摘要

一个文本转语音的合成系统通常包含多阶段处理，例如文本分析前端，声学模型和声音合成模块。构建这些组件常常需要大量的领域专业知识，而且设计选择也可能很脆弱。在这篇论文里，我们提出了Tacotron，一种端到端的生成式文本转语音模型，可以直接从字符合成语音。在<文本,声音>配对数据集上，该模型可以完全从随机初始化从头开始训练。我们提出了几个可以使seq2seq框架在这个高难度任务上表现良好的关键技术。Tacotron 在美式英语测试里的平均主观意见评分达到了3.82分（总分是5分），在合成自然度方面优于已在生产中应用的参数模型。另外，由于Tacotron是在帧层面上生成语音，所以它大幅度快于样本级自回归方式的模型。

1.介绍

现代文本转语音（TTS）的流水线比较复杂(Taylor,2009)，举例来说，通常基于统计参数的TTS系统包含：一个用来抽出多种语言特征的文本分析前端，一个音长模型（语音持续时间模型），一个声学特征预测模型，和一个复杂的基于信号处理的声码器(Zen et al.,2009; Agiomyrgiannakis,2015)。这些组件都基于大量的领域专业知识因而其设计很艰难。并且，这些组件都是单独训练的，所以产生自每个组件的错误会有叠加效应。因此，现代TTS系统的设计复杂度导致在构建新系统时需要投入大量的工程努力。

正因如此，集成一个能在少量人工标注的<文本，语音>配对数据集上训练的端到端的TTS系统，会带来诸多优势。首先，这样一个系统减少了艰难特征工程的必要，而正是这些特征工程可能会导致启发式错误和脆弱的设计选择。其次，这样的系统允许基于各种属性来进行多样化的调节，比如不同说话人，不同语言，或者像语义这样的高层特征，这是因为调节不只是出现在特定几个组件中，而是在模型的最开始就发生了。类似的，拟合新数据也将变得更容易。最后，相比会出现错误叠加效应的多阶段模型，单一模型倾向于更鲁棒。这些优势意味着，一个端到端的模型能够允许我们在现实世界容易获取的大量的丰富生动的同时也很嘈杂的数据上执行训练。

TTS是一个大规模的逆问题：把一份高度压缩的文本源解压缩成语音。由于同一份文本可以对应到不同的发音或讲话风格，对一个端到端的模型来说，这是一个异常困难的学习任务：给定一个输入它必须处理在信号层面处理大量变种。而且，不像端到端的语音识别(Chan et al.,2016)或者机器翻译(Wu et al.,2016)，TTS的输出是连续的，并且输出序列通常比输入序列长很多。这些属性造成了预测错误的快速累积。在这篇论文中，我们提出了Tacotron，一个端到端的基于带注意力范式(Bahdanau et al.,2014)的序列到序列（seq2seq）(Sutskever et al.,2014)生成式TTS模型。该模型使用几个可以改善普通seq2seq模型能力的技术，输入字符直接输出原始声谱图。给定<文本，语音>配对数据，Tacotron可以完全从随机初始化从头开始训练。由于不需要音素层面的对齐，因此它可以很容易的使用大量带有转录文本的声学数据。使用一个简单的波形合成技术，Tacotron在美式英语评估数据集上得到了3.82的平均意见得分（MOS），在合成自然度方面优于已在生产中应用的参数模型。(语音样本展示参照：https://google.github.io/tacotron)

2. 相关工作

WaveNet(van den Oord et al.,2016)是一个强大的声音生成模型。它在TTS中表现良好，但是样本水平自回归的天性导致其速度慢。它还要求在既存TTS前端生成的语言特性上进行调节，因此不是端到端的，它只替换了声码器和声学模型部分。另外一个最近开发的神经模型是DeepVoice (Arik et al.,2017)，它分别用一个神经网络替换了典型TTS系统流水线中的每一个组件，然而它的每个组件都是单独训练的，要把系统改成端到端的方式不那么简单。

据我们所知，Wang et al. (2016)是最早使用带有注意力范式的seq2seq方法尝试端到端TTS系统的。但是，首先，它需要一个预先训练好的隐马尔可夫（HMM）对齐器来帮助seq2seq模型学习如何对齐。所以很难说seq2seq本身学到了多少对齐能力。其次，为了训练模型使用了几个技巧，作者指出这些技巧有损于合成韵律。第三，它预测了声码器参数作为中间特征表达，因此需要一个声码器。最后，该模型训练的输入是音素数据并且实验结果好像有点受限。

