CSLT TECHNICAL REPORT-20190417 [Friday 10th May, 2019]

PHONETIC-ATTENTION SCORING FOR DEEP SPEAKER VERIFICATION

Jiawei Yu[1], Lantian Li[1] and Dong Wang[1*]

*Correspondence: wang-dong99@mails.tsinghua.edu.cn
[1]Center for Speech and Language Technology, Research Institute of Information Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China
Full list of author information is available at the end of the article

摘要

这个工作是蓝天学长工作的延续，从之前的工作里我们了解到，不同的 d-vector 对对于区分说话人的重要性是不同的，那么到底哪些是更重要的呢？我们认为当说话人说的内容相同时，这些 d-vector 的距离更有代表性，所以我们选择使用 phonetic 信息来描述到底哪些 d-vector 对的距离更有代表性。但在计算两帧 phonetic 的相似度时我们的计算是固定的，不会根据我们最后的训练目标去动态调整每一对 d-vector 对于说话人区分的重要程度，所以我们选择，在原有实验的基础上，加入一个神经网络，使其根据训练目标动态调整不同 d-vector 对在区分说话人时的权重。

Keywords: speaker recognition; attention

1 Introduction

近来的研究表明通过神经网络产生的帧层级的话者特征向量可以用于区分说话人。具体的方法就是将每一帧的说话人向量加和取平均，得到句子层级的说话人向量，这一向量通常称为 d-vector。但是这种简单的加和取平均的方式无法突出一些区分性的说话人信息（eg:）。因此，由于受到机器翻译领域里注意力机制研究的启发，我们引入注意力机制到说话人识别任务中。具体来说（通常第一个会想到的是，在 n 帧说话人特征里分别给这些帧赋予不同的权重），我们通过计算 phonetic 帧之间的相似度（ASR-DNN 的 posterior，BNF），来作为 d-vector 帧之间的权重，以此实现不同 d-vector 帧之间的对齐关系，从而突出那些 phonetic 信息相同的 d-vcetor（即，让这样的 d-vector 之间的距离权重更大），将文本无关的说话人识别任务转化为文本相关的说话人识别任务，提升说话人识别系统的性能（通常在说话人识别这个任务上，文本相关的系统性能是好于文本无关的系统性能）。

在 [1] 中，两帧 d-vector 之间距离的权重是 asr 系统得到的 phonetic 帧之间的 KL 距离，这个距离是直接根据 KL 距离公式计算得到的（即：手动计算）。这种方法没有办法自动“注意”哪一对 d-vector 对于话者区分更重要，只是根据 phonetic 帧之间的距离来决定权重大小。但实际区分说话人时，我们希望可以

通过神经网络自动学习、调整每一对 d-vector 距离的权重，做出一个对系统区分说话人最优的决定。

- 我们直觉上想，不同说话人帧的重要程度或者说区分能力是不同的，但是否有研究论证这一假设？或者这不需要研究去论证这一想法？

2 相关工作

关于 attention 在说话人识别领域的应用，有以下一些工作。google 的 attention SRE 系统 [2] 如图 1所示。另一篇 attention 在说话人识别领域里的应用来自 microsoft[3]，系统流程如图 2所示。

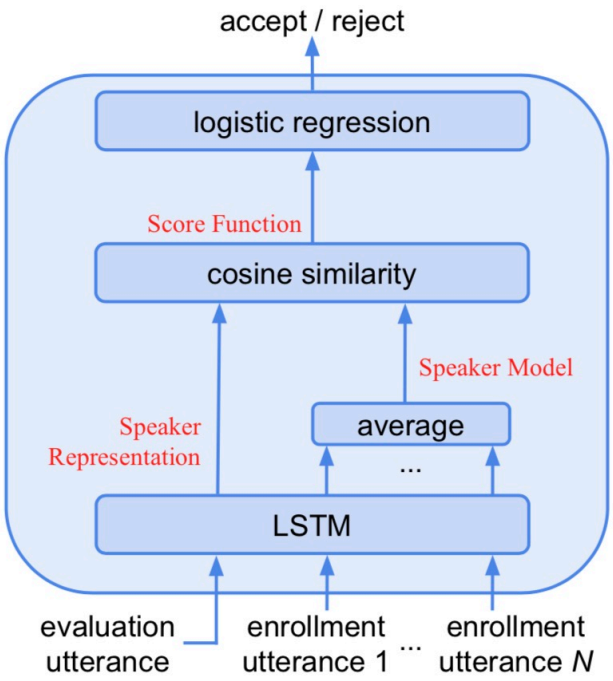

Fig. 1: Our baseline end-to-end training architecture as introduced in [7].

