arXiv:2305.05271v1 [cs.CL] 9 May 2023

神经转导器语音识别中的稳健声学与语义上下文偏置

Xuandi Fu†、*Kanthashree Mysore Sathyendra*†、*Ankur Gandhe*、*Jing Liu*、*Grant P. Strimel*、*Ross McGowan*、*Athanasios Mouchtaris*

亚马逊 Alexa

摘要

基于注意力的上下文偏置方法已显著提升端到端自动语音识别（E2E ASR）系统（如神经转导器）对通用和/或个人稀有词的识别效果。这些方法采用交叉注意力，使模型偏向于作为偏置短语注入模型的特定上下文实体。以往方法通常依赖子词编码器对偏置短语进行编码。然而，子词分词较为粗略，无法捕获细粒度的发音信息，而这些信息对于基于声学相似度进行偏置至关重要。本文提出使用轻量级字符表示来编码细粒度发音特征，从而改进由音频与上下文实体之间的声学相似度引导的上下文偏置（称为*声学偏置*）。我们进一步整合基于预训练神经语言模型（NLM）的编码器，对话语的语义上下文及上下文实体进行编码，从而根据话语的语义上下文实施偏置（称为*语义偏置*）。在 LibriSpeech 数据集上使用 Conformer 转导器模型进行的实验表明，与基线上下文模型相比，采用我们提出的声学与语义偏置方法后，在不同偏置列表大小下，词错误率相对降低了 4.62% 至 9.26%。在大规模内部数据集上，与基线模型相比，我们观察到词错误率相对降低了 7.91%。在长尾话语上，改进更为显著：在 LibriSpeech 稀有词和内部测试集上，词错误率分别相对降低了 36.80% 和 23.40%。

索引词—— 上下文偏置、注意力机制、RNN-T、Conformer、端到端自动语音识别、神经转导器、个性化

1. 引言

端到端自动语音识别系统因其单体式架构和易于训练而日益普及，因此成为部署于商用语音助手（VAs）的理想候选方案。然而，端到端模型通常依赖词片段词汇表，导致罕见实体经常被拆分为训练数据中不常见的目标序列，使模型难以正确识别这些实体 [1–5]。为了提供最佳用户体验，语音助手需要纳入每位用户的自定义环境和偏好，并利用这些信息改进对个性化请求的识别，例如“拨打 *[联系人姓名]*”，以及“在 *Spotify* 上播放我的 *[播放列表]*”。此外，识别其他类型的罕见短语也很有用，例如热门词（如“冠状病毒”），或屏幕上出现的罕见词（如语音助手屏幕所显示维基百科页面中的罕见短语，或先前文本转语音回复中的罕见短语）。上下文偏置被广泛用于注入此类词汇上下文，以引导模型的预测倾向于这些词汇上下文。词汇上下文的例子包括用户定义术语（如用户播放列表、用户联系人）和通用罕见短语（屏幕显示的罕见词、热门词等）。

端到端自动语音识别模型的神经上下文偏置方法大致分为两类——图融合方法（例如深度浅层融合、字典树和基于 WSFT 的方法） [6–8]以及全神经注意力方法 [2, 3, 9–12]。在第一类方法中， [1, 13]分别提出了深度浅层融合，以及结合深度个性化语言模型融合的基于字典树的上下文偏置。近来，全神经注意力上下文偏置方法因其能够提升罕见词识别效果，且易于与端到端神经推理引擎集成，正变得日益流行。文献 [2, 14] 探索了面向 LAS 模型的神经上下文偏置：通过双向长短期记忆编码器对偏置短语和/或上下文实体进行编码，并利用位置感知注意力机制使模型倾向于这些内容 [15–17]。针对 RNN-T 模型， [3, 9, 10] 提出了全神经注意力上下文偏置方法。 [9, 10] 针对 Transformer 转导器（T-T）和 Conformer 转导器（C-T）的音频编码器和基于文本的预测网络，提出了上下文偏置方法。 [10] 利用上下文适配器探索了预训练 RNN-T 和 C-T 模型的上下文适配；这种适配器速度更快、成本更低，且数据效率更高。

