**语音合成项目立项报告**

## 一、 研究目的、意义及国内外概况

### 1.1 研究目的及意义

在科技高速发展，而聋哑人却未得到高度重视的当下，该项目由千博信息有限公司与湖南省计算媒体研究院发起，旨在关注聋哑人的生活，为他们的生活带去便利。

语音，在人类的发展过程中，起到了巨大的作用。语音是语言的外部形式，是最直接地记录人的思维活动的符号体系，也是人类赖以生存发展和从事各种社会活动最基本、最重要的交流方式之一。而让机器开口说话，则是人类千百年来的梦想。语音合成是人类不断探索、实现这一梦想的科学实践，也是受到这一梦想不断推动、不断提升的技术领域。语音合成是将文本转化为语音技术。语音合成技术的应用场景非常广泛，在语音助理里面的Apple Siri就用到了语音合成技术，语音合成是语音助理的重要组成部分；智能音响、地图导航、新闻播报、智能客服、呼叫中心等也都用到了语音合成技术。

近年来，随着人工智能概念的推广，语音交互成为了一个热点，智能助手、智能客服等应用层出不穷。语音交互中，主要有三个关键技术，语音识别、语音合成和语义理解，语音合成在其中的作用显而易见。语音合成技术，已经成功应用在很多领域，包括语音导航、信息播报等。对于语音合成的应用前景，中天智领有着自己的看法。因为中天智领既是智慧交互信息化系统整体解决方案服务商，同时也是语音合成整体解决方案提供商，所以对于语音合成的应用前景，也做过很多思考。目前语音合成的声音，从合成效果上，已经可以满足大多数用户的需求。语音识别技术的出现使得机器可以听懂人类的语言并且按照人类的口头命令行动，实现人机的语音交流。语音合成技术就是通过已经采集的数据样本，通过一系列的机器学习最终合成出需求的声音。这是一门交叉学科，正逐步成为信息技术中人机接口的关键技术，语音识别技术与语音合成技术的结合，将使人们能够甩掉键盘，通过语音命令进行操作。语音技术的应用已经成为一个具有竞争性的新兴高技术产业。

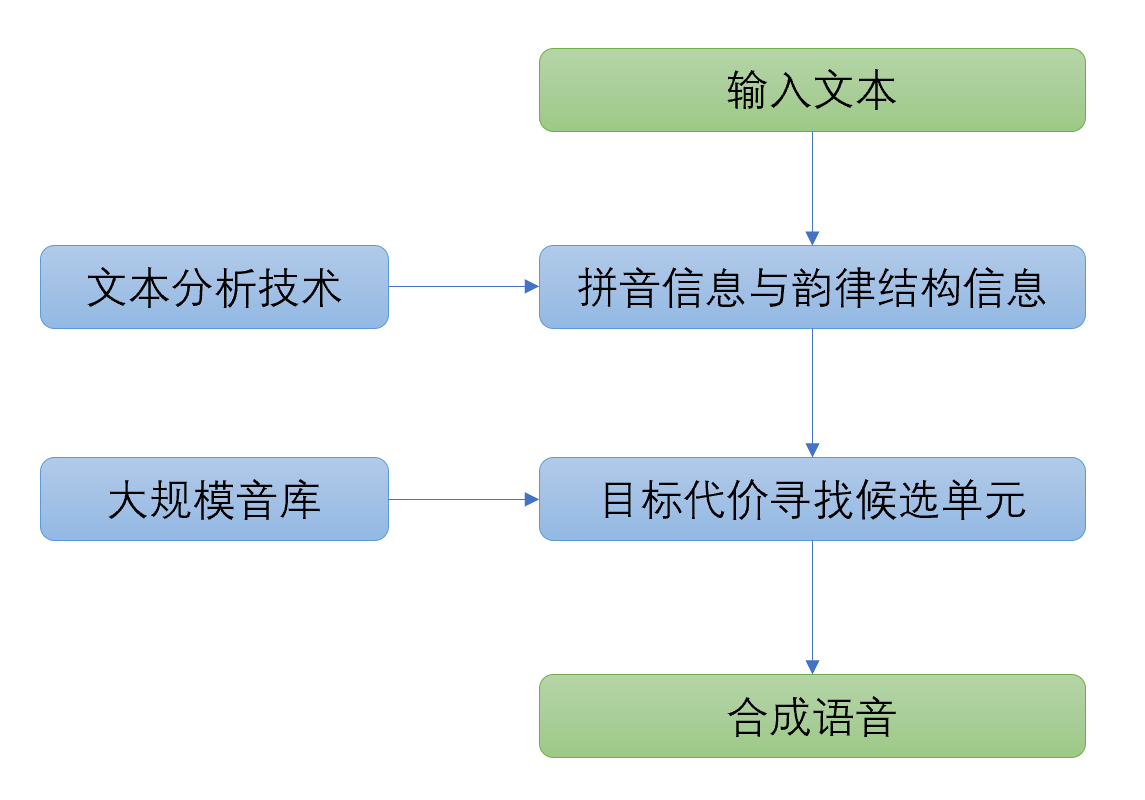
### 1.2 国内外概况

纵观语言合成技术的研究已有二百多年的历史，但是真正有实用意义的近代语音合成技术是随着计算机技术和数字信号处理技术的发展而发展起来的，主要是让计算机能够产生高清晰度、高自然度的连续语音。近几十年来国际和国内的研究主要集中在按规则文语转换，即将书面语言转换成口头语言。最近几年，一种新的基于数据库的语音合成方法正引起人们的注意。在这个方法中，合成语句的语音单元是从一个预先录下的庞大的语音数据库中挑选出来的， 不难想象只要语音数据库足够大，包括了各种可能语境下的语音单元，理论上讲有可能拼接出任何语句。由于合成的语音基元都是来自自然的原始发音，合成语句的清晰度和自然度都将会非常高。

目前国内的语音系统基本上都是采用基于PSOLA方法的时域波形拼接技术，其合成汉语普通话的可懂度、清晰度达到了很高的水平。然而同国外其它语种的文语转换系统一样，这些系统合成的句子及篇章语音机器味较浓，其自然度还不能达到用户可广泛接受的程度，从而制约了这项技术的大规模进入市场。

最近中国科技大学推出了KD-2000汉语文语转换系统，不仅在语音合成技术方面有进一步的发展，特别是在文本预处理中围绕层次化结构思想，运用大量的统计和规则的方法，较好地解决了三个大的处理环节：特殊符号处理，分词处理和拼接处理，使得汉语文语转换系统的整体性能有很大提高。以KD-2000文语转换为核心的“畅言2000”智能汉语平台软件已开始进入市场。

