V} \cup$ '–' 表示输出序列中每一个时刻的标签, $\mathbb{V}$ 表示源语言词汇表, '-' 表示空标签. 假设每一个时刻的输出与其他时刻的输出是条件独立的, 那么给定隐层状态序列 $\boldsymbol{h}$, 任何一条输出序列 $\boldsymbol{\pi}$ 的概率可以表示为:

$$p(\boldsymbol{\pi}|\boldsymbol{s}) = \prod_{t=1}^{T_s} p(\pi_t|\boldsymbol{s}) \tag{5}$$

$$p(\pi_t|\boldsymbol{s}) = softmax(\boldsymbol{W}_{\mathrm{CTC}}^{\mathrm{T}}\boldsymbol{h}_t + \boldsymbol{b}_{\mathrm{CTC}}) \tag{6}$$

其中, $\boldsymbol{W}_{CTC} \in \mathbb{R}^{d\times(|\mathbb{V}|+1)}$ 是将隐层状态序列映射到源语言文本向量空间的参数矩阵.

CTC 中的输出路径集合包含多条序列 $\boldsymbol{\pi}$, 每条序列最终都可以被映射到预测序列 $\boldsymbol{x}$. 这里定义了一个多对一的函数映射 $\mathbb{B}$, 即 $\mathbb{B}(\boldsymbol{\pi}) = \boldsymbol{x}$. 该函数的作用是仅保留连续重复标签中的一个标签并且去掉空标签. 其中, 映射后的序列长度不长于映射前的序列长度. 例如, $\mathbb{B}(\mathrm{a}-\mathrm{abbb}-) = \mathrm{aab}$, 表示根据隐层状态序列预测得到的输出序列 $\boldsymbol{\pi} = \mathrm{a}-\mathrm{abbb}-$ 通过函数 $\mathbb{B}$ 被映射到预测序列 $\boldsymbol{x} = \mathrm{aab}$. 那么正确标签序列的预测概率等于所有能够被映射为正确标签序列的预测序列概率之和, 即:

$$p(\boldsymbol{x}|\boldsymbol{s}) = \sum\nolimits_{\boldsymbol{\pi}\in\mathbb{B}^{-1}(\boldsymbol{x})} p(\boldsymbol{\pi}|\boldsymbol{s}) \tag{7}$$

那么, 在整个数据集上的 CTC 损失函数为:

$$\mathcal{L}_{\mathrm{CTC}} = -\sum\nolimits_{(\boldsymbol{s},\boldsymbol{x})\in\mathcal{D}_{\mathrm{ST}}} \log p(\boldsymbol{x}|\boldsymbol{s}) \tag{8}$$

由于 CTC 中的预测序列每一个标签与隐层状态序列中的每一个时刻相对应, 因此预测序列的长度与输入的隐层状态序列长度一致, 仍然远远长于对应的文本序列, 导致隐层状态序列难以与文本特征表示进行融合. 但事实上, CTC 中的触发时刻位于一个特定单词的发声范围内, 只需要触发时刻对应的隐层状态就可以映射到对应的文本. 因此可以认为触发时刻对应的隐层状态包含了文本的先验信息, 而 CTC 预测序列中非触发时刻往往对应着空标签和重复标签, 这些时刻包含的冗余信息不利于语义的建模. 为此, 本文提出了一种冗余信息过滤器用于提取 CTC 触发时刻对应的隐层状态. 具体做法如图 2 所示, 本文设定了一个触发阈值 β, 如果 CTC 路径中非空标签对应的预测概率大于触发阈值, 则将该时刻的状态标记为 1, 否则将该时刻的状态标记为 0[25]. 最终从原隐层状态序列 $\boldsymbol{h}$ 中抽取标记为 1 的时刻对应的隐层状态组成新的状态序列 $\widetilde{\boldsymbol{h}}$. 该过程可以表示为:

$$POS(i) = \begin{cases} 1, & 1-p_b \geqslant \beta \\ 0, & 1-p_b < \beta \end{cases} \tag{9}$$

$$\widetilde{\boldsymbol{h}} = [h_i \in \boldsymbol{h} | POS(i) = 1,\ i = 1,2,\ldots,T_s] \tag{10}$$

其中, $POS(i)$ 表示隐层状态序列的第 i 个时刻, p_b=$p(\pi_t$='−'$|\boldsymbol{s})$ 对应公式 (6) CTC 路径中的空标签的预测概率, 则非空标签的预测概率可以表示为 $1-p_b$, 我们仅抽取 $POS(i)=1$ 的时刻对应的隐层状态.