Char2Wav (Sotelo et al.,2017)是一个独立开发的可以在字符数据上训练的端到端模型。但是，Char2Wav在使用SampleRNN神经声码器(Mehri et al.,2016)前也预测了声码器参数，而Tacotron直接预测原始声谱图。另外，seq2seq和SampleRNN需要单独进行预训练，但我们的模型可以从头开始训练。最后，我们对普通seq2seq范式进行了几个关键变更，我们后面会展示，普通的seq2seq模型对字符输入不太奏效。

3. Tacotron模型架构

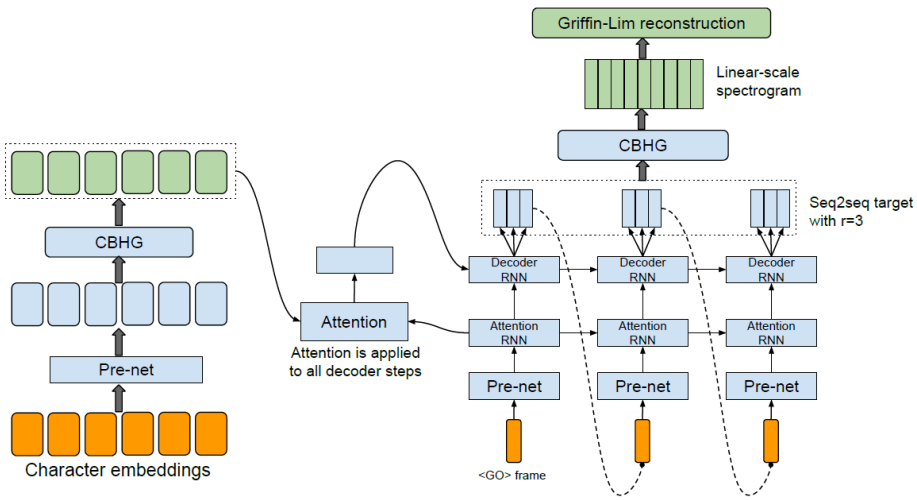


Figure 1: *Model architecture. The model takes characters as input and outputs the corresponding raw spectrogram, which is then fed to the Griffin-Lim reconstruction algorithm to synthesize speech.*

https://blog.csdn.net/lujian1989

Tacotron的骨干部分是一个有注意力机制的(Bahdanau et al.,2014; Vinyals et al.,2015)seq2seq模型。图1描绘了该模型架构，它包含一个编码器，一个基于注意力机制的解码器和一个后处理网络。从高层面上说，我们的模型把字符作为输入，产生的声谱帧数据随后被转换成波形。下面详细描述这些组件。