为嵌入上下文实体，先前的方法依赖于文本编码器（通常基于长短期记忆网络），并使用基于字素的子词/SentencePiece 分词 [2, 3, 10, 12]。然而，两个声学相似词可能采用完全不同且互不重叠的子词分词。例如，单词“seat”和“meat”虽然声学上相似，但其子词分词却大不相同： [sea, t] 和 [meat]。假设“seat”是一个新出现的未登录词，它可能会被错误地预测为“sit”（后者在训练数据中可能更常见）。字符级编码器或许能更好地捕捉“seat”和“meat”在发音上的相似性。 [16, 17] 还提出了字素与音素联合上下文嵌入，以纳入发音特征。然而，这需要发音词典或字素到音素的转换网络。此外，一个单词可能有多种发音，从而导致偏置短语列表更长。为解决这些问题，我们采用字符嵌入模型，它不仅能捕捉细粒度的发音信息，而且由于字符词表较小，模型也很轻量，所需的模型参数更少。

也有研究提出利用语言模型的语义上下文特征，提升自动语音识别系统对稀有词的识别能力 [18, 19]。在 [3, 9, 10], 中，转导器预测网络的输出被直接用作预测网络侧上下文偏置层的查询。然而，预测网络编码语义上下文的能力受其训练数据限制，对于长尾话语和稀有词，其中的语义上下文并未得到充分训练 [20]。因此，有必要集成一个基于大量纯文本数据训练、能够更好地编码语义的预训练语言模型（PLM）

†共同贡献

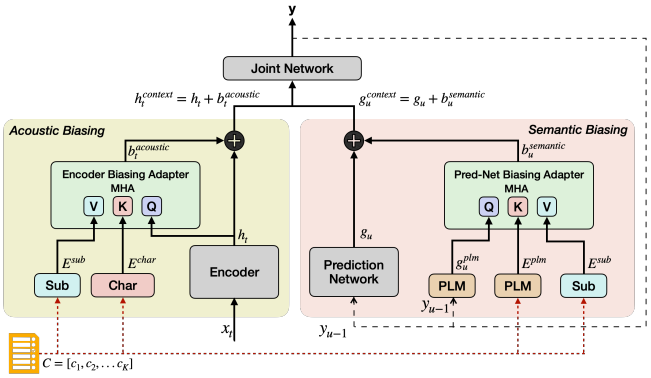


图 1： 所提出的声学与语义上下文偏置模型。字符、子词和预训练语言模型分别表示字符、子词和预训练语言模型编码器。相同颜色表示共享参数。

。将预训练语言模型的表示用作预测网络侧偏置层的查询，有助于实现更好的语义偏置。更广义地说，语义偏置旨在解决这样一个问题："给定先前解码出的子词单元，列表中的哪个上下文实体最有可能接着出现？"它确保模型预测的子词单元与所选上下文实体之间在语义上连贯一致。

本文提出在编码器侧训练基于字符的上下文表示，以捕捉可泛化的声学上下文特征，从而根据上下文实体与输入音频之间的声学相似度进行偏置。我们还引入神经语言模型编码器，对用于偏置的话语上下文和词汇上下文进行编码，以利用话语的语义和含义，从而将其推广到开放领域的应用场景。

2. 方法

神经转导器模型通常由编码器网络（H）、预测网络（P）和联合网络（J）组成 [21]。编码器网络的输入是长度为 T 的音频特征，$\boldsymbol{X} = (x_0, x_1, .., x_T)$。编码器网络 H（通常由堆叠的循环神经网络或 Conformer 模块构成）将音频帧 x_t 编码为 D 维中间表示 $h_t = H(x_t)$。预测网络将先前预测的非空白子词单元序列 $\boldsymbol{Y} = (y_0, y_1, .., y_{u-1})$编码为隐藏状态 g_u。联合网络融合时间步 t 的编码器隐藏状态和步骤 u 的预测网络隐藏状态，以计算下一个子词单元的概率，如下所示：$z_{t,u} = J(h_t + g_u)$。 J 是一系列可选用非线性激活的前馈层。转导器模型通过 RNN-T损失进行优化，该损失使用前向后向算法计算对齐概率 [21]。为使转导器模型具备上下文信息并帮助识别上下文实体，我们对包含 K 个偏置短语的列表 $\boldsymbol{C} = \{c_1, c_2, ..., c_K\}$进行编码，并将其注入转导器模型 [9, 10]。如图 1 所示，编码器侧偏置适配器和预测网络侧偏置适配器分别通过学习具备声学和语义上下文感知能力的编码器表示与预测网络表示，使模型偏向正确的上下文实体，$h_t^{context}$和 $g_u^{context}$。