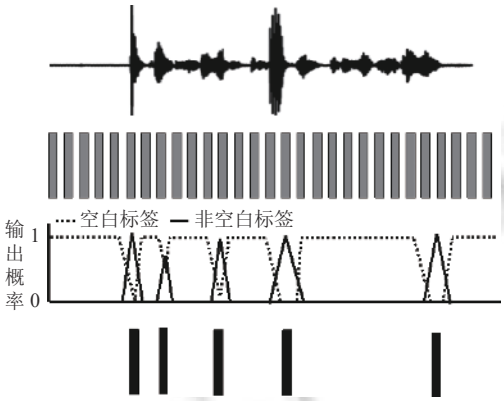

图 2 CTC 触发时刻与冗余信息过滤器

3.3 语义特征编码器与解码器

隐层状态序列经过过滤后的长度可与源语言文本匹配, 但其被映射到的隐层空间仍然缺乏翻译任务所需的语义知识. 为了使隐层状态序列获取更多的语义信息, 本文使用语义特征编码器对过滤后的隐层状态序列 $\widetilde{\boldsymbol{h}}$ 进一步编码从而将其映射到更高层次的语义表示空间. 该过程使用多层自注意力网络编码器 Enc_s 进行编码, 具体如下:

$$\boldsymbol{h}_s = Enc_s\left(\widetilde{\boldsymbol{h}}\right) \tag{11}$$

本文使用的解码器遵循 Transformer 模型框架. 首先通过查询词向量矩阵 $\boldsymbol{W}_{\boldsymbol{y}}$ 将目标语言文本映射到文本序列的向量表示 $\boldsymbol{h}_{\boldsymbol{y}}$, 继而通过自注意力网络和跨注意力网络关注到语义编码器输出的语义状态序列, 最终被前馈网络映射到目标语言文本对应空间 $\boldsymbol{h}_{\boldsymbol{d}}$. 上述过程可表示为:

$$\boldsymbol{h}_{\boldsymbol{y}} = Emb(\boldsymbol{y}) \tag{12}$$

$$\boldsymbol{h}_{\boldsymbol{d}} = Dec\left(\boldsymbol{h}_{\boldsymbol{s}}, \boldsymbol{h}_{\boldsymbol{y}}\right) \tag{13}$$

目标语言文本序列的预测概率为:

$$p(\boldsymbol{y}|\boldsymbol{s}) = \prod_{t=1}^{T_y} p(y_t|\boldsymbol{s}) \tag{14}$$

$$p(y_t|\boldsymbol{s}) = softmax(\boldsymbol{W}_{\boldsymbol{y}}^{\mathrm{T}}\boldsymbol{h}_{\boldsymbol{d}} + \boldsymbol{b}_{\boldsymbol{y}}) \tag{15}$$

最终, 在整个数据集上语音翻译的目标函数为:

$$\mathcal{L}_{\mathrm{ST}} = -\sum\nolimits_{(\boldsymbol{s},\boldsymbol{y})\in\mathcal{D}_{\mathrm{ST}}} \log p(\boldsymbol{y}|\boldsymbol{s}) \tag{16}$$

3.4 共享的文本机器翻译模型

经过冗余信息过滤器后, 声学隐层状态序列的长度被缩减到与源语言文本长度一致, 使得语音模态和文本模态的特征有了可融合的基础. 为了使语音对应的语义特征表示和文本对应的语义特征表示映射到同一个空间, 本文将文本机器翻译模型的参数与语音翻译模型中的语义特征编码器和解码器的参数进行共享, 即由语音翻译模型中的语义特征编码器和解码器组成一个完整的机器翻译模型. 在进行文本翻译任务时, 首先通过查询词向量矩阵 $\boldsymbol{W}_{\boldsymbol{x}}$ 将源语言文本映射到文本序列的向量表示, 然后基于语义编码器提取文本语义特征表示 $\boldsymbol{h}_{\boldsymbol{x}}$, 解码器的过程与上述语音翻译模型中一致. 由于语音翻译模型中语义编码器的输入来自声学特征序列 $\widetilde{\boldsymbol{h}}$, 而文本机器翻译模型中语义编码器的输入来自词向量矩阵, 如果不加约束两者可能分属不同的空间. 为了使语音模态和文本模态的特征更好地融合, 本文将文本机器翻译模型中的词向量矩阵与 CTC 模块中 Softmax 层的权重进行参数共享, 即 $\boldsymbol{W}_x = \boldsymbol{W}_{\mathrm{CTC}}$. 最终, 在整个数据集上文本翻译任务的目标函数为:

$$\mathcal{L}_{\mathrm{NMT}} = -\sum\nolimits_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathrm{ST}}} \log p(\boldsymbol{y}|\boldsymbol{x}) \tag{17}$$

3.5 不同模态的特征迁移策略

为了进一步缩小语音的语义表示和文本的语义表示之间的距离, 本文提出了两种特征迁移策略将语义知识从文本的表示迁移到语音的表示中. 本文通过最小化文本表示 $\boldsymbol{h}_x$ 和语音表示 $\boldsymbol{h}_s$ 之间的距离对两者的空间进行约束. 具体地, 本文提出了序列级别和词级别两种特征迁移策略. 目标函数可以表示为:

$$\mathcal{L}_{\mathrm{AD}} = \begin{cases} \text{序列级别}: \sum\nolimits_{(\boldsymbol{s},\boldsymbol{x})\in\mathcal{D}_{\mathrm{ST}}} \|\bar{\boldsymbol{h}}_{\boldsymbol{s}} - \bar{\boldsymbol{h}}_{\boldsymbol{x}}\|_2 \\ \text{词级别}: \sum\nolimits_{(\boldsymbol{s},\boldsymbol{x})\in\mathcal{D}_{\mathrm{ST}}} \|\boldsymbol{h}_s - \boldsymbol{h}_x\|_2 \end{cases} \tag{18}$$

其中, $\|\cdot\|_2$ 表示 L2 范数, $\bar{\boldsymbol{h}}_s$ 和 $\bar{\boldsymbol{h}}_x$ 分别对应语音和文本的序列级的语义表示, 该结果通过计算上下文表示的均值获得. 对于词级别的特征迁移策略, 如果文本序列和过滤后的声学隐层状态序列的长度不完全一致, 较短的序列将通过补零操作被补齐.

