基于语音文本跨模态表征对齐的端到端语音翻译

周国江[1,2]，董凌[1,2]，余正涛*[1,2]，高盛祥[1,2]，王文君[1,2]，马候丽[1,2]
1.昆明理工大学，信息工程与自动化学院，昆明，650500
2.昆明理工大学，云南省人工智能重点实验室，昆明，650500
1845716340@qq.com,46761956@qq.com,ztyu@hotmail.com,
gaoshengxiang.yn@foxmail.com,175360805@qq.com,1341584939@qq.com

摘要

端到端语音翻译需要解决源语言语音到目标语言文本的跨语言和跨模态映射，有限标注数据条件下，建立语音文本表征间的统一映射，缓解跨模态差异是提升语音翻译性能的关键。本文提出语音文本跨模态表征对齐方法，对语音文本表征进行多粒度对齐并进行混合作为并行输入，基于多模态表征的一致性约束进行多任务融合训练。在MuST-C数据集上的实验表明，本文所提方法优于现有端到端语音翻译跨模态表征相关方法，有效提升了语音翻译模型跨模态映射能力和翻译性能。

关键词： 端到端语音翻译；跨模态；多任务；表征对齐

End-to-end Speech Translation Based on Cross-modal Representation Alignment of Speech and Text

Guojiang Zhou[1,2],Ling Dong[1,2],Zhengtao Yu[1,2],Shengxiang Gao[1,2], Wenjun Wang[1,2],Houli Ma[1,2]
1. Faculty of Information Engineering and Automation,
Kunming University of Science and Technology, Kunming 650500, China
2. Yunnan Key Laboratory of Artificial Intelligence,
Kunming University of Science and Technology, Kunming 650500, China
1845716340@qq.com,46761956@qq.com,ztyu@hotmail.com,
gaoshengxiang.yn@foxmail.com,175360805@qq.com,1341584939@qq.com

Abstract

End-to-end speech translation aims to address cross-language and cross-modal mapping from source language speech to target language text. Under the limitation of labeled data, establishing an unified mapping between speech and text representation and alleviating cross-modal differencials become the keys to improve speech translation performance. In this paper, we propose a cross-modal representation alignment method for speech and text. The representation of speech and text are aligned at multiple granularities and mixed as parallel inputs to model, and multi-task training is performed based on the consistency constraint of multi-modal representation. Experiments on MuST-C dataset show that the proposed method outperforms existing related methods in end-to-end speech translation, and it effectively improves the cross-modal mapping capability and speech translation performance.

Keywords: end-to-end speech translation , cross-modal , multitasking , representation alignment

*余正涛（通信作者）：ztyu@hotmail.com
基金项目：国家自然科学基金（U21B2027, 61972186）；云南高新技术产业发展项目（201606）；云南省重大科技专项计划（202103AA080015）；云南省基础研究计划（202001AS070014）；云南省科技人才与平台计划（202105AC160018）

1 引言

端到端语音翻译任务将源语言语音直接翻译为目标语言文本，在多语言视频字幕、多语言会议同传等场景中具有广阔的应用前景。相较于先对源语言语音进行识别再翻译为目标语言文本的级联系统，端到端语音翻译系统具有更低的延迟和更少的参数量，避免了错误传播问题(Bérard et al., 2016)，因此备受研究者关注(Inaguma et al., 2020; Wang et al., 2020; Zhao et al., 2020)。

目前，面向端到端语音翻译任务的标注数据相对较少，有限标注数据条件下，输入语音和输出文本间的模态差异在较大程度上影响着语音翻译模型的性能(Liu et al., 2020)。这种模态差异主要表现在：语音长度远远大于其对应的文本长度，同时语音和文本的结构不同，语音是连续的时序信号，而文本是离散的符号序列，导致模型难以学习到语音和文本的对齐关系(Xu et al., 2021)。目前端到端语音翻译大多利用机器翻译、语音识别领域中较为丰富的数据通过预训练(Weiss et al., 2017; Alinejad and Sarkar, 2020; Stoian et al., 2020)，多任务训练(Tang et al., 2021; Ye et al., 2021)，知识蒸馏(Liu et al., 2019; Inaguma et al., 2021)等方式进行语音翻译辅助训练。然而机器翻译中的训练数据仅为文本模态，语音识别中的训练数据并不具备跨语言特性，故使用这类数据进行语音翻译辅助训练易导致编解码器跨模态映射能力不匹配(Cheng et al., 2022)，因此，如何有效缓解语音和文本之间的模态差异，提升语音翻译模型的跨模态映射能力是端到端语音翻译任务面临的一个重要问题。