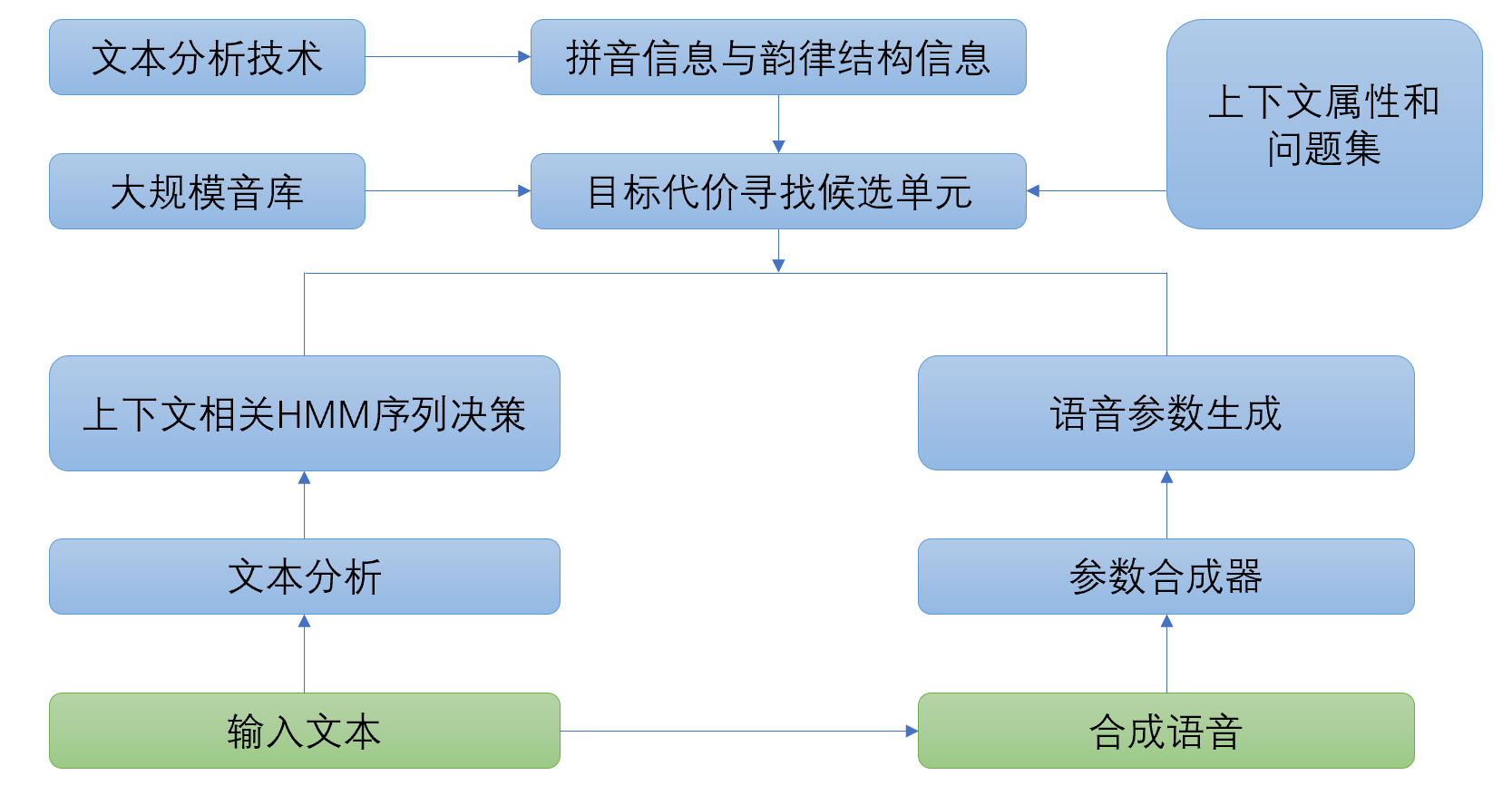
**第一代：基于拼接合成**



**拼接合成：**为了更好的拟合人声，拼接合成技术需要一个大规模的真人音库，音库内容按照音素和不同特征进行标注，合成时根据语言学特征寻找符合的音素，拼接起来完成合成

**优点：**效果接近真人，运算量低

**缺点：**拼接而成，语音不连贯，依赖音库，需要人工介入制定很多挑选规则和参数，制作成本高



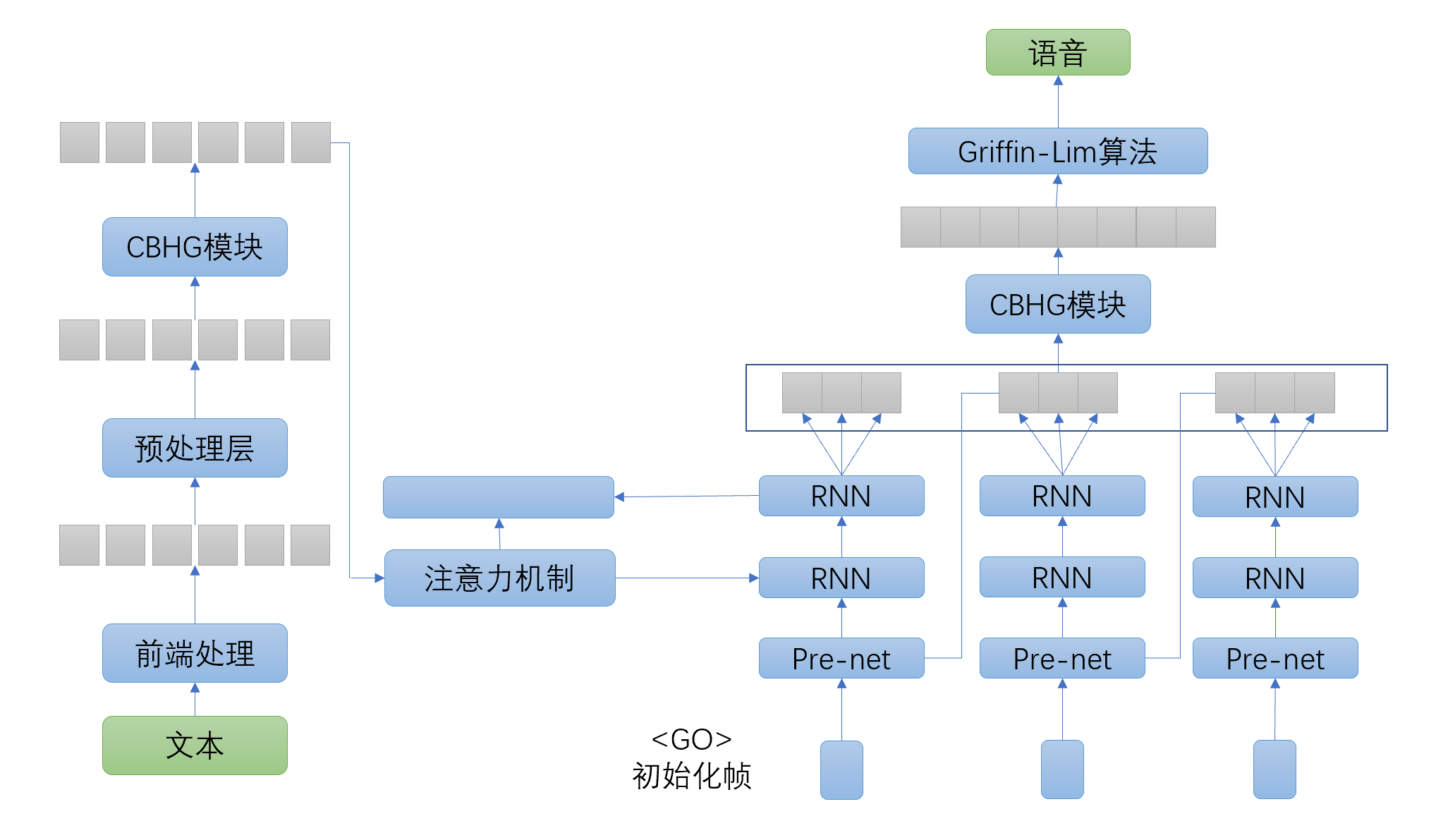
**第二代：基于参数合成**

**参数合成∶**通过深度学习构建文本特征和音库之间的映射关系，构建参数合成模型，当输入一个语言学特征时，基于神经网络给出声频特征，后通过声码器合成语音波形。

**优点：**音库数量要求不多，合成语音连接平稳，质量高

**缺点：**对声码器依赖程度高，同时，由于传统参数系统建模时存在信息损失，限制了合成语音表现力的进一步提升

**第三代：基于端到端（以Tactron为例）**



**端到端合成∶**端到端语音合成一定程度上解决了拼接合成和参数合成存在的部分缺陷。端到端合成系统直接输入文本或者注音字符，通过文本或者文本特征和语音直接建模，跳过声码器阶段，减少了对声码器的依赖，弱化前端概念。

**优点∶**降低对语言学知识的要求，可方便的在不同语种上复制，批量实现几.十种甚至更多语种的合成系统，语音自然程度高

**缺点∶**运算量大，无法人工调优，实时性低

## 二、 前期科研（开发）工作情况、现有基础和条件

### 2.1 Tacotron2端到端神经网络语音合成模型

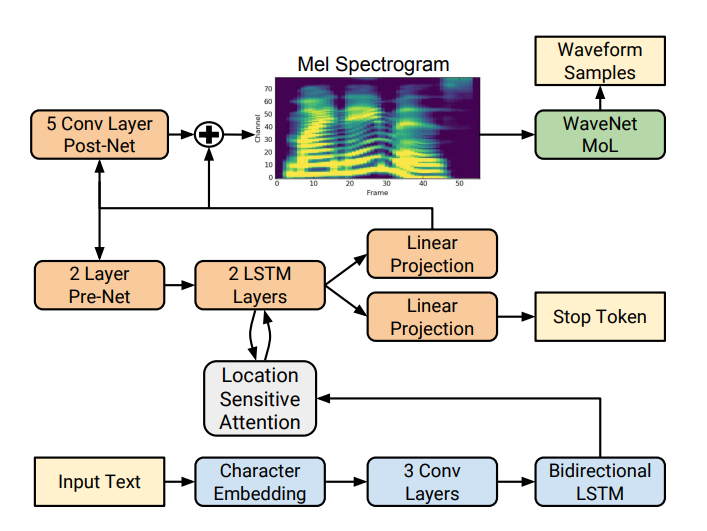
Tacotron2是由Google Brain 2017年提出来的一个语音合成框架。