3.6 训练过程

本文需要使用 CTC 损失计算过程中产生的输出路径过滤冗余信息并抽取触发时刻对应的隐层状态序列. 同时, CTC 已经被广泛证明可以有效地加速模型训练和收敛. 因此, 本文将 CTC 的损失与语音翻译任务中的最大似然估计联合作为目标函数.

$$\mathcal{L}_1 = \lambda_1 \mathcal{L}_{\mathrm{CTC}} + (1-\lambda_1)\mathcal{L}_{\mathrm{ST}} \tag{19}$$

其中, 超参数 λ_1 用来控制 CTC 损失所占的权重.

传统的多任务学习方法在语音翻译任务中仅能共享模型的部分参数, 例如在同时训练语音翻译任务与文本翻译任务时只能共享解码器, 编码器由于源端模态对应的长度和特征空间不一致而无法共享. 而本文使用了冗余信

息过滤器与语义空间约束机制使不同模态的特征编码到同一空间, 通过将整个文本机器翻译模型嵌入到语音翻译模型中实现全部参数的共享. 本文使用多任务训练的方式同时训练语音翻译任务和文本翻译任务, 并对模型的参数空间进行联合优化. 最终的目标函数是 CTC 损失, 语音翻译任务的最大似然估计, 文本翻译任务的最大似然估计和模态迁移损失之和.

$$\mathcal{L}_2 = \lambda_1 \mathcal{L}_{\text{CTC}} + (1-\lambda_1)\mathcal{L}_{\text{ST}} + \lambda_2 \mathcal{L}_{\text{NMT}} + \lambda_3 \mathcal{L}_{\text{AD}} \tag{20}$$

由于语音翻译任务训练困难收敛较慢, 本文将训练过程分成预训练和微调两步. 在预训练阶段, 首先使用语音翻译语料库 $\mathcal{D}_{\text{ST}}$ 中的源语言语音-源语言转录文本 $(\boldsymbol{s},\boldsymbol{x})$ 对声学特征提取器进行预训练, 然后利用源语言语音-源语言转录文本-目标语言翻译文本三元组 $(\boldsymbol{s},\boldsymbol{x},\boldsymbol{y})$ 微调整个模型, 其中源语言转录文本仅在训练时使用. 本文所提模型兼具模块化和灵活性, 其中声学编码器和 CTC 模块可组成独立的语音识别模型, 而语义编码器和解码器可组成独立的文本机器翻译模型. 因此模型可以很容易地利用额外的数据进行训练, 例如利用额外的语音识别数据集 $\mathcal{D}_{\text{ASR}}$ 预训练声学特征提取器, 或利用额外的文本翻译数据集 $\mathcal{D}_{\text{MT}}$ 预训练语义编码器和解码器.

4 实验与分析

4.1 实验数据

本文在公开的语音翻译数据集上验证所提方法的有效性. 实验在英语-法语 (英-法), 英语-德语 (英-德) 和英语-汉语 (英-中) 3 个语言方向上进行. 各数据集对应的训练集, 开发集及测试集的音频时长和平行句对规模如表 1.

表 1 数据集中音频时长与句子数目统计

数据集	英-法 LibriSpeech		英-德 MuST_C		英-中 TED	
	时长 (h)	句数	时长 (h)	句数	时长 (h)	句数
训练集	100	94 542	400	229 703	500	291 526
开发集	2	1 071	2.5	1 423	2.5	1 357
测试集	4	2 048	4	2 641	2.3	1 220

表 1 中, 英-法 Augmented LibriSpeech 数据集[26]通过将法语电子书与英语语音进行对齐收集得到的, 总时长为 236 小时, 包含英语语音, 英语转录文本, 自动对齐的法语翻译文本和谷歌翻译的法语翻译文本. 与之前的工作相同[4], 我们只使用其中 100 小时的干净训练集, 并将自动对齐的法语翻译文本和谷歌翻译的法语翻译文本进行拼接, 将训练规模扩大一倍. 对于声学特征提取器的预训练, 本文使用语音识别领域常用数据集 LibriSpeech 语料库作为扩展数据[27], 该数据集包含 960 个小时的语音; 对于文本机器翻译模型的预训练, 本文从 WMT 英-法评测数据集中随机抽取了 100 万个双语平行句对作为外部的文本翻译数据, 并在 Augmented LibriSpeech 语料库中的双语句对上进行微调.

英-德 MuST_C 数据集[28]来自 TED 演讲, 包括英语语音, 英语转录文本以及不同语言的翻译文本, 我们在其中的英语-德语方向上进行实验. 由于 TED 演讲是现场录制的, 该数据集中的语音包含较多噪声.

此外, 为检验模型在语种差异较大的语言对上的性能, 本文另外选取了英-中 TED 数据集[19]进行实验, 该语料同样来自 TED 演讲.

4.2 实验设置

本文使用 80 维 log-Mel 滤波器组提取语音特征, 帧移为 10 ms, 每帧窗口大小为 25 ms, 并使用均值方差归一化操作. 本文将相邻的 3 帧语音特征拼为一帧, 并采用降采样技术每 3 帧采样一次.