3.1 CBHG 模块

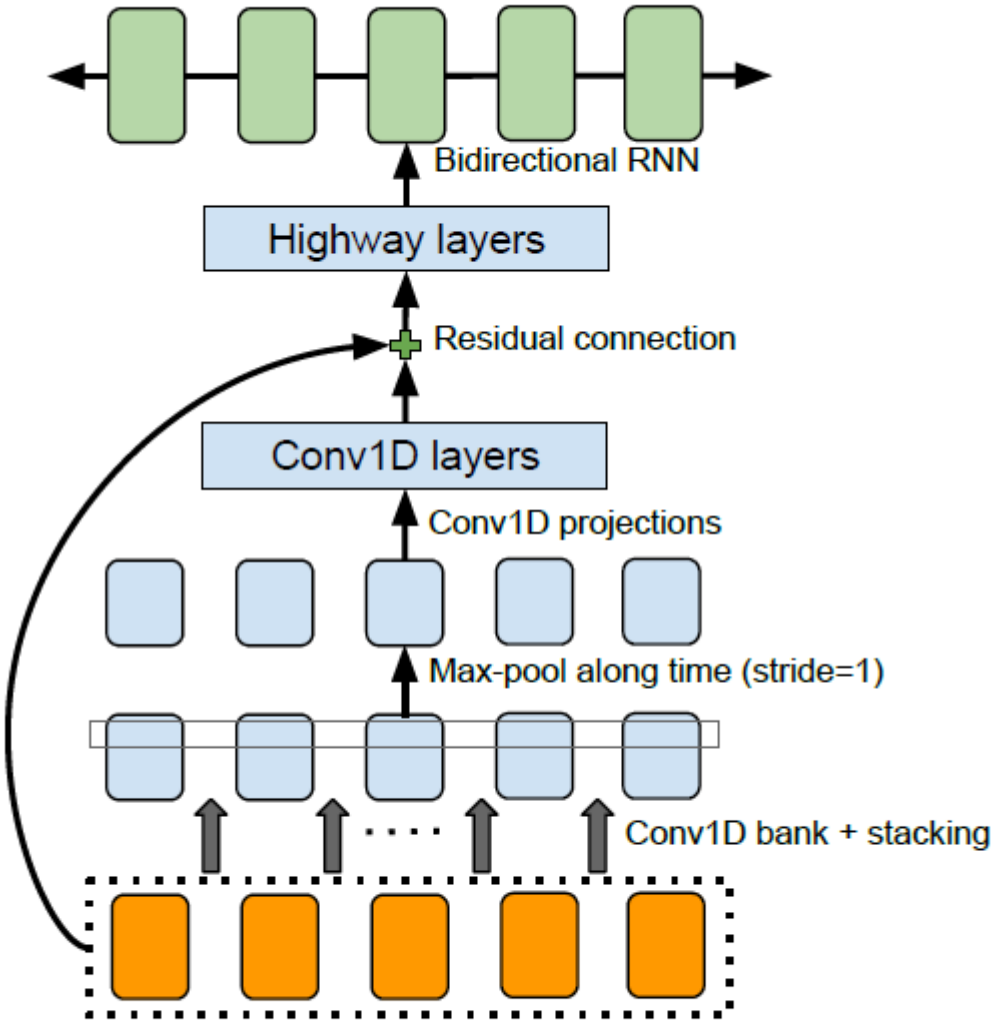


Figure 2: *The CBHG module adapted from [8].*

https://blog.csdn.net/lujian1989

我们首先描述称为CBHG的模块，如图2所示。CBHG包含一个一维卷积滤波器组，后跟一个高速公路网络(Srivastava et al.,2015)和一个双向门控循环单元（GRU）(Chung et al.,2014)循环神经网络（RNN）。CBHG是一个强大的模块以提取序列的特征表达。首先在输入序列上用K组一维卷积核进行卷积，这里第k组包含Ck个宽度是k（即k = 1,2,...,K）的卷积核。这些卷积核显式地对局部上下文信息进行建模（类似于unigrams, bigrams, 直到K-grams）。卷积输出结果被堆叠在一起然后再沿时间做最大池化以增加局部不变性，注意我们令步长=1以维持时间方向的原始分辨率。然后把得到的结果序列传给几个定长一维卷积，其输出结果通过残差连接(He et al.,2016)和原始输入序列相叠加。所有卷积层都使用了批标准化(Ioffe & Szegedy,2015)。卷积输出被送入一个多层高速公路网络来提取高层特征。最上层我们堆叠一个双向GRU RNN用来前后双向提取序列特征。CBHG是受机器翻译(Lee et al.,2016)论文的启发，我们与(Lee et al.,2016)的不同包括使用非因果卷积，批标准化，残差连接以及步长=1的最大池化处理。我们发现这些修改提高了模型的泛化能力。

3.2 编码器

编码器的目的，是提取文本的鲁棒序列表达。编码器的输入是字符序列，输入的每个字符都是个一个one-hot向量并被嵌入一个连续向量中。然后对每个字符向量施加一组非线性变换，统称为“pre-net”。在这次工作中，我们使用带dropout的瓶颈层（bottleneck layer）作为pre-net以帮助收敛并提高泛化能力。CBHG模块将prenet的输出变换成编码器的最终表达，并传给后续的注意力模块。我们发现基于CBHG的编码器不仅减少了过拟合，它还能比标准的多层RNN编码器产生更少的错音（参看链接网页上的合成语音样本）。