是一个完整神经网络语音合成方法。模型主要由三部分组成：

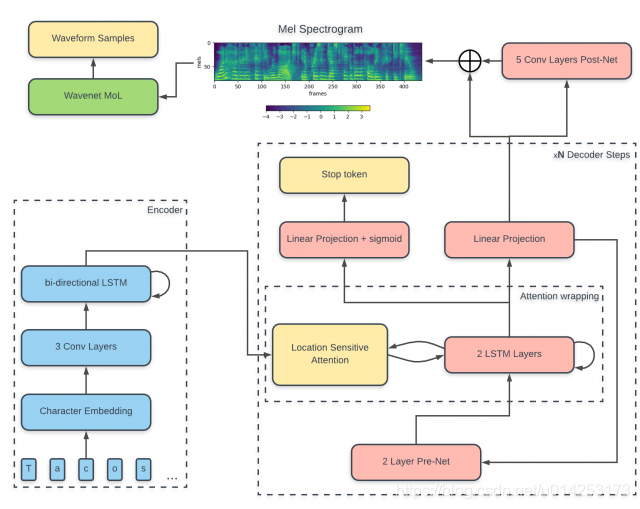
**声谱预测网络**：一个引入注意力机制（attention）的基于循环的Seq2seq的特征预测网络，用于从输入的字符序列预测梅尔频谱的帧序列。

**声码器（vocoder）**：一个WaveNet的修订版，用预测的梅尔频谱帧序列来生成时域波形样本。

**中间连接层**：使用低层次的声学表征-梅尔频率声谱图来衔接系统的两个部分。

**[[1]](#footnote-1)**

**2.1.1****声谱预测网络**

声谱预测网络主要包含一个编码器和一个包含注意力机制的解码器。编码器把字符序列转换成一个隐层表征，解码器接受这个隐层表征用以预测声谱图。[[2]](#footnote-2)

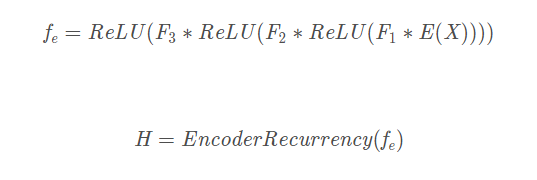
**2.1.2编码器**

编码器模块包含一个字符嵌入层（Character Embedding），一个3层卷积，一个双向LSTM层。

输入字符被编码成512维的字符向量；

然后穿过一个三层卷积，每层卷积包含512个5x1的卷积核，即每个卷积核横跨5个字符，卷积层会对输入的字符序列进行大跨度上下文建模（类似于N-grams），这里使用卷积层获取上下文主要是由于实践中RNN很难捕获长时依赖；卷积层后接批归一化（batch normalization），使用ReLu进行激活；

最后一个卷积层的输出被传送到一个双向的LSTM层用以生成编码特征，这个LSTM包含512个单元（每个方向256个单元）。

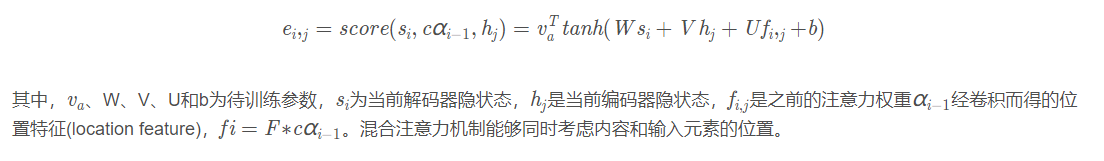


其中，F1、F2、F3为3个卷积核，ReLU为每一个卷积层上的非线性激活，E表示对字符序列X做embedding，EncoderRecurrency表示双向LSTM。

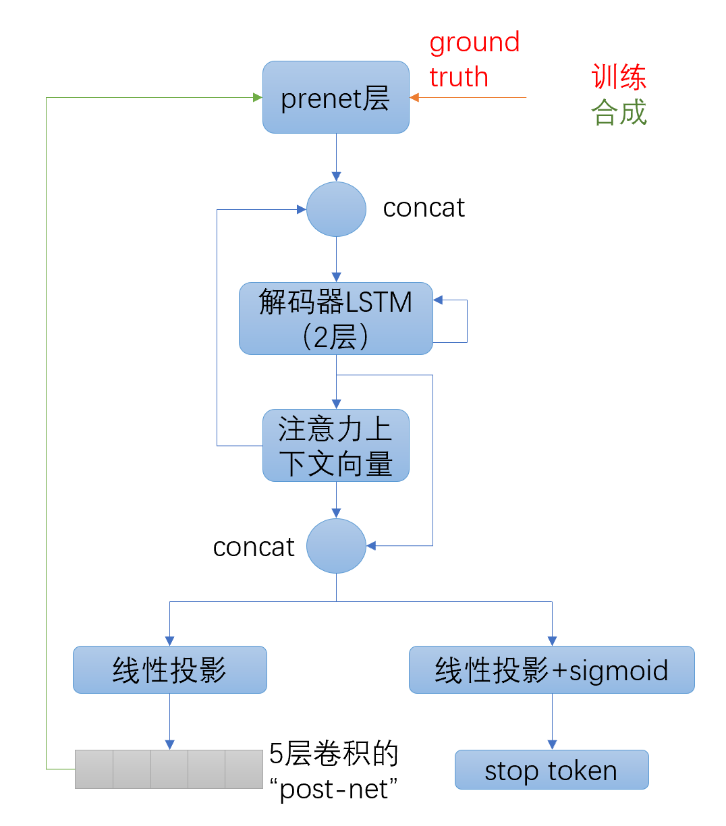
**2.1.3注意力网络**

Tacotron2中使用了基于位置敏感的注意力机制（Attention-Based Models for Speech Recognition），是对之前注意力机制的扩展（Neural machine translation by jointly learning to align and translate）；这样处理可以使用之前解码处理的累积注意力权重作为一个额外的特征，因此使得模型在沿着输入序列向前移动的时候保持前后一致，减少了解码过程中潜在的子序列重复或遗漏。位置特征用32个长度为31的1维卷积核卷积得出，然后把输入序列和为位置特征投影到128维隐层表征，计算出注意力权重。

Tacotron2中使用的是混合注意力机制，在对齐中加入了位置特征。



**2.1.4解码器**



解码器是一个自回归循环神经网络，它从编码的输入序列预测输出声谱图，一次预测一帧。

1.上一步预测出的频谱首先被传入一个“pre-net”，每层由256个隐藏ReLU单元组成的双层全连接层，pre-net作为一个信息瓶颈层（boottleneck）,对于学习注意力是必要的。

2.pre-net的输出和注意力上下文向量拼接在一起，传给一个两层堆叠的由1024个单元组成的单向LSTM。LSTM的输出再次和注意力上下文向量拼接在一起，然后经过一个线性投影来预测目标频谱帧。

3.最后，目标频谱帧经过一个5层卷积的“post-net”来预测一个残差叠加到卷积前的频谱帧上，用以改善频谱重构的整个过程。post-net每层由512个5X1卷积核组成，后接批归一化层，除了最后一层卷积，每层批归一化都用tanh激活。

4.并行于频谱帧的预测，解码器LSTM的输出与注意力上下文向量拼接在一起，投影成一个标量后传递给sigmoid激活函数，来预测输出序列是否已经完成的概率。

**2.1.5 与Tacotron对比**

1.Tacotron 2使用了更简洁的构造模块，在编码器和解码器中使用是普通的LSTM和卷积层;Tacotron中使用的是“CBHG”堆叠结构和GRU循环层；

2.Tacotron2在解码器的输出中没有使用“缩小因子（reduction factor）”，即每个解码步骤只输出一个单独的频谱帧。

### 2.2声码器模型

声码器是一种将声学参数转换成语音波形的工具。声码器利用频谱等声学特征，生成语音样本点并重建时域波形，如：将梅尔频谱恢复为对应的语音。当前常用的声码器模型算法有纯信号处理：Griffin-Lim算法，基于自回归神经网络模型：ＷaveNet，WaveRNN，基于非自回归神经网络模型：WaveGlow和基于生成对抗网络：MelGAN。

**2.2.1 纯信号处理：Griffin-Lim算法**

Griffin-Lim算法是一种已知幅度谱，未知相位谱，通过迭代生成相位谱，并用已知的幅度谱和计算得出的相位谱重建语音波形的方法。由于要对抛弃相位信息的语音波形进行恢复，因此需要通过迭代尽可能“猜测”出原始信息。这种声码器不需要训练，不需要预知相位谱，而是通过帧与帧之间的关系估计相位信息，从而重建语音波形。

算法步骤：

1)随机初始化一个相位谱

2)用这个相位谱与已知的幅度谱（来自MEL谱）经过ISTFT（逆傅里叶变换）合成新的语音波形

4)用合成语音做STFT， 得到新的幅度谱和新的相位谱

5)丢弃新的幅度谱，用已知幅度谱与新的相位谱合成新的语音

6)重复2,3,4多次，直至合成的语音达到满意的效果或者迭代次数达到设定的上限

**2.2.2 基于自回归神经网络模型**

**ＷaveNet**

WaveNet是一种典型的自回归生成模型。所谓自回归生成模型，即是利用前面若干时刻变量的线性组合来描述以后某时刻变量的线性回归模型。模型输入若干历史采样点，输出下一采样点的预测值，也就是根据历史预测未来，不对语音做任何先验假设，而是利用神经网络从数据中学习分布，不直接预测语音样本值，而是通过一个采样过程来生成语音。

