

学校代码: 10126

学号: 21609006

分 类 号: _____

编 号: _____

内蒙古大学

论文题目

基于深度学习的蒙古语语音合成研究

学 院: 计算机学院

专 业: 计算机应用技术

研究方向: 智能信息处理

姓 名: 刘瑞

指导教师: 高光来 教授

2020 年 4 月 27 日

原创性声明

本人声明：所呈交的学位论文是本人在导师的指导下进行的研究工作及取得的研究成果。除本文已经注明引用的内容外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得内蒙古大学及其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名: 刘端
日 期: 2020.4.21

指导教师签名: 高峰
日 期: 2020.4.30

在学期间研究成果使用承诺书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：内蒙古大学有权将学位论文的全部内容或部分保留并向国家有关机构、部门送交学位论文的复印件和磁盘，允许编入有关数据库进行检索，也可以采用影印、缩印或其他复制手段保存、汇编学位论文。为保护学院和导师的知识产权，作者在学期间取得的研究成果属于内蒙古大学。作者今后使用涉及在学期间主要研究内容或研究成果，须征得内蒙古大学就读期间导师的同意；若用于发表论文，版权单位必须署名为内蒙古大学方可投稿或公开发表。

学位论文作者签名: 刘端
日 期: 2020.4.21

指导教师签名: 高峰
日 期: 2020.4.30

基于深度学习的蒙古语语音合成研究

摘要

语音合成解决的主要问题就是如何将文字信息转化为可听的声音信息，它涉及声学、语言学、数字信号处理、计算机科学等多个学科技术，可广泛应用于智能家居、虚拟主播、语音导航、信息播报、阅读教育、泛娱乐等领域，是人机交互的重要组成部分。近年来，越来越多的研究人员使用深度学习技术对蒙古语智能信息处理相关问题展开深入研究。得益于深度学习模型强大的建模能力，蒙古语语音合成的整体质量得到了显著提升。但是，与汉语、英语等主流语种的语音合成技术相比，蒙古语语音合成研究还有很大的探索空间，要想满足合成语音质量的实用需求，还需要更进一步的深入研究。当前蒙古语语音合成系统与真实语音相比，自然度和表现力还是明显不足，主要表现在：韵律节奏缺乏表现力，合成语音音质不够高。其中，韵律建模和声学建模能力的不足是导致这些问题的主要原因。为了提高蒙古语语音合成系统的整体合成表现，本文从基于深度学习的蒙古文韵律建模和声学建模两个方面开展研究工作。在蒙古文韵律建模方面，采用深度学习技术并充分利用蒙古语语言特点和韵律建模相关任务的知识，提出了融合蒙古文形态学与音系学知识和基于多任务学习的蒙古文韵律建模方法；在声学建模方面，对端到端声学模型进行改进，提出了基于知识蒸馏和融合显式韵律信息的端到端声学建模方法。本文的创新点和主要贡献体现在以下几个方面：

1. 提出了融合蒙古文形态学和音系学知识的蒙古文韵律建模方法。为了提升蒙古语语音合成模型的整体韵律表现，采用循环神经网络对传统蒙古文韵律建模的模型输入和模型结构进行改进，提出了两种蒙古文韵律建

模方法。第一种方法为基于词素单元的蒙古文韵律建模方法，该方法将蒙古文单词切分转化为词素单元表示，以子词单元代替蒙古文整词作为建模单元进行韵律预测。第二种方法为融合形态向量和音系向量的蒙古文韵律建模方法，该方法将蒙古文单词向量与词素向量、音系向量一起输入蒙古文韵律模型以提高蒙古语韵律预测模型的精度。实验证明两种方法可以有效提高蒙古语语音合成的自然度。

2. 提出了基于多任务学习的蒙古文韵律建模方法。蒙古文韵律建模任务和蒙古文字母转音素任务具有天然的相关性，传统的蒙古文韵律建模方法没有考虑两者的关系，针对传统蒙古文韵律建模过程中缺乏相关任务信息指导的问题，该方法利用多任务学习机制，将蒙古文韵律建模与蒙古文字母转音素整合到同一个训练框架，通过两个任务联合训练的方式，可以提升蒙古文韵律建模的精度，进而提升蒙古语语音合成的韵律节奏表现。

3. 提出了基于知识蒸馏的端到端声学建模方法。针对端到端声学模型中解码器本身自回归性质的解码方式引起的曝光偏差问题，该方法采用“教师-学生”训练框架，首先训练使用真实语音参数作为解码器输入的教师模型，之后训练使用前一时刻预测得到的语音参数作为解码器输入的学生模型，在学生模型的训练过程中，通过知识蒸馏策略，使得学生模型同时学习到教师声学模型解码器输出的隐状态和真实的语音参数分布。实验证明该方法可以使得端到端声学模型产生更加稳定的声学参数，从而合成自然度更高的语音，并且很好的缓解了合成过程中跳词、漏词、重复等问题。

4. 提出了融合显式韵律信息的端到端声学建模方法。端到端声学模型对<文本，语音>的映射关系进行学习，但是其韵律建模过程被隐式的包含其中，使得模型在训练过程中缺乏显式的韵律信息的指导，从而限制了其自然度的提升。该方法分别从特征级别和模型级别将韵律信息融入到声学模型。特征级别韵律信息融合方法中，将韵律向量和字符向量进行融合后

输入端到端声学模型的文本编码器和声学解码器进行参数预测；模型级别韵律信息融合方法中，将韵律信息融入声学模型内部结构，具体来说，是将韵律生成器得到的韵律向量与文本编码器输出的字符向量进行融合后输入声学解码器，且韵律生成器与端到端声学模型联合训练。实验证明两种方法可以有效提升蒙古语语音合成模型的整体韵律表现。

综上所述，本文通过研究韵律和声学建模，使得蒙古语语音合成系统的性能达到可用水平，为蒙古语上游语音交互系统提供基础服务，对黏着语语音合成研究有一定的启示作用。同时，本文工作也将对促进蒙古文智能信息处理和少数民族地区的人工智能技术发展贡献力量。

关键词：蒙古语；语音合成；深度学习；韵律建模；声学建模

MONGOLIAN SPEECH SYNTHESIS BASED ON DEEP LEARNING

ABSTRACT

The main task of speech synthesis (or Text-to-Speech (TTS)) is to map an input text to a waveform file. It involves acoustics, linguistics, digital signal processing, computer science, and other subjects. This technology can be widely used in the smart home, virtual anchors, voice navigation, information broadcasting, education, pan-entertainment, and other fields and it is significant in human-computer interaction. Recently, researchers have shown an increased interest in applying deep learning technology to conduct in-depth research on Mongolian intelligent information processing related issues. Due to the powerful performance of the deep learning model, Mongolian TTS has gained a significant improvement. However, compared with other mainstream languages, such as Chinese and English, the performance of current Mongolian TTS is not mature enough, and meanwhile, the further deep research is necessary to meet the practical requirements about the quality of synthesized speech. Compared with the natural speech, the synthesized speech generated by the Mongolian TTS system is still far from perfect and lacks naturalness and expressiveness, for its poor performance of prosody and quality. And the reason mainly lies in the fact that there exists a certain deficiency in prosody modeling and acoustic modeling capability. To address these issues, this dissertation proposes some deep learning-based methods in terms of these two aspects. For the prosody model, we make full use of Mongolian knowledge and utilize multi-task learning skills to improve

the prosody model. For the acoustic model, we utilize knowledge distillation strategy and explicit prosodic knowledge to improve the end-to-end acoustic model. The innovation and main contributions of this dissertation are summarized as follows:

1. We proposed to combine Mongolian morphological and phonological knowledge to model the Mongolian prosody structure. In order to improve the prosodic performance of the Mongolian TTS model, this work proposed two prosody modeling methods incorporating Mongolian morphological and phonological knowledge. The first method called "morpheme units based Mongolian prosody model". This method transforms Mongolian words into morpheme units and then uses these morpheme units to predict the Mongolian prosody structure. The second method called "Mongolian prosody model using morphological and phonological embeddings", which takes Mongolian word embeddings, morphological and phonological embeddings as a joint input to improve the accuracy of the Mongolian prosody model. Experiment results show that these two methods can improve the prosodic performance of the Mongolian TTS model.
2. We proposed a Mongolian prosody modeling method with multi-task learning. Mongolian prosody model and the Mongolian grapheme-to-phoneme (G2P) have a natural correlation. The traditional Mongolian prosody model does not consider the relationship between these two tasks and ignore the relevant task information. To solve this problem, this method uses a multi-task learning mechanism to integrate the Mongolian prosody model and Mongolian G2P task into a unified training framework. Through the joint training of these two tasks, the accuracy of Mongolian prosody modeling can be improved. Experiment results show that this method can improve the naturalness of the Mongolian TTS system effectively.

3. We proposed a robust end-to-end acoustic model using a knowledge distillation strategy. To solve the exposure bias problem caused by the autoregressive decoding method of the decoder in the end-to-end acoustic model, this method uses the "teacher-student" training framework to train the acoustic model. We first train the teacher model that uses natural speech parameters as the decoder input called "teacher-forcing" decoding mode. Then we train the student model which takes the estimated speech parameters at the previous time step as the decoder input called "free-running" decoding mode. During the student model training process, the student model learns the decoder's hidden states of the teacher model and the natural speech parameter distribution at the same time through the knowledge distillation strategy. The experiment proves that this method can make the end-to-end acoustic model more stable and robust, and it can alleviate some problems such as the word skipping, missing, and repetition in the synthesis process.

4. We proposed an acoustic modeling method with explicit prosodic information guidance. The end-to-end acoustic model is designed to learn the mapping relationship of <text, speech>, but the prosody model is included implicitly, which makes the model lack the guidance of explicit prosodic information during the training process and may limit its prosody performance as well. This method integrates prosody information into the acoustic model using the feature-level and model-level strategy respectively. For the feature-level strategy, a pre-trained prosody generator was used to obtain the prosody embeddings. Then we combine the prosody embeddings and character embeddings together to feed the text encoder and acoustic decoder. For the model-level strategy, we first use a prosody generator to obtain the prosody embeddings, then we use text encoder to obtain the high-level character embeddings. At last, these two embeddings are concatenated together into a

single feature representation to feed the acoustic decoder. Note that the prosody generator was trained jointly with the acoustic model. Experiment results show that these two methods can improve the overall performance of the end-to-end Mongolian TTS model effectively.

In summary, these proposed methods involved in this dissertation are feasible to improve the prosody model and acoustic model and let the Mongolian TTS system meet the practical requirements. It also sheds new light on future speech synthesis research about other agglutinative languages. At the same time, these works in this dissertation also contribute to the promotion of Mongolian intelligent information processing and the development of artificial intelligence technology in minority areas of China.

KEYWORDS: Mongolian; Speech synthesis; Deep Learning; Prosody Modelling; Acoustic Modelling

目 录

摘要	I
ABSTRACT	IV
目录	VIII
图目录	XII
表目录	XV
第一章 绪论	1
1.1 研究背景与意义	1
1.2 语音合成研究概述	2
1.2.1 传统语音合成方法	2
1.2.2 统计参数语音合成方法	3
1.2.3 深度学习语音合成方法	7
1.3 蒙古语语音合成研究现状	11
1.4 本文研究内容与创新点	13
1.5 论文结构安排	14
第二章 深度学习蒙古语语音合成系统	16
2.1 蒙古语语言特点	16
2.2 蒙古语语音合成语音库的建立	19
2.3 基于深度神经网络的蒙古语语音合成系统	22
2.3.1 前端模块	22
2.3.2 后端模块	29
2.4 实验	31
2.4.1 实验配置	31
2.4.2 实验设计	32
2.4.3 实验结果与分析	33
2.5 本章小结	35
第三章 融合蒙古文形态学与音系学知识的韵律建模方法	36

3.1 引言	36
3.2 相关技术	37
3.2.1 循环神经网络	37
3.2.2 词向量表示	38
3.2.3 子词向量表示	40
3.3 基于词素单元的蒙古文韵律建模方法	42
3.3.1 蒙古文词素向量	43
3.3.2 LSTM 韵律模型	45
3.4 融合形态向量和音系向量的蒙古文韵律建模方法	47
3.4.1 蒙古文形态向量与音系向量	48
3.4.2 BiLSTM 韵律模型	50
3.5 实验	52
3.5.1 实验配置	52
3.5.2 建模单元的比较	53
3.5.3 不同向量融合效果的比较	56
3.5.4 注意力层有效性的验证	57
3.5.5 不同输出层的比较	57
3.5.6 对集外词鲁棒性效果的验证	58
3.5.7 主观测听实验	60
3.6 本章小结	62
第四章 基于多任务学习的蒙古文韵律建模方法	63
4.1 引言	63
4.2 相关技术	63
4.2.1 “编码器-解码器”网络	63
4.2.2 多任务学习	66
4.3 基于多任务学习的蒙古文韵律建模方法	68
4.3.1 文本编码器	68
4.3.2 韵律解码器	70

4.3.3 音素解码器	70
4.4 实验	71
4.4.1 实验配置	71
4.4.2 实验设计	72
4.4.3 实验结果与分析	73
4.5 本章小结	77
第五章 基于知识蒸馏的端到端声学建模方法	78
5.1 引言	78
5.2 端到端声学建模方法	79
5.3 基于知识蒸馏的端到端声学建模方法	82
5.3.1 教师模型	83
5.3.2 学生模型	84
5.3.3 模型训练	84
5.4 实验	86
5.4.1 实验配置	86
5.4.2 实验设计	87
5.4.3 实验结果与分析	88
5.5 本章小结	91
第六章 融合显式韵律信息的端到端声学建模方法	92
6.1 引言	92
6.2 融合显式韵律信息的端到端声学建模方法	92
6.2.1 特征级别融合方法	93
6.2.2 模型级别融合方法	95
6.3 实验	97
6.3.1 实验配置	97
6.3.2 实验设计	97
6.3.3 实验结果与分析	98
6.4 本章小结	104

第七章 总结与展望	105
7.1 本文工作总结	105
7.2 未来工作展望	106
参考文献	108
致谢	120
攻读学位期间发表的学术论文	122
攻读学位期间参加的科研项目	124

图目录

图 1.1 基于统计参数模型的语音合成方法的基本框架	3
图 1.2 基于 HMM 统计参数的语音合成模型训练流程	5
图 1.3 基于神经网络声学模型的语音合成模型基本框架	9
图 1.4 基于端到端声学模型的语音合成模型基本框架	10
图 2.1 蒙古文单词构成示意图	16
图 2.2 蒙古文（拉丁表示）不同层次单元的表示形式比较	17
图 2.3 蒙古文韵律结构示意图	18
图 2.4 数据标注界面	20
图 2.5 TextGrid 文件格式示意图	21
图 2.6 基于深度神经网络的蒙古语语音合成模型结构图	22
图 2.7 基于规则的蒙古文字母转音素方法流程图	24
图 2.8 蒙古文单词序列的韵律短语标签示例	25
图 2.9 蒙古文文本处理流程图	27
图 2.10 声学参数构成示意图	30
图 2.11 蒙古语多种结构 DNN 模型与最优 HMM 基线系统的主观 MOS 评测结果 （置信度 95%）	34
图 3.1 循环神经网络结构示意图	38
图 3.2 长短时记忆循环神经网络门控机制示意图	38
图 3.3 CBOW 模型结构图	40
图 3.4 Skip-Gram 模型结构图	40
图 3.5 CNN 字符向量学习模型结构图	41
图 3.6 BiLSTM 字符向量学习模型结构图	42
图 3.7 蒙古文词素单元的韵律短语标签示例	42
图 3.8 基于词素单元的蒙古文韵律建模框架	43
图 3.9 融合形态向量和音系向量的蒙古文韵律建模框架	47
图 3.10 基于 BiLSTM 的蒙古文形态向量训练框架	48