对于文本数据, 我们采用了小写化, 标点规范化等处理, 英语转录文本中的标点符号被去除. 英-法和英-德翻译中英语和目标语言共享一套词表, 词表大小为 8 000. 英-中翻译中的英语和汉语词表大小分别设置为 8 000 和 14 000. 所有单词均经过字节对编码 (byte pair encoding, BPE)[29]预处理, 切分为子词.

本文方法和所有的基线模型都选用 Transformer_base 的模型参数[24], 其中声学特征提取器, 语义特征编码器

和解码器中的层数设为 6, 多头注意力机制中的头数为 8, 隐层变量维度为 512, 前馈网络的维度为 2 048, dropout 为 0.1. 在训练时, 使用 Adam 算法更新参数, 初始学习率设置为 0.1. 本文在进行多任务训练时, 将语音翻译任务和文本翻译任务的重要性同等对待, 因此将公式 (20) 中文本翻译任务的权重 λ_2 经验性地设置为 1. 模态迁移损失的权重 λ_3 同样设置为 1. 在测试时, 使用集束搜索算法, 集束的大小设置为 4. 我们每训练 1 000 步保存一次模型, 并将最后保存的 5 个模型进行参数平均作为最终的模型. 本文使用词错误率 WER 作为语音识别任务的评价指标, 使用 BLEU 作为翻译任务的评价指标, 其中英-法和英-德翻译任务中以词为单位进行计算, 英-中翻译任务以字为单位进行计算.

本文使用的模型均基于 Transformer 模型框架. 参与对比的基线系统包括:

● 文本机器翻译模型: 采用标准的 Transformer 模型, 使用 Transformer_base 的参数设置[24], 包含 6 层编码器和解码器. 文本机器翻译模型使用 $\mathcal{D}_{\mathrm{ST}}$ 中转录文本-目标语言翻译文本 $(\boldsymbol{x},\boldsymbol{y})$ 进行训练, 在测试时使用人工转录的源语言文本.

● 级联式翻译系统: 由语音识别模型和上述文本机器翻译模型串联组成, 在测试时将语音识别模型的输出作为文本机器翻译模型的输入. 其中, 语音识别模型基于 Transformer 模型进行构建[19], 模型的输入来自由梅尔滤波器组提取到的语音频域特征, 经过前端网络被映射到与模型相同维度的隐状态, 之后的流程与标准的 Transformer 模型一致. 这里使用 $\mathcal{D}_{\mathrm{ST}}$ 中的源语言语音-转录文本 $(\boldsymbol{s},\boldsymbol{x})$ 训练语音识别模型.

● 端到端的语音翻译基线模型: 该模型与上述语音识别模型的结构一致, 但是目标输出是目标语言的翻译文本. 这里使用上述语音识别模型中的编码器初始化语音翻译模型中编码器的参数. 具体地, 端到端的语音翻译模型使用 $\mathcal{D}_{\mathrm{ST}}$ 中的源语言语音-目标语言翻译文本 $(\boldsymbol{s},\boldsymbol{y})$ 进行训练.

● 传统的多任务训练: 基于 Transformer 模型进行复现, 该方法将上述文本机器翻译模型与端到端的语音翻译基线模型进行组合并联合训练, 两个模型共用解码器但使用独立的编码器.

4.3 主要实验结果

本节讨论所提方法的有效性. 表 2 列出了语音识别任务和文本翻译任务在 3 个测试集上的性能, 其中文本机器翻译模型的结果可视为语音翻译模型的上界.

表 2 语音识别和文本机器翻译在各数据集上的结果

实验内容	英-法 LibriSpeech	英-德 MuST_C	英-中 TED
语音识别 (WER↓)	10.47	18.26	13.69
机器翻译 (BLEU↑)	21.25	29.01	25.32

与英-法 LibriSpeech 数据集相比, 英-德 MuST_C 和英-中 TED 数据集中语音识别模型的性能较差, 这是因为 LibriSpeech 数据集中的语音是在录音棚环境下录制的, 而 TED 数据集中的语音来自现场演讲, 包含了较多噪音, 如笑声, 鼓掌声和欢呼声等.

表 3 列出了基线系统与本文所提方法在 3 个测试集上语音翻译的 BLEU 值. 其中, 在英-法 LibriSpeech 数据集上有 2 种设定, 分别是仅使用语音翻译数据集的设定 (基本数据) 和使用外部语音识别数据进行预训练的设定 (扩展数据).

与端到端的语音翻译基线模型相比, 本文所提方法使用了冗余信息过滤机制和语义编码器, 可以对声学信息进一步编码得到更好的语义信息, 因此翻译质量获得了显著提升. 在英-法 LibriSpeech, 英-德 MuST_C 和英-中 TED 数据集上, 本文所提方法的 BLEU 值分别提升了 1.2, 1.6, 1.1 和 1.8.

多任务训练方法通过共享模型参数, 联合训练语音翻译和文本机器翻译任务, 相比端到端的基线模型可以获得显著提升. 而在本文所提方法的基础上加入多任务训练方法, 可以获得更多的提升, BLEU 值分别提升了 0.7, 0.8, 1.3 和 1.5. 与传统的多任务训练方法不同, 本文所提方法可以将整个机器翻译模型嵌入到语音翻译模型中, 因此机器翻译模型的全部参数可以得到更充分的利用, 翻译性能提升的幅度也更加明显.

