计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210900007

基于 wav2vec 预训练的样例关键词识别

李昭奇　黎　塔
中国科学院声学研究所语言声学与内容理解重点实验室　北京 100190
中国科学院大学　北京 100049
(lizhaoqi@hccl.ioa.ac.cn)

摘　要　样例关键词识别是将语音关键词片段与语音流中的片段匹配的任务。在低资源或零资源的情况下,样例关键词识别通常采用基于动态时间规正的方法。近年来,神经网络声学词嵌入已成为一种常用的样例关键词识别方法,但神经网络的方法受限于标注数据数量。使用 wav2vec 预训练可以减少神经网络对数据量的依赖,提升系统的性能。使用 wav2vec 模型提取的预训练特征直接替换梅尔频率倒谱系数特征后,在 SwitchBoard 语料库中提取的数据集上使双向长短时记忆网络的神经网络声学词嵌入系统的平均准确率提高了 11.1%,等精度召回值提高了 10.0%。将 wav2vec 特征与梅尔频率倒谱系数特征相融合以提取嵌入向量的方法进一步提高了系统的性能,与仅使用 wav2vec 的方法相比,融合方法的平均准确率提高了 5.3%,等精度召回值提高了 2.5%。

关键词:声学词嵌入;孤立词识别;wav2vec 预训练;样例查询;语音片段查询
中图法分类号　TP181

Query-by-Example with Acoustic Word Embeddings Using wav2vec Pretraining

LI Zhao-qi and LI Ta
Key Laboratory of Speech Acoustics and Content Understanding, Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China
University of Chinese Academy of Sciences, Beijing 100049, China

Abstract　Query-by-Example is a popular keyword detection method in the absence of speech resources. It can build a keyword query system with excellent performance when there are few labeled voice resources and a lack of pronunciation dictionaries. In recent years, neural acoustic word embeddings has become a commonly used Query-by-Example method. In this paper, we propose to use wav2vec pre-training to optimize the neural acoustic word embeddings system, which is using bidirectional long short-term memory. On the data set extracted in SwitchBoard, the features extracted by the wav2vec model are directly used to replace the Mel frequency cepstral coefficient features, which relatively increases the system's average precision rate by 11.1% and precision recall break-even point by 10.0%. Subsequently, we tried some methods to fuse the wav2vec feature and Mel frequency cepstral coefficient feature to extract the embedding vector. The average precision rate and precision recall break-even point of the fusion method is a relative increase of 5.3% and 2.5% compared to the method using only wav2vec.

Keywords　Acoustic word embedding, Isolated word discrimination, wav2vec pretraining, Query-by-example, Spoken term detection

1　引言

基于样例的(Query-by-Example,QbE)关键词识别是一项在目标语音数据库中搜索与请求的查询语音序列相似的子序列的任务。与常见的关键词系统使用文本来搜索不同,QbE 直接使用语音片段进行查询,这使它无需构建完整的语音识别系统即可进行与语言无关的关键词搜索。无论语言和具体的语义信息如何,此任务都只通过语音数据的相似度进行搜索,因此它可以在资源较少的情况下很好地执行查询,而不必担心语音资源不足的问题。

QbE 系统的搜索过程主要包括以下两个步骤:首先,将待查询语音和目标语音库转换为相同的表示形式,然后,使用这些表示形式来计算与查询相同的关键字作为子序列出现在目标语音库中某处的可能性。常用的表示形式主要有 3 种:频谱特征[1-3]、语音后验概率[4-6]和瓶颈特征[7-10]。在特征提取之后,动态时间规正(Dynamic Time Warping,DTW)是 QbE 系统中常用的序列相似度比较方法。

DTW[1]是一种搜索路径的方法,该路径可以表示两个语

到稿日期:2021-09-01　返修日期:2021-10-09
基金项目:国家重点研发计划(2020AAA0108002)
This work was supported by the National Key R&D Program of China(2020AAA0108002).
通信作者:黎塔(lita@hccl.ioa.ac.cn)

音序列之间的最小距离。路径搜索是在二维平面上逐帧进行的。平面上点的坐标是两个待比较的语音序列的帧序号，将每个点中两帧的相似度用作路径成本。逐帧搜索使得该方法能够得出准确的结果，而路径搜索的方式使比较过程不易受语音速度变换的影响，因此 DTW 至今仍然是最准确的相似度比较算法之一。但是，在二维平面上进行搜索只能实时地进行动态规划计算，需要耗费大量时间。对于 DTW 的并行性、搜索范围和准确性，已有多个优化的变体形式被提出：如斜率约束 DTW[4]、分段 DTW[5]、加权 DTW[11]、非分段 DTW[12]、子序列 DTW[13]、子空间正则化 DTW[14]、根据发音类型加权的 DTW[15]。