图 1: end-to-end attention SRE system.

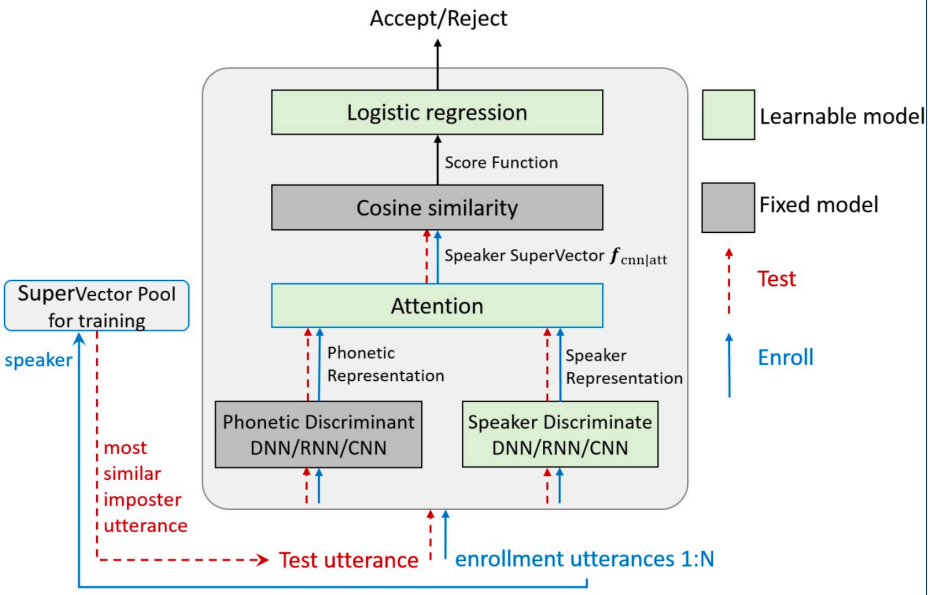

图 2: Block diagram of phonetic-attention SRE system.

3 实验描述

基于 phonetic-attention 的手动计算说话人识别系统流程如图 3所示；自动的 phonetic-attention 说话人识别系统说如图 4所示。

我们的 attention SRE 系统与上述模型的区别主要是：上述模型，attention 主要应用于将帧层级的说话人特征转化为句子层级的说话人特征；我们采用成对训练的方式，两个句子通过 attention 机制直接得到一个相似度的分数。