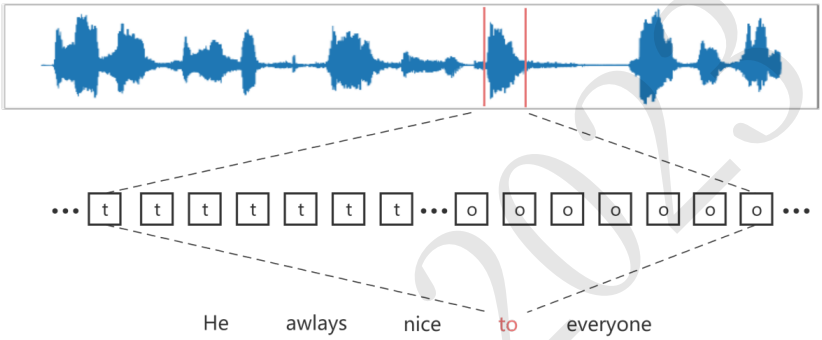


图 1. 语音与文本间的长度与结构差异

近期Liu et al. (2020)等人的工作表明，实现语音和文本的跨模态表征有助于减缓端到端语音翻译中文本与语音间的模态差异。Wei et al. (2022)和Yao et al. (2022)使用不成对的语音文本数据进行跨模态预训练以统一语音和文本的编码表示，这种多模态预训练模型相较于单模态预训练模型取得了更好的结果；Wang et al. (2022)提出了一种离散跨模态对齐方法，利用一个共享的离散词汇空间来容纳和匹配语音和文本模态，利用非平行数据对有效提升了端到端语音翻译的性能。Ye et al. (2022)提出利用对比学习对长度不一致的语音与文本表征进行约束，在此基础上Ouyang et al. (2022)通过对比学习来统一语音与文本两种模态的词级表征，证明了在不同粒度上进行跨模态一致性表征的可行性。针对语音与文本因长度造成的模态差异，Zeng et al. (2022)通过引入预测词边界任务使模型学习到语音与文本间词级的长度对齐信息，Han et al. (2021)将语音与文本表征映射为统一且固定大小的抽象表征，有效统一了语音与文本的表征空间，但固定长度的表征空间可能限制了模型的表达。Fang et al. (2022)通过混合语音与文本的浅层表征作为并行输入，增强数据的同时缓和了模态差异，但其混合表征与语音表征之间仍存在较大的长度差异。

本文提出词级和句子级的语音文本表征对齐方法，在对齐基础上对语音和文本表征进行交叉混合得到混合表征作为模型输入，将语音与文本映射到同一表征空间。针对语音和文本表征因长度造成的模态差异，使用长度归一化融合模块对混合表征与语音表征进行长度统一。针对不同模态的表征内容差异问题，在多任务联合训练框架对语音文本模态表征进行一致性约束。从而实现语音表征和对应文本表征的跨模态对齐，从而提升模型跨模态映射能力。

本文主要贡献如下：（1）本文提出的语音文本表征对齐方法和多粒度表征混合方法，验证了跨模态一致性表征对语音翻译的积极作用。（2）本文提出长度归一化融合方法，验证了其对缓解语音文本模态间长度差异的有效性。（3）在MuST-C数据集上的实验表明，在相同数据条

件下，本文所提方法优于现有端到端语音翻译跨模态表征相关方法。

2 方法

模型由声学编码器、语音文本多模态表征对齐模块、翻译解码器构成，训练阶段语音通过声学编码器得到语音表征，文本通过嵌入得到文本表征，语音表征与文本表征通过多粒度混合得到句级和词级的混合表征。语音文本多模态表征对齐模块由长度归一融合模块、共享语义编码层、门控融合模块组成，长度归一化模块用于统一各表征的长度以减轻混合表征与语音表征之间由长度导致的模态差异，共享语义编码层用于提取抽象语义表征，门控融合模块则将经过编码模块的两个混合表征再次进行融合以降低解码难度，经对齐的语音表征与混合表征并行输入翻译解码器进行解码，输出目标语言文本词序列，模型架构如图 2所示。

训练使用的数据集包含语音、转录文本以及翻译文本三元组数据，记为D(s，x，y)。模型训练分为预训练与多任务训练两阶段，在预训练阶段，使用源语言语音s和目标语言文本y对共享语义编码层和翻译解码器进行文本翻译预训练，在多任务训练阶段使用转录文本x作为辅助输入，在推理阶段以源语言语音s作为输入，不使用转录文本x。

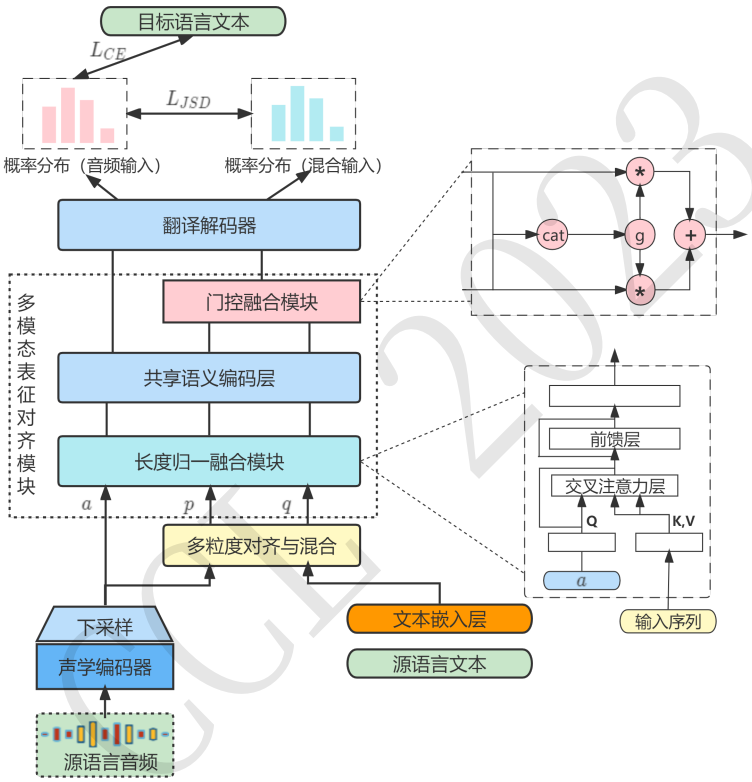


图 2. 基于语音文本跨模态表征对齐的端到端语音翻译

2.1 文本嵌入与声学编码

源语言语音s经声学编码器编码得到语音表征作为长度归一融合模块的输入。转录文本x经文本嵌入得到文本表征，与语音表征进行对齐并混合后作为模型并行输入。

声学编码器:主要用于提取语音信号的浅层表征。近期的一些研究(Gállego et al., 2021)表明使用经预训练的Hubert(Hsu et al., 2021)作为声学编码器可以提高语音翻译的性能，因此本文采用与其相同的声学编码方式$Hubert(\cdot)$。基于Ye et al. (2021)的研究，本文在Hubert的基础上加入两个卷积层对输出的语音表征进行下采样$DS(\cdot)$。编码过程如式 (1)所示，下采样倍数为4，编码后得到语音表征序列$a = [a_1, a_2, ..., a_{l_a}]$，其中$a \in \mathbb{R}^{d_a}$,$d_a$为语音表征维度，$l_a$为序列长度。