近年来，神经网络声学词嵌入（Neural Acoustic Word Embeddings，NAWEs）已逐渐成为一种流行的 QbE 方法。与 DTW 相比，NAWEs 方法不使用复杂的相似度比较算法，而是将方法的复杂性转移到计算固定长度嵌入向量的步骤中。已有多项研究将使用三元组损失（triplet loss）的孪生网络[2,16-19]用作嵌入函数。该方法通过带有弱标签（指示关键词之间是否相同）的成对关键词数据训练神经网络。通过训练，使得在嵌入向量表示的情况下，相同关键词的嵌入向量的距离近，不同关键词的向量距离远。因此，只需设定距离阈值即可判定二词是否相同，相似度比较过程简单快速。

在 QbE 任务中，多语言或单语言瓶颈特征[9]是一种常用的预训练方法，这种方法通过构建一个简单的声学模型，并将其中某个中间层的输出作为特征相使用。这种方法的一个明显缺点是，QbE 作为一种低资源任务通常不适合建立声学模型。由于标注的质量低和对齐的不正确，可能会在提取瓶颈特征的过程中引入一些错误。

wav2vec[20]是无监督的预训练方法，可用于改善语音任务。wav2vec 预训练可以使用无标注的语音数据执行，与样例关键词识别场景匹配。即使在相同的数据量下，wav2vec 预训练也可以减少训练时间并提高性能。wav2vec 已在语音识别系统中显示了出色的性能[21-22]。

本文使用 wav2vec 预训练改进了基于双向长短时记忆网络（Bidirectional Long Short-Term Memory，BLSTM）的样例关键词识别的性能，并通过实验确定了最佳的网络结构和训练参数，通过与梅尔频率倒谱系数（Mel Frequency Cepstral Coefficient，MFCC）特征相融合，进一步提升了系统的性能。

2 神经网络声学词嵌入样例关键词系统

如图 1 所示，神经网络声学词嵌入样例关键词系统通过神经网络将所有可变长度语音序列嵌入到固定长度向量中。

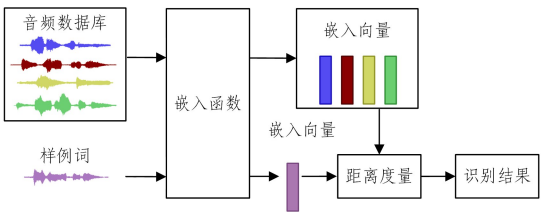


图 1 神经网络声学词嵌入样例关键词系统

Fig. 1 Neural acoustic word embeddings query by example system

通过使用至少具有弱标签的关键词对来训练神经网络，使网络可以将相同的关键词映射到彼此接近的向量，并将不同的关键词映射到相距较远的向量。使用以此方式训练好的神经网络作为嵌入函数，在将语音序列嵌入到向量中之后，直接比较嵌入向量之间的距离即可判定片段是否为相同的关键词。

具体来说，本文中的系统使用 3 层的 BLSTM 作为嵌入函数，并将两个方向网络的最后一帧输出向量拼接为嵌入向量：

$$\boldsymbol{E}=[\boldsymbol{hf}_T;\boldsymbol{hb}_1] \tag{1}$$

其中，$\boldsymbol{hf}_T$ 和 $\boldsymbol{hb}_1$ 是前向和后向的长短时记忆网络的最后一帧的输出向量，T 是语音序列的长度。使用三元组损失函数[23]训练神经网络，定义为：

$$L_{\text{triplet}}(\boldsymbol{Y}_a,\boldsymbol{Y}_p,\boldsymbol{Y}_n)=\max\{0,m+d(\boldsymbol{x}_a,\boldsymbol{x}_p)-d(\boldsymbol{x}_a,\boldsymbol{x}_n)\} \tag{2}$$

其中，$\boldsymbol{Y}_a$ 和 $\boldsymbol{Y}_p$ 是同一关键词的两个不同语音片段，$\boldsymbol{Y}_n$ 是另一个不同关键词的语音片段，$\boldsymbol{x}_a,\boldsymbol{x}_p,\boldsymbol{x}_n$ 是对应的神经网络嵌入的向量。三元组损失旨在在指定的距离度量 d 内使相同关键词映射到的向量距离近，而不同的关键词映射的向量距离远。m 是用于限制类间距离足够大于类内距离的余量参数。本文使用余弦距离作为距离度量：

$$d(\boldsymbol{x}_1,\boldsymbol{x}_2)=d_{\cos}(\boldsymbol{x}_1,\boldsymbol{x}_2)=1-\cos(\boldsymbol{x}_1,\boldsymbol{x}_2) \tag{3}$$