图 3.11 基于 BiLSTM 的蒙古文音系向量训练框架.....	49
图 3.12 采用“BES”格式的标注文件示意图	52
图 3.13 蒙古语 DNN、DNN-LEB 与 DNN-WMP 系统主观 MOS 评测结果（置信度 95%）	61
图 3.14 蒙古语 DNN、DNN-LEB 与 DNN-WMP 系统 A/B 倾向性测试结果（置信度 95%）	61
图 4.1 “编码器-解码器”网络结构示意图	64
图 4.2 基于注意力机制的“编码器-解码器”网络结构图	65
图 4.3 参数硬共享机制示意图	67
图 4.4 参数软共享机制示意图	67
图 4.5 基于多任务学习的蒙古文韵律建模框架	68
图 4.6 蒙古语 DNN、DNN-WMP 与 DNN-MTL 系统主观 MOS 评测结果（置信度 95%）	76
图 4.7 蒙古语 DNN、DNN-WMP 与 DNN-MTL 系统 A/B 倾向性测试结果（置信度 95%）	76
图 5.1 “教师激励”与“自由运行”解码模式示意图	78
图 5.2 基于端到端声学建模的的蒙古语语音合成模型框架图	80
图 5.3 基于知识蒸馏的端到端声学建模框架图	83
图 5.4 蒙古语 End2End 与 End2End-KD 系统主观 MOS 评测结果（置信度 95%）	88
图 5.5 蒙古语 End2End 与 End2End-KD 系统 A/B 倾向性测试结果（置信度 95%）	89
图 5.6 汉语和英语 End2End 与 End2End-KD 系统主观 MOS 评测结果（置信度 95%）	90
图 5.7 汉语和英语 End2End 与 End2End-KD 系统 A/B 倾向性测试结果（置信度 95%）	90
图 6.1 特征级别韵律信息融合方法示意图	93
图 6.2 韵律向量上采样操作示意图	94

图 6.3 模型级别韵律信息融合方法示意图	95
图 6.4 蒙古语 End2End、End2End-PE 与 End2End-MTL 系统主观 MOS 评测结果 (置信度 95%)	99
图 6.5 蒙古语 End2End、End2End-PE 与 End2End-MTL 系统 A/B 倾向性测试结果 (置信度 95%)	99
图 6.6 蒙古语 End2End、End2End-PE 与 End2End-MTL 系统对不同长度文本韵律 建模的 A/B 倾向性测试实验结果 (置信度 95%)	101
图 6.7 汉语 End2End、End2End-PE 与 End2End-MTL 系统主观 MOS 评测结果 (置信度 95%)	102
图 6.8 汉语 End2End、End2End-PE 与 End2End-MTL 系统 A/B 倾向性测试结果 (置信度 95%)	103
图 6.9 汉语 End2End、End2End-PE 与 End2End-MTL 系统 A/B 倾向性测试结果 (置信度 95%)	103

表目录

表 2.1 蒙古文字母及其拉丁字母对照表	23
表 2.2 部分特殊字符与其蒙古文形式对照表	24
表 2.3 蒙古语音素集	24
表 2.4 蒙古语上下文特征集	28
表 2.5 语音合成主观评价 MOS 评分标准	32
表 2.6 不同结构 DNN 蒙古语语音合成模型的客观评价结果	33
表 3.1 不同条件下基于 CRF 的蒙古文韵律建模的 F 值比较	55
表 3.2 不同条件下基于 CRF 且以向量为输入的蒙古文韵律建模的 F 值比较	55
表 3.3 不同条件下基于 LSTM 的蒙古文韵律建模的 F 值比较	55
表 3.4 不同向量融合效果的比较结果	57
表 3.5 注意力层有效性的验证结果	57
表 3.6 不同输出层的比较结果	57
表 3.7 对集外词鲁棒性效果的验证结果	58
表 3.8 不同系统预测效果的举例说明	60
表 4.1 基于多任务学习的蒙古文韵律建模方法的有效性验证结果	73
表 4.2 基于多任务学习的蒙古文韵律建模方法的消融实验结果	74
表 5.1 蒙古语 End2End 与 End2End-KD 系统词错误率比较结果	89
表 5.2 汉语和英语 End2End 与 End2End-KD 系统词错误率比较结果	90
表 6.1 蒙古文韵律生成器在 End2End-PE 与 End2End-MTL 系统的表现比较	101
表 6.2 汉语韵律生成器在 End2End-PE 与 End2End-MTL 系统的表现比较	103

第一章 绪论

1.1 研究背景与意义

随着互联网技术、信息处理技术和人工智能技术等领域的蓬勃发展，越来越多的新兴交互方式出现在了大众视野，人们享受着计算机、智能手机、平板电脑、汽车导航等智能终端设备带来的前所未有的便捷性。传统的人机交互方式多以键盘输入、屏幕输出为主，这样的交互方式显然在灵活性和自然性上具有明显不足。为了使人机交互方式更接近于人与人之间的无障碍交流方式，研究人员致力于让计算机实现人类“能听会说”的高级功能。因此，以语音识别、语音合成、自然语言理解为基础的新一代人机语音交互技术给人们带来了更加自然的交互体验。语音合成技术作为其中的核心组成部分在研究和技术应用角度都取得了飞速发展，再加上研究人员对语音学、语言学、统计建模技术、深度学习技术等相关技术的深入研究，语音合成逐渐成为了人机语音交互的研究热点。

语音合成解决的主要问题就是如何将文字信息转化为可听的声音信息^[1]，它涉及声学、语言学、数字信号处理、计算机科学等多个学科技术，可广泛应用于智能家居、虚拟主播、语音导航、信息播报、阅读教育、泛娱乐等领域，是人机交互的重要组成部分。

虽然语音合成技术已经发展相对成熟，应用也越来越广泛，但是目前语音合成技术的整体表现与真实语音相比还存在一定的差距，不管是英语、汉语等大语种（Rich-resource Language），还是其它小语种（也叫“低资源语言”， Low-resource Language），合成语音和真实语音可以被人耳很轻易的分辨出来。其原因主要源于两个方面：（1）语音合成前端模块中韵律建模效果不理想：韵律模型得到的韵律特征精度不够高，不能准确的反映出语音的停顿、快慢等信息，如果产生错误的停顿甚至会造成语义表达的错误；（2）语音合成后端模块中声学建模效果不理想：声学模型预测得到的声学参数与真实声学参数有一定差距，其合成语音因此不够清晰，错误的声学参数甚至也会造成不可预知的语义表达错误。

蒙古语作为小语种中的一员，其语音合成研究同样面临以上两个方面的问题，另外

又限于蒙古语独特的黏着语语言特点，导致其在这两个方面表现出的问题更加严重。蒙古语作为中国的少数民族语言之一，有着悠久的历史和丰厚的底蕴。它的使用人群分布在当今世界各地，包括内蒙古自治区、甘肃、西藏自治区等中国八省区，蒙古国及俄罗斯等世界不同地区。随着研究学者对文字智能信息处理研究的不断深入，蒙古语智能信息处理相关问题也受到越来越多的关注。经过研究人员的不懈努力，蒙古语语音合成技术取得了长足发展，但是相对于主流语种相关研究，蒙古语语音合成技术仍不够成熟，语音合成的前端模块和后端模块还有很多问题亟待解决。针对语音合成研究中面临的两个问题，结合蒙古语语言特点等实际情况，本文从蒙古语语音合成的韵律建模和声学建模两个方面出发，使用深度学习技术对蒙古语语音合成的整体表现进行全面优化，从研究和技术应用角度进一步推动蒙古语智能信息处理的发展，同时也为少数民族地区的人工智能技术发展贡献力量。

本章首先概述语音合成技术的发展历程，包括传统语音合成方法、统计参数语音合成方法和深度学习语音合成方法。之后对蒙古语语音合成研究的现状进行介绍，然后引出本文的研究内容与出发点。最后说明本论文的结构安排。

1.2 语音合成研究概述

1.2.1 传统语音合成方法

早期的语音合成技术通过模拟人类的发声机理来达到合成语音的目的。早在 1779 年，学者 Kratzenstein 就利用这一思路实现了一种机械式语音合成器^[2]，该合成器使用风箱、簧片和用皮革制成的共振腔来分别模拟人类的肺、声带和声道。具体的，它通过控制共振腔的形状来实现不同的元音的发音。这种合成器原理直观，对语音合成技术的研究具有一定的启发作用。但是受限于其有限的发音功能，这种合成器并不具有很大的实际意义。为了更好的模拟人类发声的激励系统，研究人员提出了电子式合成器。它是由贝尔实验室于 1939 年首先提出^[3]，该合成器更多关注的是对产生清音和浊音的激励系统进行模拟，而不是对具体的生理器官进行模拟。具体的，它包括脉冲发射器和噪声发射器，合成语音时，通过对多个带通滤波器进行手动控制来模拟声道系统，最后通过放大器输出语音信号。该合成器进一步推动了语音合成技术的发展，但是由于它包含

有限个带通滤波器，因此对自然语音的频谱特征刻画能力有限。为了对声道系统进行更加精细的刻画，1953年，Lawrence等人提出另一种声道系统模拟语音合成方法，叫做共振峰参数合成器^[4,5]，该合成器利用共振峰频率和宽度等声道的谐振特性来构建声道滤波器，将声道系统视作一个谐振腔，对语音的声道特性进行更好的刻画。然而，由于该语音合成方法涉及大量实验参数需要进行人工手动调整来测试，实用性大打折扣，难以在实际场景部署。上世纪90年代，随着计算机运算能力的大幅提升，基于波形拼接的语音合成技术^[6]应运而生，也叫做单元挑选语音合成方法。基于波形拼接的语音合成系统的基本原理是首先录制并标注一个大的音库，然后通过对待合成文本进行文本分析，根据分析结果从音库中挑选出合适的声学单元，最后通过最小化拼接代价将挑选出的声学单元进行拼接后得到最终合成语音。早期的波形拼接语音合成系统由于受到音库大小、挑选代价计算方法、拼接代价计算方法等的限制，合成语音并没有达到很高的质量。1990年，研究人员为了提高波形拼接语音合成的质量，使用基于时域同步叠加波形修改算法（Pitch Synchronous Overlap Add, PSOLA）^[6]对波形拼接语音合成进行改进。波形拼接语音合成的声学单元拼接处不平滑问题得到了很大的缓解。更进一步，早期的波形拼接语音合成又在大数据的支撑下，逐渐发展为基于大语料库的单元挑选语音合成方法^[8,13]，由于该合成方法生成的语音直接来源于音库中存储的真实语音片段，因此其合成语音的音质与真实的自然语音相比很接近。但是由于其合成质量直接取决于音库的质量和规模，因此该方法的灵活性、扩展性受到很大制约。

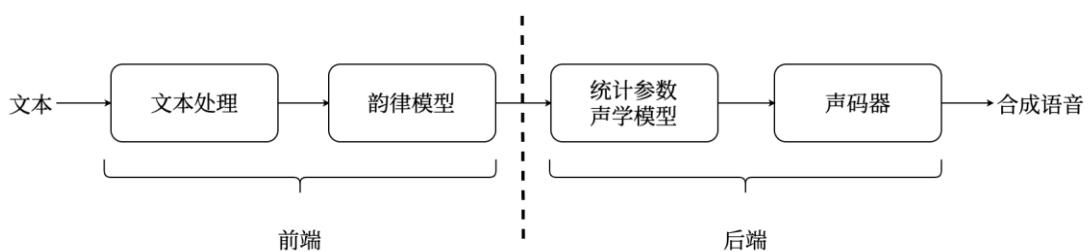


图 1.1 基于统计参数模型的语音合成方法的基本框架

Fig. 1.1 The schematic diagram of statistical parametric speech synthesis model.

1.2.2 统计参数语音合成方法

随着统计建模理论的完善^[13]，以及语音信号处理等相关学科的发展，研究人员提出了灵活性更强的基于统计参数的语音合成方法^[15-17]。基于统计参数的语音合成方法按照

功能划分可以分为两个模块，分别是前端模块和后端模块，其基本框架如图 1.1 所示。前端模块包括文本处理和韵律模型^[18,19]，文本处理的主要工作是将输入文本处理为规范的文字序列，通常包括文本正则化、自动分词、单词转音素等；韵律模型的主要工作是从语法语义层面挖掘文本的超音段特征，对文字序列中的韵律结构进行建模。最终前端模块输出文本对应的包含韵律特征的语言学信息，送入后端模块进行合成。后端模块包括声学模型和声码器，声学模型根据前端模块得到的语言学信息对声学参数进行预测，最后声码器将预测得到的声学参数转换为最终的语音波形。

对于前端模块，韵律结构生成是其最终目的。因此，韵律模型是整个前端模块的关键组成部分^[20]。早期韵律建模大多采用基于规则的方法对文本的韵律结构进行预测^[20-23]，研究者们基于对大量数据的分析和规则总结后，发现韵律结构的划分与文本的语法信息高度相关。但是由于规则的发现总结需要大量语种相关的语言学知识的储备，因此基于规则的韵律建模方法具有一定的局限性。为了摆脱基于规则方法的限制，研究者们开始将统计建模相关模型应用到韵律建模，提出了基于支持向量机（Support Vector Machines, SVM）、决策树模型以及最大熵模型（Maximum Entropy Model）、隐马尔可夫（Hidden Markov Model, HMM）和条件随机场（Condition Random Field, CRF）等方法^[24-32]。这些模型以词性（Part-of-Speech, POS）等特征为特征输入，从大量的训练数据中有监督的学习韵律结构知识，并最终实现韵律结构的自动预测。

对于后端模块，声学模型是实现语音合成的关键。基于 HMM^[33]的统计参数声学模型是统计参数语音合成模型中发展最为成熟的一种。HMM 模型已经在解决信号处理相关问题等领域中持续应用了将近四十年的时间，并且随着其广泛的应用，其理论基础也在不断完善和成熟，广泛的应用也进一步促进了其理论的发展。例如，在语音识别领域，研究人员成功将 HMM 理论与自动语音识别结合并且大获成功^[15]。自然而然的，研究人员将 HMM 模型引入语音合成问题的研究也取得了突破性进展。其中，最为成熟的案例就是基于 HMM 模型建立的统计参数语音合成声学建模方法^[33-37]。其原理与语音识别相反，在训练阶段，利用基于 HMM 的统计模型对输出语音的声学参数进行表征建模。在语音合成的推理阶段，输入任意待合成的文本序列，使用 HMM 声学模型对语音声学参数进行预测，最后声码器将预测得到的声学参数转换为最终的合成语音波形^[38]。

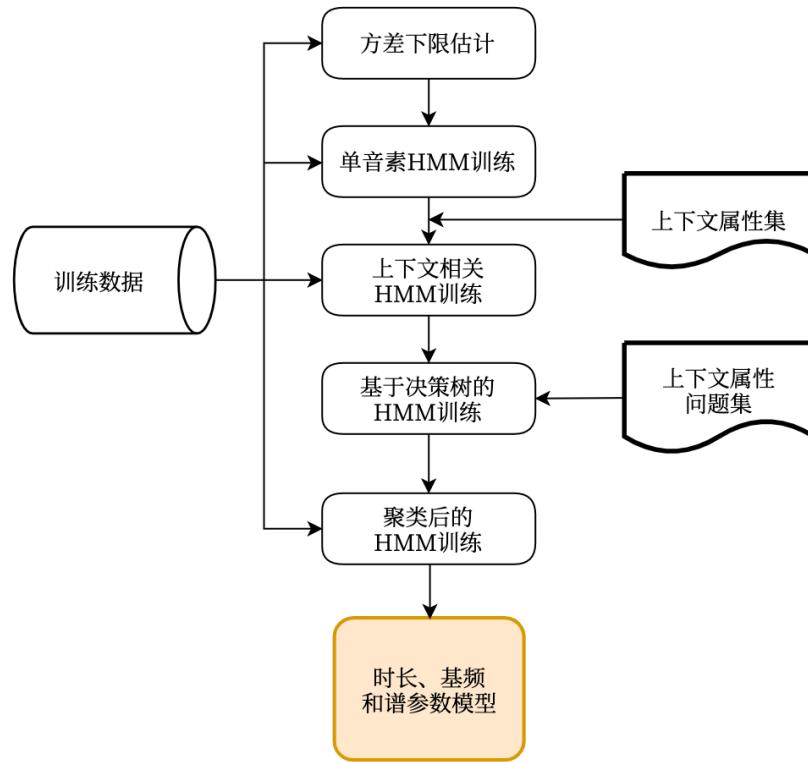


图 1.2 基于 HMM 统计参数的语音合成模型训练流程

Fig. 1.2 Training process of HMM-based TTS model.