2.1. 声学偏置

编码器偏置模块依靠当前音频特征与偏置列表之间的声学相似度，将注意力集中在声学上最相似的上下文实体上。基于注意力的偏置适配器以编码器状态作为查询，以偏置列表作为键和值，计算跨模态（来自音频和文本表示）的相似度。我们使用字符编码器对偏置短语进行编码，并将其用作编码器偏置适配器中注意力计算的键，从而关联声学隐藏状态 h_t 与外部上下文。所提出的编码器上下文适配器由三个组件构成：字符级键编码器、子词级值编码器，以及用于将上下文信息融入编码器隐藏状态的基于交叉注意力的偏置适配器。

2.1.1. 字符键编码器

字符键编码器将每个上下文实体 c_k 编码为字符上下文嵌入 e_k^{char}。每个上下文实体 c_k 首先被拆分为字符序列 c_k^{char}，并依次通过嵌入层和堆叠的双向长短期记忆网络层；双向长短期记忆网络输出的最终状态被用作键嵌入，记为 $E^{char} = [e_1^{char}, e_2^{char}, ..., e_K^{char}] \in \mathbb{R}^{K \times D^c}$ $e_k^{char} =$，其中 $BiLSTM\ Embedding\ c_k^{char}$ D^c（（））， 是字符嵌入的嵌入大小。

2.1.2. 子词值编码器

子词编码器用于为基于注意力的偏置层生成值，从而为模型的子词预测提供信息（因此使用子词嵌入）。为与转导器模型 [22, 23] 的输出单元保持一致，我们使用与 C-T 模型相同的句子片段模型。分词后的子词序列依次通过嵌入层和堆叠的双向长短期记忆网络层，双向长短期记忆网络的最终状态被传递为子词级上下文嵌入，记为 $E^{sub} = [e_1^{sub}, e_2^{sub}, ..., e_K^{sub}]$，且每个嵌入都是一个 D^s 维向量。

2.1.3. 用于声学偏置的编码器偏置适配器（*EBA*）

EBA 依靠音频与偏置列表 [10] 之间的声学相似度，学习向相关上下文实体进行偏置。多头注意力（MHA）模块将音频/编码器隐藏状态 h_t 作为查询，并将偏置短语 E^{char} 的字符上下文嵌入列表作为键，以计算每个实体的注意力分数 a_t^k（公式 1），用于 $t \in [0, T]$。查询 和键 首先分别通过 $\boldsymbol{W}^k$ 和 $\boldsymbol{W}^q$ 进行投影。通过缩放点积注意力机制计算得到的注意力分数 a_t^k，衡量时间步 t 的音频特征与偏置短语 c_k 之间的相似度。偏置向量 $b_t^{acoustic}$ 通过对投影后的值 嵌入（经由 $\boldsymbol{W}^v$ 投影）进行注意力加权求和得到，并加回到 h_t 中，以获得上下文感知特征 $h_t^{context}$（公式 2）。

$$a_t^k = Softmax(\frac{\boldsymbol{W}^k e_k^{char} \boldsymbol{W}^q h_t}{\sqrt{d}}) \tag{1}$$

$$b_t^{acoustic} = \sum_k^K a_t^k \boldsymbol{W}^v e_k^{sub}; \quad h_t^{context} = h_t + b_t^{acoustic} \tag{2}$$

2.2. 语义偏置

端到端模型的一个局限在于，解码器/预测网络必须基于带转写标注的音频数据进行训练；与大规模纯文本语料库相比，这类数据所涵盖的载体短语或语义话语上下文较为有限。由于声学偏置学习的是上下文嵌入与音频帧之间的相似性，因此往往会对同音词给予同等关注，无法区分它们。在这些情况下，来自话语语义上下文的强指示可以帮助模型在具有相似发音的词语中选择正确的实体进行偏置。为此，我们提出了一种

Acoustic Biasing Only	Acoustic Biasing + Semantic Biasing
show me that janie cow	show me that giant cow
what's the simple recipe for recipe again	what's the simple syrup recipe again
what is laser beam latest video	what is laser beam's latest video
find richard samsung tv	find richard's samsung tv
computer play station extreme my radio from tunein	computer play station extreme on my radio from tunein
what is cognag made out of	what is cognac made out of
order gillette sport triumph triumph deodorant for men	order gillette sport triumph deodorant for men
play songs by francisco cabrel	play songs by francis cabrel
play nobody's better by petit wap	play nobody's better by fetty wap
read the book owen by kevin hunks	read the book owen by kevin henkes