通过最小化三元组损失，目标样本嵌入向量 $\boldsymbol{x}_a$ 和正样本嵌入向量 $\boldsymbol{x}_p$ 的类内距离 $d(\boldsymbol{x}_a,\boldsymbol{x}_p)$ 减小，而 $\boldsymbol{x}_a$ 和负样本嵌入向量 $\boldsymbol{x}_n$ 的类间距离 $d(\boldsymbol{x}_a,\boldsymbol{x}_n)$ 增加。训练时，每个训练迭代周期随机排列训练集，随后依次对训练集中的每一个词选择一定个数的正样本和负样本进行训练。选择负样本时对训练集进行采样，然后选择最难区分的负样本进行训练。选择正样本时用同样的方法选择最难区分的正样本进行训练。训练期间不使用任何文本信息，只需要用来确定单词是否相同的类别标签。同样地，测试也不需要文本信息，因此训练好的系统实际上可以适用于任何语言，不受训练语种的限制。

3 wav2vec 无监督预训练

在许多应用场景中，足够的带标签数据和准确的词典通常代价高昂或难以获得，使得有效的语音识别系统和关键词系统难以构建。QbE 关键词识别不需要构建语音识别系统，并且在缺少资源的情况下表现出色。与 QbE 任务一样，训练 wav2vec[20]模型的数据成本也很少，只需要大量没有标注的语音数据即可。wav2vec 特征非常适用于改善 QbE 系统的性能。

wav2vec 是一种预训练的模型，通过无监督训练，该模型可使网络将原始语音样本映射到更能代表数据特征的特征空间。使用计算出的特征向量来替代 MFCC 等传统特征，可以改善后续的任务，比如语音识别或者 QbE。wav2vec 模型包含两个卷积神经网络，一个将原始输入音频信号映射到隐藏空间的编码器网络，另一个结合了编码网络的多个时间步输出的上下文网络。编码器为每个时间步 i 生成一个表示 z_i，而上下文网络将多个编码器时间步长输出组合为每个时间步 i 的一个新的表示 $\boldsymbol{c}_i$。通过最小化如下对比损失来训练模型：

$$L_k=-\sum(\log\sigma(\boldsymbol{z}_{i+k}^T h_k(\boldsymbol{c}_i))+\lambda\mathbb{E}_{\tilde{z}\sim p_n}[\log\sigma(-\tilde{\boldsymbol{z}}^T h_k(\boldsymbol{c}_i))]) \tag{4}$$

其中，$k=1,\cdots,K$ 为预测的步长，$h_k(\boldsymbol{c}_i)=\boldsymbol{W}_k\boldsymbol{c}_i+\boldsymbol{b}_k$ 是每个时间步计算的 $\boldsymbol{c}_i$ 仿射变换，$\boldsymbol{z}_{i+k}$ 是时间步 i 后第 k 步计算的真实结果。训练时，损失函数的优化旨在增加 $\boldsymbol{c}_i$ 能够通过这个仿射变换正确预测 $\boldsymbol{z}_{i+k}$ 的概率。$\sigma(x)=1/(1+\exp(-x))$ 是 sigmoid 函数，$\sigma(\boldsymbol{z}_{i+k}^{T}h_k(\boldsymbol{c}_i))$ 是 $\boldsymbol{c}_i$ 通过仿射变换计算后与真实样本 $\boldsymbol{z}_{i+k}$ 相同的概率。$\tilde{\boldsymbol{z}}$ 是在 wav2vec 模型训练期间从每个语音中自动采样的干扰因素，并且符合 $p_n=1/T$ 的平均分布，其中 T 是序列长度。在计算预测错误的期望值 $\mathbb{E}[\log\sigma(-\tilde{\boldsymbol{z}}^{T}h_k(\boldsymbol{c}_i))]$ 时，通过在同一条语音中选择 10 个干扰项来近似期望值。λ 是一个负的常数。

最后，通过优化每个步长的损失函数之和 $L=\sum_{k=1}^{K}L_k$ 来训练 wav2vec 模型。在训练之后，由上下文网络产生的表示 $\boldsymbol{c}_i$ 可以用来替代传统的声学特征，并且表现出比常见的滤波器组特征（如 MFCC）更好的性能。