WaveNet 最初由 DeepMind 推出，是基于 CNN 的采样点自回归模型，主要成分是因果卷积，确保了模型输出不会违反数据的顺序。为了在扩大感受野和控制参数量间寻找平衡，引入 “扩展卷积”，即卷积核在比自身大的数据上进行卷积时跳步的卷积方法，有效地使网络可以执行粗粒度的卷积操作。随着深度增加，扩大因子的指数增长可以使感受野呈指数级增大，将多组这样的卷积模块堆叠起来会进一步增大模型容量和感受野大小。训练好的 WaveNet 合成的音频波形需要添加 condition ，condition 包含了文本的语义信息，形式并不唯一，用于帮助 WaveNet 合成需要的波形。

其优势是：能根据声学特征，生成高质量的语音，训练快、效果好、网络结构清晰简洁。单一模型的WaveNet可以以相同的保真度捕获很多说话人的特征，并可以针对说话者进行训练后在多人之间切换。

缺陷则是：1)每次预测一个采样点，速度慢；2)WaveNet并非完整的TTS方案，依赖其余模块提供高层特征，前端分析出错，直接影响合成效果；3)用于TTS时，初始采样点的选择很重要；4) inference 性能差，在 CPU 平台通常需要数十秒时间合成一秒语音

**2.2.3 WaveRNN**

WaveRNN采用一种全新的声码器架构，结构尤其簡单，主体仅由一个单层循环神经网络（RNN）组成，并且RNN利用稀疏矩阵进一步降低计算量，使用简化模型、稀疏化、并行序列生成等技术显著提升了序列生成速度,相比ＷaveNet合成速度提升10倍左右。

**2.2.4 基于非自回归神经网络模型:WaveGlow**

WaveGlow:一种依靠流的从梅尔频谱图合成高质量语音的网络。结合了Glow和WaveNet，生成的快、好、高质量的韵律，不需要自动回归，实现也只是一个单网络，仅使用单个成本函数进行训练：最大化训练数据的可能性，使训练过程简单且稳定。

**2.2.5 基于生成对抗网络：MelGAN**

MelGAN采用非自回归前馈卷积架构，在不引入额外蒸馏和感知损失的前提下，依然能够产生高质量的语音，是第一种利用GAN合成时域波形的方法。MelGAN的速度明显快于其他mel谱图反演的替代方法，而音频质量没有明显下降。

MelGAN模型结构：每个上采样层都是一个转置卷积，其内核大小是步幅的两倍（与该层的上采样率相同）。256x上采样分8个阶段进行，分别是8x，8x，2x和2x升采样。每个残差的扩张卷积堆栈都具有三层，其扩张1、3和9的内核大小为3，总感受野为27个时间步长。使用leaky-relu进行激活。每个判别器块具有4个步幅为4的跨步卷积。

其优势是：模型计算复杂度低，并行度高，因而合成速度极快。

**2.2.6 WaveFlow模型**

WaveFlow来自百度研究院的论文WaveFlow: A Compact Flow-based Model for Raw Audio，飞桨复现了该语音合成模型。根据官网介绍，只有5.9M参数量，比经典的WaveGlow语音合成模型小了15倍，同时语音合成效果也非常好。WaveFlow和WaveGlow都是基于流的生成模型，它和GAN都属于生成模型家族。

**需要注意的是，WaveFlow是个vocoder（声码器，一种将声学参数转换成语音波形的工具）,不能直接实现文字转语音，需要与Parakeet库中的TTS模型Deep Voice 3、Transformer TTS或FastSpeech模型结合，实现文字转语音的拟声合成。**

### 2.3 文本预处理

这一步做的事情是把文本转化成音素序列，并标出每个音素的起止时间、频率变化等信息。

作为一个预处理步骤，它的重要性经常被忽视，但是它涉及到很多值得研究的问题，比如拼写相同但读音不同的词的区分、缩写的处理、停顿位置的确定，等等。

文本预处理主要有三个步骤，文字的规范化，语音分析，还有韵律分析。

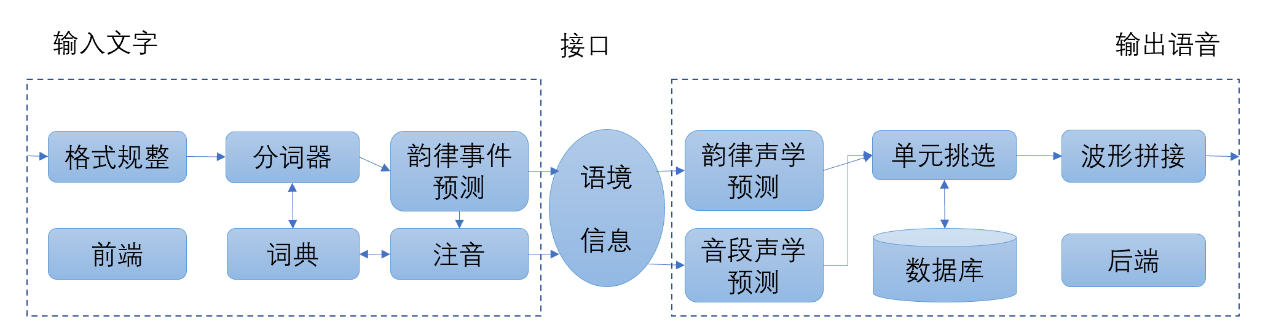


图1 TTS系统结构

图2展示了目前TTS系统中典型的文本处理模块的原理框图。图中显示，模块的输入为待合成语音的文本，输出的是发音描述符号，包括音段层面的注音符号与超音段层面的韵律符号。

文本首先通过自动分词器(Parser),被切分为带有POS标记的词汇LW(Lexical word)序列。分词及其POS构成了韵律预测器和自动标音器输入特征的主要成分，通过自动标音器，标出每个合成语音单元的注音符号；通过韵律预测器，预测出合成语音的韵律符号描述。

文本处理的三个计算模型：自动分词器、自动标音器、韵律预测器的构建，则普遍采用了与规则库相结合的数据驱动的学习模型；分别基于标注了语言学、语音学信息的文本数据库及言语数据库，完成模型的参数训练，之后利用模型完成由输入文本到发音描述符号的转换计算。

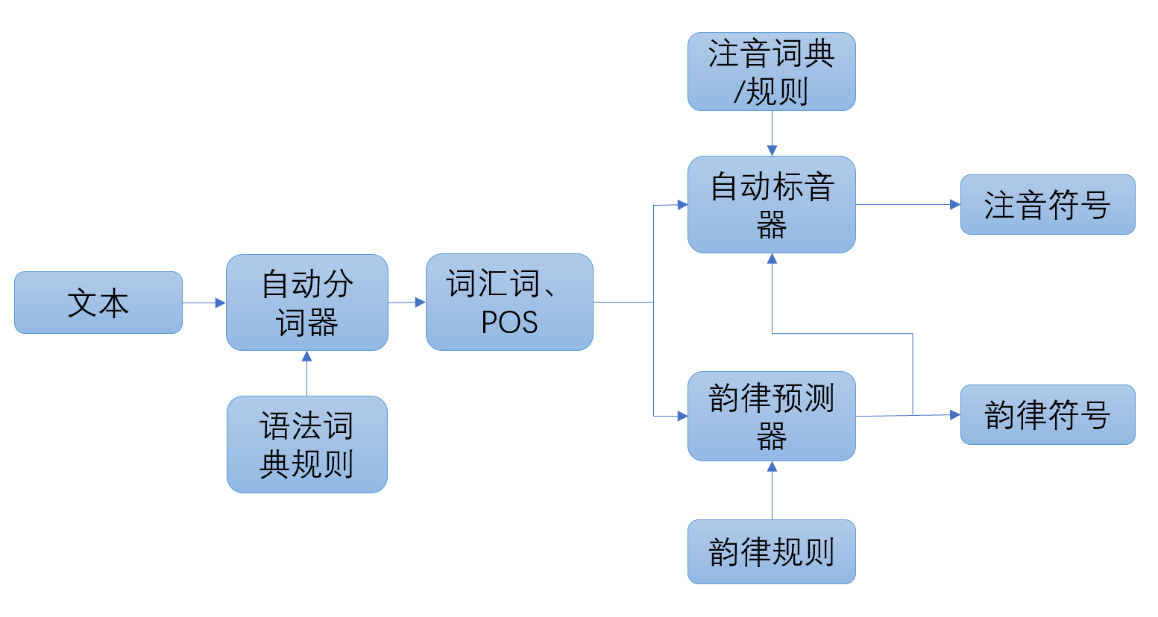


图2 TTS系统中文本处理模块原理框

**自动分词器的实现**

自动分词器就是完成对输入文本的分词和词性标注。

对于中文词法分析这是一个非常困难的问题，其难点主要体现在以下几方面：

1. **词语切分**：由于汉语词语之间没有空格分开，需要从连续的汉字串中正确辨认汉语的词语，常见的歧义现象如：“的确切”可能是“的确/切”或者“的确切”。这些类型的歧义现象在汉语中非常常见，会对汉语词语切分造成极大的干扰
2. **未定义词识别**：词典中不可能收录所有的词语，大量的人名、地名、机构名、外来语译名、新词语等等，都需要通过软件来自动识别，而在汉语中这些未定义词没有空格作为边界，其组成成分又是有意义的普通汉字，因此识别难度很大。
3. **词性标注：**汉语中词语兼类情况非常常见，比如说“领导”可以是动词，也可以是名词，要正确标注出每个词的词性，也有很多困难。