图 2： 语义改进（绿色）、句法改进（红色）、拼写纠正（蓝色）以及来自PLM模型的外部知识整合（黄色）示例。

基于预训练语言模型（PLM）的查询和键编码器，用于预测网络偏置。

2.2.1. 基于*PLM*的查询-键编码器

基于PLM的编码器（下文简称PLM 编码器）对先前预测的词片段（y_0，y_1，……，y_{u-1}）进行编码，以生成话语语义上下文向量 $g_u^{plm} \in \mathbb{R}^{1\times D^p}$，其中 D^p 表示PLM嵌入维度。同一个PLM 编码器也用于对偏置短语进行编码，从而使查询 和键处于共享嵌入空间中。 C 中的偏置短语首先使用与转导器相同的SentencePiece模型进行分词，拆分为子词单元，然后通过PLM编码，得到 $E^{plm} \in \mathbb{R}^{K\times D^p}$。

2.2.2. 用于语义偏置的预测网络偏置适配器（*PNBA*）

PNBA采用与编码器侧类似的缩放点积注意力（公式 1），计算话语上下文与偏置短语之间的语义相似度 。查询 是由 PLM 生成的语义嵌入 g_u^{plm}，用于编码话语语义上下文；键 是由 PLM 生成的上下文嵌入 E^{plm}。对于值，我们使用与编码器侧偏置适配器共享的子词值嵌入 E^{sub}。随后，按照公式 2 中针对 u 的类似方式，计算语义预测网络偏置向量 $b_u^{semantic}$，并将其与预测网络隐藏状态融合，得到 $g_u^{context} = g_u + b_u^{semantic}$。生成的上下文感知隐藏状态 $g_u^{context}$ 被传递至联合网络，以预测下一个词片段 $z_t^{context}$。 $P(z_t|h_t^{context}, g_u^{context}) = J(h_t^{context} + g_u^{context})$ 的概率。偏置适配器采用 [10] 提出的方法，通过上下文自适应方法使用 RNN-T 损失进行训练。

3. 实验结果

3.1. 数据集和评估指标

我们的实验在 LibriSpeech [25] 和一个内部大规模语音助手数据集上进行。对于 LibriSpeech 数据集，我们按照 [19] 的方法，从训练词汇表中移除最常见的 5000 个词，以构建全局稀有词列表。训练期间，从全局稀有词列表中随机抽取 K= 50 个干扰项，并将其与正确词和特殊的 <无偏置> 标记 [2, 10] 一同加入偏置列表。评估期间也以类似方式构建偏置列表，其中包括正确实体和 <无偏置>。此外，我们还针对 K = {50, 100, 500, 1000} 的情况，使用更大的偏置列表进行评估。

内部数据集包含 1.8 万小时的去标识化音频数据，其中包含来自多个领域的罕见词，例如歌曲名称、地点名称、设备名称、播放列表名称 以及其他长尾实体。模型在四个测试集上进行评估，K = 100 偏置列表大小。通用：测试话语按原始训练数据分布抽样（包含罕见和非罕见话语）。罕见词：包含至少一个罕见词的测试话语（即不在训练数据中出现频率最高的 1000 个词之列）。零样本罕见词：包含长尾话语的测试集，这些话语中的罕见词未出现在训练数据中。知识：为了测试模型的语义偏置性能，我们还准备了一个包含多样语义上下文的测试集，汇集了来自维基百科、知识 和图书等领域的话语。我们还在数据集上测试模型在提供个性化上下文时的性能——专有名称 和设备。

我们使用四项指标评估模型性能：(1) 词错误率：整个测试集上的总体词错误率；(2)稀有词词错误率：需要偏置的稀有词（RW）的词错误率；(3)非稀有词词错误率：不需要偏置的非稀有词的词错误率；(4) 零样本识别词错误率：零样本稀有词的词错误率，即训练期间未见过的词。对于内部数据集，结果以相对错误率降低幅度[1]（*WERR*、*R-WERR*、*NR-WERR* 和 *ZSR-WERR*）[2]呈现。

3.2. 基线和模型配置

所提出的方法与仅使用子词编码器对上下文实体 [10] 进行编码的上下文偏置适配器基线方法进行比较。我们遵循 [10] 中相同的适配器式训练策略，首先训练核心转导器模型，然后在冻结基础转导器参数的情况下，使用 混合 数据集训练适配器，以避免降低对通用词汇的识别性能。