基于 HMM 的统计参数语音合成模型能够同时对语音的基频、频谱和时长参数进行建模^[35]，生成连续、流畅且可懂度高的语音。整个 HMM 声学模型的训练流程如图 1.2 所示，模型训练步骤包括：（1）方差下限估计：由于训练数据的实际规模等原因，每个上下文相关模型不可能都有大量的训练数据与其对应，如果与其对应的训练数据非常稀少会导致模型方差的计算结果趋近于零。为了尽量减少类似情况出现，我们在模型训练之前先基于全部训练数据对全局方差进行计算，那么模型的方差下限可以通过对全局方差乘以一个系数（例如 0.05）获得。（2）单音素模型训练：模型的方差下限确定之后，需要先对所有单音素 HMM 的模型参数进行合理的初始化，初始化方法一般选择 K-Means 算法。初始化参数确定后，选择 EM 迭代算法对其进行训练并不断更新。（3）上下文相关模型训练：上一步中，单音素模型的模型参数通过不断迭代达到最优。这一步要将单音素模型进行扩展，以预先设计好的上下文属性集合为依据，最后同样选择 EM 迭代算法对扩展后的模型参数进行不断迭代和参数更新。（4）基于决策树的模型聚类：上下文相关模型的参数训练结束后，为了提高模型的鲁棒性以及为了平衡模型复

杂程度与训练数据规模之间的关系，这一步中，需要对上下文相关模型进行聚类。聚类方法选择基于决策树聚类的方法，聚类依据为预先设计好的语种相关上下文属性问题集，聚类准则采用最小描述距离（Minimal Description Length, MDL）准则^[39]。（5）聚类后模型训练：上下文模型聚类后的模型参数还需要进一步训练和更新，与之前训练方法相同，同样采用 EM 算法。不同的是，在参数更新的同时还要对各个状态模型的状态停留时间等信息进行统计计算并且输出。（6）时长建模：上下文相关模型的时长模型也是声学建模的一部分，其方法是根据上一步中统计得到的状态持续时间对时长模型的参数进行初始化，同样需要采用决策树的聚类方法实现时长模型的聚类。通过以上六个步骤的完整训练过程，最后我们得到的 HMM 声学模型包括：谱、基频和时长特征的聚类 HMM 模型，以及在模型训练过程中产生的各自的决策树。

统计参数语音合成方法与传统语音合成相比具有更好的灵活性和扩展性，采用该方法可以快速的构建可用的语音合成系统，并且全过程几乎不需要人工干预。更重要的是，统计参数语音合成模型可以在发音人、发音风格甚至语种等方面进行方便的扩展，具有很好的鲁棒性，因而被广泛应用。但是该方法对应的系统仍有若干不足与局限性：

（1）对数据的依赖性：和单元挑选与波形拼接方法对大规模音库的高度依赖性不同，统计参数语音合成方法所依赖的数据规模相对较少。但是其最终的合成语音质量也与训练数据的规模和质量有直接关系。另外，统计参数模型的非线性建模能力有限，对于语种相关的语音、语言学以及人的主观听觉感知经验等知识很难在这种方法中被充分学习和建模。

（2）合成语音的韵律平淡：合成语音缺乏丰富的表现力，韵律变化较为单一。前端模块中，文本处理只是对文本进行正则化处理以得到规范的文本序列，因此该问题主要来源于韵律模型。首先，统计参数语音合成中的韵律模型只是使用词性等词法特征为输入，而这些浅层的特征表示都是只对语法语义信息的简单符号化表示，无法表达深层次的语法语义知识，限制了韵律模型的建模能力；其次，现有的韵律建模方法多采用浅层机器学习模型。在韵律建模时，上下文的语义知识对当前词的韵律属性有很大的影响，而浅层模型无法对韵律相关的上下文语义属性进行充分建模，从而导致韵律建模结果的不准确。建模精度有限的韵律模型不能生成丰富的韵律结构信息，从而使得合成语音缺

乏表现力，当测听者收听表现力不强的合成语音时容易产生困乏感，如果类似的语音时间延长，那么测听者更是无法忍受。

(3) 合成语音音质不高：合成语音音质不够清晰，整体自然度欠佳。后端模块中，声码器只是接收声学模型输出的声学参数并将其转换为语音波形，因此该问题主要来源于声学模型。首先，基于 HMM 统计参数声学模型的语音合成方法在训练过程中需要对上下文相关模型进行聚类时，在聚类方法具体实现过程中，模型的输入声学特征空间被分为若干个类别。具体的，针对决策树聚类方法，决策树中的叶子节点就表示这些分类类别，他们用来表示对应声学参数轨迹上的一种模式。在语音合成推理阶段，如果模型中有限的叶子节点无法对现有预测得到的模式进行表示，那么会造成分类错误的情况发生。其次，待合成文本与其对应声学参数之间有着复杂的非线性映射关系，决策树无法很好表征此类复杂关系，在聚类过程中，有时会发生某些上下文相关属性不起作用的情况，从而难以在声学建模中发挥应有的作用。另外，由于统计参数模型是由前端模块和后端模块串联而成，模块的串联会导致误差随流水线不断积累。如果前端模块不能得到精确的语言学特征表达，那么串联结构会将前端模块的误差传递给后端模块。因此，限制了后端模块中声学模型的声学参数预测能力，从而产生细节表征不够精细的声学参数，进一步限制了合成语音音质的提高。

1.2.3 深度学习语音合成方法

近年来，GPU 已广泛应用于神经网络训练时的并行计算，大大提升了模型训练速度。另一方面，运算能力的增加也直接推动神经网络更进一步的发展，从而也使得机器学习领域的重要研究方向——深度学习（Deep Learning）^[40]更加广泛的应用。深度学习主要通过对多种非线性变换进行组合实现有监督或无监督的特征提取及转换、模式分析与识别等问题的机器学习方法。相对于 HMM、CRF 等浅层模型，“深度”模型具有更深的内部结构和更强的建模能力。浅层模型结构简单，具有较小的参数规模并且训练难度很小，因此在解决许多传统问题中多被采用且取得不错表现。然而对于另一些较为复杂的非线性建模任务来说难以保持其一贯优势，如处理语音合成、语音识别、语音翻译等问题时难以表现出很好的效果。随着深度学习技术在语音信号处理等任务中越来越广泛的应用，基于深度学习的语音合成方法凭借其出色的表现正逐渐取代统计参数语音

合成方法，目前已经成为语音合成领域的主流方法。本文将以上使用深度学习技术对统计参数语音合成中各个模块进行改进的语音合成方法统称为深度学习语音合成方法。一方面，针对前端模块，利用深层神经网络强大的非线性建模能力，从挖掘韵律相关的更深层次的语法语义信息和提升韵律模型的上下文深层次建模能力等角度，对统计参数语音合成方法中的文本处理、韵律建模进行改进^[41-47]；另一方面，在后端模块中，使用深度学习技术对统计参数语音合成方法中的声学模型等进行优化。研究者们提出使用神经网络声学模型代替 HMM 声学模型用于提高声学建模的精度^[58,48,49,50]，更进一步提出了端到端声学模型减少了前端模块对后端模块的负面影响，大大简化了声学建模流程^[51,52]。接下来对上述工作做详细介绍。

1.2.3.1 韵律建模改进

针对模型输入，传统的韵律模型只以词性等简单的符号化特征表示作为输入，对韵律结构进行预测。而符号化的特征表示只能表达浅层的语法语义信息，对深层次的韵律相关的语法语义信息表示不足。而随着深度学习技术的发展，表示学习（Representation Learning）^[53,54]为文本的语法语义信息的挖掘提供了一种全新的思路和方法。其中，词向量（Word Embedding）技术开始受到广泛关注。与以往离散化表示的符号特征相比，词向量技术可以将文本中的每个单词表示成一个定长的连续的实数向量，通过这样的表示可以将不同单词映射到同一个向量空间，从而用向量之间的距离来衡量单词之间的语义相关程度。单词的词向量表示具有更深层次的语法语义信息，使用词向量做韵律模型的输入可以更加充分的对深层次的语法语义信息进行特征描述。具体的，例如文献[41]提出了使用词向量的韵律模型，使用单词的词向量特征表示代替词性等特征从而取得了更好的韵律预测结果；文献[45,46,55]提出了语义表达更加丰富的词向量表示作为韵律模型的输入提高了韵律建模的精度。

针对模型结构，浅层的韵律模型建模能力有限，而深度学习中的深度神经网络模型凭借其更深的模型结构和强大的非线性建模能力在各种任务中大放异彩。相应的，针对韵律建模任务，文献[55-57]等工作相继提出了基于循环神经网络（Recurrent Neural Network, RNN）、长短时记忆循环神经网络（Long short-term memory Recurrent Neural Network, LSTM）和双向长短时记忆循环神经网络（Bi-directional LSTM Recurrent Neural

Network, BiLSTM) 等模型的韵律预测方法。这些模型可以对上下文语义进行充分的建模从而显著提升了韵律建模的精度。

1.2.3.2 声学建模改进

为了消除 HMM 声学模型在统计参数语音合成模型中的局限性，神经网络声学模型被用来作为 HMM 声学模型的替代，从而对文本到声学参数的条件概率密度函数进行建模^[58]。如图 1.3 所示为基于神经网络声学模型的语音合成系统的整体框架。

基于神经网络声学模型的语音合成方法将神经网络声学建模技术引入，成功取代之前的基于 HMM 的统计参数声学模型。前端模块对文本进行语言学信息和韵律特征编码后将其表示为文本特征向量。之后文本特征向量输入到神经网络声学模型，选择对应的声学特征作为神经网络声学模型的训练目标，通过神经网络强大的非线性运算能力很好的对文本与声学特征之间复杂非线性映射关系进行表征和刻画，并且避免了统计参数方法中的声学特征分类过程。

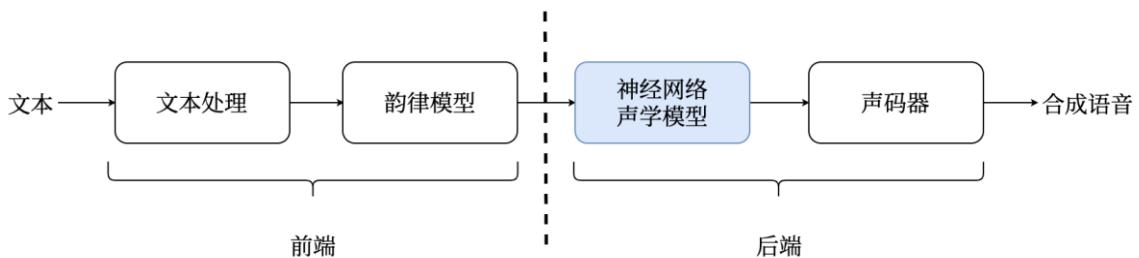


图 1.3 基于神经网络声学模型的语音合成模型基本框架

Fig. 1.3 The schematic diagram of neural network-based TTS model

具体地，H.Zen 提出了基于深度神经网络（Deep Neural Network, DNN）的语音合成方法^[48]。为了进一步提高统计参数语音合成中声学建模的精确性，更好的考虑上下文声学特征的相关性，Fan 提出了一种基于 BiLSTM 的语音合成声学模型^[59]，该模型充分考虑到当前帧与左侧、右侧双向输入帧的关系，提升了声学模型的整体表现。更进一步，研究者们结合 DNN、RNN、LSTM 等多种网络的优势对神经网络声学建模进行了更深入的研究^[60-67]。基于神经网络的语音合成模型通常采用最小均方误差（Minimum Mean Square Error, MMSE）训练准则，采用反向传播（Back propagation, BP）算法和随机梯度下降算法进行参数更新，使模型正向计算得到的声学参数尽可能接近训练数据中的真实声学参数。在语音合成推理阶段，首先输入任意给定的待合成文本，之后对其进行

文本特征向量计算，神经网络声学模型根据文本特征向量预测其对应的声学参数，最终的合成语音波形是由声码器对声学参数转换得到。基于神经网络声学模型的语音合成方法显著的提高了合成语音的整体语音自然度，但是该方法仍然面临若干问题。例如，前端模块需要计算文本特征向量，要想准确的对文本特征进行描述和刻画，较强的语言学背景知识必不可少。但是不同语种具有差异明显的语言学知识体系，系统构建时需要相应的语言学专家提供支持，无形中增加了系统的构建难度；另一方面，基于神经网络声学模型的语音合成方法与传统语音合成方法都是由一系列复杂的子模块串联构成，模块的串联导致前期工作的误差会随着流水线不断积累而放大，导致信息损失从而导致合成语音质量效果欠佳。为了解决上述问题，研究人员提出了全新的解决思路，基于端到端声学模型的语音合成方法应运而生。如图 1.4 所示为基于端到端声学模型的语音合成模型基本框架。

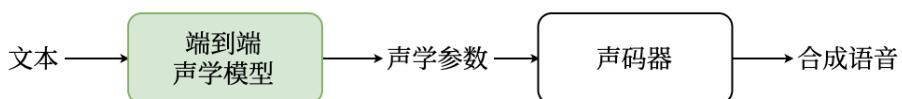


Fig. 1.4 The schematic diagram of end-to-end TTS model

研究者希望能够使用端到端建模技术使得语音合成系统的流水线建模过程尽量简化。端到端语音合成系统使用端到端声学建模方法，直接输入文本的字母表示或者音素表示，输出语音波形。使整个语音合成声学建模流程得到极大简化，其中前端模块甚至可以直接省略掉。端到端声学建模相比于之前的统计参数声学模型和神经网络声学模型，降低了对语言学知识的要求，具有更好的语种扩展性。借助于深度学习模型的强大表达能力，端到端声学建模可以生成接近真人发音的合成语音。具体的，文献[68]提出了基于 WaveNet 的语音合成方法，WaveNet 基于卷积神经网络实现，该卷积神经网络包含扩展卷积机制，采用自回归的解码方式对语音波形采样点之间的关系进行建模。在模型推理阶段，基于训练好的模型参数以历史样本点信息为输入，对当前样本点的信息进行估计，该方法可以产生接近原始音频的合成语音。但是由于其模型的自回归限制，导致其很难实现实时合成，从而限制了其实用性。另一方面，基于“编码器-解码器”框架