这里我们采用的自动分词器，可以直接“借用”自然语言处理NLP( Natural Language Processing)所应用的 Parser技术。较早期的自动分词方法基本上都是依赖事先编制的词典，通过规则或统计计算做出字词切分的判断。近期出现的基于字标注( Character-based Tagging)的分词方法，较好地解决了未登录词Oo( Out-Of-Vocabulary)的划定，并在分词评测中获得了较好的成绩。其技术路线是机器学习方法，通过训练语料构建统计模型，分词的精度决定于训练语料的精度、训练语料与测试语料的相关程度、利用大规模训练语料构建的统计模型。

语音分词的目的是根据人们在表达上的习惯和语流的停顿及强弱变化，在每个词之间插入长度不等的空语音符号(停顿)。提高语流的节奏和自然度，以利于听者的理解。建立一个好的自动分词系统，有两个关键：一是词库，二是分词算法。对于一个实时文语转换系统来说，要求语音分词速度快，能处理歧义词串和未登录词。词库不宜过大，一般的基本分词词库中包括二字词、三字词和四字词。

**词性标注**

现阶段在TTS文本处理模块中，对合成文本的分析，除了分词结果自身之外，还利用了每个词的词性信息作为后续的韵律预测、自动标音模型的输入特征。

我们在分词的基础上标注词性。一种方式是基于手工标注数据库，统计得到每个词典词的词性出现频度及相邻词的词性组合出现频度，构建词性标注概率模型。在对词的标注过程中，根据分析文本每个词的词性与相邻词的词性结合概率，从各种可能的组合中找出概率最大化的词性串作为标注结果。

如，例句带有词性的分析结果，由于“飞行”被标注为动词(vg)词性，将影响其与“甲板”合并构成一个与“直升飞机”等同的韵律单元。

例：直升飞机ng)准确ag)地(us)降落(vg)在(p)飞行(vg)甲板(ng上(f)。

N元文法模型

在分词过程以及下文的模型训练中常用到N元文法模型，其作用是预测一个单词序列出现的概率。N元语法假设一个单词出现的概率分布只与这个单词前面的n-1个单词有关，与更早出现的单词无关。这样，为了描述这个概率分布，需要使用一个n维数组，这个数组中每一维长度为单词的个数m,这个数组中元素的个数为mn,其中元素(ai1,ai2...ain)的含义为：在单词串(W1,W2...Wn-1)后面出现单词Wn的概率，也就是P(Wn|W1W2...Wn-1).

基于概率的分词模型可以这样描述，在给定字符序列S=c1c2…cn的前提上，求最优词语序列Wopt=w1w2…wn,使得

Wopt = argmax P(W|S)

利用 Bayes公式和概率乘积公式，N元文法的假设正好可以用到这里。假设单词序列符合一元文法，单词间相互独立，那么可以得到:



假设单词序列符合二元文法，每个单词只依赖于它前面的一个单词，则得到



下文讲述的原有模型以N元文法为理论基础，再结合动态规划DP( Dynamic Programming)算法以完成分词和后期层级结构的预测。

DP算法的原理就是根据分析文本每个词的词性与相邻词的词性结合概率，从各种可能的组合中找出概率最大化的词性串作为标注结果。其中的概率是根据N元文法模型得到的概率。

具体的分词过程(即词汇词LW的生成)及算法思想参照如下方案进行实施： 为了实现文本处理，我们构建了一个词汇表。词汇表中的每个条目都包含以下信息：

1. 词的拼写：wi(i=1,…,k),K是词汇表的长度。
2. 词频：Fre(wi),这个频率是由统计结果得到的，即：



1. 词的所有可能词性：POSj(j=1…N),N是词的词性数.
2. 词的各个词性之间的权重：λ(POSj(wi))

### 2.4多人语音合成模型

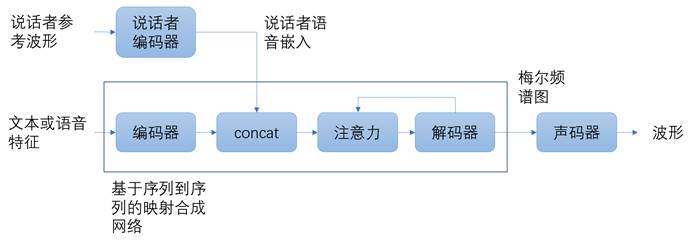
由三个独立训练的神经网络组成，如下图所示:

(1)语音特征编码器，提取说话者的声音特征信息。将说话者的语音嵌入编码为固定维度的向量，该向量表示了说话者的声音潜在特征。

(2) 基于序列到序列的映射合成网络，基于Tacotron 2的映射网络，通过文本和语音特征编码器得到的向量来生成log mel spectrogram（梅尔谱图将谱图的频率标度Hz取对数，转换为梅尔标度，使得人耳对声音的敏感度与梅尔标度承线性正相关关系）

(3) WaveNet自回归语音合成网络，将梅尔频谱图（谱域）转化为时间序列声音波形图（时域），完成语音的合成。

　　需要注意的是，这三部分网络都是独立训练的，声音编码器网络主要对序列映射网络起到条件监督作用，保证生成的语音具有说话者的独特声音特征。



**声音特征编码器**

　　编码器主要将目标说话人的参考语音 嵌入编码到固定维度的向量空间，并以此为监督，使映射合成网络能生成具有相应特征的梅尔频谱。编码器的关键作用在于相似性度量，对于同一说话者的不同语音，其在嵌入向量空间中的向量距离（余弦夹角）应该尽可能小，而对不同说话者应该尽可能大。此外，编码器还应具有抗噪能力和鲁棒性，能够不受具体语音内容和背景噪声的影响，提取出说话者声音的潜在特征信息。在与文本无关的说话人验证任务上训练的说话人辨别模型满足这些要求，因此可以进行迁移学习。

　　一个高度可扩展并且准确的说话人验证网络框架。该网络可以从任意长度的语音中计算出对数梅尔谱图帧序列，从而映射到固定维嵌入向量，称为d-vector[20，9]。该网络使用广义端到端说话人验证损失训练，使得来自同一说话人的话语的embedding具有高余弦相似性，而来自不同说话人的话语的嵌入在嵌入空间中相距很远。训练数据集由 分割成1.6秒的语音示例和相关的说话者身份标签 组成。

　　编码器的输入是40通道数的 log-mel spectrograms，网络结构主要由3层 256个单元的LSTM 构成。最后一层是全连接层，全连接层输出经过L2正则化处理后，即得到整个序列的嵌入向量表示。实际推理时，任意长度的输入语音信号都会被800ms的窗口分割为多段，重叠50%，每段得到一个输出，该网络在每个窗口上独立运行，输出被平均和归一化以创建最终的嵌入向量。