$$a = Ds(Hubert(\mathrm{s})) \tag{1}$$

文本嵌入:如式 (2)所示，对于训练时的文本输入，使用Unigram SentencePiece[1] 学习双语词表，并对文本输入x进行编码$Unigram(\cdot)$，编码后经文本嵌入$Emb(\cdot)$得到文本表征$e = [e_1, e_2, ..., e_{l_e}]$，其中$e \in \mathbb{R}^{d_e}$,$d_e$为文本嵌入表征维度，$l_e$为序列长度。

$$e = Emb(Unigram(\mathrm{x})) \tag{2}$$

2.2 语音文本表征的对齐与混合

针对语音长度远大于其对应的文本长度，导致对齐关系难以学习的问题，本节将语音与文本表征在句子级和词级上进行混合作为多模态表征对齐模块的并行输入，让模型学习到跨模态的词级和句子级对齐信息，混合过程如图 3所示。

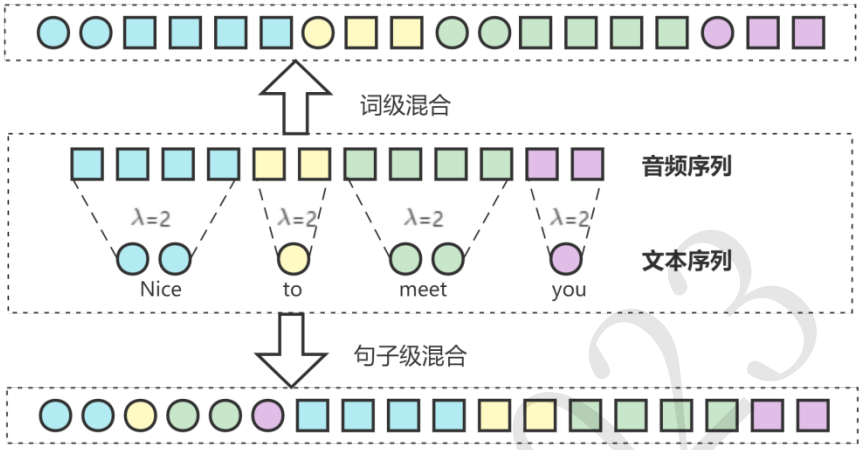


图 3. 语音文本表征的对齐与混合

词级语音文本表征混合：将编码后的语音表征序列a与文本表征序列e按一定比例切割对齐，如式 (3)，先计算出训练集中所有语音编码表征序列长度l_a与文本嵌入序列长度l_e的数学期望之比λ ，$E(\cdot)$用于计算数学期望，$Int(\cdot)$用于向下取整。

$$\lambda = Int(E(l_a)/E(l_e)) \tag{3}$$

对于文本表征序列e中的任意单词序列e_j，如式 (4)，使用λ进行对齐得到其对应语音表征序列的起止位置u_j和v_j，$|\cdot|$用于计算单词序列长度。

$$u_j = \sum_{\mathrm{i}=1}^{\mathrm{j}-1} \lambda \left|\mathrm{e_i}\right|, \quad v_j = u_j + \lambda \left|\mathrm{e_j}\right| = \sum_{\mathrm{i}=1}^{\mathrm{j}} \lambda \left|\mathrm{e_i}\right| \tag{4}$$

根据位置信息u_j和v_j对语音进行对齐得到m_j。对整个语音表征序列进行对齐后可表示为$m = [m_1, m_2, ..., m_{l_m}]$，其中$m \in \mathbb{R}^{d_a}$,$l_m$为序列长度。

$$m_j = \begin{cases} \left[a_{u_j} : a_{v_j}\right] & v_j \le l_a \text{ and } j \le l_e \\ \left[a_{v_{j-1}} :\right] & j > l_e \\ \left[a_{u_j} :\right] & v_j > l_a \end{cases} \tag{5}$$

如式 (5)所示，对齐后的语音表征序列m_j为原音频表征序列a中对应起始位置u_j与终止位置v_j之间的序列。当最后一个单词序列对应的终止位置小于l_a时，对于剩余还未对齐的音频序列，有$j > l_e$，取剩余音频序列作为m_j。当终止位置v_j大于语音表征序列长度l_a时，取起始位置u_j之后的所有序列作为m_j，若$u_j > l_a$，则a_{u_j}与m_j均为空序列。

将对齐后的序列混合后进行拼接$Concat(\cdot)$，得到词级混合表征p，过程如式 (6)所示。

[1]https://github.com/google/sentencepiece

$$p = Concat(\mathrm{m}_1, \mathrm{e}_1,\ \mathrm{m}_2, \mathrm{e}_2 \ldots, \mathrm{m}_{l_m}, \mathrm{e}_{l_e}) \tag{6}$$

句子级语音文本表征混合：句子级混合不需要对齐,混合过程如式 (7)所示，q表示句子级混合表征，词级与句子级混合表征序列长度相同。

$$q = Concat(\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_{l_a}, \mathrm{e}_1, \mathrm{e}_2 \ldots, \mathrm{e}_{l_e}) \tag{7}$$

2.3 多模态表征对齐模块

语音文本多模态表征对齐模块由长度归一融合模块、共享语义编码层、门控融合模块组成。训练阶段以语音表征a和两种混合表征p，q作为输入，通过长度归一化模块映射为具有相同维度的表征，经共享语义编码层得到抽象语义表征，词级与句级的混合语义表征通过门控融合模块融合为多粒度语义表征。