4 实验与结果

4.1 数据集

本文用来训练嵌入网络的数据截取自 SwitchBoard 电话会话英语语料库[24]。将 Kaldi[25] 在每帧中提取的标准的 39 维 MFCC+Δ+Δ 特征用于基线系统和融合系统。训练集包含 1959 个词的语音片段，而开发集和测试集分别包含 14997 个和 11951 个词，测试集、训练集和开发集与文献[13]中的相同。

4.2 基线系统

本文的基线系统结构[2]是一个 3 层的 BLSTM 网络，每层中都有 0.3 的概率神经元随机失活。本文中所有模型都在 TensorFlow 中实现[26]。BLSTM 在每一层的每个方向上包含 256 个隐藏单元，网络输入可变长度的 MFCC 特征序列，最后输出 512 维固定长度的嵌入向量。在所有实验中，三元组损失的余量 m 设置为 0.6，并使用 Adam 优化器，其初始学习率为 0.001，每批处理 32 个数据。

在每个训练的迭代周期将整个训练集随机排列，并为每个单词随机选择 1 个正样本和 5 个负样本，然后使用当前网络下最难以区分的负样本来计算三元组损失。每种结构训练 200 个迭代周期后，选择开发集上性能最佳的一个周期作为最终模型。

4.3 评价指标

本文通过在测试集上进行孤立词识别任务来进行对系统的性能评价，作为样例关键词性能判定的替代任务，该方法在以往已多次使用[2,4,13]。

对于测试集的所有可能词对，通过神经网络嵌入到向量后，计算余弦距离。余弦距离小于阈值则判定为相同词，否则判定为不同词。遍历所有可能阈值并画出精度召回曲线，精度召回曲线下方的面积即为平均准确率（Average Presion，AP）。AP 越大，代表系统在所有阈值下的性能越好。精度召回曲线上精度与召回率之差的绝对值最小的点为等精度召回点（Break-even Point，BEP），本文取该点上精度和召回率中较小的一个作为 BEP 的值。BEP 越大，代表系统在精度和召回率均衡时的性能越好。

4.4 wav2vec 系统

为了用 wav2vec 系统提取特征替换 MFCC 特征输入的 BLSTM 作为新的嵌入函数，本文使用文献[20]中进行训练和公开的性能最佳的“Wav2Vec Large”模型，以便于实验结果的复现。该模型编码器网络增加了两个线性变换，上下文网络为卷积核大小从 2 递增至 13 的 12 层卷积神经网络。该模型使用 960 h 的 Librispeech[27] 数据进行训练。

本文首先研究了使用 wav2vec 特征的 BLSTM 层数的影响。随着层数的加深，网络参数和复杂性将增加，这将使网络性能得到提升但难以训练，计算也更加耗时。表 1 所列为堆叠 1 到 4 层 BLSTM 作为嵌入函数的性能，表中第一列表示嵌入网络输入的特征和网络的层数。在这些实验中，使用 wav2vec 特征的结构明显优于使用 MFCC 特征的最佳结果。考虑到与 3 层 BLSTM 相比，4 层 BLSTM 不但难以训练，并且在增加训练时间的同时性能并未得到明显改善，因此我们选择了性能和培训时间相对平衡的 3 层 BLSTM，用于后续测试。

表 1 不同层数下使用 wav2vec 的 BLSTM 系统的性能比较

Table 1 Performance comparison of BLSTM systems using wav2vec with different layers

LayerNum	*AP*/*BEP*/%	Time per epoch/s
3($MFCC_{39}$)	73.82/71.12	405
1(wav2vec)	77.21/74.37	216
2(wav2vec)	81.48/77.92	309
3(wav2vec)	82.01/78.23	420
4(wav2vec)	82.02/78.28	477

使用 wav2vec 特征训练的网络作为嵌入函数，与 MFCC 特征相比，*AP* 提高了 11.1%。在此基础上，本文将 wav2vec 特征与 MFCC 特征相融合输入嵌入网络。

4.5 嵌入网络输入的特征融合

将 wav2vec 和 MFCC 特征直接拼接作为嵌入网络的输入，在不同随机失活概率下的测试结果如表 2 所示。相比只使用 wav2vec 作为输入，将 wav2vec 和 MFCC 特征直接拼接输入不仅没有改善网络的 *AP*，而且在神经网络失活为 0 时其 *AP* 有所降低。这是因为将 512 维 wav2vec 特征和 39 维 MFCC 特征拼接在一起（为方便表示，表中记为 $MFCC_{39}$，用加号表示拼接），MFCC 特征的尺寸与 wav2vec 的维度不匹配。