表 3 基线模型和本文所提方法在各数据集上的结果

翻译模型		英-法 LibriSpeech		英-德 MuST_C	英-中 TED
		基本数据	扩展数据		
基线系统	级联式翻译系统	17.65	19.30	22.45	22.07
	端到端基线模型	15.96	16.55	20.10	19.58
	+多任务训练	16.50	17.27	20.82	20.60
本文的方法	所提语音翻译模型	17.11	18.17	21.25	21.42
	+多任务训练	**17.81**	18.97	**22.55**	**22.93**

尽管端到端的语音翻译模型相比级联式翻译系统具有低延迟, 计算少, 存储小等优势, 但在一个模型中同时学习源端语音的声学信息以及声学信息与目标语言文本之间的对齐关系, 训练难度仍然很大, 因此端到端的语音翻译基线模型与级联式系统相比仍存在较大性能差距. 在英-法 LibriSpeech 数据集中由于语音数据中包含的噪声少, 语音识别效果较好, 级联式系统面临的错误累积问题不严重, 因此其性能明显优于端到端的语音翻译模型; 而英-德 MuST_C 和英-中 TED 数据集中语音包含较多噪声, 识别错误对级联式系统的性能有明显影响. 而本文提出的方法在加入多任务训练后可以达到与级联式系统可比甚至更好的结果.

4.4 CTC 权重设置和触发阈值设置

本节讨论 CTC 权重和触发阈值在不同的取值情况下对模型性能的影响. 本文在英-法 LibriSpeech 数据集上进行训练, 在开发集上进行调参. λ_1 和 β 的取值范围为 [0.1, 0.3, 0.5, 0.7] 和 [0.3, 0.5, 0.7, 0.9], 结果如表 4 所示.

表 4 不同 CTC 权重 λ_1 和触发阈值 β 对模型性能的影响

CTC权重 λ_1	触发阈值 β			
	0.3	0.5	0.7	0.9
0.1	18.27	18.54	**18.94**	18.17
0.3	18.14	18.13	18.06	18.41
0.5	17.64	17.42	18.56	18.49
0.7	16.81	16.52	16.36	16.06

CTC 权重和触发阈值在不同方面影响模型的性能. CTC 权值 λ_1 用于控制声学特征提取器的性能, 而触发阈值 β 用于确定触发时刻的数量, 决定了冗余信息过滤器的效果, 该值过小会导致冗余信息不能被完全过滤, 该值过大则会导致部分非冗余信息被过滤. 如表 4 所示, 在 CTC 权重 λ_1 保持不变的情况下, 触发阈值 β 取值 0.7 时翻译性能达到峰值; 而当触发阈值 β 的取值固定时, 权重 λ_1 越大翻译性能会不断下降. 当 $\lambda_1=0.1, \beta=0.7$ 时, 翻译结果的 BLEU 值最高. 因此, 在本文的实验中, 设置超参数 $\lambda_1=0.1, \beta=0.7$.

4.5 消融实验

相比基线端到端模型, 本文提出了冗余信息过滤器和语义特征编码器等模块并将所提模型与文本机器翻译任务联合训练. 本小节对本文中的模型进行了消融实验, 从而评估所提方法中不同模块对模型性能的贡献. 表 5 列出了模型在使用不同模块后在英-法 LibriSpeech 数据集上的性能.

表 5 中的结果表明, 本文所提出的不同模块和方法对模型性能均能带来正向提升, 全部使用时可以达到最优结果. 其中模态迁移策略和共享文本翻译模型的方法分别可以带来 0.5 和 0.3 个 BLEU 值的提升, 同时该方法能够加快语音翻译模型的收敛速度. 如果去除语义特征编码器模块, 直接将过滤后的声学状态输入解码器, 翻译的性能大幅下降. 这意味着声学状态缺乏语义信息, 语义特征编码器可以使其获得更好的语义表征能力, 证明了将语音翻译模型编码器解耦为声学特征提取器和语义特征编码器两部分的必要性. 本文与具有 12 层编码器的端到端强基线模型进行了比较, 该基线模型的 BLEU 值为 17.21, 与所提方法相比 BLEU 值相差 1.7, 证明了本文所提方法的性能提升并非仅因为参数量的增加而获得. 如果进一步去除冗余信息过滤器, 翻译模型的 BLEU 值将进一步下降,

证明了空白标签和重复标签等冗余信息不利于模型建模输出序列和输入序列之间对齐关系. 最后, 如果移除 CTC 损失模型性能将继续降低 0.4 个 BLEU 值.

表 5 在英-法 LibriSpeech 数据集上的消融实验

模型	BLEU
本文方法	**18.97**
–模态迁移策略	18.43
–共享文本翻译模型	18.17
–语义特征编码器	16.91
–冗余信息过滤器	16.55
–CTC损失函数	16.19

4.6 冗余信息过滤器的作用

本小节讨论冗余信息过滤器对 CTC 中输出序列长度的影响. 通过计算过滤后声学状态序列的长度与对应转录文本长度之间的差值, 分析冗余信息过滤器的过滤效果, 结果如图 3 所示.