长度归一融合模块：本文使用长度归一的融合模块再次对语音表征与混合表征进行融合,融合编码方法为多头交叉注意力$CMHA(\cdot)$ 。如式 (8)所示，语音表征a、词级混合表征p、句子级混合表征q经长度归一融合后分别得到融合表征h_a，h_p，h_q，其中W_q，W_k，W_v均为随机初始化的参数矩阵,输入Q始终为语音表征a，K、V则为对应的输入表征。

$$h_{out} = CMHA(QW_q, KW_k, VW_v) \tag{8}$$

共享语义编码层：本文使用的编码层遵循Transformer编码层的结构，层数为6，每一层都包含一个自注意、残差、前馈和归一化模块。如式 (9)所示，共享语义编码层$encoder(\cdot)$的输入h_{input}分别为长度归一融合模块的输出h_a、h_p、h_q，经语义编码后得到对应的抽象语义表征h_a^A、h_p^A、h_q^A。

$$h_{out}^A = encoder(h_{input}) \tag{9}$$

门控融合模块：该模块将词级与句级的混合语义表征h_p^A、h_q^A进一步融合，综合两种混合表征的特点，同时降低翻译解码器解码压力。如式 (10)所示，$*$表示计算矩阵乘法，先将h_p^A、h_q^A在隐层维度拼接，使用可学习的W_g进行线性映射得到门控单元系数γ，γ的隐层维度为1，激活函数$\sigma(\cdot)$为sigmoid，最后使用γ对h_p^A、h_q^A进行融合得到多粒度融合表征h_g。

$$\gamma = \sigma(Concat(h_p^A, h_q^A) * W_g), h_g = \gamma * h_p^A + (1-\gamma) * h_q^A \tag{10}$$

2.4 翻译解码器

解码器由6层transfomer解码层构成。语音输入s经声学编码器得到语音表征，通过长度归一融合模块和共享语义编码层得到抽象语义表征，再通过翻译解码器生成目标语言的单词序列，简化表示为h(s)。同理，语音输入s与其转录文本输入y计算得到多粒度融合表征，经翻译解码器生成目标语言的单词序列，简化表示为h(s, x)。对于输入序列s和期望输出y，我们定义损失函数如式 (11)。

$$\mathrm{L_{CE}}(\mathrm{h(s)}, \mathrm{y}) = \sum_{\mathrm{i=1}}^{\mathrm{|y|}} \log\left(\mathrm{P}_\theta\left(\mathrm{y_i} \mid \mathrm{y_{<i}}, \mathrm{h(s)}\right)\right) \tag{11}$$

使用交叉熵损失作为语音输入得到结果与目标语言文本的损失，语音表征和融合表征之间使用Jensen-Shannon散度$JSD(\cdot)$计算得到一致性约束损失，下文简称为JSD损失，计算过程如式 (12)所示。本文没有引入混合表征作为输入时其结果与目标语言文本的约束损失，使模型更加关注以语音作为输入时的结果，同时降低模型拟合的难度。

$$\mathrm{L_{JSD}}(\mathrm{h(s)}, \mathrm{h(s,x)}, \mathrm{y}) = \sum_{i}^{\mathrm{|y|}} JSD\left(\mathrm{P}_\theta\left(\mathrm{y_i} \mid \mathrm{y_{<i}}, \mathrm{h(s)}\right), \mathrm{P}_\theta\left(\mathrm{y_i} \mid \mathrm{y_{<i}}, \mathrm{h(s,x)}\right)\right) \tag{12}$$

综上，微调阶段总损失L(s, x, y)如式 (13)所示，β为JSD损失权重系数。

$$L(s,x,y) = L_{CE}(h(s),y) + \beta * L_{JSD}(h(s),h(s,x),y) \quad (13)$$

3 实验设置与结果分析

3.1 实验设置

3.1.1 数据集

实验使用了MuST-C(Di Gangi et al., 2019)数据集，该数据集包含英语到多个语言对数据，本文使用了英语(En)到越南语(Vi)、德语(De)、意大利语(It)、俄罗斯语(Ru)、西班牙语(Es)、法语(Fr)、罗马尼亚语(Ro)、荷兰语(Nl)和葡萄牙语(Pt)9个语言对，具体参数如表 1所示。实验中使用MuST-C中的dev集作为验证集，tst-COMMON作为测试集。

语言	En-De	En-Es	En-Fr	En-Nl	En-It	En-Pt	En-Ro	En-Ru	En-Vi
时长（h）	408	504	492	442	465	385	432	489	441
句数（k）	234	270	280	253	258	211	241	270	230

表 1. MuST-C数据集

3.1.2 数据预处理

为保证实验公平性，文本采用Fairseq(Ott et al., 2019)中对MuST-C数据集的预处理方式，并使用Unigram SentencesPiece学习得到每个语言对的源、目标语言双语共享词表，词表大小为10000。语音输入为16bit，16kHz的单通道原始语音，使用开源Hubert[2]作为声学编码器，提取语音的768维语音表征，并用两层卷积神经网络对其进行下采样，卷积核大小为5，步长为2，隐藏层维度为1024。根据语音长度与声学编码器中的维度变化计算语音经表征序列长度，与文本嵌入表征长度信息结合，计算得到λ，λ的值为3，根据λ进行对齐得到位置信息u，v。

3.1.3 模型配置与评价指标

为保证实验结果可比性，本文所做实验均基于Fairseq框架。共享语义编码层与解码层的层数均为6，多头注意力头数为8，隐层变量维度为512，前馈层网络维度为2048，dropout为0.1。