ASR模型。 输入音频特征为64维LFBE特征，每10毫秒提取一次，窗口大小为25毫秒。随后，将每帧特征与左侧两帧堆叠，再以3倍因子进行下采样，以实现较低帧率，得到192维特征。为了训练偏置层，我们使用包含稀有词的话语与包含常见词的通用话语组成的混合数据集，教模型学习上下文实体和 <no-bias> 标记的表示。上下文偏置适配器采用流式Conformer-transducer (C-T)架构进行训练，以协同解码子词单元。对于LibriSpeech，我们在音频编码器中使用14个Conformer模块 [24] 。Conformer编码器的隐藏状态大小 d_{model}、投影维度 d_{ff} 和注意力头数 h 分别为256、1024和4。对于预测网络，转录文本首先由包含2500个子词单元的SentencePiece分词器 [23],进行分词，然后经过一个256维嵌入层，再输入包含640个单元、输出大小为512维的单层长短期记忆网络。对于内部数据集，音频编码器包含12个Conformer模块，具有 $d_{model} = 512$、$d_{ff} = 512$ 和 $h = 4$，输出大小为512维。预测网络使用512维嵌入层对子词单元进行编码，并采用包含736个单元的双层长短期记忆网络，输出投影至512维。

字符-子词编码器偏置模型 对于 LibriSpeech 的字符-子词偏置模型，上下文实体首先使用包含 38 个字符的分词器进行分词，然后通过 64 维嵌入层进行嵌入，接着经过一个包含 64 个单元的双向长短期记忆网络层。对于子词表示，上下文实体使用包含 2500 个词片的 SentencePiece 分词器进行分词，并通过 64 维嵌入层进行编码，随后经过一个包含 64 个单元的双向长短期记忆网络层。声学偏置层中多头注意力层的查询、键和值被投影为 128 维张量。对于内部数据集，我们使用大小相同的字符级上下文编码器和子词上下文编码器，而上下文实体则通过包含 4000 个词片的 SentencePiece 分词器进行分词

[1]由于公司内部政策，我们无法报告内部数据上的绝对词错误率指标[2]给定模型 A 的词错误率（WER_A）和基线 B 的词错误率（WER_B），A 相对于 B 的 WERR 计算方式为 $WERR = (WER_B - WER_A)/WER_B$。

表 1：Librispeech。模型在 K = {50, 100, 500, 1000} 条件下的绝对词错误率。编码器偏置适配器的输入形式为（键、值），Pred-Net 偏置适配器的输入形式为（查询、键、值）。*"子"*、*"字符"*、*"预训练语言模型"*和*"PredNet"*分别是子词嵌入 E^{sub}、字符嵌入 E^{char}、预训练语言模型嵌入 E^{plm} 和预测网络状态 g_u 的缩写。

模型类型	编码器偏置适配器输入（***K***、***V***）	***Pred-Net*** 偏置适配器输入（***Q***、***K***、***V***）	***test-clean***				测试集-其他			
			K=50	*K=100*	*K=500*	*K=1000*	*K=50*	*K=100*	*K=500*	*K=1000*
C-T [24]	-	-	6.08				14.01			
基线（*B1*） *[10]*	（子词，子词）	-	4.70	4.75	5.05	5.19	12.08	12.21	12.75	13.05
字符-*I*	*(字符, 字符)*	-	4.98	5.06	5.30	5.43	12.67	12.70	13.20	13.55
字符-*II*	*(字符, 子词)*	-	4.56	4.63	4.82	5.08	11.80	12.06	12.51	12.89
字符-子词	*(字符, 子词)*	*(PredNet, 子词, 子词)*	4.27	4.31	4.67	4.95	11.16	11.34	12.21	12.64
子词-预训练语言模型	*(子词, 子词)*	*(PredNet, 子词, 子词)*	4.22	4.32	4.63	4.95	**11.01**	11.22	12.11	12.72
Char-PLM	*(字符, 子词)*	*(预训练语言模型, 预训练语言模型, 子词)*	**4.10**	**4.11**	**4.52**	**4.83**	**11.01**	**11.21**	**12.09**	**12.60**

表 2：内部数据集。各种内部测试集上的 WERR。推理上下文实体 (C) 是罕见词 *(RW)* 、专有名称 *(PN)* 或设备名称 *(D)* 之一。*K=100*。

模型	通用	***RW***	***ZSRW***	知识 专有名称 设备		
	C = (RW)	*C =(RW)*	*C =(RW)*	*C = (RW)*	*C = (PN)*	*C = (D)*
基线（*B1*）	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
字符-*I*	-4.11%	-9.54%	-17.45%	-5.18%	-13.78%	+3.10%
字符-*II*	+0.91%	+1.19%	+5.52%	+2.38%	+3.24%	+1.65%
字符-子词	+5.18%	+8.48%	+16.40%	+6.63%	+12.24%	**+5.58%**
子词-预训练语言模型	+7.00%	+12.58%	+15.80%	+9.02%	+8.91%	+3.10%
Char-PLM	**+7.91%**	**+14.97%**	**+23.40%**	**+11.40%**	**+13.13%**	+4.75%