表 2 在第一层输入进行特征融合的性能

Table 2 Performance of feature fusion in input layer

feature	dropout	*AP*/*BEP*/%
wav2vec	0	81.51/77.96
wav2vec	0.1	82.05/78.53
wav2vec	0.2	81.76/78.11
wav2vec	0.3	82.01/78.23
wav2vec+$MFCC_{39}$	0	81.41/77.89
wav2vec+$MFCC_{39}$	0.1	82.22/78.38
wav2vec+$MFCC_{39}$	0.2	82.03/78.25
wav2vec+$MFCC_{39}$	0.3	82.41/78.42
wav2vec+$MFCC_{512}$	0	83.23/79.15
wav2vec+$MFCC_{512}$	**0.1**	**83.23/79.18**
wav2vec+$MFCC_{512}$	0.2	82.88//78.63
wav2vec+$MFCC_{512}$	0.3	83.13/79.10

为了使 MFCC 特征在拼接后起到更大的作用，将 MFCC

通过一个全连接层映射到512维(表中记为 $MFCC_{512}$),再与wav2vec拼接在一起,将得到的1024维向量作为网络新的输入。与仅使用wav2vec的网络效果相比,拼接输入的网络的 AP 最多提高了1.18%,BEP 最多提高了0.65%。

4.6 在不同层进行输入的特征融合

wav2vec特征是通过神经网络提取的深层特征,而MFCC是通过滤波器组提取的浅层特征,二者适用的最佳网络层数可能不同。为了探索最佳的特征融合方式,本文通过在BLSTM网络的不同层输入一种或多种特征来进行特征融合,具体结构如图2所示。在图2中,$X=x_1\cdots x_T$ 是第一层的输入特征,$Y=y_1\cdots y_T$ 是第二层的输入特征。Y 和第一层的输出逐帧地拼接在一起,然后输入到第二层。类似地,$Z=z_1\cdots z_T$ 与第二层的输出拼接,并输入到第三层。在后续的实验中,X 一定输入,Y 和 Z 不一定输入。构建整个网络的特征输入后,训练整个网络作为嵌入函数。

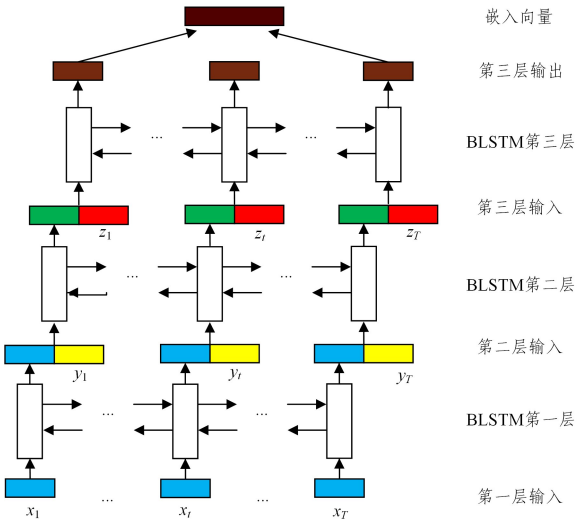


图2 多层特征融合结构

Fig. 2 Structure of feature fusion at different layers

首先,将 $MFCC_{39}$ 或 $MFCC_{512}$ 作为第一层的特征输入,同时在第二层或者第三层额外输入wav2vec特征,将这个网络结构按第2节的方法进行训练,随后将其作为嵌入函数进行样例关键词识别的结果,如表3所列。

表3 第一层输入MFCC,第二、三层输入wav2vec的性能

Table 3 Performance of input MFCC to the first layer and input wav2vec to the second or third layers

Input feature at			Dropout	AP/BEP/%
Layer 1	Layer 2	Layer3		
$MFCC_{39}$	wav2vec	—	0	82.05/78.59
$MFCC_{39}$	wav2vec	—	0.1	82.82/79.03
$MFCC_{39}$	wav2vec	—	0.2	82.37/78.83
$MFCC_{39}$	—	wav2vec	0	78.99/75.49
$MFCC_{39}$	—	wav2vec	0.1	77.23/74.43
$MFCC_{39}$	—	wav2vec	0.2	78.41/75.17
$MFCC_{512}$	wav2vec	—	0	82.08/78.61
$MFCC_{512}$	wav2vec	—	0.1	82.74/78.96
$MFCC_{512}$	wav2vec	—	**0.2**	**83.32/79.23**
$MFCC_{512}$	—	wav2vec	0	74.54/72.04
$MFCC_{512}$	—	wav2vec	0.1	73.94/71.15
$MFCC_{512}$	—	wav2vec	0.2	76.16/73.37