在文本翻译预训练阶段，使用源语言-目标语言对来对翻译编解码器进行训练，设置学习率为7e-4，每批可使用的序列长度最多为4k。

在多任务训练阶段，每批使用最多2M的源语音帧，学习率为1e-4，JSD权重系数β为4，设置每8个批次进行一次梯度更新以模拟使用8张显卡进行计算。为避免过拟合，设置最大训练周期为30，若验证集上的损失在十个周期内没有减少，提前停止训练。上述两个训练阶段所使用优化器均为Adam(Kingma and Ba, 2014)，设置β_1 为0.9，β_2 为0.98，交叉熵损失标签平滑率为0.1。学习率预热步长为4000，4000步后学习率将与步数的平方成反比下降。

在推理阶段，对最后十个周期得到的模型参数进行平均以用于评估。解码使用大小为5的束搜索算法，使用区分大小写的SacreBLEU[3] (Post, 2018) 作为模型性能的评价指标，所有训练过程均在1张Tesla V100 GPU上进行。

3.2 实验结果

3.2.1 与其它方法的对比实验

为验证所提方法的有效性，在MuST-C数据集上进行对比实验，实验参数与所提方法一致。选择了以下几个端到端语音翻译方法：Fairseq-ST (Wang et al., 2020) 、Chimera(Han et al., 2021)、STEMM(Fang et al., 2022)、ConST(Ye et al., 2022)、W2V2-ST，具体基线模型介绍如下：

（1）Fairseq-ST：使用语音识别任务进行预训练，在端到端语音翻译任务上微调。

（2）Chimera：引入了一个共享的语义空间映射层，将语音和文本映射成固定维度的语义表示并用于翻译。

[2]https://github.com/facebookresearch/Fairseq/tree/main/examples/hubert
[3]https://github.com/mjpost/sacrebleu

（3）STEMM：将语音和文本的浅层表征进行混合作为翻译任务的并行输入，使用自学习训练框架约束训练结果。

（4）ConST：在多任务交替训练中引入对比损失约束语音和文本表征的一致性，让语义相似的语音和文本具有相似的表示。

（5）W2V2-ST：使用Wav2vec2.0(Baevski et al., 2020)模型提取语音表征，使用transformer编解码器进行训练，结果取自(Cheng et al., 2022)。

模型	额外数据			BLEU								
	Speech	MT	MFA	Vi	De	Es	Fr	It	Nl	Pt	Ro	Ru	Avg.
Fairseq-ST	×	×	×	—	22.7	27.2	32.9	22.7	27.3	28.1	21.9	15.3	24.8
Chimera	√	√	×	—	**27.1**	30.6	35.6	25.0	29.2	30.2	24.0	17.4	27.4
STEMM	√	×	√	—	25.6	30.3	36.1	25.6	30.1	31.0	24.3	17.1	27.5
ConST	√	×	×	—	25.7	30.4	36.8	26.3	30.6	32.0	24.8	17.3	28.0
W2V2-ST	√	×	×	23.2	24.3	29.6	35.2	25.1	29.1	30.3	23.4	16.5	26.7
本文所提方法	√	×	×	**24.7***	26.6*	**31.0***	**37.2***	**26.4***	**30.8***	**32.4***	**25.1***	**17.8***	**28.4***

表 2. 在MuST-C数据集多语言对上与其它方法的比较

如表 2，*表示本文所提方法结果强于W2V2-ST基线模型结果，加粗部分表示达到了最佳翻译效果，speech表示使用外部语音数据，MT(Machine Translation)表示使用外部翻译数据，MFA[4](McAuliffe et al., 2017)表示使用外部强制对齐模型。因其它方法在英语-越南语语言对上实验数据的缺乏，本文对除越南语外的8个语言对的BLEU求均值得到Avg.结果。

本文方法与W2V-ST基线模型相比，BLEU值平均提高了1.8，验证了本文所提语音文本跨态表征对齐方法的有效性。与使用了额外文本翻译数据的Chimera相比，在可比较语言对上BLEU值平均提升了1.0，本文方法同样将语音和文本映射到了同一长度但没有将输出限制到固定大小，验证了基于语音长度归一化融合的有效性。

STEMM方法在进行混合时使用了外部强制对齐模型，本文则根据数据集本身语音与文本表征长度进行混合。相较于STEMM，本文方法进行了多粒度混合且得到的混合序列更长，但经长度一致性融合与门控融合后仅增加了少量训练时间，并在可比较语言对上取得了平均0.9 BLEU值的提升，证明了本文所提多粒度对齐与长度归一融合方法的有效性。

在相同数据条件下，本文方法较ConST方法在可比较语言对上平均BLEU值提升了0.4，ConST为使用对比损失将语音和文本进行了平均池化，一定程度上忽略了语音和文本表征间的局部差异，本文方法在使用JSD损失进行约束时并未造成局部信息损失，表明了在JSD损失约束下的多任务训练框架有效性。

3.2.2 预训练对语音翻译结果的影响

遵循Fairseq中基于端到端语音到文本的模型设置，分别基于Fbank特征和Hubert特征进行英语到越南语翻译实验作为对比，基于Fbank特征的Fairseq-ST实验参数与Wang et al. (2020)等人一致，基于Hubert特征的实验参数与上节中提到的一致。为进一步验证文本翻译预训练对结果的影响，使用CCMatrix(Schwenk et al., 2019)中英语-越南语平行语料作为额外数据进行文本翻译预训练。

模型	预训练	训练时间	推理时间	BLEU
Fairseq-ST	语音识别	1.00x	1.00x	20.8
Hubert-Transformer	Hubert	4.23x	1.83x	22.6
Hubert-Transformer	Hubert+文本翻译	4.23x	1.83x	23.4
本文方法	Hubert+文本翻译	5.52x	1.83x	24.7
本文方法+额外翻译数据	Hubert+文本翻译	5.52x	1.83x	25.4