的端到端声学建模方法也取得突破性进展，如 Char2Wav^[69]、Tacotron（及其改进 Tacotron2 等）^[51,52]等。Char2Wav 由两部分组成，一个是读取器，一个是声码器。读取器是带有注意力机制（Attention Mechanism）的“编码器-解码器”模型，其中编码器采用双向循环神经网络对输入文本进行读取，带有注意力机制的解码器以编码器输出为输入，对声学特征进行预测并输出；声码器使用基于 SampleRNN^[68]的扩展方法生成语音样本。Char2Wav 是一个比较完整的端到端语音合成模型，但是它的训练过程需要将读取器和声码器单独训练，并没有实现充分的端到端。Tacotron^[51,52]语音合成方法的提出才真正意义上的体现了端到端声学建模的思想，它采用基于卷积神经网络、循环神经网络等结构的“编码器-解码器”框架可以直接学习文本字符序列到声学参数序列的映射关系，只需要输入待合成文本的字符序列，不需要复杂的文本分析步骤，大大简化了传统语音合成模型的流水线框架。其简洁的模型框架和高质量的合成语音受到业界一致认可并获得广泛应用。

1.3 蒙古语语音合成研究现状

蒙古语是内蒙古自治区的主体民族语言^[71]。在中国，蒙古语使用者遍布中国多个省份，包括内蒙古自治区、辽宁、青海、黑龙江、新疆维吾尔自治区、甘肃、吉林等省区。在世界范围内，除中国外，同样也有多个国家和地区将蒙古语作为其官方或主体语言，如蒙古国、俄罗斯的布里亚特共和国等。研究蒙古语语音合成技术，对中国乃至世界的教育、交通、文化、经济等方面具有重要意义^[72]。

针对蒙古语语音合成，已有一大批学者在早期开展了大量研究。敖其尔、巩政提出了一种波形拼接的蒙古语语音合成方法^[73]；高光来提出了以词为单位的波形拼接技术进一步对蒙古语语音合成方法展开研究^[74]；萨其容贵将基音同步叠加法引入并建立了多样板蒙古语语音合成音库^[75]；田会利提出了基于词干后缀的有限词条的蒙古语语音合成方法^[76]；孟和吉雅针对蒙古语动词词干词缀，提出了另一种基于词干后缀的蒙古语语音合成方法^[77]；敖敏从韵律角度出发，对蒙古语语音合成方法进行研究^[78]。近几年来，统计参数语音合成方法在英语、汉语等主流语种中取得了成功应用，其合成语音的整体表现已经与真人发音非常接近。随后，基于统计参数的蒙古语语音合成技术也相继被内蒙古地区相关研究人员提出。具体的，在韵律建模方面，李婷会等提出了基于 CRF

模型的蒙古文韵律建模方法^[79];文献[80,81]等进一步利用蒙古文单词词性等符号化特征表示提升了基于 CRF 模型的蒙古文韵律模型的性能。在声学建模方面,赵建东等提出了基于 HMM 声学模型的蒙古语语音合成的方法^[82,83],该方法首先构建了蒙古语语音语料库,结合蒙古语语言特点设计了上下文属性集以及相应于模型聚类的属性问题集,最后实现了基于 HMM 声学模型的蒙古语语音合成系统。

鉴于深度学习技术引入语音合成领域后的出色表现,一些学者开始将深度学习技术与蒙古语语音合成进行结合。针对蒙古文韵律建模,基于深度学习的蒙古文韵律建模研究还没有相关研究涉及,仍然以基于 CRF 模型的韵律模型为主。针对蒙古文声学建模,深度学习声学建模技术在蒙古语语音合成中取得了重要进展,具体的,文献[84]首次将深度学习技术引入蒙古语语音合成,使用基于 DNN 的声学模型代替 HMM 声学模型,进一步提升了蒙古语语音合成的整体表现;李劲东、刘致楠等在蒙古语语音合成中引入端到端声学建模方法^[85,86],端到端声学建模过程具有更加简洁明了和直接的建模过程,该方法进一步扩展了现有蒙古语语音语料库的规模并实现了当前最好的蒙古语语音合成效果。这些研究方法的提出,使得蒙古语语音合成技术不断深入,合成蒙古语语音的整体表现相较传统方法也获得了显著提升,但是其合成功音的韵律表现和合成音质与真实语音相比仍然具有较大差距,其韵律建模和声学建模部分仍有很多问题没有解决。

针对蒙古文韵律建模和声学建模,具体问题分析如下:

(1) 韵律模型语义建模能力不足:现有蒙古语系统在韵律表现上较为平淡、表现力不足。具体而言,韵律建模是前端模块的重要组成部分,由其得到的韵律结构特征是合成具有丰富表现力语音的关键因素。现有蒙古文韵律建模只依赖于蒙古文单词的符号化语义特征表示和 CRF 等浅层机器学习模型,严重制约了蒙古文韵律模型的语义特征挖掘能力,更重要的是,有限的蒙古文文本资源和蒙古文独特的黏着语特性导致的数据稀疏问题,限制了蒙古文前端韵律模型的建模能力,而数据稀疏问题会进一步导致训练数据中出现大量的集外词(Out of Vocabulary, OOV),也给蒙古文韵律建模带来很大挑战。

(2) 韵律建模缺乏相关任务的辅助:现有蒙古语语音合成系统在进行韵律建模时只使用单一模型从蒙古文单词序列及其语义特征中学习韵律结构。但是音素是蒙古文发音的基本单元,字母转音素任务(Grapheme to Phoneme, G2P)中对蒙古文单词发音的

音素序列进行预测，因此蒙古文单词的发音和韵律变化特征也被隐式包含其中。但是当前韵律模型并没有充分考虑两个相关联任务的深层次关系，也对韵律建模的精度产生一定影响。

(3) 声学建模鲁棒性不足：端到端声学建模方法会频繁出现跳词、漏词、重复等现象。具体而言，在推理阶段，基于端到端声学模型的蒙古语语音合成系统使用上一时间步预测的声学参数进行当前时间步的参数预测，这样的解码方式导致解码误差随着时间步的推移不断积累，造成后面解码序列的不准确，生成不准确的声学参数从而导致合成语音质量下降。

(4) 声学建模缺乏韵律信息显式指导：虽然基于端到端声学模型的蒙古语语音合成系统抛弃了复杂的蒙古文文本处理流程，可以直接学习蒙古文拉丁字符到蒙古语语音声学特征的直接映射。但是模型的韵律建模功能被隐式的包含在声学模型框架中，待合成文本的韵律信息并没有被显式建模，简单的字符表示不能表征复杂的韵律变化，因此限制了蒙古语语音合成自然度的提升。

本文对蒙古语语音合成中的韵律建模和声学建模两个方面进行深入研究，以期从前端模块和后端模块两个角度去全面提升蒙古语语音合成系统的整体表现。

1.4 本文研究内容与创新点

鉴于深度学习技术显著优于传统方法的建模能力，本文使用深度学习技术对现有蒙古语语音合成的前端模块和后端模块进行全面改进。本文围绕基于深度学习的蒙古语语音合成方法展开，以进一步全面提升蒙古语语音合成的整体表现为研究重点。结合蒙古语语言特点和深度学习相关知识，针对蒙古语语音合成中韵律建模和声学建模中发现的四个问题提出了相应的解决方案，具体研究方案分为以下四个方面：

(1) 融合蒙古文形态学与音系学知识的蒙古文韵律建模方法：为了从模型输入和模型结构两个角度对蒙古文韵律建模的建模能力进行增强，使得蒙古语深层次上下文语义特征被充分建模，从而达到提升蒙古语语音合成的合成效果的目标。使用 LSTM 网络和 BiLSTM 网络进行蒙古文韵律建模，并充分考虑蒙古文词干后缀、音节、音素等对蒙古语单词韵律发音的影响，提出了基于词素单元的蒙古文韵律建模方法和融合形态向

量和音系向量的蒙古文韵律建模方法,以提高蒙古文韵律建模在集内词和集外词的整体精度。

(2) 基于多任务学习的蒙古文韵律建模方法:为了进一步提升韵律建模的精度,结合蒙古文字母转音素任务与蒙古文韵律建模任务的天然高度相关性。将蒙古文韵律建模任务和蒙古文字母转音素任务整合到同一个框架,采用“编码器-解码器”网络构建多任务学习(Multi-Task Learning)系统,使得两个任务以联合训练的方式互相学习,以期提升蒙古文韵律建模的表现。近几年,多任务学习技术已经被成功应用于诸多领域中,如机器翻译、图像标注等,该方法可以提升蒙古语语音合成模型的合成自然度。

(3) 基于知识蒸馏的端到端声学建模方法:为了克服基于端到端声学模型的蒙古语语音合成系统的天然缺陷,考虑到端到端声学模型使用自回归属性的解码器预测语音声学参数,但是由于自回归解码器天然具有的曝光偏差(Exposure Bias)问题而导致合成语音中频繁出现跳词、漏词、重复等现象。本文提出了一种基于知识蒸馏技术的端到端声学建模方法,提升了蒙古语语音合成系统的鲁棒性和整体表现。

(4) 融合显式韵律信息的端到端声学建模方法:为了将显式韵律信息充分融入端到端声学模型的训练过程,本文提出了特征级别和模型级别两种韵律信息融入方法,通过使用文本的韵律信息对端到端声学模型的训练过程进行指导,以提升基于蒙古语语音合成的整体自然度。

1.5 论文结构安排

本文旨在进一步研究基于深度学习的语音合成技术在蒙古语语音合成中的应用,论文第二章介绍基线蒙古语语音合成系统,第三章和第四章针对韵律建模,第五章和第六章针对声学建模,具体结构安排如下:

在第二章中,详细介绍了深度学习蒙古语语音合成系统的实现过程。通过比较不同参数配置下的合成表现,最终采用最优参数搭建完成了深度学习蒙古语语音合成系统。

第三章介绍了融合蒙古文形态学与音系学知识的蒙古语韵律建模方法。利用神经网络强大的建模能力,并挖掘蒙古语的构词特点和音系知识提升蒙古文韵律模型的建模精度,提出了基于词素单元的蒙古文韵律建模方法和融合形态向量和音系向量的蒙古文建模方法,最后通过实验对比验证了方法的有效性。

第四章中，详细介绍了基于多任务学习的蒙古文韵律建模方法。通过将蒙古文韵律模型与蒙古文字母转音素两个任务联合训练，以进一步提升蒙古文韵律建模的性能表现，从而对最终的蒙古语语音合成系统起到积极地促进作用。通过实验验证了方法的有效性。

第五章中详细介绍了基于知识蒸馏的端到端声学建模方法。通过采用“教师-学生”训练框架可以很好的缓解曝光偏差问题对自回归声学解码器的影响，从而提升了蒙古语语音合成系统的合成语音自然度。通过详细的实验比较与分析证明了该方法的有效性。

第六章中详细介绍了融合显式韵律信息的端到端声学建模方法。通过采用特征级别和模型级别两种韵律信息融合方式，来将韵律信息显式融入端到端声学模型的训练过程，从而提升蒙古语语音合成系统的整体表现。通过完善的实验设置以及比较分析证明了该方法的有效性。

最后在第七章中，对全文内容进行总结，以及对蒙古语语音合成的未来工作进行展望。

第二章 深度学习蒙古语语音合成系统

本文详细介绍基于深度学习蒙古语语音合成系统的完整构建过程。这一章中，首先对蒙古语的语言特点进行归纳和总结，之后根据语音合成的实际数据要求，对蒙古语语音合成语音库的整理和建立进行详细介绍，然后介绍了深度学习蒙古语语音合成系统的具体框架。最后通过实验比较分析了不同参数配置下的性能优劣。这个深度学习蒙古语语音合成系统作为本文的基线系统，后续工作在这个语音合成系统上进行优化改进。

2.1 蒙古语语言特点

现行蒙古文拥有两种截然不同的书写系统，一种是西里尔蒙古文文字，另一种是传统蒙古文文字，中国的蒙古文使用者主要使用后者。传统蒙古文是一种拼音文字，以词为基本书写单位，单词间的间隔表示与英语相同采用空格表示，具有从上到下、从左到右、在主干线相连的独特的书写顺序。如无特别说明，本文中出现的“蒙古文”一词均指传统蒙古文。需要特别强调的是，蒙古语言文字由于其独特的黏着语特性给蒙古文文本处理带来很大挑战^[87]。下面我们将分别从形态学、音系学两个角度进行具体分析。

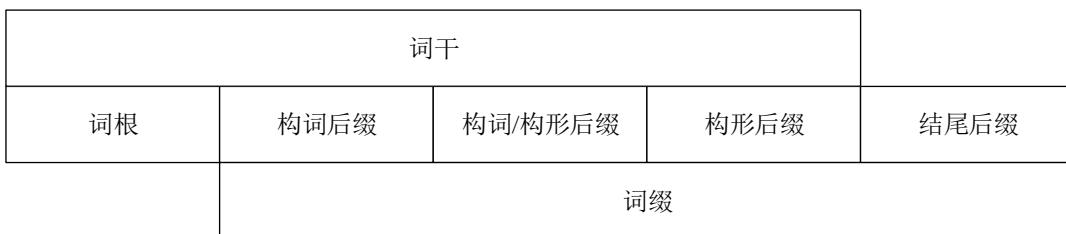


图 2.1 蒙古文单词构成示意图

Fig. 2.1 The schematic diagram of internal structure in Mongolian word

从形态学角度：蒙古文形态复杂，其构词方式独特且复杂，与汉语言文字相比具有很大的不同^[87]。汉语言文字在形态方面几乎不存在任何变化，其单词表示是由独立的字组成的，单词又进一步组成短语。蒙古文单词虽然也是由蒙古文字符直接拼接而成，但是与汉语相比其构词特点更加复杂，即蒙古文单词是通过在词根或者词干后连接后缀构造而成。如图 2.1 所示，蒙古文单词可以拆分解构为多个组成部分：包括词根、构词后缀、构形后缀和结尾后缀。这些子词（Subword）单元统称为“词素”。其中词根与连接在其后的构词后缀、构形后缀共同组成蒙古文词干。蒙古文词缀又可以进一步分类，

具体类别包括构词后缀、构形后缀和结尾后缀，这三种后缀以“前、中、后”的固定顺序组合在一起。但是他们的数量不是固定的，词根后可以添加若干个后缀形成派生词。若干个词根和与若干个后缀组合在一起形成的派生词又叫做复合词，复合词可以在句子中充当句子成分。因此，在一个蒙古文单词中，构词后缀和构形后缀的数量灵活，可以叠加若干个；结尾后缀的数量比较固定，一般只有一个。蒙古文词根是派生新词的基础，又是保存单词意义的基本单元。词根连接词缀不仅可以改变词的词性、词义，对于蒙古文句子中的上下文语境也会产生决定性的影响。值得注意的是，位于末尾的结尾后缀相较于其他形式后缀出现最为频繁，它仅表示语法含义，对单词的词汇意义不产生任何影响。

鉴于以上复杂的形态学特点，如果将所有的词干拼接不同的后缀可以构成近百万的蒙古文单词量，庞大的单词量带来严重的数据稀疏问题并且导致对大规模文本数据的严重依赖，从而给基于有限规模语料资源的蒙古文文本处理任务带来巨大挑战。