表3：少样本泛化在 LibriSpeech（*test-clean*）上的结果。稀有词的 n-shot 词错误率；$n \leq \{0, 1, 5, 10, 20, 100\}$。*K=100*。

模型	零样本识别词错误率						稀有词词错误率	非稀有词词错误率
	0样本	≤***1*** 样本	≤***5*** 样本	≤***10*** 样本	≤***20*** 样本	≤***100*** 样本		
C-T	113.6	103.9	85.4	70.0	54.5	27.7	24.6	3.8
基线（*B1*）	91.2	81.6	60.1	46.2	33.5	15.2	13.1	3.9
字符-*II*	85.5	75.2	55.2	42.1	30.4	13.7	12.0	3.8
字符-子词	79.2	67.7	48.8	37.0	26.0	11.5	10.0	3.7
子词-预训练语言模型	81.9	67.8	47.9	36.0	25.5	11.0	9.6	3.7
Char-PLM	**71.0**	**58.8**	**42.3**	**31.5**	**22.0**	**9.6**	**8.3**	**3.6**
WERR	**+22.1%**	**+27.9%**	**+29.6%**	**+31.8%**	**+34.3%**	**+36.8%**	**+36.6%**	**+7.7%**
词数	*331*	*461*	*814*	*1154*	*1772*	*4844*	*5752*	*46815*

以获得子词表示。在训练过程中，对于 LibriSpeech 模型和内部模型，我们分别使用学习率为 5e-4 和 1e-4 的 Adam 优化器。

预训练语言模型（PLM）编码器：对于 LibriSpeech，PLM 编码器采用 256 维嵌入层，后接一个包含两层长短期记忆网络（每层 256 个单元）的模型，并使用仅含文本的 Wikitext-103 语料库 [26]进行训练。对于内部数据，我们使用匿名用户交互中的实时音频，对一个领域通用神经语言模型（嵌入维度为 512，LSTM 大小为 512）进行了训练，共使用 8000 万条话语。为提高对稀有实体的覆盖率，我们还加入了艺术家和歌曲名称目录中的 2500 万个实体，以及地名目录中的 800 万个实体。

3.3. 结果

表1、表2展示了 LibriSpeech 和内部数据集上的词错误率。在 LibriSpeech 上，所提出的声学-语义偏置模型优于子词嵌入基线 [10]，且适用于不同的偏置列表大小（+9.26% 和 +4.62% 的 WERR，分别对应 test-clean K=100 和 1000）。对于内部数据集，该模型在 通用 （+7.91% WERR）和 稀有词 测试集上（+14.97% WERR）均取得了提升。**泛化能力得到提升**。 字符键编码器结合基于 PLM 的预测网络偏置，在识别零样本和少样本稀有词方面显著优于基线 （*B1*）。在 LibriSpeech 上，与 B1 相比，*Char-II* 的 0-*shot* 稀有词词错误率从 91.2% 降至 85.5%； 采用基于 PLM 的预测网络偏置后，进一步降至 71.0%

表 4： 对于 PNBA，在不同的查询-键 *(Q-、K-)* 对中，注意力机制选择正确实体进行偏置的准确率。

模型	注意力准确率 相对提升（%）	平均注意力分数 相对提升（%）
（查询预测网络，键子词）	0.00	0.00
（查询预测网络，键预训练语言模型）	+2.25	+13.80
(查询预训练语言模型, 键预训练语言模型)	**+7.49**	**+16.55**

偏置（表3）。随着稀有词训练样本数量增加，相对提升更加明显（+36.8% WERR，对应 ≤100-*shot*）。在内部 ZSRW 数据集上，*Char-II* 取得了 +5.52% 的 WERR。为进一步分析 PLM 集成带来的提升，我们比较了 *Char-PLM* （基于 PLM 的查询-键）与 *Char-Subword* 模型（以 g_u 为查询、以 子词 为键）。在 ZSRW 测试集上，WERR 从 *Char-Subword* 的 16.40% 提升至 *Char-PLM* 的 23.40%（表2）。

基于预训练语言模型的预测网络偏置的重要性。基于 PLM 的查询-键偏置层在语义、句法、拼写改进和外部知识整合方面展现出更优的识别效果。图 2 展示了一些定性示例。借助从更大文本语料库中学习到的外部知识，基于 PLM 的偏置层能够正确关注艺术家姓名*"Francis Cabrel"*。在包含来自多个领域的尾部词/稀有词的知识内部测试集中，基于 PLM 的模型相比 B1 实现了 11.40% 的 WERR。