　　虽然网络没有直接优化以学习捕获与合成相关的说话人特征的表示，但我们发现 说话人辨别任务训练的模型 生成的embedding适用于根据说话人身份 调节合成网络。

## 三、项目实施方案

### 3.1 拟解决的关键问题

#### 3.1.1自动注音的实现

造成汉语注音问题的原因主要有以下几个方面，一是命名实体的检测；二是数字串的读法；三是多音字及连读变调。

命名实体的注音关键在于在自动分词时能够将这些词或词组检测出来，之后可就交由正常自动注音模型处理。

和命名实体一样，数字串的读法也经常出错，而且听众对这类错误十分敏感。常见的解决思路是考察数字串的相邻词及其词性，通过设定有限的规则，决定数字串的读法。

而解决多音字及连读变调，则可从以下几个方面进行操作：

1. 注音表。在所有的注音模型中，都会有一个覆盖所有汉字的音节注音表。对于多音字，则无法通过简单的查表解决。
2. 分词。组成多音节词后，有相当比例的多音字的读音在词内得到固定，如“降”在“投降”这个词中只能读“Xiang2”,在“降落”这个词中只能读“Jiang4”,利用正确的分词结果，借助注音词表，可解决部分多音字的注音
3. 词性。通过附加POS标示的注音词表，多音字读音又可解决一部分，如“地道”为名词时读“di4dao4”,为形容词时读“di4dao5”。但对于具有相同POS标示的多音词，问题又出现了，如标示为动词词性的“倒”,可以读“dao4”,也可以读“dao3”;“调配”可以读“diao4pei4”,也可以读“tiao2pei4"。词义不同读音不同，仅依靠分词及词性，仍旧无法解决，需考察其出现的上下文才能确定。
4. 规则。除了通过直接查表，有时还需根据多音字的语境信息，利用规则实施注音。由于多音字数目庞大，语境又千变万化，靠手工编纂规则，工程浩大，既不经济又不可靠，因而出现了一些利用机器学习的自动生成规则的方法。
5. 统计模型。 采用TBL( Transformation- Based error-dri ven Learning)算法，利用多音词及其词性、相邻词及其词性以及位置信息等作为输入特征，基于一个由通用言语数据库调整得到的多音字训练数据库，通过训练得到了一系列转换规则。
6. 连读变调。主要涉及语句中“上上”相连前“上”变“阳平”,以及“一、不”变调。这些音节声调的确定，可在韵律结构确定之后，通过设定规则加以实施。

以上自动注音所利用的都是有关字词本身，或与之相邻的字词的特征。而注音问题的彻底解决，有时需要根据待注音词的确切语义才能确定。例如，例句中两个“长”字的注音。

例：孩子又长了，胳膊也长了

因而，自动注音器的输入特征，从分析范围上需要有全局性的背景描述，从语言学层级上需要上升到语义层面甚至语用层面。

经过自动注音器之后得到的是输入文本的一系列注音符号序列。

**3.1.2韵律结构预测的实现**

对于文本处理模块来讲，其核心是韵律预测。在言语工程界，目前通常采用韵律层级结构来描述语句内部的单位边界，一般地划分为韵律词、韵律短语、语调短语等几个层级，用以刻画感知不同的“停延”级别。

由于韵律层级结构的划分虽然与语法结构有着密切的关联，但并不完全一致，所以言语工程界独立发展了韵律结构的预测计算模型。其基本思想都是基于相当规模的言语数据库，采用数据驱动的统计模型构建韵律预测器用二元文法、DP和规则相结合的统计方法预测现在大多文献中通用的几个层级结构，从小单位到大单位依次是：词汇词LW、韵律词PW( Prosodic Word)、韵律短语PP( Prosodic phrase)、语调短语IP( Innotation phrase)、句子ST( Sentence)。

#### 3.1.2训练数据集

  采用公开数据集进行训练。希尔贝壳中文普通话语音数据库AISHELL-3

的语音时长为85小时88035句，可做为多说话人合成系统。录制过程在安静室内环境中， 使用高保真麦克风（44.1kHz，16bit）。218名来自中国不同口音区域的发言人参与录制。专业语音校对人员进行拼音和韵律标注，并通过严格质量检验，此数据库音字确率在98%以上。从中筛选不同年龄不同性别的声音数据

### 3.2拟采用的技术路线

#### 方案一：NVIDIA-tacotron2语音合成方案

基于pytorch实现文字输入后预测得到梅尔三维频谱图，再使用Waveglow对梅尔三维频谱图生成波形，最后输出语音。采用LJSpeech数据集测试，并有分布式和自动混合精度支持，合成更加可靠准确的语音。

**训练过程**

声码器

文本预处理

声音特征生成网络

文本 语音

声音特征生成网络：Tacotron2

声码器： WaveGlow

环境配置： Pytorch ，NVIDIA GPU + CUDA cuDNN

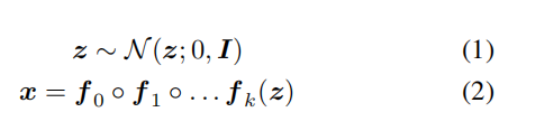
**WaveGlow**

WaveGlow 是一个生成式模型，通过从分布中采样来生成音频。

为了使用神经网络作为生成模型，我们从一个简单的分布中抽取样本。

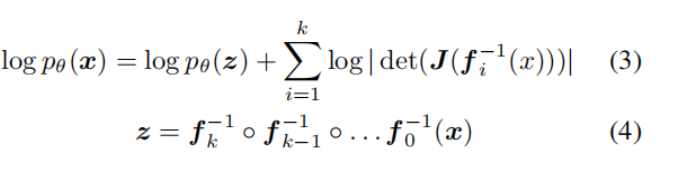
一个维数与我们期望的输出相同的零均值球形高斯函数，将这些样本通过一系列的层，将简单的分布转换为期望的分布。

在这种情况下，我们在梅尔谱图条件下模拟音频样本的分布。

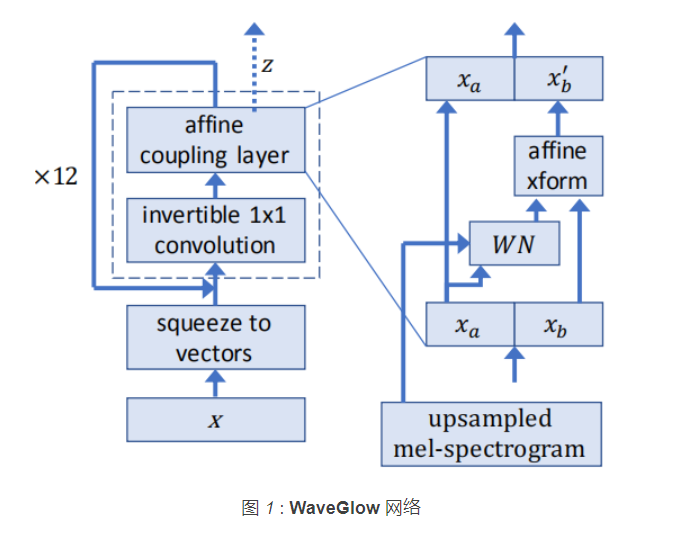


我们想通过直接最小化数据的负对数似然来训练这个模型。如果我们使用任意的神经网络，这是很难做到的。基于流的网络(《Nice: Non-linear independent components estimation》，《Density estimation using real nvp》，《Glow: Generative flow with invertible 1x1 convolutions》)通过保证神经网络映射是可逆的来解决这一问题。

通过限制每一层为双射，可以通过变量的改变直接计算可能性：



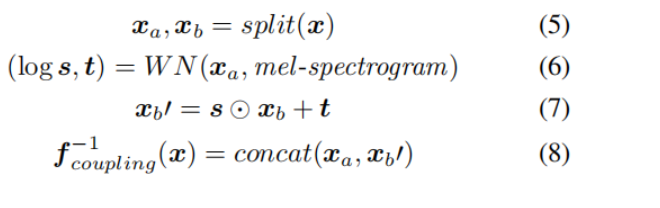
在我们的例子中，第一项是球面高斯分布的对数似然。 这一项惩罚了变换后样本的 l2 范数。第二项由变量代换产生， J 是雅可比矩阵。雅可比矩阵的对数行列式奖励任何一层在前进过程中增加空间的体积。 这一项也防止了一个层仅仅将 x 项乘以 0 来优化 l2 范数。这个转换序列也称为规范化流程。

[[3]](#footnote-3)

这里的流程包括一个可逆的 1×1 卷积，然后是一个仿射耦合层，如下所述。

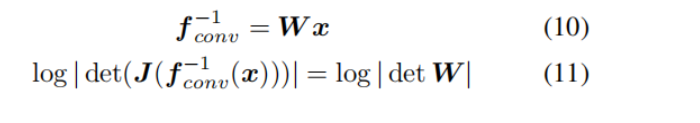
**仿射耦合层**

可逆神经网络通常使用耦合层来构建(《Nice: Non-linear independent components estimation》，《Density estimation using real nvp》，《Glow: Generative flow with invertible 1x1 convolutions》)。 在我们的例子中，我们使用了仿射耦合层(《Density estimation using real nvp》)。一半的通道作为输入，然后产生乘法和加法术语，用于缩放和转换剩余的通道：

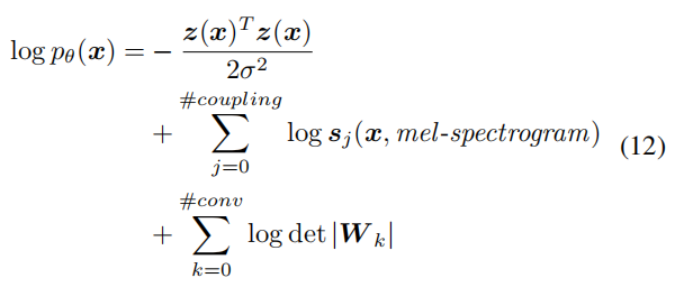


**1x1可逆的卷积**

在仿射耦合层中，同一半的通道不会直接相互修改。如果不跨渠道混合信息，这将是一个严重的限制。 在 Glow (《Glow: Generative flow with invertible 1x1 convolutions》)之后，我们通过在每个仿射耦合层之前添加一个可逆的 1x1 卷积层来混合通道上的信息。这些卷积的 W WW 个权值初始化为标准正交的，因此是可逆的。由于变量的变化，这个变换雅可比矩阵的对数行列式连接了损失函数，并且在训练网络时，还可以使这些卷积保持可逆。