表 3. MuST-C英语-越南语不同预训练方法BLEU值对比

[4]https://mfa-models.readthedocs.io/en/latest/index.html

在tst-COMMON测试集上的结果如表 3所示，训练时间与推理时间分别为训练与推理阶段花费的时长。在没有经过文本预训练的情况下，使用经预训练的Hubert作为声学编码器进行训练BLEU值达到了22.6，比在语音识别预训练下以Fbank作为特征输入进行训练得到的BLEU结果高出1.8，验证了Hubert作为声学编码器的有效性。在使用文本翻译进行预训练后，结果再次提高了0.8 BLEU，表明进行跨模态的预训练对端到端语音翻译是有效的。在此基础上，本文所提跨模态表征对齐方法训练得到的BLEU再次提升了1.3，验证了本文所提方法的有效性。引入额外翻译数据进行文本翻译预训练后BLEU结果达到了25.4，进一步验证了使用文本翻译进行跨模态预训练对语音翻译的积极作用。以Fairseq-ST结果1.00x为基准，使用Hubert作为声学编码器使训练时间增长为4.23x，推理时间增长为1.83x。本文方法引入了词级与句子级混合表征进行多任务训练，训练时间达到了5.52x,推理阶段只有音频作为输入，故推理时间与Hubert-Transformer方法相同。

3.2.3 不同并行输入特征的对比实验

为探究混合表征对语音翻译结果的影响，分别设置词级、句级混合表征以不同组合方式作为训练时的并行输入进行实验。

为了验证模型缓和跨模态差异的能力，使用tst-COMMON测试集中的英语语音-文本对作为输入，分别得到其经过共享语义编码层后的文本语义表征和语音语义表征，计算语音-文本语义表征对的平均余弦相似度作为特征相似度指标，它反应了模型在语义层面将不同模态表征进行共同映射的能力。

有无混合	并行输入	BLEU	特征相似度（%）
无混合	无	23.4	97.88
	文本表征	24.2	97.26
有混合	词级混合表征	24.5	98.14
	句子级混合表征	24.4	98.00
	句子+词级混合表征	24.7	98.28

表 4. 混合表征对英语-越南语实验结果影响

如表 4所示，在没有进行混合的情况下，使用文本表征作为并行任务的输入BLEU提高了0.8，表明在多任务训练框架下使用额外的翻译数据是有益的。在多任务训练框架下，特征相似度与BLEU结果成正相关，表明缓和模态差异对端到端多任务语音翻译的积极影响。利用词级或句级混合表征都能对语音翻译产生正向效果，因为混合表征与文本表征相比具有额外的语音信息和不同模态信息间的相对位置信息。相较于句子级的混合表征，使用词级的混合表征进行训练特征相似度提高了0.14%，BLEU提高了0.1，表明用词级的混合表征作为并行输入使模型能够捕捉到不同模态表征长度期望之间的联系，从而更好地缓解语音和文本在长度上造成的模态差异，提升翻译性能。相较于仅利用单一混合表征，使用多粒度的混合表征能够进一步提升语音翻译性能和特征相似度，表明同时使用词级混合表征与句子级混合表征使模型学习到更多不同模态间的位置信息，二者具有互补性。

3.2.4 长度归一融合模块不同输入下的对比实验

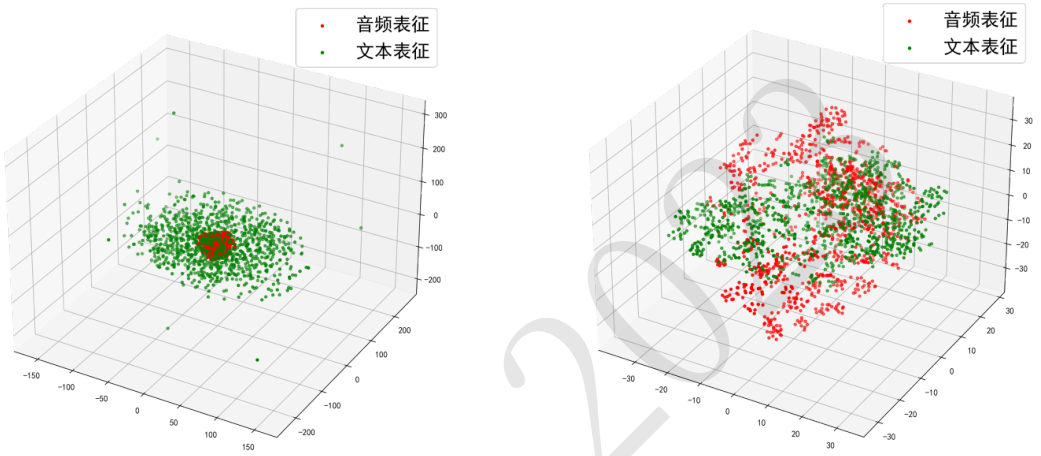
为探究融合特征与语音表征长度之间的关系对语音翻译性能的影响，在不改变主任务语音表征输入的情况下，将并行任务中长度归一融合模块的语音输入改为文本表征与语音表征分别进行了实验。为避免词级与句级混合表征共同作为输入时，两种混合表征间可能造成的影响，实验以表 4中仅使用词级混合表征为基础进行。

相较于语音表征，融合表征中还包含完整文本表征，以所含信息量评估，使用其作为输入BLEU值应更高。但如表 5所示，将语音输入更改为融合表征输入后BLEU值下降了0.1，特征相似度下降了0.73%，表明在融合表征作为辅助任务输入时，将其映射为较长的表征向量导致其与语音翻译主任务中语音表征产生较大长度差异，增加了模型跨模态映射难度和训练复杂度。将语音输入替换为文本表征之后BLEU与特征相似度下降情况更加明显，这表明将融合表征映射为与语音统一长度对缓解因长度造成的模态差异问题是有效的。