注意力机制评估。为比较所提出的语义偏置模型依据语义相似度关注正确偏置短语（而非干扰项）的能力，我们计算了另外两项指标：1）注意力准确率 *(AA)(%)*：模型关注正确上下文实体（即对正确实体的注意力分数最高）的话语所占百分比。2）平均注意力分数 *(AAS)(%)*：注意力模型为正确偏置短语赋予的平均注意力分数。我们在内部*ZSRW* 测试集上进行了比较（表 4——以相对提升幅度报告）。与使用预测网络输出作为查询相比，基于预训练语言模型的偏置模型（*(Q-PLM, K-PLM)*）的注意力准确率 和平均注意力分数 分别相对提升了 7.49% 和 16.55%。这表明该模型能够利用话语语义上下文来选择关注的实体。

4. 结论

在本研究中，我们提出利用基于字符的声学偏置和基于预训练语言模型的语义偏置模型，改进 Conformer-Transducer 偏置模型，从而将话语上下文与正确的偏置实体关联起来。通过在 LibriSpeech 和大规模内部数据集上开展实验，我们证明，所提出的声学-语义偏置模型能够编码更好的上下文特征，并学习由声学相似度和语义相似度共同引导的更优偏置模型。与基线相比，该模型在泛化至长尾稀有词和开放域应用场景时也取得了显著提升。

5. 参考文献

[1] Duc Le、Gil Keren、Julian Chan、Jay Mahadeokar、Christian Fuegen 和 Michael L Seltzer，“用于循环神经网络传感器个性化的深度浅层融合”，载于*2021*年 *IEEE* 口语技术研讨会 *(SLT)*。IEEE，2021年，第 251–257 页。[2] Golan Pundak、Tara N Sainath、Rohit Prabhavalkar、Anjuli Kannan 和 Ding Zhao，“深度上下文：端到端上下文语音识别”，载于*2018* 年 *IEEE* 口语技术研讨会 *(SLT)*。IEEE，2018年，第 418–425 页。[3] Mahaveer Jain、Gil Keren、Jay Mahadeokar、Geoffrey Zweig、Florian Metze 和 Yatharth Saraf，“面向开放领域自动语音识别的上下文循环神经网络传感器”，*Interspeech 2020* 论文集，第 11–15 页，2020年。[4] Tara N Sainath、Rohit Prabhavalkar、Shankar Kumar、Seungji Lee、Anjuli Kannan、David Rybach、Vlad Schogol、Patrick Nguyen、Bo Li、Yonghui Wu 等，“还需要词典吗？评估发音词典在端到端模型中的价值”，载于*2018*年 *IEEE* 国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2018年，第 5859–5863 页。[5] Tony Bruguier、Fuchun Peng 和 Franc.oise Beaufays，“学习联系人姓名识别的个性化发音”，2016年。[6] Ding Zhao、Tara N Sainath、David Rybach、Pat Rondon、Deepti Bhatia、Bo Li 和 Ruoming Pang，“浅层融合端到端上下文偏置”，2019年。[7] Yanzhang He、Tara N Sainath、Rohit Prabhavalkar、Ian McGraw、Raziel Alvarez、Ding Zhao、David Rybach、Anjuli Kannan、Yonghui Wu、Ruoming Pang 等，“面向移动设备的流式端到端语音识别”，载于*2019*年 *IEEE ICASSP*。IEEE，2019年，第 6381–6385 页。[8] Aditya Gourav、Linda Liu、Ankur Gandhe、Yile Gu、Guitang Lan、Xiangyang Huang、Shashank Kalmane、Gautam Tiwari、Denis Filimonov、Ariya Rastrow 等，“端到端语音识别系统的个性化策略”，载于*2021*年 *IEEE ICASSP*。IEEE，2021年，第 7348–7352 页。[9] Feng-Ju Chang、Jing Liu、Martin Radfar、Athanasios Mouchtaris、Maurizio Omologo、Ariya Rastrow 和 Siegfried Kunzmann，“用于语音识别的上下文感知 Transformer 转导器”，载于*2021*年 *IEEE* 自动语音识别与理解研讨会 *(ASRU)*。IEEE，2021年，第 503–510 页。[10] Kanthashree Mysore Sathyendra、Thejaswi Muniyappa、Feng-Ju Chang、Jing Liu、Jinru Su、Grant P Strimel、Athanasios Mouchtaris 和 Siegfried Kunzmann，“神经转导器中的上下文适配器：用于个性化语音识别”，载于*ICASSP 2022-2022 IEEE* 国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2022年，第 8537–8541 页。[11] Minglun Han、Linhao Dong、Zhenlin Liang、Meng Cai、Shiyu Zhou、Zejun Ma 和 Bo Xu，“通过细粒度上下文知识选择改进端到端上下文语音识别”，载于*ICASSP 2022-2022 IEEE* 国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2022年，第 8532–8536 页。[12] Tsendsuren Munkhdalai、Khe Chai Sim、Angad Chandorkar、Fan Gao、Mason Chua、Trevor Strohman 和 Franc.oise Beaufays，“利用神经关联记忆快速实现设备端个性化语音识别的上下文适配”，载于*2022*年 *IEEE ICASSP*，2022年，第 6632–6636 页。