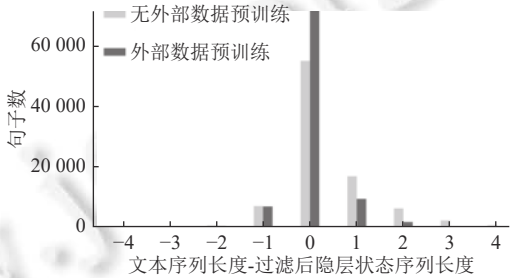

图 3 冗余信息过滤器对隐层状态序列长度的影响

图 3 中直方图显示了英-法 LibriSpeech 数据集上转录文本的长度与经过冗余信息过滤器过滤后的序列长度的差值对应的句子数目, 其中浅色柱状图表示未使用外部语音识别数据集预训练的结果, 深色柱状图表示使用了外部语音识别数据集进行预训练的结果. 差值小于零时表示过滤后的序列长度大于对应转录文本的长度, 说明部分冗余信息未被充分过滤; 差值大于零时表示部分非冗余信息被过滤. 从图 3 中可以看出, 大部分句子在经过过滤后长度与对应转录文本相等, 而基本上所有句子的预测长度与真实长度间的差值小于 2, 因此可以认为基于 CTC 中的触发时刻能够有效过滤冗余信息并准确地预测对应文本长度. 此外, 在使用外部语音识别数据预训练后, 句子的预测长度与对应文本长度更加接近, 证明使用更多语音识别数据进行预训练可以显著提升 CTC 中触发时刻的预测准确率, 进而提升冗余信息过滤器的过滤效果.

4.7 不同模态迁移策略的对比

本小节对比了两种模态迁移策略的结果, 分别对应序列级别和词级别的模态迁移策略, 结果如表 6 所示. 实验发现, 序列级别的迁移策略带来的提升效果略优于词级别的迁移策略. 因此, 本文在实验中选择使用序列级别的模态迁移策略.

表 6 在英-法 LibriSpeech 数据集上不同迁移策略的对比

模态迁移策略	开发集	测试集
序列级别迁移	**19.86**	**18.97**
词级别迁移	19.48	18.47

4.8 不同预训练数据及相关方法对比

为了验证所提方法有效性, 本小节在英-法 LibriSpeech 数据集上与其他相关方法进行了对比, 结果如表 7 所

示. 可以看出, 不管在是否使用外部数据的情况下, 所提方法的性能均超过了其他的相关方法. 具体地, 本文所提方法基于 Transformer 模型实现, 超过了所有基于 LSTM 模型的结果. 与知识蒸馏的方法相比, 本文所提方法能够在编码端融入更多的语义知识, 因此取得了更优的效果. 虽然 TCEN-LSTM 模型[20]和 LUT 模型[30]也对端到端语音翻译模型中的编码器进行了解耦, 但是前者为了解决不同模态的表示长度不一致的问题, 训练了一个额外的序列到序列的模型, 通过在文本翻译数据中插入重复或空白标签以模拟语音识别模型的输出, 影响了机器翻译模型的训练; 而后者基于外部的预训练语言模型提取文本的语义表示, 但该语义表示并不适用于翻译模型, 并且该预训练语言模型和语音翻译模型的编码器结构不同, 无法通过参数共享的方式联合优化整个模型. Wang 等人[22]提出了课程学习的方法, 使用 3 个课程分别用于学习声学表示、语义理解和双语词汇的映射, 但是该方法需要先对语音特征和源语言词语, 以及源语言词语和目标语言词语进行对齐, 操作复杂且可能引入额外的对齐错误. 与上述方法相比, 本文提出了多种策略以缓解语音和文本的表示长度和语义不一致的问题, 通过特征层面的模态迁移实现不同模型的深度结合, 能够同时优化语音识别, 文本机器翻译和语音翻译任务. 因此, 本文所提模型兼具模块化和灵活性的优点, 可以利用外部的语音识别数据对声学特征提取器进行预训练, 也可以利用外部的文本翻译数据对语义编码器和解码器进行预训练. 在使用额外的语音识别数据后, 所提方法的译文质量可以获得 1.2 个 BLEU 值的提升; 如果同时使用额外的文本翻译数据, 可以进一步获得 0.5 个 BLEU 值的提升. 在使用外部数据的情况下, 所提方法仍然优于其他融合外部数据的方法, 验证了该方法的有效性.

表 7　在英-法 LibriSpeech 数据集上相关方法的对比结果

	方法	预训练编码器	预训练解码器	BLEU
基本数据	基于LSTM的模型[4]	×	×	12.90
	+预训练+多任务训练	√	√	13.40
	ESPnet[31]	√	√	16.68
	基于Transformer的模型[19]	√	×	14.30
	+知识蒸馏	√	×	17.02
	TCEN-LSTM模型[20]	√	√	17.05
	基于Transformer的模型[22]	√	×	15.97
	+课程学习	√	×	17.66
	LUT模型[30]	×	×	17.75
	本文方法	√	×	**17.81**
扩展数据	基于LSTM的模型+频谱增强[32]	√	√	17.00
	多语言的语音翻译模型[33]	√	×	17.60
	基于Transformer的模型[22]	√	×	16.90
	+课程学习	√	×	18.01
	LUT模型[30]	√	×	18.34
	本文方法	√	×	**18.97**
		√	√	**19.43**

5 结论和未来工作

本文从源端特征编码的角度出发, 提出了多种策略以减少语音和文本模态表示的差异性, 通过不同模态特征层面的迁移, 最终实现跨模态信息融合的语音翻译方法, 显著提升了译文质量. 考虑到语音和文本模态特征表示的差异性, 本文提出将语音翻译模型中的编码器拆分为声学特征提取器, 冗余信息过滤器和语义编码器 3 部分, 通过冗余信息过滤器使语音和文本的特征长度趋于一致, 通过参数共享机制和空间约束方法将语音和文本的语义特征映射到同一空间. 实验表明, 本文所提方法可以显著提升模型的翻译性能. 同时, 本文所提模型具有较好的灵活性, 可以方便地融合外部的语音识别数据和文本翻译数据. 未来, 我们期望将这种跨模态融合的方法应用到其他自然

语言处理任务中, 并探索其他模态融合的方法.