3.3 解码器

我们使用基于内容的tanh注意力解码器（参照Vinyals et al. (2015)），在这个解码器中，一个有状态的循环层在每个时间步骤上都产生一次注意点查询。我们把上下文向量和Attention RNN单元的输出拼接在一起，作为解码器RNN的输入。我们使用带有纵向残差连接的GRUs堆栈(Wu et al., 2016)作为解码器，我们发现残差连接加速了收敛。解码器的目标输出，是一个重要的设计选择。因为我们可以直接预测原始声谱图，这对于学习语音信号和原始文本对齐的目标（这是在这个任务上使用seq2seq的真正动机）是一个高度冗余的表示。因为这个冗余，我们为seq2seq解码和波形合成选择了一个不同的目标。语音合成作为一个可训练或者可确定的逆向过程，只要能够提供足够的语音可理解性和足够的韵律信息，seq2seq的目标输出就可以被大幅度压缩。尽管类似倒谱这样更小的带宽或者更简洁的目标输出也可行，但我们采用带宽为80的梅尔刻度声谱图作为解码器的目标输出。我们使用了一个后处理网络（接下来讨论）把seq2seq的目标输出转化为波形。

我们使用一个简单的全连接输出层来预测解码器的目标输出。我们发现一个重要的技巧是，每一步解码处理可以同时预测多个非重叠的输出帧，一次预测r帧使得全体解码步骤缩小了r倍，结果是减小了模型大小，训练时间和推断时间。更重要的，我们发现这个技巧会大幅度加快收敛速度，试验中注意力模块非常迅速（并且非常稳定）的学到了如何对齐。这可能是因为每个字符通常对应了多个语音帧而相邻的语音帧具有相关性。强制每次输出一帧使得模型对同一个输入字符进行多次重复关注，而同时输出多帧允许注意力在训练中更早向前移动。Zen et al. (2016)也是用了类似的技巧，但目的主要是用来加速推断。

解码器的第一步是在一个“全零帧”上开始调节，图1中标示了”<\GO> frame”。在推断时，解码器的第t步处理，预测结果的最后一帧被作为解码器第t+1步的输入。注意这里选择最后一帧输入到下一步处理中只是一种选择而已，也可以选择一组r帧的全部作为下一步的输入。在训练中，我们取每个第r帧输入给解码器。像编码器中的处理一样，输入帧传给一个pre-net。因为没有使用预排程采样（Bengio et al.,2015）（我们发现这样做会损害声音质量）那样的技术，所以pre-net中的dropout对模型泛化很关键，因为dropout为解决输出分布中的多形态问题提供了噪声源。

3.4 后处理网络和波形合成

上面也提到了，后处理网络的任务是，把seq2seq的输出转化成可以被合成为波形的目标表达。因为使用Griffin-Lim做合成器，后处理网络要学习的是如何预测在线性频率刻度上采样的频谱幅度。构建后处理网络的另外一个动机是它可以看到全体解码结果序列，对比普通seq2seq总是从左到右运行，它可以获得前后双向信息用以纠正单帧预测错误。在这次工作中，我们使用CBHG模块作为后处理网络，尽管一个更简单的架构可能也会工作的很好。后处理网络的概念是高度通用的，它可以用来预测不同的目标输出如声码器参数，也可以作为像WaveNet那样的神经声码器(van den Oord et al.,2016; Mehri et al.,2016; Arik et al.,2017)来直接合成波形样本。

我们使用Griffin-Lim算法(Griffin & Lim,1984)从预测出的声谱图合成波形。我们发现把预测频谱的振幅提高1.2倍再输入到Griffin-Lim可以减少人工痕迹，可能是归功于其谐波增强效果。我们观察到Griffin-Lim在50次迭代后收敛（实际上大约30次迭代好像就足够了），这个速度相当快。我们在Tensorflow中实现了Griffin-Lim算法，所以它也成为了整个模型的一部分。尽管Griffin-Lim是可导的（它没有训练参数），但我们没有在其上设计任何损失。我们强调选择Griffin-Lim是为了简单，尽管它已经生成了很好的结果，我们也在开发一个快速的高品质的可训练的声谱-波形转换器。

4. 模型细节

Table 1: *Hyper-parameters and network architectures. "conv-k-c-ReLU" denotes 1-D convolution with width k and c output channels with ReLU activation. FC stands for fully-connected.*