蒙古文文本	“homun-u bey_e-yin eregul qihirag-tv tvsalan_a”
单词表示	homun-u^bey_e-yin^eregul^qihirag-tv^tvsalan_a
音节表示	ho/mun/-u^be/y_e/-yin^e/re/gul^qi/hi/rag/-tv^tv/sa/la/n_a
词素表示	homun/-u^bey_e/-yin^eregul^qihirag/-tv^tvsalan_a
音素表示	h/o/m/os/n/-n/u^b/y/el/ll/n^e/r/ul/l^q/i/r/a/g/-t/vl^t/v/s/as1/l/n/al
字符表示	h/o/m/u/n/-u^b/e/y/_e/-y/i/n^e/r/e/g/u/l^q/i/h/i/r/a/g/-t/v^t/v/s/a/l/a/n/_a

图 2.2 蒙古文（拉丁表示）不同层次单元的表示形式比较

Fig. 2.2 Comparison of different linguistic levels of Mongolian word (Latin representation)

从音系学角度：音素是蒙古语发音的基本单元，蒙古语发音是由音素决定的，音素序列相比于字符序列能够更准确地表征发音信息^[88]。音节是由一个或几个音素组成的最小的语音片段，语音的节奏一般指语句中各音节的长短快慢。另外，音节单元和词干后缀一样，同样具有区别词义的功能。在生理发音上，音节的形成主要依赖于发音器官肌肉在一个张弛周期的生理变化，在物理音响上，主要体现为音强和响度从强到弱的变化过程。音节中的每一个音素都代表一个音段，通常来说，音节的构成是元音为主，辅

音为辅，辅音作为元音的辅助与元音组合在一起形成完整音节。因此，音节的主体是元音音素，围绕在其前或后的辅音音素可以表示音节的边界。通过判断音节末尾边界位置的音素类型可以将蒙古文音节分类两类，分别是开音节和闭音节。如果末尾音素是元音音素，那么该音节为开音节，如果末尾音素是辅音音素，那么该音节为闭音节。在蒙古语语音中，蒙古语元音的变化会导致整个连续语流的变化，因此这些复杂多变的语音变化会直接影响语音合成可懂度和自然度。

综上所述，根据蒙古文形态学及音系学知识，可以将蒙古文单词划分为词素序列、音节序列、音素序列和字符序列。图 2.2 展示了蒙古文（拉丁表示）不同层级表示单元的具体形式。

另外，同汉语、英语等语言一样，蒙古语韵律结构也存在层级划分^[78]。根据可感知的语音停顿层级和语音合成训练数据的韵律标注要求，蒙古语韵律结构可以划分为三层：语调短语、韵律短语、韵律词。因为小于韵律词的单位（如音节）没有特定的发音韵律表征，因此韵律词作为层级划分中最小的单元，其具有独立的韵律价值。下面我们将按照韵律词、韵律短语、语调短语的顺序依次进行介绍。

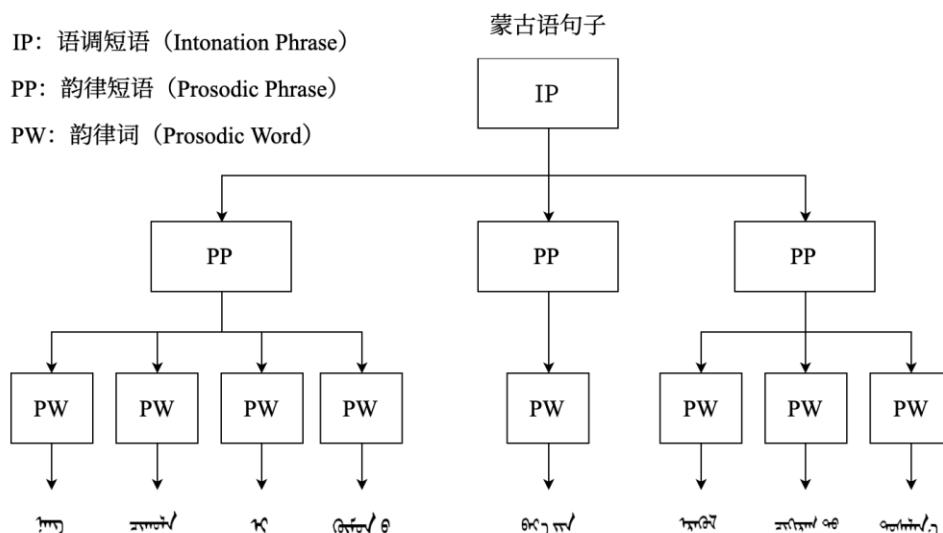


图 2.3 蒙古文韵律结构示意图

Fig. 2.3 The block diagram of Mongolian prosody structure

(1) 韵律词：由于蒙古文单词以空格作为间隔符，因此单词之间的界限明显，不需要进行额外的界限切分。韵律词内部不具有停顿，但是韵律词边界一般可以有短时停顿。

(2) 韵律短语：韵律短语一般由若干个韵律词构成，其内部的韵律词处于同一个节奏群，因此具有比较紧密的韵律关系。另外，构成韵律短语的韵律词之间通常不存在可感知的停顿，而在同级别的韵律短语之间可能会存在比较明显的短时停顿或长时停顿。

(3) 语调短语：语调短语是由若干个韵律短语构成。其一般与简单语法词组或复杂语法词组相对应。在句子中它可以承担一定的语法成分，例如：主语、谓语、宾语、定语或者状语。语调短语后一般伴随有比较明显的可感知的停顿。

以上三个层级韵律单元存在固定的从属关系，即语调短语边界一定是韵律短语边界，韵律短语边界一定是韵律词边界，并且韵律短语边界最终也与韵律词的边界重合。其树形结构如图 2.3 所示。

2.2 蒙古语语音合成语音库的建立

本节主要介绍蒙古语语音合成语音库的建立过程。该语音库包括两部分：一个是语料库，一个是语音库。语料库包含标准规范的蒙古文文本语料，语音库就是为训练蒙古语音合成声学模型而录制的语音数据。语音库的构建是进行基于深度学习的蒙古语语音合成方法研究的基础，其中包括语音库的准备、语音库的录制、语音库的标注等。

(1) 语音库准备：蒙古语语音合成语音库的准备是进行蒙古语语音合成研究的前提。语音库录制前需要准备对应的文本语料，该文本语料的设计要以尽可能覆盖所有的音素组合和发音现象为原则。语料来源包括：蒙古文人名、日常蒙古语对话文本材料、蒙古文新闻稿、蒙古文科普（文学、地理、历史）文字材料等。经过人工校正文字错误后，最终搜集整理了 2620 句蒙古语语音合成文本语料用于语音库的录制。

(2) 语音库录制：蒙古语语音合成语音库的录制是进行蒙古语语音合成研究的基础。在确定蒙古语语音合成语料库后，对其进行统一编号，分别从 0001 到 2620，作为发音人朗读的依据，邀请内蒙古大学蒙古学学院蒙古语播音主持专业的女大学生进行录制。录音时，对照蒙古语语音合成语料库，按照新闻播音标准风格进行朗读，语句之间要有明显的停顿。整个录音过程在内蒙古大学计算机学院标准录音室完成，确保蒙古语语音合成语音库的音质清晰。另外，录音过程中有监督人员进行监督，如果发生错读、漏读等现象则该蒙古文文本重新录制，在录音过程中进行监督校正可以很大程度减少蒙古语语音合成语音库的发音错误产生，从而减轻后期人工校正的负担。录音保存格式为：

采样率 44100Hz，采样精度 16-bit，单声道，wav 格式。最终录制完成时长约 2 小时的标准蒙古语语音合成都音库。

(3) 语音库标注：蒙古语语音合成都音库的标注是决定蒙古语语音合成整体表现的关键工作。数据标注过程选择 Praat 语音标注工具¹完成，标注后得到后缀名为 TextGrid 的数据文件。数据标注示例如图 2.4 所示。

图中从上到下依次为：第一层为待标记语音的语音波形，第二层为待标记语音的频谱特征图，第三层是音素标记层，第四层是韵律标记层。其中，第四层中 1 代表当前时间段内为一个蒙古语单词；3 代表当前时间段内单词是所在蒙古文韵律短语的最后一个单词，表示蒙古文韵律短语的边界；4 代表当前时间段的单词为整个待标记蒙古语语音的最后一个单词，表示蒙古语语音的末尾。

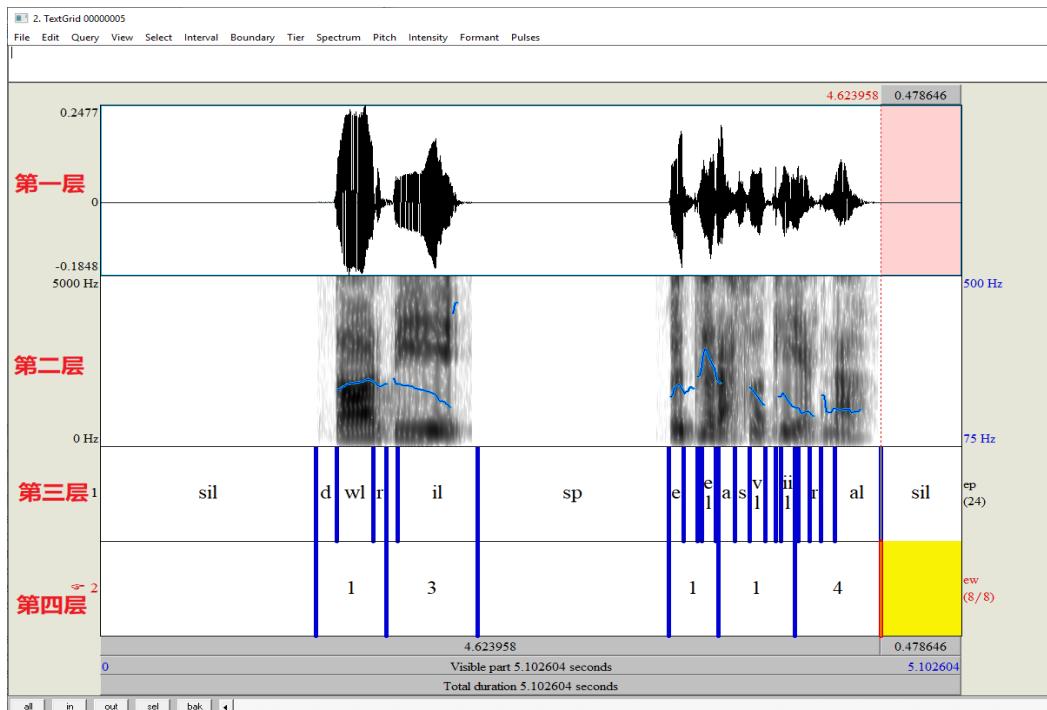


图 2.4 数据标注界面

Fig. 2.4 Interface of data labeling work

具体的，以蒙古语句子“**жийн төслийн мэдээллийн талаар**”（汉语意思：“下面讲这些问题”）为例，其音素序列为“d wl r b il e d g el r a s v l d l iil y ae r n al”，则其带有韵律标记的音

¹ <http://www.fon.hum.uva.nl/praat/>

素序列为“d wl r b il #3 e d g el r a s vl d l iil y ae r n al #4”。其 TextGrid 文件格式如图 2.5 所示。

本文邀请 10 名内蒙古大学蒙古学学院蒙古语母语本科生进行数据标注，并最终由 2 名内蒙古大学计算机学院本科生进行标注结果的检查与校正。最终整理了 2 小时的标注数据用于基于深度神经网络的蒙古语语音合成模型的训练，共包含文本数 2620 句。

```

1 File type = "ooTextFile"
2 Object class = "TextGrid"
3
4 xmin = 0
5 xmax = 5.102604166666667
6 tiers? <exists>
7 size = 2
8 item []:
9     item [1]:
10        class = "IntervalTier"
11        name = "ep"
12        xmin = 0
13        xmax = 5.102604166666667
14        intervals: size = 24
15        intervals [1]:
16            xmin = 0
17            xmax = 1.2748823031799545
18            text = "sil"
19        intervals [2]:
20            xmin = 1.2748823031799545
21            xmax = 1.4015911317566363
22            text = "d"
23        intervals [3]:
24            xmin = 1.4015911317566363
25            xmax = 1.6175371451179235
26            text = "wl"
27        intervals [4]:
28            xmin = 1.6175371451179235
29            xmax = 1.6955142350936454
30            text = "r"
31        intervals [5]:
32            xmin = 1.6955142350936454
33            xmax = 1.7627362957776211
34            text = "b"
35        intervals [6]:
36            xmin = 1.7627362957776211
37            xmax = 2.2366447246128858
38            text = "il"

```

图 2.5 TextGrid 文件格式示意图

Fig. 2.5 The details of TextGrid file

2.3 基于深度神经网络的蒙古语语音合成系统

基于深度神经网络的蒙古语语音合成模型框架如图 2.6 所示。整体框架分为前端模块和后端模块两部分，其中前端部分包括蒙古文文本处理和蒙古文韵律模型，后端模块包括 DNN 声学模型和声码器^[84]。

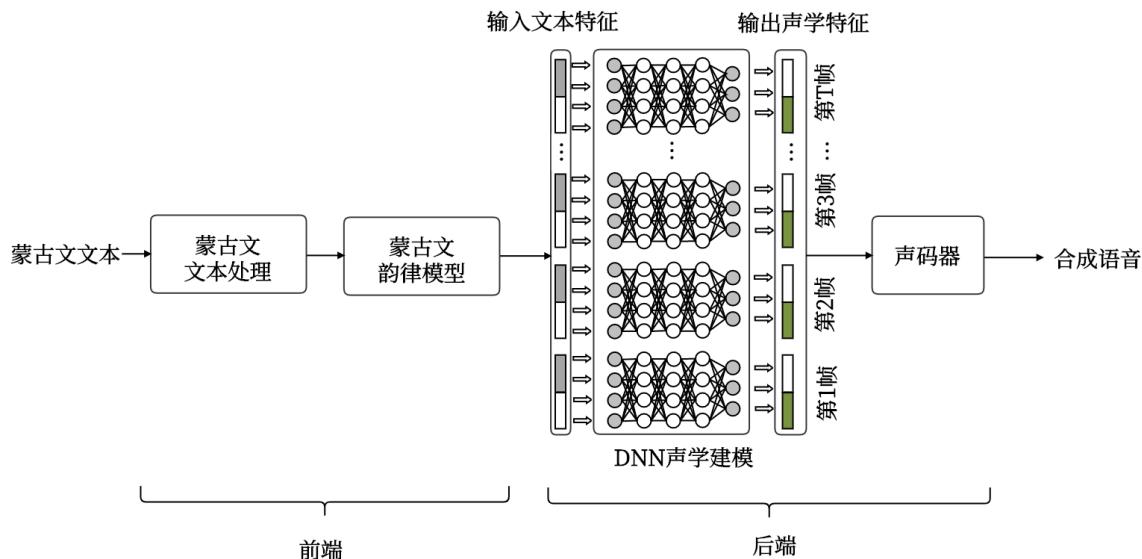


图 2.6 基于深度神经网络的蒙古语语音合成模型结构图

Fig. 2.6 The block diagram of DNN-based Mongolian TTS model

2.3.1 前端模块

2.3.1.1 蒙古文文本处理

蒙古文文本处理是将任意给定的蒙古文文本转换为规范的蒙古文文字序列表示，主要包括蒙古文文本校正、蒙古文特殊字符处理、蒙古文字母转音素、蒙古文音节划分等。

蒙古文文本校正就是将蒙古文单词序列中的错别字进行校正，之后转换为其拉丁表示。蒙古文字母在词中具有很多种不同的形式，在一个蒙古文单词中，蒙古文字母在不同的上下文环境会有不同的显现形式，因此导致蒙古文字母存在严重的形同音异现象。介于该独特的语言现象，蒙古文的文本数据中极有可能出现书写错误的情况，因此在进行蒙古文文本分析前首先需要对输入的蒙古文进行校正，校正的过程中将编码错误的蒙古文转换为其正确的显现形式。之后把经过校正得到的蒙古文统一转换为拉丁表示形式，拉丁表示根据如表 2.1 所示的蒙古文拉丁字母对照表直接转换得到。蒙古文特殊字符处