结果表明,只要在网络中融合了wav2vec特征,AP 和 BEP 就会比仅在第一层使用MFCC特征有所提升,并且将wav2vec的特征输入到第二层的性能大大优于输入到第三层的性能。

表3还比较了直接使用 $MFCC_{39}$ 和 $MFCC_{512}$ 的效果,可以看出,$MFCC_{512}$ 的性能更好。在后续的实验中,仅尝试将特征输入到前两层中,使用MFCC时,仅尝试 $MFCC_{512}$。

表4列出了仅在第一层中输入wav2vec而在第二层中输入 $MFCC_{512}$ 或wav2vec的结果。实验结果显示,相比只在第一层输入wav2vec特征,额外增加 $MFCC_{512}$ 特征输入,反而会导致网络性能的下降。表2中将wav2vec特征和 $MFCC_{39}$ 拼接输入第一层,在神经网络失活概率为0时网络性能也比只在第一层输入wav2vec特征更低。这表明在训练嵌入网络的过程中,不正确的融合方法会降低网络的性能。

表4 第一层输入wav2vec,第二层中输入MFCC或wav2vec的性能

Table 4 Performance of input wav2vec in the first layer and input MFCC or wav2vec in the second layer

Input Feature at		Dropout	AP/BEP/%
Layer 1	Layer 2		
wav2vec	$MFCC_{512}$	0	79.90/76.22
wav2vec	$MFCC_{512}$	0.1	80.62/76.53
wav2vec	$MFCC_{512}$	0.2	79.27/75.65
wav2vec	wav2vec	0	81.25/77.67
wav2vec	wav2vec	0.1	80.88/76.59
wav2vec	wav2vec	**0.2**	**81.80/78.01**

表5列出将 $MFCC_{512}$ 和wav2vec特征进行拼接输入到第一层,并在第二层输入 $MFCC_{512}$ 或wav2vec或二者的拼接结果。结果表明,在第二层输入MFCC特征将导致网络性能下降,而在第二层输入wav2vec特征可以使网络性能得到提升。同时在第一层和第二层都输入 $MFCC_{512}$ 和wav2vec的方法获得了最佳性能。取得最佳性能的方法与仅在第一层输入特征的方法相比,AP 最多增加了3.7%,BEP 最多增加了1.3%,与仅使用wav2vec特征的方法相比,AP 增加了4.35%,BEP 增加了1.95%。

表5 第一层输入融合特征,在第二层输入不同特征的性能

Table 5 Performance of input the fusion features in the first layer and input different features in the second layer

Input Feature at		Dropout	AP/BEP/%
Layer 1	Layer 2		
wav2vec+$MFCC_{512}$	$MFCC_{512}$	0	81.84/77.98
wav2vec+$MFCC_{512}$	$MFCC_{512}$	0.1	82.59/78.13
wav2vec+$MFCC_{512}$	$MFCC_{512}$	0.2	81.81/77.95
wav2vec+$MFCC_{512}$	$MFCC_{512}$	0.3	81.59/77.96
wav2vec+$MFCC_{512}$	wav2vec	0	82.90/78.21
wav2vec+$MFCC_{512}$	wav2vec	0.1	82.27/78.05
wav2vec+$MFCC_{512}$	wav2vec	0.2	83.59/78.35
wav2vec+$MFCC_{512}$	wav2vec	0.3	83.67/78.23
wav2vec+$MFCC_{512}$	wav2vec+$MFCC_{512}$	0	85.33/79.51
wav2vec+$MFCC_{512}$	wav2vec+$MFCC_{512}$	0.1	84.99/79.53
wav2vec+$MFCC_{512}$	wav2vec+$MFCC_{512}$	**0.2**	**86.40/80.48**
wav2vec+$MFCC_{512}$	wav2vec+$MFCC_{512}$	0.3	85.62/79.76

表6列出了本文最佳方法与基准方法的性能和耗时的比较。可以看出,文献[13]中的DTW方法无论在性能还是速度方面,相比神经网络方法都没有优势。与文献[28]中仅使

用 MFCC 特征的方法相比，本文最佳方法的 *AP* 增加了 17.0%，BEP 增加了 13.2%，而计算消耗的时间只增加了 25%。

表 6 本文最佳方法与基准方法的性能和耗时的比较

Table 6 Performance and run times comparison of the best method in this article and benchmark methods

Method	*AP*/*BEP*/%	run times/s
DTW Method in [13]	64.51/52.37	35437.1
Neural Net Method in [28]	73.82/71.12	193.5
Our Best Method	**86.40/80.48**	240.3