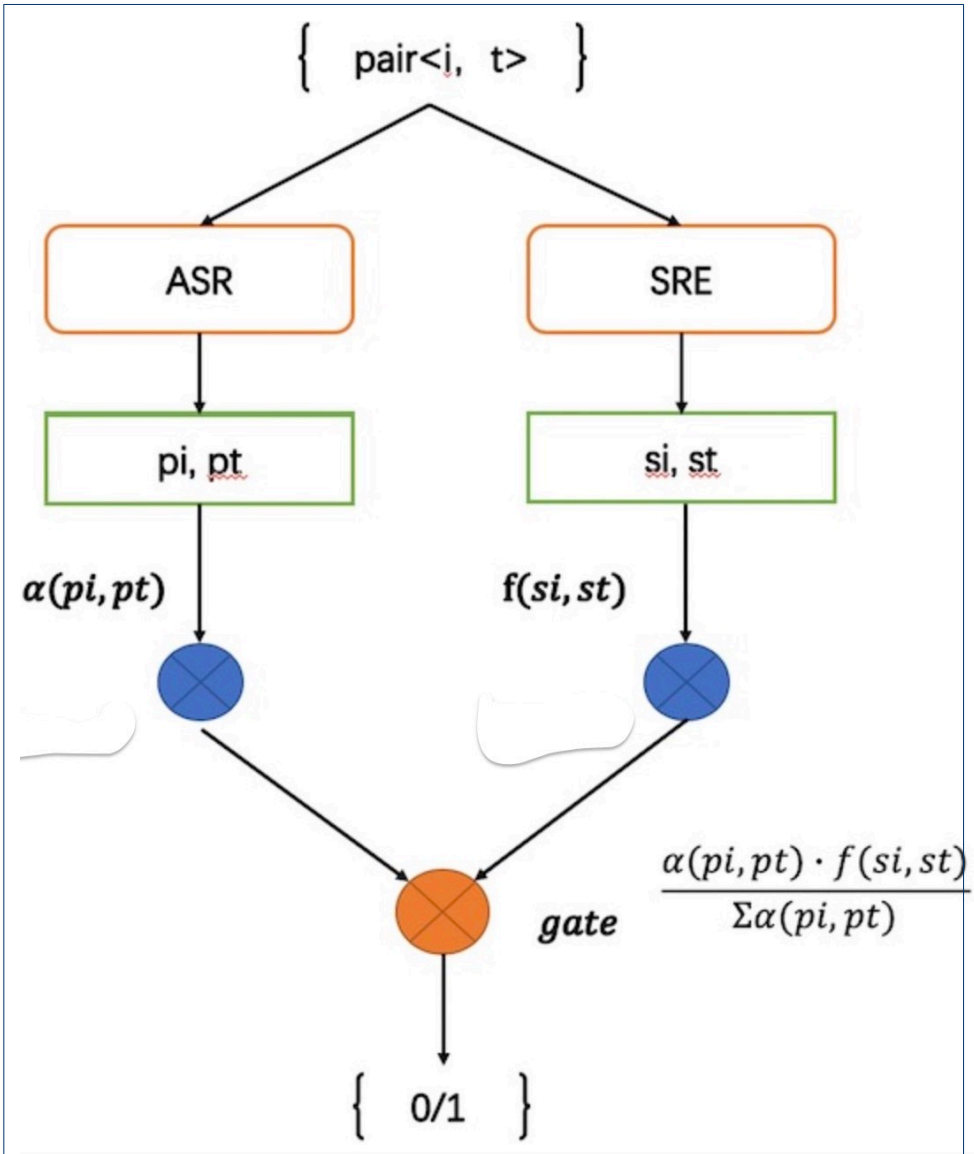

图 3: Block diagram of fixed phonetic-attention SRE system.