Q	K，V	BLEU	特征相似度（%）
词级混合表征	词级混合表征	24.4	97.41
文本表征	词级混合表征	23.4	80.03
语音表征	词级混合表征	24.5	98.14

表 5. 长度归一融合模块输入特征类型对英语-越南语翻译效果影响

使用T-SNE(Van der Maaten and Hinton, 2008)将各表征的隐层维度由512维简化为3维，对经过长度归一融合模块进行融合前后的文本和语音表征进行三元核密度估计可视化。如图 4所示，在进行长度归一化融合之前语音表征分布更为集中，导致模型难以学习到跨模态的对齐关系，表现了语音与文本不同模态表征之间存在的分布差异，而融合后两种表征的分布更为均匀，表明长度归一化融合模块可以有效将两种模态的表征映射到同一表征空间，缓解了不同模态间的分布差异，提升了翻译性能。



(a) 融合前表征分布　　(b) 融合后表征分布

图 4. 长度归一融合模块对表征分布的影响

3.2.5 门控融合模块对翻译性能影响

为验证门控融合模块在多任务训练框架下的作用，使用双JSD损失约束代替门控融合模块进行实验，经共享语义编码层的词级和句级融合表征都作为翻译解码器的并行输入，分别得到词级混合表征和句子级混合表征对应的目标文本词序列，过程分别简化表示为$h_1(x,y)$，$h_2(x,y)$，使用两个JSD损失对其进行约束，训练损失$L_1(s,x,y)$如式 (14)所示。

$$L_1(s,x,y) = L_{CE}(h(s),y) + \beta * L_{JSD}(h(s),h_1(s,x),y) + \beta * L_{JSD}(h(s),h_2(s,x),y) \quad (14)$$

为验证门控融合模块的有效性，对两个经共享语义编码层的混合表征求均值作为解码层输入，进行对比实验。

融合	JSD约束对象	BLEU	特征相似度（%）
无	句子与词级表征	24.1	97.09
均值融合	多粒度融合表征	24.5	98.11
门控融合	多粒度融合表征	24.7	98.28

表 6. 不同融合方式对英语-越南语翻译结果的影响

结果如表 6所示，在不对词级与句子级混合表征进行融合的情况下BLEU值为24.1，而使用取均值进行融合使BLEU提高了0.4，特征相似度提高了1.02%，表明过多的并行输入和损失约束增加了解码器的解码压力，导致翻译性能的下降。与均值融合相比，使用门控融合方法提高了0.2BLEU值，表明门控融合模块能够关注到词级与句子级混合表征间不同的模态与位置差异，实现对两种混合表征的有效整合。

3.2.6 多任务训练框架下不同损失约束的对比实验

为验证训练过程中损失函数的作用，如式 (15)所示，新增多粒度融合表征的交叉熵损失，用以评估模型在不同损失下的翻译效果。

$$\mathrm{L}_{\mathrm{CE}_m}(\mathrm{h}(\mathrm{s},\mathrm{x}),\mathrm{y}) = \sum_{\mathrm{i}=1}^{|\mathrm{y}|} \log\left(\mathrm{P}_\theta\left(\mathrm{y_i} \mid \mathrm{y_{<i}}, \mathrm{h}(\mathrm{s},\mathrm{x})\right)\right) \tag{15}$$

L_{CE_m}	L_{JSD}	BLEU	特征相似度（%）
×	×	23.4	97.88
√	×	23.6	93.00
√	√	24.2	97.81
×	√	24.7	98.28

表 7. 多任务训练中损失函数对英语-越南语翻译结果影响

如表 7所示，当使用交叉熵对融合表征作为输入的辅助任务进行约束时其BLEU值提升了0.2，但使模型更加关注于翻译效果更好的文本输入，导致特征相似度的下降。在此基础上加入JSD损失规范两个输出的预测，BLEU值进一步提升了0.6，特征相似度提升了4.81%，表明使用JSD损失进行一致性约束在多任务框架中对缓和模态差异的积极作用。当仅使用JSD损失进行一致性约束时，特征相似度与BLEU值得到了进一步提高，表明仅使用JSD损失与针对语音的交叉熵损失使模型更加关注语音翻译任务，同时降低了模型拟合的难度。

JSD损失是约束输出一致性的重要因素，为选定最优JSD损失比例，我们将JSD权重系数β分别置为0、1、2、3、4、5进行实验。由图 5可知，在β为4时得到最佳翻译效果。

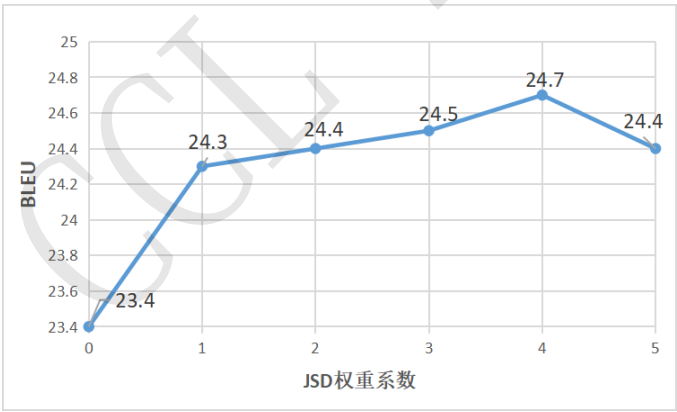


图 5. JSD权重系数对英语-越南语翻译结果影响

4 结论

针对端到端语音翻译过程中存在的跨模态问题，本文根据音频与文本之间长度的关系，提出语音文本跨模态表征对齐方法，使用多粒度混合特征作为模型并行输入，对不同模态的表征进行归一化融合与对齐，使用改进了损失约束的多任务训练框架约束混合表征与音频表征的一致性，使模型将语音与文本输入映射到同一表征空间。实验和分析表明了本文提出方法在不同层面对缓解跨模态表征差异的有效性，提高了端到端语音翻译的性能。未来的工作将在现有跨模态一致性表征工作的基础上对跨模态数据的利用进行研究，探索在使用更多外部数据的情况下，对端到端语音翻译进行增强的同时保持其跨模态表征的一致性。

参考文献

Ashkan Alinejad and Anoop Sarkar. 2020. Effectively pretraining a speech translation decoder with machine translation data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8014–8020.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. 2016. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744*.