图 3 显示了在训练期间 MFCC 基线系统和最佳的融合系统在开发集上作为嵌入函数进行样例关键词识别的 *AP*。可以观察到，整个训练过程中最佳融合系统的性能的始终领先基线系统的性能 10%以上。

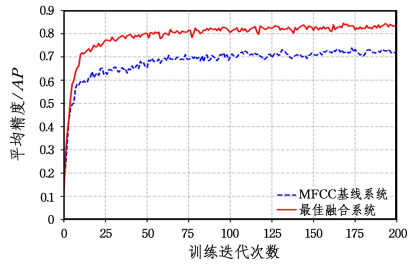


图 3 孤立词识别任务上两个系统的 *AP* 随迭代次数的变化

Fig. 3 Progression of *AP* on acoustic word discrimination of two systems

4.7 结论

本文使用 wav2vec 预训练特征来改进使用三元组损失的 BLSTM 网络作为嵌入函数的 NAWEs 的 QbE 关键词系统。作为一种无监督的预训练系统，wav2vec 特征可以在语音资源量较小时优化目标任务。本文通过实验选择了与 MFCC 基线相同且性价比更高的 3 层网络来进行 wav2vec 实验。在这种情况下，*AP* 相比 MFCC 基线增加了 11.1%，*BEP* 提升了 10.0%。

本文还融合了 wav2vec 特征和 MFCC 特征。实验发现，与只使用单个特征输入的方法相比，错误的融合方法将导致网络性能下降。使用全连接层将 MFCC 特征映射到与 wav2vec 特征相同的 512 维后融合性能更佳。将 wav2vec 特征和 512 维的 MFCC 特征拼接并输入到第一层中的系统，与仅使用 wav2vec 特征并输入第一层的系统相比，*AP* 提高了 1.4%。实验中得出的最佳融合系统是将 wav2vec 与使用全连接层映射到 512 维的 MFCC 拼接后同时输入第一层和第二层。最佳融合系统的 *AP* 比仅使用 wav2vec 特征的系统提高了 5.3%，比 MFCC 基线系统提高了 17.0%，而 *BEP* 则分别提升了 2.5%和 13.2%。

结束语 样例关键词识别一直是低资源情况下语音关键词识别的常用方法，基于动态时间规正的样例关键词方法，可以在零资源的情况下进行关键词识别。随着对基于神经网络的样例关键词方法的探索，数据量逐渐成为样例关键词系统训练神经网络的一大难题。本文采用预训练和特征融合的方式，改进了基于三元组损失和双向长短时记忆网络的样例关键词识别系统。通过实验得出了最佳的网络层数、神经元随机失活概率和特征融合方式，降低了基于神经网络的样例关键词系统对数据量的依赖。

在接下来的工作中，我们将进一步研究针对语种的预训练和跨域的预训练对样例关键词的影响，以更合理地在特定情况下提升模型性能，增加模型的实际应用价值。

参 考 文 献

[1] ITAKURA F. Minimum prediction residual principle applied to speech recognition [J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1975, 23(1): 67-72.

[2] SETTLE S, LEVIN K, KAMPERH, et al. Query-by-example search with discriminative neural acoustic word embeddings [C] // Proc. Interspeech. Stockholm, Sweden, 2017: 2874-2878.

[3] SHAH N, SREERAJ R, MADHAVI M C, et al. Query-By-Example Spoken Term Detection Using Generative Adversarial Network [C] // Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). IEEE, 2020: 644-648.

[4] HAZEN T J, SHEN W, WHITE C. Query-by-example spoken term detection using phonetic posteriorgram templates [C] // IEEE Workshop on Automatic Speech Recognition & Understanding. Merano, Italy, 2009: 421-426.

[5] ZHANG Y D, GLASS J R. Unsupervised spoken keyword spotting via segmental dtw on gaussian posteriorgrams [C] // IEEE Workshop on Automatic Speech Recognition & Understanding. Merano, Italy, 2009: 398-403.

[6] MA M, WU H, WANG X, et al. Acoustic word embedding system for code-switching query-by-example spoken term detection [C] // 12th International Symposium on Chinese Spoken Language Processing (ISCSLP). IEEE, 2021.

[7] CHEN H J, LEUNG C C, XIE L, et al. Unsupervised bottleneck features for low-resource query-by-example spoken term detection [C] // Proc. Interspeech. San Francisco, USA, 2016: 923-927.

[8] YUAN Y G, LEUNG C C, XIE L, et al. Pairwise learning using multi-lingual bottleneck features for lowresource query-by-example spoken term detection [C] // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). New Orleans, USA, 2017: 5645-5649.