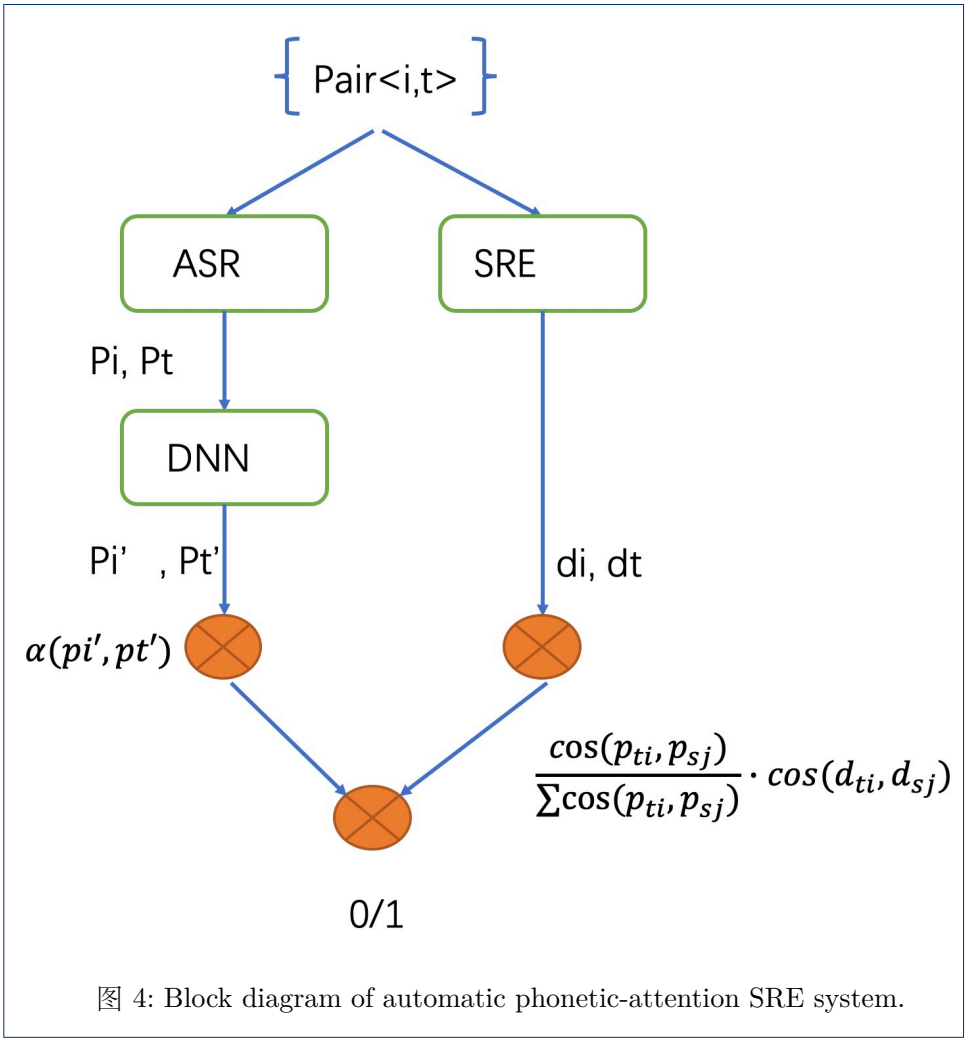

图 4: Block diagram of automatic phonetic-attention SRE system.

3.1 数据准备

使用 thch30 分别提取帧层级的 phonetic 特征（100 维 bnf）和 400 维的 d-vector。

thch30 训练集中一共有 10000 句话，分别来自 50 个人。这里我随机选出 50000 对句子的组合，这 50000 对组合中正例（来自于同一个说话人）和负例（不同的说话人）各占一半。这相当于在构造正例和负例的数据时每个人分别取 500 句话。格式如下：

utt1 utt2 1

utt3 utt4 0

……

对 thch30 在帧层级的 dvector 上画出了 t-SNE 图，图 5和图 6分别显示了 50 个人和 10 个人情况下的帧分布情况。从这两张图中我们不难发现这些 dvector 帧分的还是比较开的。验证了 thch30 这一数据集的 dvector 分布后，我们发现在这一数据集上可以继续开展 attention 的实验。