Spectral analysis	*pre-emphasis*: 0.97; *frame length*: 50 ms; *frame shift*: 12.5 ms; *window type*: Hann
Character embedding	256-D
Encoder CBHG	*Conv1D bank*: K=16, conv-k-128-ReLU *Max pooling*: stride=1, width=2 *Conv1D projections*: conv-3-128-ReLU $\rightarrow$ conv-3-128-Linear *Highway net*: 4 layers of FC-128-ReLU *Bidirectional GRU*: 128 cells
Encoder pre-net	FC-256-ReLU $\rightarrow$ Dropout(0.5) $\rightarrow$ FC-128-ReLU $\rightarrow$ Dropout(0.5)
Decoder pre-net	FC-256-ReLU $\rightarrow$ Dropout(0.5)$\rightarrow$ FC-128-ReLU $\rightarrow$ Dropout(0.5)
Decoder RNN	2-layer residual GRU (256 cells)
Attention RNN	1-layer GRU (256 cells)
Post-processing net CBHG	*Conv1D bank*: K=8, conv-k-128-ReLU *Max pooling*: stride=1, width=2 *Conv1D projections*: conv-3-256-ReLU $\rightarrow$ conv-3-80-Linear *Highway net*: 4 layers of FC-128-ReLU *Bidirectional GRU*: 128 cells
Reduction factor (r)	2

https://blog.csdn.net/lujian1989

表1列出了超参数和模型架构。我们使用对数幅度谱，汉明窗，帧长50毫秒，帧移12.5毫秒，2048点傅里叶变换，我们还发现预加重(0.97)也有用。对所有的试验我们使用24k赫兹采样率。

在论文的MOS评分中使用r=2（解码器输出层的缩小因子），更大的r也运行的很好（例如r=5）。我们使用Adam优化器(Kingma & Ba,2015)，学习率从0.001开始500K步后降低到0.0005，1M步后降到0.0003，2M步后降到0.0001。我们采用简单的L1损失同时囊括seq2seq解码器（梅尔刻度声谱图）和后处理网络（线性刻度声谱图）。两部分损失的权重相同。

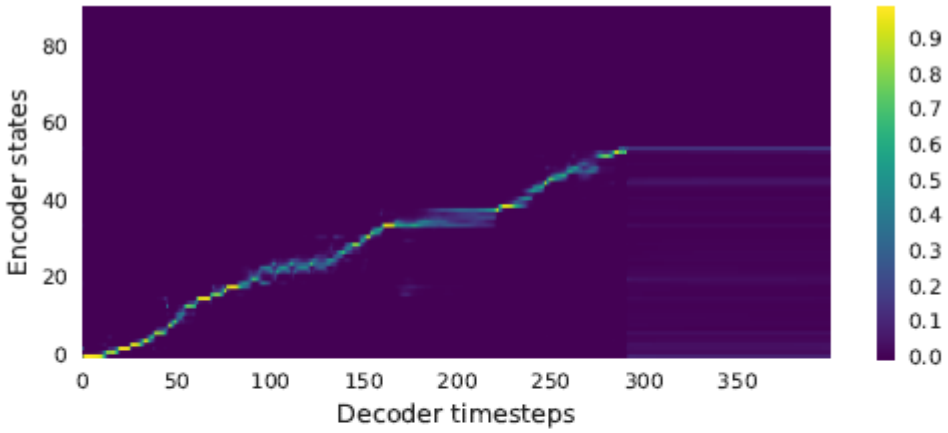
训练的批大小设定为32，所有的序列都**补齐到最大长度。**在训练中使用损失屏蔽是一个常用做法，在补零的数据帧上屏蔽损失。然而我们发现这样训练出的模型不知道何时停止输出，导致靠近结尾会有重复的声音。解决这个问题的一个简单技巧是对补零的数据帧也进行波形重建。