References:

[1] Zong CQ. Statistical Natural Language Processing. 2nd ed., Beijing: Tsinghua University Press, 2013. 399–415 (in Chinese).

[2] Du JH, Zhang M, Zong CQ, Sun L. Opportunities and challenges for machine translation research in China-summary and prospects for the 8th China workshop on machine translation. Journal of Chinese Information Processing, 2013, 27(4): 1–8 (in Chinese with English abstract). [doi: 10.3969/j.issn.1003-0077.2013.04.001]

[3] Weiss RJ, Chorowski J, Jaitly N, Wu YH, Chen ZF. Sequence-to-sequence models can directly translate foreign speech. In: Proc. of the 18th Annual Conf. of the Int'l Speech Communication Association (Interspeech). Stockholm: ISCA, 2017. 2625–2629. [doi: 10.21437/Interspeech.2017-503]

[4] Bérard A, Besacier L, Kocabiyikoglu AC, Pietquin O. End-to-end automatic speech translation of audiobooks. In: Proc. of the 2018 Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP). Calgary: IEEE, 2018. 6224–6228. [doi: 10.1109/ICASSP.2018.8461690]

[5] Sperber M, Neubig G, Niehues J, Waibel A. Attention-passing models for robust and data-efficient end-to-end speech translation. Trans. of the Association for Computational Linguistics, 2019, 7: 313–325. [doi: 10.1162/tacl_a_00270]

[6] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. In: Proc. of the 3rd Int'l Conf. on Learning Representations (ICLR). San Diego, 2015.

[7] Chan W, Jaitly N, Le Q, Vinyals O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In: Proc. of the 2016 Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP). Shanghai: IEEE, 2016. 4960–4964. [doi: 10.1109/ICASSP.2016.7472621]

[8] Graves A, Fernández S, Gomez F, Schmidhuber J. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In: Proc. of the 23rd Int'l Conf. on Machine Learning (ICML). Pittsburgh: Association for Computing Machinery, 2006. 369–376. [doi: 10.1145/1143844.1143891]

[9] Schultz T, Jou SC, Vogel S, Saleem S. Using word latice information for a tighter coupling in speech translation systems. In: Proc. of the 8th Int'l Conf. on Spoken Language Processing (Interspeech). Jeju Island: ISCA, 2004. 41–44. [doi: 10.21437/Interspeech.2004-32]

[10] Sperber M, Neubig G, Niehues J, Waibel A. Neural lattice-to-sequence models for uncertain inputs. In: Proc. of the 2017 Conf. on Empirical Methods in Natural Language Processing (EMNLP). Copenhagen: Association for Computational Linguistics, 2017. 1380–1389. [doi: 10.18653/v1/D17-1145]

[11] Zhang P, Ge NY, Chen BX, Fan K. Lattice transformer for speech translation. In: Proc. of the 57th Annual Meeting of the Association for Computational Linguistics (ACL). Florence: Association for Computational Linguistics, 2019. 6475–6484. [doi: 10.18653/v1/P19-1649]

[12] Tsvetkov Y, Metze F, Dyer C. Augmenting translation models with simulated acoustic confusions for improved spoken language translation. In: Proc. of the 14th Conf. of the European Chapter of the Association for Computational Linguistics (EACL). Gothenburg: Association for Computational Linguistics, 2014. 616–625. [doi: 10.3115/v1/E14-1065]

[13] Cheng Y, Tu ZP, Meng FD, Zhai JJ, Liu Y. Towards robust neural machine translation. In: Proc. of the 56th Annual Meeting of the Association for Computational Linguistics (ACL). Melbourne: Association for Computational Linguistics, 2018. 1756–1766. [doi: 10.18653/v1/P18-1163]

[14] Kano T, Sakti S, Nakamura S. Structured-based curriculum learning for end-to-end English-Japanese speech translation. In: Proc. of the 18th Annual Conf. of the Int'l Speech Communication Association (Interspeech). Stockholm: ISCA, 2017. 2630–2634.

[15] Anastasopoulos A, Chiang D. Tied multitask learning for neural speech translation. In: Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). New Orleans: Association for Computational Linguistics, 2018. 82–91. [doi: 10.18653/v1/N18-1008]

[16] Zong CQ, Huang TY, Xu B. The technical analysis on automatic spoken language translation systems. Journal of Chinese Information Processing, 1999, 13(2): 57–66 (in Chinese with English abstract).

[17] Duong L, Anastasopoulos A, Chiang D, Bird S, Cohn T. An attentional model for speech translation without transcription. In: Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). San Diego: Association for Computational Linguistics, 2016. 949–959. [doi: 10.18653/v1/N16-1109]

[18] Bérard A, Pietquin O, Besacier L, Servan C. Listen and translate: A proof of concept for end-to-end speech-to-text translation. In: Proc. of the 2016 NIPS Workshop on End-to-end Learning for Speech and Audio Processing. Barcelona, 2016.