表 2.1 蒙古文字母及其拉丁字母对照表

Table 2.1 The alphabet of Mongolian and its Latin form

蒙古文 字母	拉丁 字母	蒙古文 字母	拉丁 字母	蒙古文 字母	拉丁 字母	蒙古文 字母	拉丁 字母
ᠶ	a	ᠶ	N	ᠶ	t	ᠶ	K
ᠱ	e	ᠱ	b	ᠱ	d	ᠱ	c
ᠶ	i	ᠶ	p	ᠶ	q	ᠶ	Z
ᠳ	w	ᠳ	h	ᠳ	j	ᠳ	H
ᠳ	v	ᠳ	g	ᠳ	y	ᠳ	R
ᠤ	o	ᠤ	m	ᠤ	r	ᠤ	L
ᠤ	u	ᠤ	l	ᠤ	W	ᠤ	Z
ᠴ	E	ᠴ	s	ᠴ	f	ᠴ	C
ᠶ	n	ᠶ	x	ᠶ	k		

理的效果直接影响到下游文本处理相关任务的表现。特殊字符如果不经过处理直接输入合成系统，则会产生不可预知的错误，严重影响语音合成整体表现。针对蒙古文特殊字符的特点，我们总结了出现频率较高的几类特殊字符：时间、日期、非蒙古语符号、英文单词和阿拉伯数字等，针对以上具体问题分析并设计了相对应的正则表达式。通过正则表达式过滤后，将含有特殊字符的蒙古文文本转换为只含有蒙古文文字的蒙古文文本以做进一步处理。表 2.2 列举了部分特殊字符与其对应的正确蒙古文形式。

蒙古文字母转音素就是将蒙古文单词的拉丁表示转换为其对应的音素序列。音素是蒙古语语音合成系统的基本合成单元，因此音素序列是否准确直接关系到最后蒙古语语音合成系统输出语音的词汇传达是否准确。本文采用的蒙古语音素集包含常用蒙古语音素 60 个，如表 2.3 所示，其中元音音素 35 个，其余 23 个为辅音音素，另外还有长时静音音素标记和短时静音音素标记 2 个。

为了得到蒙古文字母序列对应的正确的音素序列，飞龙等人提出基于规则的蒙古文字母转音素方法^[90]。基于规则的蒙古文字母到音素转换方法如图 2.7 所示。该方法主要根据蒙古文独特的发音规则：元音变异规则、辅音绑定规则和元音和谐规则对蒙古文字母序列进行处理。元音变异规则是指蒙古文单词中的长元音、前化元音和复合元音的复

杂变化形式，以及短元音之间的复杂形式演变。辅音结合规则是指在有些特殊情况下，若干个辅音音素可以在不依赖元音音素的情况下直接相连构成一个复辅音音节。元音和谐规则主要体现在两方面：一是同一个蒙古文单词中前后独立元音具有协调关系，二是前独立元音与其之后的依附元音具有制约关系。虽然基于这三种蒙古语发音规则可以成功对蒙古文字符进行音素转换，但是这三种发音规则仍然无法完全覆盖蒙古语复杂的发音现象，对于一些规则以外的蒙古文单词仍然无能为力。另外，由于蒙古语是典型的黏着语，同一个词根附加不同的后缀时其词根或者后缀的发音都有可能随之发生巨大变化。这些都严重制约了基于规则的蒙古文字符转音素方法的准确率。

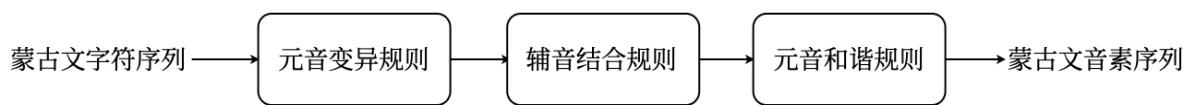


图 2.7 基于规则的蒙古文字符转音素方法流程图

Fig. 2.7 The flowchart of Rule-based Mongolian Grapheme to Phoneme (G2P)

表 2.2 部分特殊字符与其蒙古文形式对照表

Table 2.2 Table of some special characters and their corresponding Mongolian scripts

特殊字符 (非蒙古语符号)	蒙古文形式	特殊字符 (英文单词)	蒙古文形式
+	ᠶ	K	ᠺ
-	ᠶ	O	ᠶ
*	ᠶ	Au	ᠶ
/	ᠶ	Na	ᠶ
=	ᠶ	Fe	ᠶ
≈	ᠶ/ᠶ	Al	ᠶ/ᠶ

表 2.3 蒙古语音素集

Table 2.3 Phonemes set of Mongolian

元音音素	辅音音素	长时静音 音素标记	短时静音 音素标记
a ae ael al asl as2 e e2 eel el es i ii il il o oe oel ol os u ue ui ul v va vae vi vl w wi wl ws Yl	b c cc d f g h j k l m n nn p q r s t ww x y z zz	sil	sp

针对以上基于规则的 G2P 转换方法所存在的问题，飞龙等人提出了基于联合序列模型的 G2P 方法^[90]，刘致楠等人提出了基于“编码器-解码器”模型的 G2P 方法^[91]，这些方法与传统方法相比显著的提升了字母转音素的准确率，达到了实用要求。

蒙古文音节划分就是在蒙古文音素序列的基础上，明确划分其音节边界^[92]。蒙古文的音节组成符合一定的规则，通过总结发现蒙古文的音节组成符合以下六种情况：（1）单元音；（2）辅音+元音；（3）元音+辅音；（4）辅音+元音+辅音；（5）元音+复辅音；（6）辅音+元音+复辅音。采用基于规则的方法划分蒙古文音节完全可以满足实际需要。

2.3.1.2 蒙古文韵律模型

蒙古文韵律模型就是对蒙古文单词序列的韵律层级进行标注，它是蒙古语语音合成系统前端模块的重要组成部分，其韵律信息预测的准确与否直接影响蒙古语语音合成系统的自然度和表现力。

在蒙古文韵律层级中，韵律短语相对于韵律词和语调短语来说具有更加重要的韵律价值。具体的，韵律短语内部的若干韵律词一般基于相对稳定的韵律发音模式进行组合。在人类发声的韵律生成过程中，人类通过对音高、停顿等进行控制来表现丰富的节奏信息。因此，本文的蒙古文韵律建模主要指对蒙古文文本的韵律短语边界进行预测。能否从前端文本中准确估计出韵律短语的边界位置，关系到声学建模中基频曲线和时长模式的准确预测，并最终对合成语音的自然度起到关键作用^[78]。图 2.8 所示为蒙古文单词序列对应的蒙古文韵律短语边界标签序列示例（“B”表示停顿，“NB”表示非停顿）。

Mongolian	English Trans:
ᠮᠱᠳ ᨃᠳ	Most importantly, it is good for human health.
ᠮᠱᠳ ᨃᠳ	Latin:
ᠮᠱᠳ ᨃᠳ	neN qihvla ni homun-u bey_e-yin erekul qihirag-tv tvsalan_a.
ᠮᠱᠳ ᨃᠳ	Phrase Break Label:
ᠮᠱᠳ ᨃᠳ	neN [NB] qihvla [NB] ni [B] homun-u [NB] bey_e-yin [B]
ᠮᠱᠳ ᨃᠳ	erekul [NB] qihirag-tv [NB] tvsalan_a [B].
ᠮ	

图 2.8 蒙古文单词序列的韵律短语标签示例

Fig. 2.8 A prosody phrase break label example for Mongolian word units

蒙古文韵律建模即对单词序列的韵律标签进行预测，是一个标准的序列标注任务。传统蒙古文韵律建模方法均选择蒙古文单词作为基本建模单元，输出蒙古文单词的韵律停顿标签，其输入长度与输出长度相同。为了对蒙古文文本中的韵律结构进行准确的建模，刘瑞等提出了基于多种蒙古文特征和 CRF 模型的蒙古语韵律建模方法^[80]。

条件随机场最初是由 Lafferty 等人于 2001 年提出的一种无向图模型^[27]，它主要解决的就是序列标注和序列切分的问题。假设给定观察序列 X 和输出序列 Y ，其核心思想是对给定观察序列 X 进行分析，然后计算整个输出序列 Y 的联合概率，最终计算条件概率分布 $P(Y|X)$ 。假设 $G = (V, E)$ 是无向图， $Y = \{y_v | v \in V\}$ 是以 G 中结点 v 为索引的随机变量 y_v 构成的集合。如果每个随机变量 y_v 服从马尔可夫属性，则称 (X, Y) 是一个 CRF。设 $C = \{(x_c, y_c)\}$ 是图 G 中所有的团构成的集合，根据 CRF 的理论基础，对于一个输入观测序列 x ，那么这个观测序列条件下的标注序列 y 的条件概率则为 $P(Y|X)$ ，如公式 (2-1) 所示：

$$P(Y|X) = \frac{1}{Z(x)} \prod_{c \in C} e^{\sum_k \lambda_k f_k(y_c, x_c)} \quad (2-1)$$

上式中， $f_k(y_c, x_c)$ 是特征函数，特征函数权重的集合 $\Lambda = \{\lambda_k\}$ 共同构成了 CRF 模型的全部参数。 $Z(x)$ 是一个归一化因子，用于保证所有状态序列的概率的总和为 1，其计算方式如公式 (2-2) 所示：

$$Z(x) = \sum_y \prod_{c \in C} e^{\lambda_k f_k(y_c, x_c)} \quad (2-2)$$

需要说明的是，状态序列的特征函数包含两种类型，一种是状态特征函数，一种是转移特征函数。它们都需要预先初始化并使用极大似然估计等方法从训练数据中不断学习最终确定模型的最优参数。之后可以使用最优 CRF 模型参数来推断任意输入序列 x 对应的标注序列， x 最可能的韵律短语标记序列 \hat{y} 表示为公式 (2-3)：

$$\hat{y} = \operatorname{argmax}_y p_\wedge(y/x) = \operatorname{argmax}_y \sum_{c \in C} \sum_k \lambda_k f_k(y_c, x_c) \quad (2-3)$$

其中 \hat{y} 可以用动态规划的 Viterbi 算法查找。

CRF 模型中一个重要问题就是模型输入特征的选取。基于 CRF 模型的韵律建模方法中选取单词作为基本特征,为了充分考虑上下文信息对当前单词韵律标签预测的影响,模型中加入了“特征窗口”,窗口内容即以当前单词为中心的上下文信息。窗口长度越大,包含的上下文信息越丰富,对韵律短语预测越有利。但窗口长度太大,可能出现过拟合现象,而窗口长度过小,则加入特征不充分,包含信息有限,会忽略一些有用信息。常用的特征窗口包括 Unigram 和 Bigram 两种,其中 Unigram 包括前一个单元、当前单元和后一个单元, Bigram 包括前两个单元、当前单元和后两个单元。

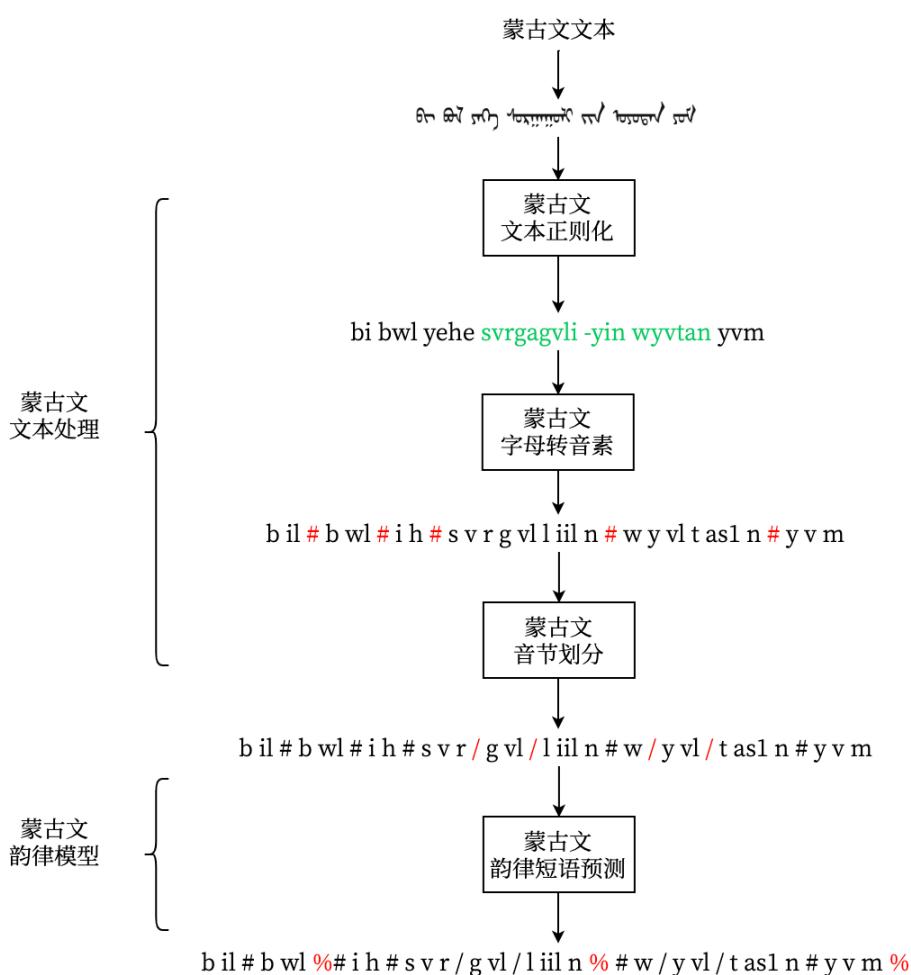


图 2.9 蒙古文文本处理流程图

Fig. 2.9 Flowchart of Mongolian text processing

2.3.1.3 蒙古文文本特征设计

图 2.9 以“**БИ БҮЛ ҮЕХЭ СҮРГАГВЛИЙН УҮСЧӨЛТӨНДҮЙН**”（汉语意思：“我是一名大学生”）为例，展示了具体的蒙古文文本处理过程。首先对其进行正则化得到其拉丁表示输出“bi bwl yehe svrgagvli -yin wyvtan yvm”，其中“svrgagvli -yin wyvtan”三个单词为在正则化过程中进行了编码校正后正确的单词表示；之后蒙古文字符转音素将其转换为对应的音素序列“b il # b w l # i h # s v r g v l l i i l n # w y v l t a s1 n # y v m”，其中“#”表示单词的边界；蒙古文音节划分后将音素序列的音节边界进行标注“b il # b w l # i h # s v r / g v l / l i i l n # w / y v l / t a s1 n # y v m”，其中“/”表示音节边界；最后通过蒙古文韵律建模处理后输出其带有韵律短语停顿标记的文本序列“b il # b w l %# i h # s v r / g v l / l i i l n%# w / y v l / t a s1 n # y v m%”，“%”表示预测得到的韵律短语边界。