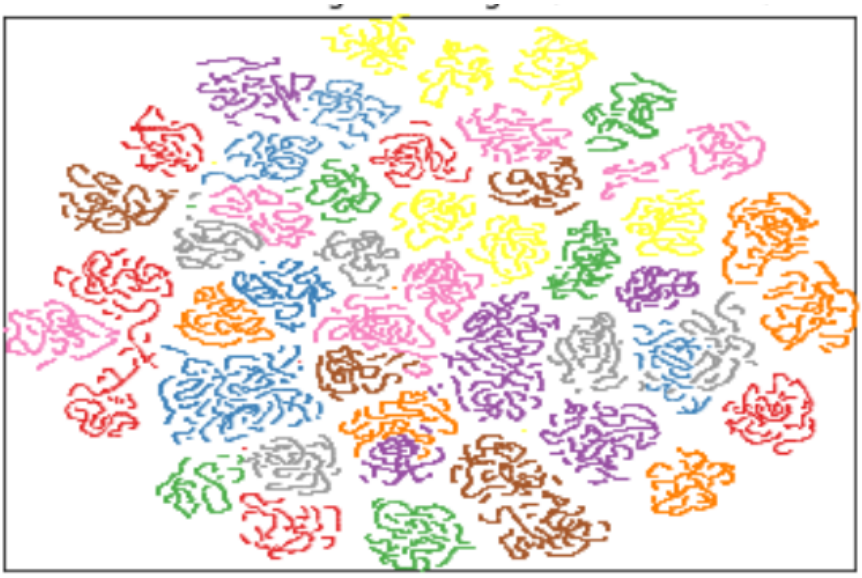

图 5: 50 人的 t-SNE

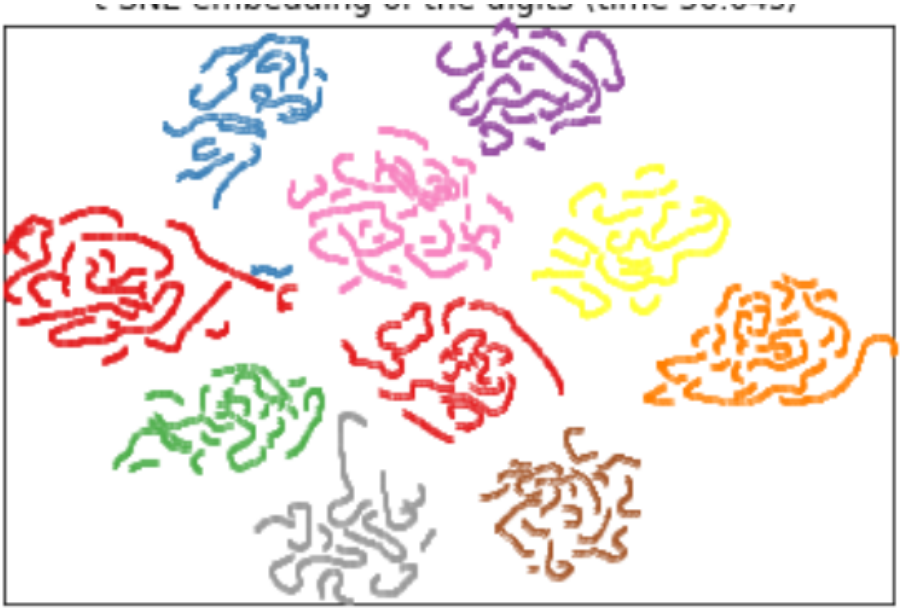

图 6: 10 人的 t-SNE

3.2 模型训练

phonetic-attention 的 loss function 如公式 1 所示：

$$L_T(s_j\ , t_k) = \delta(j,k)\sigma(similarity) + \big(1-\delta(j,k)\big)\big(1-\sigma(similarity)\big). \quad (1)$$

具体的实现代码如图 7所示。当 $j == k$ 时，$\delta(j,k) == 1$，即当两句话是同一个人说的时候，我们 minimize loss，相当于让公式（2）（公式 2 的中的 d_{ti}, d_{sj} 分别表示 target 端和 source 端的一帧 dvector；p_{ti}, p_{sj} 则分别表示一帧 phonetic 帧。）整体增大，而 $cos(d_{ti}, d_{sj})$ 是定值，所以如果想让公式（2）整体最大，则需让 dvector 帧对的余弦距离矩阵，如表格 1中的大的值（如 0.93）这样的值权重更大，而小的值（如：0.74）权重应小一些，这样在将乘过权重的矩阵中的所有值加和后才能能保证公式（2）整体最大。如果是不同的人，则上述过程应完全相反，即当 dvector 的距离值大的时候，权重应该赋予一个小值，距离小的时候，权重赋予一个大值。

$$\frac{cos(p_{ti}, p_{sj})}{\sum_{j=1}^{n} cos(p_{ti}, p_{sj})} \cdot cos(d_{ti}, d_{sj}). \quad (2)$$

这样看来整个过程，便相当于是让神经网络学到某种模式，根据如表 1的这样的一个 d-vector 距离矩阵这样的分布情况，来鉴别出这样的 dvector 矩阵是表示同一个人，还是不同的人（或者说发现这样的数据后，应该如何赋予权重），但这样的数据给定后真的能学到权重吗？或者说能区分出不同的人吗？

表 1: dvector 帧对的余弦距离矩阵

Frame	t1	t2	t3	...
s1	0.93	0.85	0.88	...
s2	0.75	0.92	0.74	...
s3	0.85	0.92	0.83	...
...	...	...	...	...

```
def f1():
    loss = 0
    for idx in range(batch_size):
        loss += tf.sigmoid(cos_sim(idx))
    return -tf.log(loss / batch_size)

def f2():
    loss = 0
    for idx in range(batch_size):
        loss += (1 - tf.sigmoid(cos_sim(idx)))
    return -tf.log(loss / batch_size)

return tf.cond(tf.equal(labels, 1), f1, f2)
```

图 7: loss function 代码

3.3 模型测试

由于我们模型测试得到的分数都是成对的句子之间的分数，所以测试时得到的是句子之间的分数，需要先将句子与句子的分数转化为句子和人之间的分数 (将相同的行的分数加和取平均)。句子与句子的分数：

utt1 utt2 1

utt3 utt4 0

……

句子和人的分数：

utt1 spk1 score1

utt1 spk2 score2

……

3.4 实验结果

在本实验中，我们采用 DNN 来学习 attention，DNN 包括两个隐藏层，第一个隐藏层有 300 个结点，第二个隐藏层有 100 个结点，输出层 50 个结点。phonetic 信息采用 BNF 特征，100 维；dvector400 维。实验结果如图 8所示。其中 num repeats 表示每一个话者所取的句子数。Frame500 pooling50 表示每句话截取 500 帧，每 50 帧 pooling 一次，Frame600 pooling100 表示每句话截取 600 帧，每 100 帧 pooling 一次，依此类推。Baseline EER 表示当不使用 phonetic 信息，仅仅使用 dvector 在 pair 方式下的 EER。图 8所示的实验结果中，attention 的 weight 均是有变化的，可以看到高山低谷。