在添加了来自耦合层的所有项之后，最终的可能性就变成了:



第一项来自球形高斯分布的对数似然。

*σ2*项是高斯分布的假设方差，其余项是变量的变化。

#### 方案二：百度飞桨平台基于PaddleSpeech的中文语音合成方案

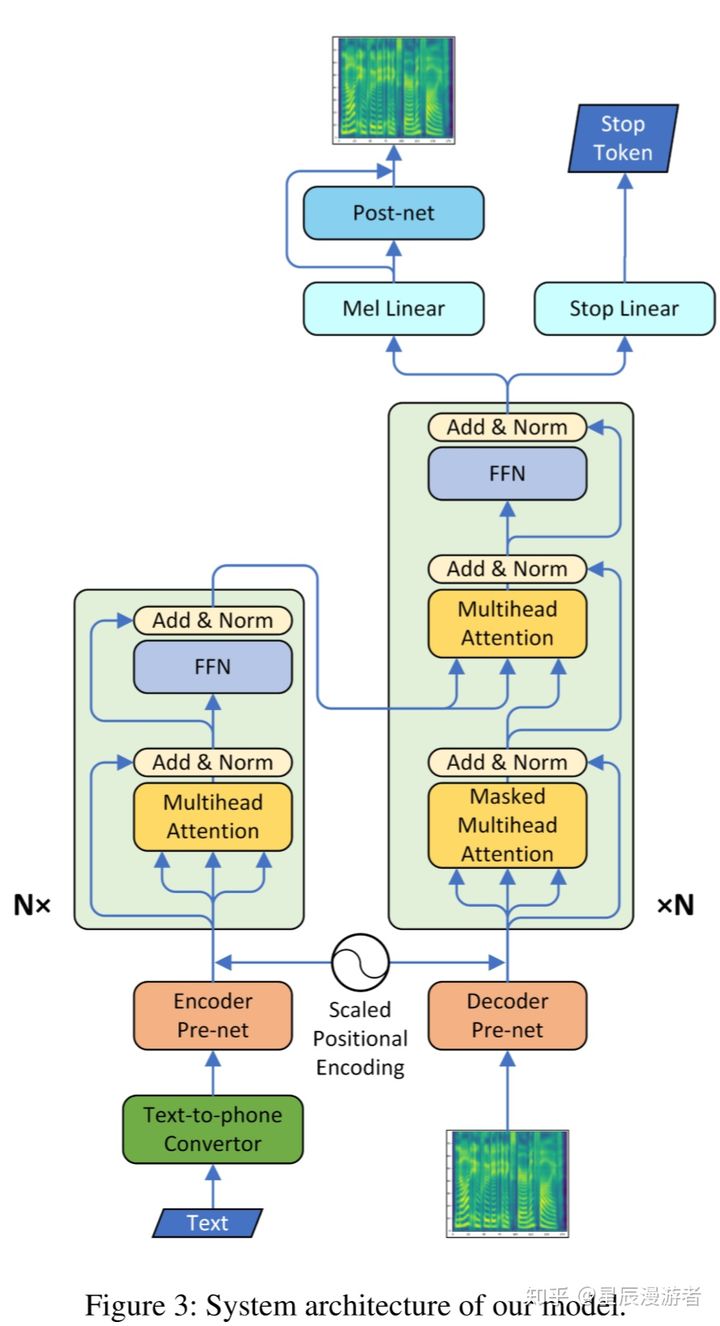
**Parakeet模型库介绍**

Parakeet是飞桨近期上新的语音合成套件，用于实现端到端的语音合成。如果您使用过各类读书app或者某些浏览器、插件的朗读功能，这些都是典型的TTS（Text To Speech）场景。

本项目将使用WaveFlow语音合成模型完成相关任务，并结合Transformer TTS验证语音合成效果。

**Transformer TTS文字转语音模型**

Parakeet使用PaddlePaddle动态图复现了Transformer TTS, 根据论文*[Neural Speech Synthesis with Transformer Network](https://arxiv.org/abs/1809.08895" \t "_blank)*实现了基于Transformer的语音合成系统。

[[4]](#footnote-4)

在这篇论文中，作者把Transformer和Tacotron2融合，形成了TransformerTTS。

模型的主体还是Original Transformer，只是在输入阶段和输出阶段为了配合语音数据的特性做了改变。首先是Encoder的Input阶段，先将text逐字符转化为编号，方便Embedding，然后进入Encoder PreNet，这层网络由一个Embedding layer和三层卷积层构成，转化为512维的向量后，进入Transformer Encoder。其次是Transformer的Decoder部分，分为Input和Output。Input通过一个PreNet，将80维的梅尔声谱图转化为512维向量，这里的PreNet是一个三层的全连接网络（个人认为论文中应当解释一下为什么Encoder的PreNet是用卷积设计的，而Decoder的PreNet由全连接网络就可以解决问题）；Output部分与Tacotron2的设计完全一致。

Parakeet库中Transformer TTS支持两种vocoder，分别是Griffin-Lim algorithm和WaveFlow。WaveFlow属于基于深度神经网络的声码器，而Griffin-Lim是在仅知幅度谱、不知道相位谱的条件下重建语音的算法，属于经典声码器，算法简单，高效，但是合成的声音比较颤抖，机器感较强。

**训练过程**

数据集准备

导入预训练模型

环境配置

训练数据集

实现文字到语音转换

声音特征生成网络：TransformerTTS

声码器： Griffin-Lim/WaveFlow

环境配置： PaddlePaddle

**第一步：下载Parakeet模型库**

**Notebook环境**

和PaddleDetection不同，目前Parakeet模型库目前文件还是比较少的，因此尽管在gitee上没有镜像，直接去github上拉取不需要花太长时间（一般也不会中断）

**脚本任务环境**

在脚本任务中，需要逐个文件夹创建、逐个文件上传。并且，由于脚本任务提交时，后台实际上执行的是命令行python -u train.py，因此train.py文件是不可删除的，只能修改内容。由于文件内容比较多，建议只上传必须的waveflow示例目录和parakeet安装必须的目录文件。

另外可以使用print(os.getcwd())确认一下脚本任务代码界面的路径，是/root/paddlejob/workspace/code

**第二步：安装依赖库**

NoteBook的依赖库有 libsndfile1，PaddlePaddle，Parakeet，CMUdict for nltk

以CMUdict for nltk的安装为例



脚本环境的依赖库有libsndfile1，PaddlePaddle，Parakeet，CMUdict for nltk

**第三步：准备训练集和预训练模型**

参考官方文档，Parakeet项目提供了WaveFlow模型在64, 96和128隐藏层大小（比如 128 则意味着每个隐藏层的大小是 128）下的预训练模型。本文将使用隐藏层大小为128的模型。

**第四步：环境配置和YAML文件准备**

**Notebook环境**

*%set\_env CUDA\_VISIBLE\_DEVICES=0（Notebook环境）、export CUDA\_VISIBLE\_DEVICES=0*或*os.environ['CUDA\_VISIBLE\_DEVICES'] = '0'*指定GPU

直接修改Parakeet/examples/waveflow/configs/waveflow\_ljspeech.yaml文件或预训练模型的waveflow\_ljspeech.yaml文件内容，如调整learning\_rate、batch\_size。

在**脚本任务**中，默认的数据集存放目录和Notebook环境挂载的位置不同，参考示例说明



第五步：开始训练

第六步：使用Transformer TTS + WaveFlow实现文字转语音

## 四、参考文献

### 4.1文献综述

微软亚洲研究院的研究员们通过调研了450余篇语音合成领域的文献，发表了迄今为止语音合成领域几乎最详尽的综述论文 “[A Survey on Neural Speech Synthesis](https://arxiv.org/pdf/2106.15561.pdf)”。在文中，研究员们还整理收集了语音合成领域的相关资源如数据集、开源实现、演讲教程等，同时也对语音合成领域未来的研究方向进行了探讨和展望。

论文从核心模块：文本分析（textanalysis）、声学模型（acoustic model）、声码器（vocoder）、完全端到端模型（fully end-to-end model）等方面进行了介绍。