Xuxin Cheng, Qianqian Dong, Fengpeng Yue, Tom Ko, Mingxuan Wang, and Yuexian Zou. 2022. M3st: Mix at three levels for speech translation. *arXiv preprint arXiv:2212.03657*.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. MuST-C: a Multilingual Speech Translation Corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2012–2017, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Qingkai Fang, Rong Ye, Lei Li, Yang Feng, and Mingxuan Wang. 2022. Stemm: Self-learning with speech-text manifold mixup for speech translation. *arXiv preprint arXiv:2203.10426*.

Gerard I Gállego, Ioannis Tsiamas, Carlos Escolano, José AR Fonollosa, and Marta R Costa-jussà. 2021. End-to-end speech translation with pre-trained models and adapters: Upc at iwslt 2021. *arXiv preprint arXiv:2105.04512*.

Chi Han, Mingxuan Wang, Heng Ji, and Lei Li. 2021. Learning shared semantic space for speech-to-text translation. *arXiv preprint arXiv:2105.03095*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Hirofumi Inaguma, Shun Kiyono, Kevin Duh, Shigeki Karita, Nelson Enrique Yalta Soplin, Tomoki Hayashi, and Shinji Watanabe. 2020. Espnet-st: All-in-one speech translation toolkit. *arXiv preprint arXiv:2004.10234*.

Hirofumi Inaguma, Tatsuya Kawahara, and Shinji Watanabe. 2021. Source and target bidirectional knowledge distillation for end-to-end speech translation. *arXiv preprint arXiv:2104.06457*.

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

Yuchen Liu, Hao Xiong, Zhongjun He, Jiajun Zhang, Hua Wu, Haifeng Wang, and Chengqing Zong. 2019. End-to-end speech translation with knowledge distillation. *arXiv preprint arXiv:1904.08075*.

Yuchen Liu, Junnan Zhu, Jiajun Zhang, and Chengqing Zong. 2020. Bridging the modality gap for speech-to-text translation. *arXiv preprint arXiv:2010.14920*.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. 2017. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech*, volume 2017, pages 498–502.

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. *arXiv preprint arXiv:1904.01038*.

Siqi Ouyang, Rong Ye, and Lei Li. 2022. Waco: Word-aligned contrastive learning for speech translation. *arXiv preprint arXiv:2212.09359*.

Matt Post. 2018. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2019. Ccmatrix: Mining billions of high-quality parallel sentences on the web. *arXiv preprint arXiv:1911.04944*.

Mihaela C Stoian, Sameer Bansal, and Sharon Goldwater. 2020. Analyzing asr pretraining for low-resource speech-to-text translation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7909–7913. IEEE.

Yun Tang, Juan Pino, Changhan Wang, Xutai Ma, and Dmitriy Genzel. 2021. A general multi-task learning framework to leverage text data for speech to text tasks. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6209–6213. IEEE.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Changhan Wang, Yun Tang, Xutai Ma, Anne Wu, Sravya Popuri, Dmytro Okhonko, and Juan Pino. 2020. fairseq s2t: Fast speech-to-text modeling with fairseq. *arXiv preprint arXiv:2010.05171*.

Chen Wang, Yuchen Liu, Boxing Chen, Jiajun Zhang, Wei Luo, Zhongqiang Huang, and Chengqing Zong. 2022. Discrete cross-modal alignment enables zero-shot speech translation. *arXiv preprint arXiv:2210.09556*.

Kun Wei, Long Zhou, Ziqiang Zhang, Liping Chen, Shujie Liu, Lei He, Jinyu Li, and Furu Wei. 2022. Joint pre-training with speech and bilingual text for direct speech to speech translation. *arXiv preprint arXiv:2210.17027*.

Ron J Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-sequence models can directly translate foreign speech. *arXiv preprint arXiv:1703.08581*.

Chen Xu, Bojie Hu, Yanyang Li, Yuhao Zhang, Qi Ju, Tong Xiao, Jingbo Zhu, et al. 2021. Stacked acoustic-and-textual encoding: Integrating the pre-trained models into speech translation encoders. *arXiv preprint arXiv:2105.05752*.

Zhuoyuan Yao, Shuo Ren, Sanyuan Chen, Ziyang Ma, Pengcheng Guo, and Lei Xie. 2022. Tessp: Text-enhanced self-supervised speech pre-training. *arXiv preprint arXiv:2211.13443*.

Rong Ye, Mingxuan Wang, and Lei Li. 2021. End-to-end speech translation via cross-modal progressive training. *arXiv preprint arXiv:2104.10380*.

Rong Ye, Mingxuan Wang, and Lei Li. 2022. Cross-modal contrastive learning for speech translation. *arXiv preprint arXiv:2205.02444*.

Xingshan Zeng, Liangyou Li, and Qun Liu. 2022. Adatrans: Adapting with boundary-based shrinking for end-to-end speech translation. *arXiv preprint arXiv:2212.08911*.

Chengqi Zhao, Mingxuan Wang, Qianqian Dong, Rong Ye, and Lei Li. 2020. Neurst: Neural speech translation toolkit. *arXiv preprint arXiv:2012.10018*.