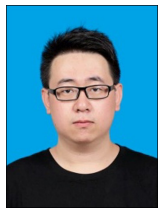
[9] RAM D, MICULICICH L, BOURLARD H. Multilingual bottleneck features for query by-example spoken term detection [C] // IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). Sentosa, Singapore, 2019: 621-628.

[10] RAM D, MICULICICH L, BOURLARD H. Neural network based end-to-end query by example spoken term detection [J]. IEEE/ACM Transactions on Audio, Speech. and Language Processing, 2020, 28: 1416-1427.

[11] LEVIN K, JANSEN A, VAN DURME B. Segmental acoustic indexing for zero resource keyword search [C] // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Brisbane, Australia, 2015: 5828-5832.

[12] CHUNG Y A, WU C C, SHEN C H, et al. Audio word2vec: Unsupervised learning of audio segment representations using se-

quence-to-sequence autoencoder[C]// Proc. Interspeech. San Francisco, USA, 2016: 765-769.
[13] MÜLLER M. Dynamic time warping [M]// Information Retrieval for Music and Motion. Berlin: Springer, 2007: 69-84.
[14] DHANANJAY R, AFSANEH A, HERV B. I Sparse subspace modeling for query by example spoken term detection[J]. IEEE/ACM Trans. Audio, Speech, Lang. Process., 2018, 26(6): 1130-1143.
[15] ZHAN J, HE Q, SU J, et al. A Stage Match for Query-by-Example Spoken Term Detection Based On Structure Information of Query[C]// IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2021). IEEE, 2021: 6833-6837.
[16] HE W J, WANG W R, LIVESCU K. Multi-view recurrent neural acoustic word embeddings[C]// Proc. ICLR. Toulon, France, 2017.
[17] JUNG M, LIM H, GOO J, et al. Additional shared decoder on siamese multi-view encoders for learning acoustic word embeddings[C]// IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). Sentosa, Singapore, 2019: 629-636.
[18] AUDHKHASI K, ROSENBERG A, SETHY A, et al. End-to-end asr-free keyword search from speech[J]. IEEE Journal of Selected Topics in Signal Processing, 2017, 11(8): 1351-1359.
[19] KAMPER H, LIVESCU K, GOLDWATER S. An embedded segmental k-means model for unsupervised segmentation and clustering of speech[C]// IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). Okinawa, Japan, 2017: 719-726.
[20] SCHNEIDER S, BAEVSKI A, COLLOBERT R, et al. wav2vec: Unsupervised pre-training for speech recognition[C]// Proc. Interspeech. Graz, Austria, 2019: 3465-3469.
[21] BAEVSKI A, AULI M, MOHAMED A. Effectiveness of self-supervised pre-training for asr[C]// International Conference on Acoustics, Speech and Signal Processing (ICASSP). Barcelona, Spain, 2020: 7694-7698.
[22] RIVIÈRE M, JOULIN A, MAZARÈ P E, et al. Unsupervised pretraining transfers well across languages[C]// International Conference on Acoustics, Speech and Signal Processing (ICASSP). Virtual Barcelona, Spain, 2020: 7414-7418.
[23] HOFFER E, AILON N. Deep metric learning using triplet network[C]// International Workshop on Similarity-based Pattern Recognition. Cham: Springer, 2015: 84-92.
[24] GODFREY J J, HOLLIMAN E C, MCDANIE L J. SWITCHBOARD: telephone speech corpus for research and development[C]// IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). San Francisco, USA, 1992: 517-520.
[25] POVEY D, GHOSHAL A. The Kaldi Speech Recognition Toolkit[C]// IEEE Workshop on Automatic Speech Recognition and Understanding(ASRU). Big Island, USA, 2011: 1-14.
[26] ABADI M, AGARWAL A, BARHAM P, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems[EB/OL]. (2016-3-16)[2021-08-31]. https://arxiv.org/abs/1603.04467.
[27] PANAYOTOV V, CHEN G, POVEY D, et al. Librispeech: an asr corpus based on public domain audio books[C]// IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Brisbane, Austrlia, 2015: 5206-5210.
[28] SETTLE S, LIVESCU K. Discriminative acoustic word embeddings: Tecurrent neural network-based approaches[C]// 2016 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2016: 503-510.



LI Zhao-qi, born in 1995, Ph. D. His main research interests include query by example spoken term detection and speech recognition.

LI Ta, born in 1982, Ph. D, professor. His main research interests include large vocabulary continuous speech recognition, keyword search, speaker recognition, pronunciation evaluation, emotion recognition, speech classification and analysis, and human-computer speech interaction technology.

(责任编辑:李亚辉)