研究员们根据神经语音合成系统的核心模块提出了一个分类体系。每个模块分别对应特定的数据转换流程：

1）文本分析模块将文本字符转换成音素或语言学特征；

2）声学模型将语言学特征、音素或字符序列转换成声学特征；

3）声码器将语言7学特征或声学特征转换成语音波形；

4）完全端到端模型将字符或音素序列转换成语音波形。

#### 文本分析

文本分析模块几个常见的任务包括文本归一化、分词、词性标注、韵律预测、字形转音形以及多音字消歧等。

#### 声学模式

在声学模型部分，统计参数合成里用到的基于神经网络的声学模型，重点是端到端模型的神经声学模型，包括基于 RNN、CNN 和Transformer 的声学模型以及其它基于 Flow、GAN、VAE、Diffusion 的声学模型。

#### 声码器

声码器的发展分为两个阶段，包括传统参数合成里的声码器如 STRAIGHT和 WORLD，以及基于神经网络的声码器。相关工作分为以下几类，包括：

1）自回归声码器（WaveNet，SampleRNN，WaveRNN，LPCNet 等）；

2）基于 Flow 的声码器（WaveGlow，FloWaveNet，WaveFlow，Par. WaveNet 等）；

3）基于 GAN 的声码器（WaveGAN，GAN-TTS，MelGAN，Par. WaveGAN，HiFi-GAN，VocGAN，GED，Fre-GAN 等）；

4）基于 VAE 的声码器（WaveVAE等）；

5）基于 Diffusion 的声码器（DiffWave，WaveGrad，PriorGrad 等）。

#### 完全端到端模型

端到端模型的发展经历了以下几个阶段：

阶段0：在统计参数合成方法中，使用文本分析、声学模型和声码器三个模块级联；

阶段1：在统计参数合成方法中，将前两个模块合起来形成一个声学模型；

阶段2：直接从语言学特征生成最终的波形，例如 WaveNet；

阶段3：声学模型直接从字符或音素生成声学模型，然后利用神经声码器生成波形；

阶段4：完全端到端的神经网络模型。

### 4.2参考文献

1. [Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions](https://arxiv.org/pdf/1712.05884.pdf) by Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, Yonghui Wu.

2. [Tacotron: Towards End-to-End Speech Synthesis](https://arxiv.org/pdf/1703.10135.pdf) by Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, Quoc Le, Yannis Agiomyrgiannakis, Rob Clark, Rif A. Saurous.

3. [WaveGlow: A Flow-based Generative Network for Speech Synthesis](https://arxiv.org/pdf/1811.00002.pdf) by Ryan Prenger, Rafael Valle, Bryan Catanzaro.

4. [Neural Speech Synthesis with Transformer Network](https://arxiv.org/pdf/1809.08895.pdf) by Naihan Li, Shujie Liu, Yanqing Liu, Sheng Zhao, Ming Liu, Ming Zhou.

5. [WaveNet: A Generative Model for Raw Audio](https://arxiv.org/pdf/1609.03499v2.pdf) by Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu.

6. [MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis](https://arxiv.org/pdf/1910.06711.pdf) by Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, Aaron Courville.

7. [Multi-band MelGAN: Faster Waveform Generation for High-Quality Text-to-Speech](https://arxiv.org/pdf/2005.05106.pdf) by Geng Yang, Shan Yang, Kai Liu, Peng Fang, Wei Chen, Lei Xie.

8. [WaveFlow: A Compact Flow-based Model for Raw Audio](https://arxiv.org/pdf/1912.01219.pdf) by Wei Ping, Kainan Peng, Kexin Zhao, Zhao Song.

9. [FastSpeech 2: Fast and High-Quality End-to-End Text to Speech](https://arxiv.org/pdf/2006.04558.pdf) by Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, Tie-Yan Liu.

10. [Deep Voice 3: Scaling Text-to-Speech with Convolutional Sequence Learning](https://arxiv.org/pdf/1710.07654.pdf) by Wei Ping, Kainan Peng, Andrew Gibiansky, Sercan O. Arik, Ajay Kannan, Sharan Narang, Jonathan Raiman, John Miller.

11. [Hierarchical Prosody Modeling for Non-Autoregressive Speech Synthesis](https://arxiv.org/pdf/2011.06465v1.pdf) by Chung-Ming Chien, Hung-yi Lee.

12. [Unified Mandarin TTS Front-end Based on Distilled BERT Model](https://arxiv.org/pdf/2012.15404.pdf) by Yang Zhang,Liqun Deng and Yasheng Wang.

13. [FeatherWave:An efficient high-fidelity neural vocoder with multi-band linear prediction](http://yqli.tech/pdf/tts_paper/FeatherWave%20An%20efficient%20high%20fidelity%20neural%20vocoder%20with%20multiband%20linear%20prediction.pdf) by Qiao Tian,Zewang Zhang,Heng Lu ,Ling-Hui Chen and Shan Liu

1. [An Efficient Subband Linear Prediction for LPCNet-based Neural Synthesis](https://indico2.conference4me.psnc.pl/event/35/contributions/3023/attachments/694/732/Thu-1-1-6.pdf) by Yang Cui,Xi Wang,Lei He and Frank K.Soong.

15.[Gaussian Lpcnet for multsample speech synthesis](http://yqli.tech/pdf/tts_paper/GAUSSIAN%20LPCNET%20FOR%20MULTISAMPLE%20SPEECH%20SYNTHESIS.pdf) by Vadim Popov ,Mikhail Kudinov and Tasnima Sadekova.

16.[One Model,Many Languages:Meta-learning for Mutilingual Text-to-Speech](http://yqli.tech/pdf/tts_paper/2020%20One%20Model%20Many%20Languages%20%20Meta%20learning%20for%20Multilingual%20Text%20to%20Speech.pdf) by Tomas Nekvinda and Ondrej Dusek.

17.[Flowtron: an Autoregressive Flow-based Generative Network for Text-to-Speech Synthesis](https://nv-adlr.github.io/Flowtron) by Rafael Valle, Kevin shih, Ryan Prenger and Bryan Catanzaro.

18.[Char2Wav:End-toEnd Speech Synthesis](http://www.josesotelo.com/speechsynthesis/) by Jose Sotelo,Soroush Mehri,Kundan Kumar, Joao Felipe Santos ,Kyle Kastner ,Aaron Courville and Yoshua Bengio.

19.[Audio samples from “Fully-hierarchical Fine-grained Prosody Modelling for Interpretable Speech Synthesis](https://google.github.io/tacotron/publications/hierarchical_prosody/index.html) by Guangzhi Sun,Yu Zhang,Ron J. Weiss,Yuan Cao,Heiga Zen and Yonghui Wu

1. [DiffGan-TTS: High-Fidelity and Efficient Text-to-Speech with Denoising Diffusion GANs](https://arxiv.org/pdf/2201.11972.pdf) by Songxiang Liu,Dan Su and Dong Yu.

21.语音合成论文合集 <http://yqli.tech/page/tts_paper.html>

22. [A Survey on Neural Speech Synthesis](https://arxiv.org/pdf/2106.15561.pdf) by Xu Tan∗ , Tao Qin, Frank Soong, Tie-Yan Liu {xuta,taoqin,frankkps,tyliu}@microsoft.com Microsoft Research Asia

## 五、 申报单位研究基础

**4.1 单位概况**

**4.2 承担单位分工**

**4.3 项目主要负责人和研究人员情况**

## 六、进度安排

**1、2022年2月11日-2022年3月31日**

完成模型的测试和结果评估，确定合适的模型进行下一步开发。

**2、2022年4月1日-2022年6月30日**

模型的改进和训练，使满足项目需求

1）探索低数据量样本指导的语音合成算法，实现采集用户一个小时内语音语料样本,进行用户声音特点的语音合成能力；

2）语音合成时同步提供中文音素序列的生成结果，以json或其他格式返回；

**3、2022年7月1日-2022年7月31日**

项目测试与试运行阶段。与马栏山计算媒体研究院对接，完成部署文档等。

## 七、项目实施机制、保障措施及风险分析

包括项目的内部组织管理方式，协调机制等，保障项目实施的政策、组织和资源等，预期的经济、社会、环境效益分析，技术、市场、管理等风险分析及其预防措施等

## 八、 其他需说明的问题

1. 图片来源：Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, Yonghui Wu, ”Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions,”in Proc. *ICASSP*， 2018. [↑](#footnote-ref-1)
2. 图片来源https://github.com/Rayhane-mamah/Tacotron-2 [↑](#footnote-ref-2)
3. 图片来源：Ryan Prenger, Rafael Valle, Bryan Catanzaro NVIDIA Corporation *WAVEGLOW: A FLOW-BASED GENERATIVE NETWORK FOR SPEECH SYNTHESIS* [↑](#footnote-ref-3)
4. 4图片来源：Naihan Li, Shujie Liu, Yanqing Liu, Sheng Zhao, Ming Liu, Ming Zhou, “Neural Speech Synthesis with Transformer Network,”in Proc.AAAI,2018. [↑](#footnote-ref-4)