5. 实验

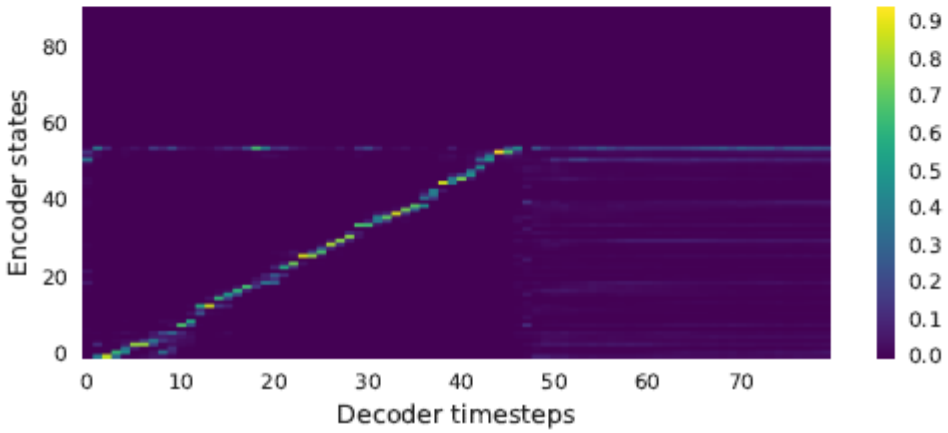
我们在一个内部北美英语数据集上训练Tacotron，这个数据集包含大约24.6小时的语音数据，由一个专业女性播讲。所有单词都进行了标准化处理，如“16”被转换成“sixteen”。

5.1 剥离分析

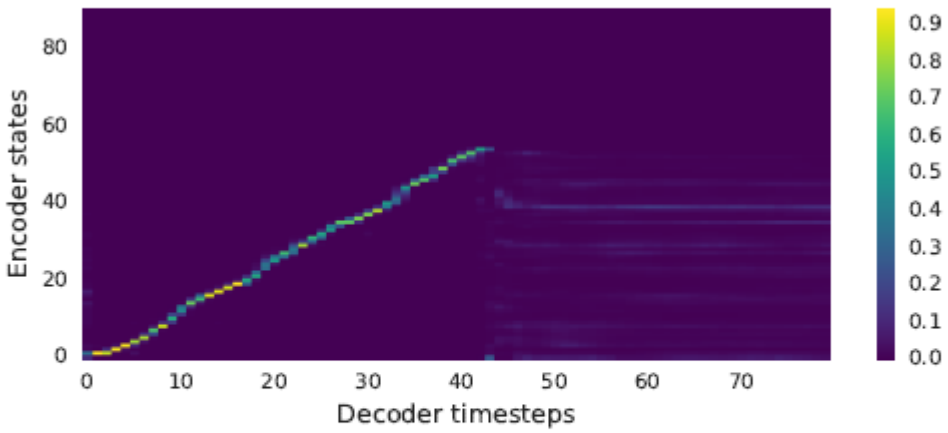
为理解模型的关键组件我们实施了几个剥离研究。对于生成式模型，基于客观度量的模型比较是很困难的，这些客观度量不能与感知很好地匹配(Theis et al.,2015)。相反的我们主要依赖视觉比较。我们强烈推荐读者听一下我们提供的语音样本。



(a) Vanilla seq2seq + scheduled sampling



(b) GRU encoder



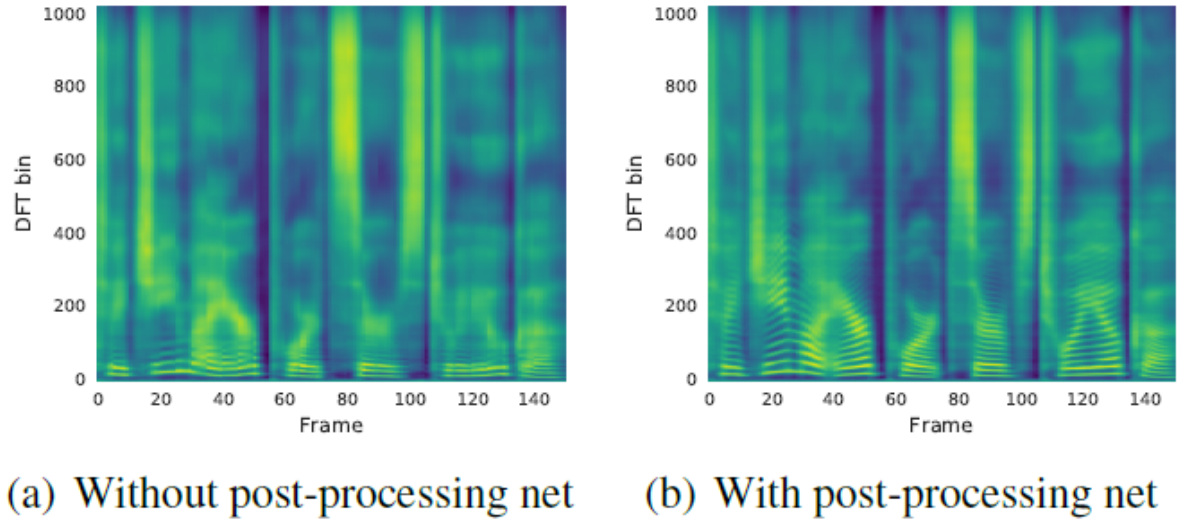
(c) Tacotron (proposed)