Num_repeats	EER Frame500 _pooling100	EER Frame500 _pooling50	EER Frame600 _pooling100	EER Baseline
100	7.8（loss基本没有下降 0.75/0.62）	7.1（loss基本没有变化 0.81/0.54）	11.32	5.54
1000	5.42（loss几乎没有下降 0.73/0.63）	5.31（loss变化不大）	8.46	4.97
2000	4.94（loss几乎没有下降）	4.82（loss几乎没有下降）	7.03	4.81
3000	4.49（loss略有下降大概从 0.53/0.82->0.54/0.80 ）	4.53（同左）	6.92	4.75
6000	4.45（loss情况和上面相同）	4.47（同左）	6.94	4.89

图 8: 结果展示

表 2为传统的基于 ivector、dvector 的说话人识别结果

表 2: ivector, dvector 说话人识别结果

approach	ivector	dvector
cosine	0.67	3.61
lda	0.07	0.94
plda	0.07	2.21

从以上结果中我们可以发现：

- 随着训练数据的增多，基于 phonetic attention 的说话人识别方法，其识别的准确率会提升，对比 num repeats 等于 100 和等于 6000 的时候，EER 从 7.8 降到 4.45，取得了 phonetic attention 方法的最好表现（EER Frame500 pooling100）。但同时我们可以看到 num repeats 从 2000 到 6000 的时候，EER 的变化就已经不明显了，再此，我认为在这种条件下对于 thch30 的训练已经充分了。

- 尽管将 pooling 的步长缩小为 50 帧（EER Frame500 pooling50），这时 attention 的精细度有所增加，但随着训练数据的增多，这种性能提升的表现并不明显。
- 当每句话选取更多的语音时长的时候（EER Frame600 pooling50），系统性能反而下降，我猜想原因是，thch30 数据每句话的平均时长在 5 到 10 秒不等，当每句话选择 6 秒的时候，我的处理办法，是不够 6 秒钟的就补零，所以相比于每句话选择 5 秒钟进行训练，每句话选 6 秒会有更多的句子进行补零操作，这一行为影响了系统的性能。
- 最后我们发现，本实验中的 phonetic attention 方法性能无法和传统的 ivector、dvector 方法相比，性能表现差距较大。
- 实验中我发现，loss 在各种实验参数下都下降不明显，具体表现为：在 num repeats 值较小时，loss 不会下降，当 num repeats 大于 3000 之后 loss 略微下降（大概从 0.75/0.54->0.73/0.54）。
-

4 问题

- 通过 3.2 节的模型训练过程的分析，以及 3.4 的实验结果，我不确定这样的模型结构设计是否能学到 attention weight。
- 本实验中的 loss function 是否鼓励 phonetic 信息对齐？也就是说是否按着我们最初的设想，将文本无关的说话人识别问题，转换为文本相关的说话人识别。

Acknowledgement

Author details

[1]Center for Speech and Language Technology, Research Institute of Information Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China. [2]Center for Speech and Language Technologies, Division of Technical Innovation and Development, Tsinghua National Laboratory for Information Science and Technology, ROOM 1-303, BLDG FIT, 100084 Beijing, China. [3]Department of Computer Science and Technology, Tsinghua University, ROOM 1-303, BLDG FIT, 100084 Beijing, China.

参考文献

1. Lantian Li, Zhiyuan Tang, Ying Shi, and Dong Wang, "Phonetic-attention scoring for deep speaker features in speaker verification," *arXiv preprint arXiv:1811.03255*, 2018.
2. FA Chowdhury, Quan Wang, Ignacio Lopez Moreno, and Li Wan, "Attention-based models for text-dependent speaker verification," *arXiv preprint arXiv:1710.10470*, 2017.
3. Shi-Xiong Zhang, Zhuo Chen, Yong Zhao, Jinyu Li, and Yifan Gong, "End-to-end attention based text-dependent speaker verification," in *2016 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2016, pp. 171–178.