[13] Ding Zhao、Tara N. Sainath、David Rybach、Pat Rondon、Deepti Bhatia、Bo Li 和 Ruoming Pang，《浅层融合端到端上下文偏置》，载于*Interspeech 2019* 论文集，2019年，第 1418–1422 页。[14] William Chan、Navdeep Jaitly、Quoc V. Le 和 Oriol Vinyals，《聆听、注意与拼写：一种用于大词汇量会话语音识别的神经网络》，载于*ICASSP*，2016年。[15]Jan Chorowski、Dzmitry Bahdanau、Dmitriy Serdyuk、Kyunghyun Cho 和 Yoshua Bengio，《基于注意力的语音识别模型》，*arXiv* 预印本 *arXiv:1506 07503*，2015年。。[16] Zhehuai Chen、Mahaveer Jain、Yongqiang Wang、Michael L Seltzer 和 Christian Fuegen，《用于上下文端到端自动语音识别的联合字素和音素嵌入》，2019年。[17] Antoine Bruguier、Rohit Prabhavalkar、Golan Pundak 和 Tara N Sainath，《Phoebe：面向端到端语音识别的发音感知上下文化》，载于*ICASSP 2019-2019 IEEE* 国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2019年，第 6171–6175 页。[18] Ashish Shenoy、Sravan Bodapati、Monica Sunkara、Srikanth Ronanki 和 Katrin Kirchhoff，《为会话代理中的自动语音识别重打分调整长上下文神经语言模型》，*arXiv* 预印本 *arXiv:2104.11070*，2021年。[19] Duc Le、Mahaveer Jain、Gil Keren、Suyoun Kim、Yangyang Shi、Jay Mahadeokar、Julian Chan、Yuan Shangguan、Christian Fuegen、Ozlem Kalinli 等，《结合基于字典树的深度偏置和浅层融合的上下文化流式端到端语音识别》，*arXiv* 预印本 *arXiv:2104.02194*，2021年。[20] Mohammadreza Ghodsi、Xiaofeng Liu、James Apfel、Rodrigo Cabrera 和 Eugene Weinstein，《采用无状态预测网络的循环神经网络转导器》，载于*ICASSP 2020*。IEEE，2020年，第 7049–7053 页。[21] Alex Graves，《采用循环神经网络的序列转导》，*arXiv* 电子预印本，第 arXiv–1211 页，2012年。[22] Taku Kudo，《子词正则化：通过多个子词候选项改进神经网络翻译模型》，载于计算语言学协会第 *56* 届年会论文集（第 *1* 卷：长篇论文），2018年，第 66–75 页。[23] Taku Kudo 和 John Richardson，《SentencePiece：一种用于神经文本处理的简单、语言无关的子词分词器和去分词器》，载于*2018* 年自然语言处理实证方法会议论文集：系统演示，2018年，第 66–71 页。[24] Anmol Gulati、James Qin、Chung-Cheng Chiu、Niki Parmar、Yu Zhang、Jiahui Yu、Wei Han、Shibo Wang、Zhengdong Zhang、Yonghui Wu 等，《Conformer：用于语音识别的卷积增强型转导器》，*Interspeech 2020* 论文集，第 5036–5040 页，2020年。[25] Vassil Panayotov、Guoguo Chen、Daniel Povey 和 Sanjeev Khudanpur，《LibriSpeech：基于公共领域有声读物的自动语音识别语料库》，载于*2015*年 *IEEE* 国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2015年，第 5206–5210 页。[26] Stephen Merity、Caiming Xiong、James Bradbury 和 Richard Socher，《指针哨兵混合模型》，载于国际学习表征会议，2017年。