Figure 3: *Attention alignments on a test phrase. The decoder length is Tacotron is shorter due to the use of the output reduction factor r=5.*

https://blog.csdn.net/lujian1989

首先，与普通的seq2seq模型比较。编码器和解码器都是用2层残差RNN，每层包含256个GRU单元（我们也尝试了LSTM，结果类似），不使用pre-net和后处理网络，解码器直接预测线性对数幅度声谱图。我们发现，预排程采样（采样率0.5）对于这个模型学习对齐和泛化是必要的。我们在图3中展示了学到的注意力对齐，图3(a)揭示了普通的seq2seq学到的对齐能力很微弱，一个问题是其中有一段注意力卡住了，这导致了语音合成结果的可理解度很差，语音自然度和整体音长也都被摧毁了。相对的，我们的模型学到了清晰平滑的对齐，如图3(c)所示。

其次，我们比较了用一个2层残差GRU编码器替换CBHG后的模型，包括编码器的pre-net在内模型的其余部分不变。比较图3(b)和图3(c)可以看到，GRU编码器的对齐有些噪声。试听语音合成结果，我们发现这些对齐噪声会导致发音错误。CBHG编码器减少了过拟合并且在长的复杂短语上泛化能力很好。



(a) Without post-processing net (b) With post-processing net

Figure 4: *Predicted spectrograms with and without using the post-processing net.*

https://blog.csdn.net/lujian1989

图4(a)和图4(b)展示了使用后处理网络的好处。我们训练了一个除了不包含后处理网络其余部分都一样（解码RNN修改成预测线性声谱图）的模型，拥有了更多的上下文信息，后处理网络的预测结果包含更好的谐波（比如在100~400之间的高频谐波）和高频共振峰结构，这会减少合成的人工痕迹。

5.2 MOS测试

Table 2: *5-scale mean opinion score evaluation.*

	mean opinion score
Tacotron	3.82 ± 0.085
Parametric	3.69 ± 0.109
Concatenative	4.09 ± 0.119

https://blog.csdn.net/lujian1989

我们做了平均意见得分（MOS）测试，由测试者对合成语音的自然程度进行 5 分制的李克特量表法（Likert scale score）评分。MOS 的测试者均为参与众包的母语人群，共使用 100 个事先未展示的短语，每个短语获得 8 次评分。当计算MOS评分时，只有佩戴耳机时打出的评分被计算在内。我们对Tacotron与参数式（parametric）系统（基于LSTM（Zen et al.,2016））和拼接式（concatenative）系统（Gonzalvo et al.,2016）做了对比，后两者目前均已投入商业应用。测试结果如下表显示：Tacotron 的 MOS 分数为 3.82，优于参数系统。由于参照基准已经非常强大，以及由Griffin-Lim 引起的人工痕迹，这一新方法具有非常好的前景。

6. 讨论

我们提出了Tacotron，一个集成的端到端的生成式TTS模型，它以字符序列作为输入，输出对应的声谱图。后接一个简单的波形合成模块，模型在美式英语上的MOS得分达到了3.82，在自然度方面超越了已经投入生产的参数式系统。Tacotron是基于帧数据的，因此推断要大大快于样本水平的自回归方法。Tacotron不像之前的研究工作那样需要人工工程的语言特征或者像HMM对齐器这样复杂的组件，它可以从随机初始化开始从头进行训练，只是进行了简单的文本标准化处理，但是最近在文本标准化学习的进步(Sproat & Jaitly,2016)表明这一步处理未来也可以去掉。

我们的模型的很多方面还有待调查，很多早期的设计决定一直保持原样。输出层，注意力模块，损失函数，以及Griffin-Lim波形合成器的改善时机都已经成熟。例如，Griffin-Lim的输出听起来含有人工合成的痕迹已经广为人知，我们现在正在开发一个快速的高品质的基于神经网络的声谱图逆变换网络。