表 2.4 蒙古语上下文特征集

Table 2.4 Format of Mongolian Context-dependent feature

标记 符号	上下文属性	标记 符号	上下文属性
p1	前音素	e1	前单词的音节个数
p2	当前音素	e2	前单词是否短停（0 或 1）
p3	后音素	f1	当前单词的音节个数
p4	当前音素在音节中的前向绝对位置	g1	后单词的音节个数
p5	当前音素在音节中的后向绝对位置	g2	后单词是否短停（0 或 1）
a1	前音节是否重读（0 或 1）	h1	前短语的音节个数
a2	前音节的音素个数	h2	当前短语的单词个数
b1	当前音节是否重读（0 或 1）	h3	当前单词在短语中的前向绝对位置
b2	当前音节的音素个数	h4	当前单词在短语中的后向绝对位置
b3	当前音节在单词中的前向绝对位置	i1	当前短语的音节个数
b4	当前音节在单词中的后向绝对位置	i2	当前短语的单词个数
b5	当前音节在短语中的前向绝对位置	i3	当前短语在句子中的前向绝对位置
b6	当前音节在短语中的后向绝对位置	i4	当前短语在句子中的后向绝对位置
c1	当前短语中前向重读音节个数	i5	当前短语的边界调
c2	当前短语中后向重读音节个数	j1	后短语的音节个数
c3	当前音节在短语中到前重读音节的距离	j2	后短语的单词个数
c4	当前音节在短语中到后重读音节的距离	k1	句子的音节总数
c5	当前音节中元音音素名	k2	句子的单词总数
d1	后音节是否重读（0 或 1）	k3	句子的短语总数
d2	后音节的音素个数	l1	当前音节后的边界类型

经过以上的文本分析流程，我们得到了最终处理后的蒙古文文本序列。根据前端模块得到的结果，要根据特定语种的相关语言学知识对与声学参数（谱、基频和时长）有一定影响的语言学特征进行选择，比如前后调、前后元音辅音等。语言学文本特征向量的设计要充分考虑声学参数的动态变化特性。比如为了对谱参数变化特性进行很好的表征，需要将当前音素的前后音素作为特征表示加入属性集，而针对基频参数变化特性，则需要在属性集中添加前后调类型。需要说明的是，由于设计的上下文语言学特征用于声学参数的生成，而不同语种或不同发音风格具有完全不同的声学参数变化特性，因此上下文特征选择时要充分考虑不同的语种（或发音风格）的独特特点。通过比较与分析，我们筛选出与蒙古语语音合成相关度较高的、对模型训练有利的上下文特征。表 2.4 就是最终的蒙古语语音合成系统采用的上下文特征。

我们根据设计好的蒙古文上下文特征集将该集合分为二值（0 或 1）属性和数值属性两部分。其中二值属性包括“当前音素是否为元音”、“前单词是否短停”等，数值属性包括“前音节的音素个数”、“后短语包含的音节数”、“当前音节在短语中的前向绝对位置”等。我们根据特征集将蒙古文单词序列转换为语言学特征向量输入到神经网络声学模型。

2.3.2 后端模块

2.3.2.1 神经网络声学模型

神经网络声学建模中，由于深度神经网络声学模型对声学参数进行逐帧建模，帧级别的文本特征向量和声学参数向量分别作为神经网络的输入和输出，因此首先需要将文本特征和声学特征进行时间规整，本文采用上下文相关 HMM 模型进行帧级别对齐操作^[62]，然后将帧级别的文本特征加入之前的文本特征序列得到最终的帧级别的文本特征向量 $x = [x_1, x_2, \dots, x_T]$ 。

在基于深度神经网络的蒙古语语音合成中，选择频谱特征和基频特征共同组成声学模型的训练目标。为了充分考虑声学特征在发音时的复杂变化特性，我们在训练目标设计时添加了声学参数的动态特征表示，即根据差分计算公式计算得到的一阶、二阶差分。通过将声学参数的动态特征加入训练目标，模型参数能够对声学特征的变化特性进行描述^[38]。另外在 2.4.2.2 节的声码器部分，通过对声学参数的动态特征进行处理可以生成

更加平滑的合成语音。具体的，假设 y_t 为第 t 帧的静态声学特征， Δy_t 为第 t 帧的一阶动态声学特征， $\Delta^2 y_t$ 为第 t 帧的二阶动态声学特征。其中每一帧声学参数都包括频谱参数静态值及其一阶二阶差分和基频参数静态值及其一阶二阶差分。其中一阶差分、二阶差分运算如公式(2-4)、(2-5)所示。

$$\Delta y_t = \frac{\partial y_t}{\partial t} \approx \frac{1}{2}(y_{t+1} - y_{t-1}) \quad (2-4)$$

$$\Delta^2 y_t = \frac{\partial^2 y_t}{\partial t^2} \approx y_{t+1} - 2y_t + y_{t-1} \quad (2-5)$$

因此，基于深度神经网络的蒙古语语音合成的输出声学参数构成如图 2.10 所示。分别包括频谱特征的静态值、一阶和二阶差分以及基频特征的静态值、一阶和二阶差分。其中字母 c 表示频谱特征，字母 f 表示基频特征。

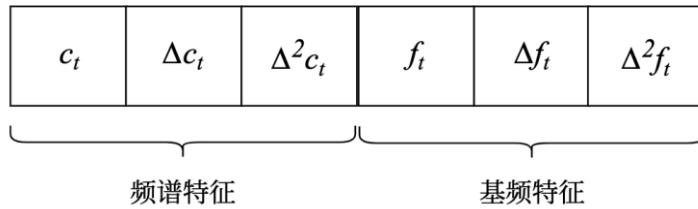


图 2.10 声学参数构成示意图

Fig. 2.10 The details of acoustic feature

最终，声学参数向量采用 35 阶的梅尔广义生成系数（Mel-generalized cepstral coefficient, MGC），1 阶对数基频（LogF0）和各自的一阶差分和二阶差分。由于基频与连续的频谱参数不同，因此需要采用插值方法将离散的基频轨迹处理为连续的基频轨迹，之后在进行声学模型的训练。DNN 声学模型需要对文本特征和对应声学特征的条件概率密度函数进行描述，具体采用高斯分布函数。我们采用最小化基于均方误差的损失函数作为模型训练准则，使用随机梯度下降和反向传播算法对 DNN 模型的权重参数进行训练^[93]，促使模型学习训练集中输入文本特征与输出声学参数的映射关系。在模型推阶段，先使用前端模块将给定文本转换为对应的文本特征向量，然后输入到训练好的 DNN 声学模型，声学模型预测出最优的静态声学特征序列及其一阶二阶动态声学参数，最后利用最大似然参数生成算法生成出连续的声学特征。

2.3.2.2 声码器

声码器部分中，DNN 声学模型预测出的声学特征被输入到声码器中进行进一步处理，从而得到最终的合成语音波形输出。本文选择 STRAIGHT 语音分析算法^[94]作为声码器的具体实现。STRAIGHT 算法可以对语音文件进行处理，实现语音特征提取和语音波形重建等任务。在语音特征提取时，该算法可以计算得到多种高质量的声学特征序列；在语音波形重建时，该算法能够利用声学模型预测得到的声学特征序列从而恢复出对应的高质量的语音波形文件。具体的，该算法通过以下方法实现：

(1) 去除周期性影响的谱估计：在对语音信号进行谱分析时，第一步需要对语音信号的短时谱做加窗操作，这一过程会导致窗内部的短时谱在时间轴和频率轴上都表现出与基音周期相关的周期性。STRAIGHT 语音分析算法可以分别去除时间轴和频率轴上的周期性，从而精确估计谱参数。

(2) 可靠的基频提取：基频提取需要对语音的频谱进行细致的谐波分析，该算法可以实现这一过程并得到浊音段上精确和稳定的基频轨迹。

(3) 合成器的实现：在语音波形恢复的过程中，该算法采用基音同步叠加法以及最小相位冲激响应的方法实现高质量的语音合成过程。

2.4 实验

本节将对基于深度神经网络声学模型的蒙古语语音合成系统的合成语音表现进行评估，并测试不同模型参数条件下的模型表现。最终采用最优参数搭建完成了基于深度学习的蒙古语语音合成系统。

2.4.1 实验配置

实验采用 2.2 节中的 2 小时标注数据用于基于深度神经网络的蒙古语语音合成模型的训练。其中包括 2620 句蒙古语语音，训练时将语音采样率转换到 16kHz，90% 作为实验训练集，5% 作为实验测试集，另外 5% 作为最终测试集。

实验的评价指标一共涉及两类指标。第一类是主观评价指标，使用主观意见平均分 (Mean Opinion Score, MOS) 对合成语音进行打分。MOS 是语音合成领域公认的一种主观语音质量评价方法，评分分值从 1 分到 5 分。在测听者对合成结果进行测听时，要

严格按照表 2.5 的评分标准进行评分，最终通过计算所有测听者分数的平均值得到最终的 MOS 分数。

第二类是客观评价指标，包括谱参数 MGC 与基频参数 logF0 的均方误差（Root Mean Squared Error, RMSE）。RMSE 是一种常用的评价生成参数与目标参数之间误差的方法，它用来衡量合成语音与目标说话人语音之间的参数扰动。计算公式如公式（2-6）所示。

$$\text{RMSE} = \sqrt{\sum_i^N (f(i) - f_e(i))^2 / N} \quad (2-6)$$

其中 N 是所有合成语音的帧数， $f(i)$ 是目标语音参数， $f_e(i)$ 是模型估计得到的语音参数。

表 2.5 语音合成主观评价 MOS 评分标准

Table 2.5 The criteria of MOS Objective test for speech synthesis

MOS	质量	语音情况
5	优	接近真人，十分自然
4	良	比较自然，接近真人
3	中	比较自然，可以接受
2	差	比较不自然，比较可以接受
1	劣	不能接受

2.4.2 实验设计

为了验证基于深度学习的蒙古语语音合成方法的有效性，本文构建了两个系统：

(1) HMM：基线系统蒙古语语音合成方法，其韵律模型采用 CRF，声学模型采用 HMM。具体的，上下文相关音素 HMM 模型的参数配置采用 5 状态、自左向右无跳转的结构。每个 HMM 状态的建模表示均采用以协方差矩阵为对角线的单高斯模型。用于上下文聚类决策树建立的问题集规模为 693 个。选择最大似然准则下对 HMM 模型参数进行训练和更新。

(2) DNN：本文实现的深度学习蒙古语语音合成方法，其韵律模型采用 CRF，声学模型采用 DNN。根据 2.4.1.3 节设计的蒙古文上下文特征集，得到包含二值特征和数值特征的输入文本特征 693 维。训练前，将文本特征规整到 [0.01, 0.99] 范围内。输出声学特征以 25ms 帧长、5ms 帧移提取，包括 35 阶梅尔广义生成系数 MGC，1 阶对数基频 logF0 和它们的一阶、二阶动态参数以及静音/非静音标签，一共 109 维 ($3 \times (35+1)$)

+1=109）。静音/非静音标签是一个二值参数用来表示当前是语音帧是否为静音帧。为了减小训练的计算量，将训练数据中所有的静音帧裁剪到 0.3 秒。训练前，将输出声学特征规整到零均值单位协方差。实验中通过调节深度神经网络中隐含层个数及每个隐含层的节点数来测试最优的模型参数。

我们使用以上两个实验系统与两个评价指标开展下节的相关实验。

2.4.3 实验结果与分析

实验过程中，采用不同参数配置比较不同 DNN 系统的效果。通过调整神经网络隐含层个数从 1 到 5，每层隐含层的节点个数从 256、512、1024 到 2048，从而构建了 40 个不同结构的 DNN 蒙古语语音合成模型。表 2.6 展示了所有系统的谱参数均方误差（MGC RMSE）和基频均方误差（LogF0 RMSE）结果。

表 2.6 不同结构 DNN 蒙古语语音合成模型的客观评价结果

Table 2.6 The objective assessment of different structure in DNN-based Mongolian TTS systems

DNN 结构	MGC RMSE	LogF0 RMSE	DNN 结构	MGC RMSE	LogF0 RMSE
1×256	21.64	3.42	1×1024	22.24	3.44
2×256	22.97	3.35	2×1024	22.16	3.35
3×256	22.45	3.35	3×1024	23.94	3.39
4×256	22.54	3.39	4×1024	23.43	3.39
5×256	22.39	3.37	5×1024	23.48	3.40
1×512	21.82	3.43	1×2048	22.59	3.42
2×512	21.90	3.34	2×2048	22.25	3.36
3×512	23.42	3.38	3×2048	23.92	3.40
4×512	23.74	3.39	4×2048	23.49	3.40
5×512	23.06	3.39	5×2048	23.66	3.40

从表 2.6 中可以看出，对于 MGC RMSE 这一评价指标，最简单的 1×256 结构的 DNN 模型取得了最好的表现，MGC RMSE 分值为 21.64。对于 LogF0 RMSE 这一评价指标，表现最好的 DNN 模型结构是 2×512，其数值为 3.34，优于其他结构。因此，从

整体指标表现看，简单的模型结构相比于复杂的模型结构具有更好的表现。

另外，考虑到主观和客观指标的不一致性，本节还使用主观评价指标来评估模型效果。为了评估的全面性和完整性，我们选取了三个 DNN 系统与上一小节的最优 HMM 基线系统进行比较，三个 DNN 系统分别是：谱参数均方误差最小的 DNN 系统，记作 DNN (1×256)；基频参数均方误差最小的 DNN 系统，记作 DNN (2×512)；参数规模最大的 DNN 系统，记作 DNN (5×2048)。我们从测试集中随机选取 60 句蒙古文句子，用这 4 个系统分别合成蒙古语语音，选取 10 位以蒙古语为母语的测听者进行语音评价，他们的年龄在 20 岁到 30 岁之间。图 2.11 所示为 4 个系统的主观 MOS 评价结果。从表中可以看出，DNN (2×512) 系统取得了最高的 MOS 分数 3.82，而模型规模最小的 DNN (1×256) 和规模最大的 DNN (5×2048) 没有达到最好的合成效果。三个 DNN 系统中有两个的表现超过了 HMM 基线系统，因此我们认为 DNN 系统与基于决策树聚类的 HMM 基线相比可以更好地拟合文本特征到声学特征的复杂映射关系，可以实现更好地效果。

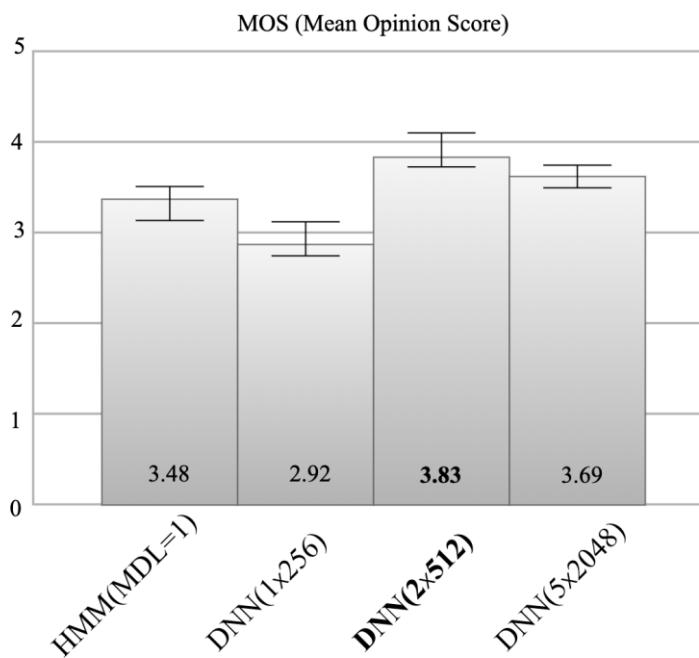


图 2.11 蒙古语多种结构 DNN 模型与最优 HMM 基线系统的主观 MOS 评测结果（置信度 95%）

Fig. 2.11 MOS results of the optimal HMM system with multiple DNN systems in Mongolian