[19] Liu YC, Xiong H, Zhang JJ, He ZJ, Wu H, Wang HF, Zong CQ. End-to-end speech translation with knowledge distillation. In: Proc. of the 20th Annual Conf. of the Int'l Speech Communication Association (Interspeech). Grazl: ISCA, 2019. 1128–1132.

[20] Wang CY, Wu Y, Liu SJ, Yang ZL, Zhou M. Bridging the gap between pre-training and fine-tuning for end-to-end speech translation. In:

Proc. of the 34th AAAI Conf. on Artificial Intelligence. New York: AAAI, 2020. 9161–9168. [doi: 10.1609/aaai.v34i05.6452]

[21] Liu YC, Zhang JJ, Xiong H, Zhou L, He ZJ, Wu H, Wang HF, Zong CQ. Synchronous speech recognition and speech-to-text translation with interactive decoding. In: Proc. of the 34th AAAI Conf. on Artificial Intelligence. New York: AAAI, 2020. 8417–8424. [doi: 10.1609/aaai.v34i05.6360]

[22] Wang CY, Wu Y, Liu SJ, Zhou M, Yang ZL. Curriculum pre-training for end-to-end speech translation. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics (ACL). Association for Computational Linguistics, 2020. 3728–3738. [doi: 10.18653/v1/2020.acl-main.344]

[23] Indurthi S, Han H, Lakumarapu NK, Lee B, Chung I, Kim S, Kim C. End-end speech-to-text translation with modality agnostic meta-learning. In: Proc. of the 2020 Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP). Barcelona: IEEE, 2020. 7904–7908. [doi: 10.1109/ICASSP40776.2020.9054759]

[24] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Proc. of the 31st Conf. on Neural Information Processing Systems (NIPS). Long Beach: Curran Associates Inc., 2017. 6000–6010.

[25] Tian ZK, Yi JY, Tao JH, Bai Y, Zhang S, Wen ZQ. Spike-triggered non-autoregressive transformer for end-to-end speech recognition. In: Proc. of the 21st Annual Conf. of the Int'l Speech Communication Association (Interspeech). Shanghai: ISCA, 2020. 5026–5030.

[26] Kocabiyikoglu AC, Besacier L, Kraif O. Augmenting librispeech with french translations: A multimodal corpus for direct speech translation evaluation. In: Proc. of the 11th Int'l Conf. on Language Resources and Evaluation Conf. (LREC). Miyazaki, 2018.

[27] Panayotov V, Chen GG, Povey D, Khudanpur S. Librispeech: An ASR corpus based on public domain audio books. In: Proc. of the 2015 Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP). South Brisbane: IEEE, 2015. 5206–5210. [doi: 10.1109/ICASSP.2015.7178964]

[28] Di Gangi MA, Cattoni R, Bentivogli L, Negri M, Turchi M. MuST-C: A multilingual speech translation corpus. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). Minneapolis: Association for Computational Linguistics, 2019. 2012–2017. [doi: 10.18653/v1/N19-1202]

[29] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics (ACL). Berlin: Association for Computational Linguistics, 2016. 1715–1725. [doi: 10.18653/v1/P16-1162]

[30] Dong QQ, Ye R, Wang MX, Zhou H, Xu S, Xu B, Li L. Listen, understand and translate: Triple supervision decouples end-to-end speech-to-text translation. In: Proc. of the 35th AAAI Conf. on Artificial Intelligence. Palo Alto: AAAI, 2021. 12749–12759.

[31] Inaguma H, Kiyono S, Duh K, Karita S, Yalta N, Hayashi T, Watanabe S. ESPnet-ST: All-in-one speech translation toolkit. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics (ACL). Association for Computational Linguistics, 2020. 302–311. [doi: 10.18653/v1/2020.acl-demos.34]

[32] Bahar P, Zeyer A, Schlüter R, Ney H. On using specaugment for end-to-end speech translation. arXiv:1911.08876, 2019.

[33] Inaguma H, Duh K, Kawahara T, Watanabe S. Multilingual end-to-end speech translation. In: Proc. of the 2019 Automatic Speech Recognition and Understanding Workshop (ASRU). Singapore: IEEE, 2019. 570–577. [doi: 10.1109/ASRU46091.2019.9003832]

附中文参考文献:

[1] 宗成庆. 统计自然语言处理. 第2版, 北京: 清华大学出版社, 2013. 399–415.

[2] 杜金华, 张萌, 宗成庆, 孙乐. 中国机器翻译研究的机遇与挑战——第八届全国机器翻译研讨会总结与展望. 中文信息学报, 2013, 27(4): 1–8. [doi: 10.3969/j.issn.1003-0077.2013.04.001]

[16] 宗成庆, 黄泰翼, 徐波. 口语自动翻译系统技术评析. 中文信息学报, 1999, 13(2): 57–66.

刘宇宸(1995－), 男, 博士, 主要研究领域为自然语言处理, 机器翻译, 语音翻译.

宗成庆(1963－), 男, 博士, 研究员, CCF 会士, 主要研究领域为自然语言处理, 机器翻译, 文本数据挖掘, 语言认知计算.