通过分析实验结果得知，神经网络模型的建模能力与网络结构的大小并不是呈正相关关系。网络结构太大，模型容易出现过拟合，并且模型中隐藏层越多，过度拟合风险

越大；网络结构如果太小，导致模型对数据拟合不足。通过比较不同 DNN 模型结构对当前蒙古语语音合成数据的泛化性能，确定了最佳的 DNN 模型结构配置。

最后我们确定参数配置为 2×512 的 DNN 系统作为本文最终的基于深度神经网络的蒙古语语音合成系统。

2.5 本章小结

本章对基于深度学习的蒙古语语音合成系统的实现过程做具体介绍。首先介绍了蒙古语的相关语言特点，之后对语音合成语音库的准备工作进行介绍。然后从前端模块到后端模块，详细介绍了基于深度神经网络的蒙古语语音合成系统的具体模型结构。最后通过实验比较不同参数配置下的模型效果，确定了最优的 DNN 模型。实验结果表明，基于深度神经网络的蒙古语语音合成方法与基于 HMM 统计参数的蒙古语语音合成方法相比具有较高的合成自然度。这个深度学习蒙古语语音合成系统将作为本文的基线系统，是本文后续章节工作的基础。

第三章 融合蒙古文形态学与音系学知识的韵律建模方法

3.1 引言

第二章详细介绍了基于神经网络声学模型的蒙古语语音合成系统的实现过程，并最终通过客观和主观实验进行对比分析，实验表明基于神经网络的蒙古语语音合成方法与基于 HMM 统计参数模型的蒙古语语音合成方法相比，可以生成音质更好、自然度更高的合成语音。这说明了 DNN 声学模型具有更强大的非线性建模能力，与 HMM 模型相比更加适用于声学模型的实现。相比于以统计建模理论支撑的 HMM 声学模型，DNN 声学模型凭借其更深的网络结构和更强的建模能力，可以很好的表征声学建模中复杂的非线性映射关系。并且这样的建模过程也摆脱了决策树聚类过程中潜在的数据分类不成功问题。然而，基于深度神经网络的蒙古语语音合成方法也有其不足之处，在韵律表现和合成音质两个方面均与真实语音有一定差距。

语音合成是一门典型的交叉学科，语种相关的语言学背景知识在语音合成系统的实现过程中发挥着举足轻重的作用^[18,19]。蒙古文韵律建模是蒙古语语音合成系统前端模块的重要部分，韵律模型的预测结果与其他语言学信息相结合被进一步用于预测待合成语音的停顿、时长、基频和频谱等声学参数，因此，韵律模型的建模准确率直接影响蒙古语语音合成系统的自然度和表现力。但是对于蒙古文韵律模型，存在很多问题需要解决：（1）现有方法只是使用简单的蒙古文符号化特征表示，其语法语义表达能力有限；（2）由于蒙古文独特的语言特点和文本资源的稀缺，造成了严重的数据稀疏问题，而且限制了蒙古文集外词的特征表达能力，给蒙古文韵律建模带来很大挑战；（3）现有方法使用浅层机器学习模型，不具有很强大的建模能力；本章结合蒙古文的语言特点，以提升蒙古语语音合成的自然度为目标，从模型输入和模型结构两个角度研究如何提升前端模块中蒙古文韵律建模的性能^[97-99]。

本章首先对使用的相关技术进行介绍，分别包括长短时记忆循环神经网络和词向量表示学习相关知识。之后详细介绍基线蒙古文韵律建模方法的具体结构，然后提出了两种基于深度学习的蒙古文韵律建模方法，分别是基于词素单元的蒙古文韵律建模方法^[97]

和融合形态向量与音系向量的蒙古文韵律建模方法。最后通过客观实验比较分析了两种方法的性能优劣,通过主观实验分析了两种方法对于提升蒙古语语音合成自然度的影响。

3.2 相关技术

3.2.1 长短时记忆循环神经网络

DNN 神经网络可以对序列数据进行建模,但是当输入数据具有很强的时间依赖性时,全连接的前馈 DNN 神经网络结果一般都不太好。DNN 的前一时刻输入和下一时刻输入之间没有任何关联,输出序列中相邻单元都是独立的,相邻单元的时序依赖关系没有被充分考虑。为了解决这一问题,循环神经网络提供了一种全新的解决思路,它与传统的 DNN 网络结构不同, RNN 对每个隐层节点中添加了自循环连接,使得隐含层的输入同时来自于多种历史信息,即包括输入层的输出和上一时刻隐含层的输出,因此可以很好的处理序列数据。图 3.1 (a) 所示是一个典型的 RNN, 沿时间 t 展开后的结构如图 3.1 (b) 所示。

上述结构使得 RNN 增加了记忆能力从而很好的进行序列建模,然而实际中,随着训练时间步的推移, RNN 训练很容易出现梯度消失(或爆炸)问题,使得 RNN 网络很难充分的考虑到距离较远的序列依赖关系。因此,长短时记忆单元循环神经网络 LSTM 被提出用来缓解这一问题^[100]。LSTM 在记忆单元中引入了多种门控机制来控制信息的流动,可以选择性的对上下文信息进行保留,缓解了梯度消失问题,可以更有效的对长序列依赖问题进行学习训练。如图 3.2 所示, LSTM 在 RNN 隐层单元的基础上加入了门控机制,包括输入门 i_t , 输出门 o_t 和遗忘门 f_t 。但是 RNN 和 LSTM 只对序列从左向右的历史信息进行记忆学习,没有序列的未来信息进行建模。为了更有效的学习序列的历史和未来信息,双向循环神经网络(Bi-directional RNN, BiRNN)及双向长短时记忆循环神经网络 BiLSTM 被提出^[101]。以 BiLSTM 为例,将从左到右的 LSTM 称为前向 LSTM, 从右到左的 LSTM 称为后向 LSTM, 前向后向 LSTM 同时接受两个方向的输入,更好的学习输入序列的全局上下文信息。

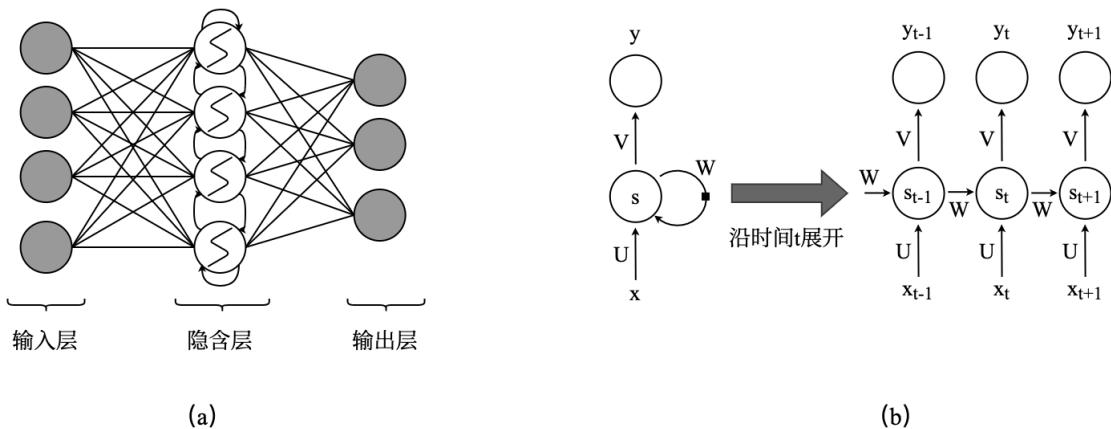


图 3.1 循环神经网络结构示意图

Fig. 3.1 The schematic diagram of recurrent neural network (RNN)

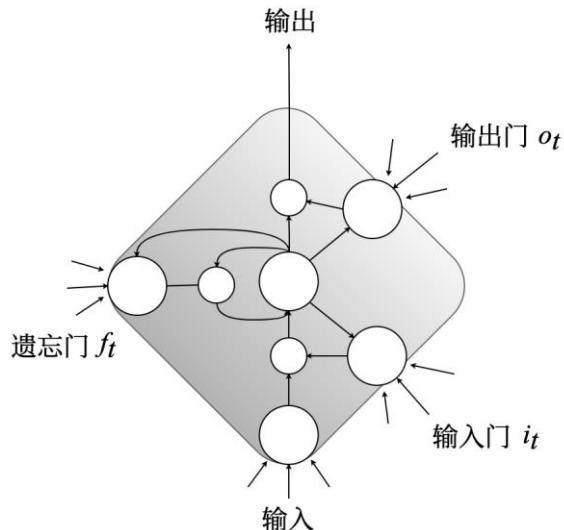


图 3.2 长短时记忆循环神经网络门控机制示意图

Fig. 3.2 The schematic diagram of Long short-term memory Recurrent Neural Network (LSTM)

3.2.2 词向量表示

表示学习^[53,54]是当今机器学习领域经常被讨论的话题，随着深度学习相关技术的崛起，越来越多的研究者开始研究表示学习相关问题。表示学习的核心思想就是用数字（向量、矩阵等）来表达现实世界中的物体，这种表达方式有利于后续的分类或者其他决策问题。与传统的特征工程（Feature Engineering）不同，特征工程中需要利用人为的经验去设计一套适合某个特定领域特征，这样的过程具有经验依赖度高、时间效率低、特征扩展性差等问题。相反，在表示学习中，我们希望通过深度学习算法自动从给定的数据

中学习合理的特征或者表示。在自然语言处理领域，词向量^[95,96]被用来表示单词的深层次语义特征。词向量的基本思想就是通过训练将维数为单词词典大小的高维空间转换为一个具有更低维度的连续向量空间，这一转换过程使得每个单词都被映射成一个固定维度的向量表示。之后，所有单词训练得到的向量表示最终形成一个完整的词向量空间，每个单词对应的词向量可以很好的表示其词法和语义等信息。

传统的词向量学习方法是基于现有训练数据统计得到一个词语共生矩阵 \mathbf{X} 。其矩阵维度为 $|V| \times |V|$ 。其中， V 表示当前训练数据中计算得到的词汇表（Vocabulary）， $|V|$ 为词汇表的大小， X_{ij} 表示在所有语料中，词汇表 V 中第 i 个词和第 j 个词同时出现的频次。之后通过采用奇异值分解（Singular Value Decomposition, SVD）的方法对矩阵 \mathbf{X} 进行矩阵分解，最后得到的 \mathbf{U} 即视为所有词的词向量，如公式（3-1）所示：

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T \quad (3-1)$$

这样的传统做法虽然可以得到单词的向量表示，但是其实现过程表现出诸多问题：

(1) 由于训练数据的规模有限，导致很多词具有极低的出现频率，甚至没有出现，从而造成矩阵非常稀疏。因此，为了达到比较理想的矩阵分解效果，需要预先对单词的词频做一些额外处理；(2) 停用词（如“而且”、“否则”等）在语料数据中会频繁出现，为了降低其对矩阵分解的负面影响，需要预先对停用词进行删除；(3) 矩阵的维度太高，导致计算复杂。为了很好的解决上述问题，研究人员提出了基于神经网络的词向量学习模型，该方法不需要对一个大规模的语料数据进行繁重的单词统计，而是使用模型训练的方法实现语义信息的自动学习，最终得到词向量作为单词的语义信息表示。主流的词向量训练方法包括：连续词袋模型（Continuous Bag of Word, CBOW）和跳字模型（Skip-Gram）^[95,96]等。

CBOW 模型通过一个词的上下文（各 N 个词）预测当前词。当 $N=2$ 时，模型如图 3.3 所示。给定单词序列 $\mathbf{W}=[w_1, w_2, \dots, w_T]$ ，CBOW 以公式（3-2）作为优化目标：

$$\sum_{t=1}^T \log P(w_t / w_{t-N}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+N}) \quad (3-2)$$

其中概率 P 的计算采用 Softmax 函数。

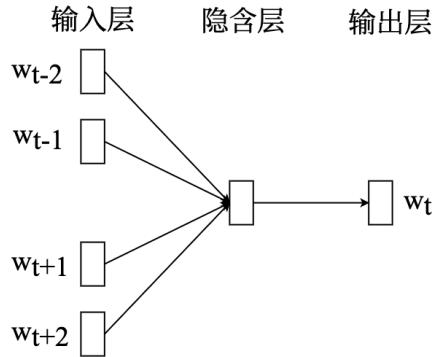


图 3.3 CBOW 模型结构图

Fig. 3.3 The schematic diagram of CBOW model

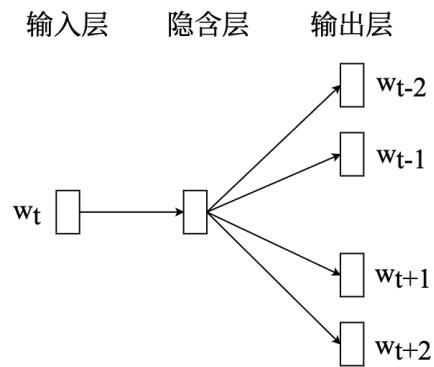


图 3.4 Skip-Gram 模型结构图

Fig. 3.4 The schematic diagram of Skip-Gram model

Skip-Gram 模型与 CBOW 模型相反，它是通过当前词去预测该词的上下文（附近 N 个词）。当 N=2 时，模型如图 3.4 所示。给定单词序列 $W=[w_1, w_2, \dots, w_T]$ ，Skip-Gram 以公式 (3-3) 作为优化目标：

$$\sum_{t=1}^T \log P(w_{t-N}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+N} | w_t) \quad (3-3)$$

其中， w_t 表示当前单词， $(w_{t-N} \dots w_{t+N})$ 表示在上下文窗口内的邻近单词。

3.2.3 子词向量表示

词向量可以使用连续的数字向量作为单词的语义表示，而词向量的训练通常需要预先设定一个词汇表，该词汇表的大小是固定不变的，对于词汇表中所有词可以学习到其对应的向量表示。而由于数据的不规范处理或者语言的复杂构词特点等问题，导致训练

数据不能对特定语种的所有词汇进行完全覆盖，在对训练数据中的单词进行向量训练时会出现一定数量的集外词。针对集外词的处理，传统的做法是设置一个特殊标记来表示所有 OOV 的词汇，比如 UNK (Unknown Token) 标记。这种做法简单粗暴，严重忽略了集外词的语法语义信息。为了更好学习到集外词的语法语义信息，研究者们从词的内部结构出发，从子词角度提出了基于字符的词向量学习方法^[102,103]。其核心思想就是，假设有两个不同的单词，但是他们可能包含相同的字符，这样对于属于集外词的单词，找到其对应的字符向量表示后，进行融合（求和或求平均值）即可得到该集外词的语法语义向量表示。借助于神经网络强大的特征表示能力，目前主流的字符向量学习方法主要有 CNN^[102] 和 BiLSTM 模型^[103] 两种。

CNN 字符向量学习模型主要包括卷积层和池化层。其结构图如图 3.5 所示。卷积层以单词的字符序列的独热编码向量为输入，计算字符序列的隐含特征表示，之后使用最大池化操作从字符特征中抽取得到固定维度的向量表示作为单词的字符向量。

BiLSTM 字符向量学习模型主要使用前向 LSTM 层和后向 LSTM 层对输入的字符序列进行处理。其结构图如图 3.6 所示。前向 LSTM 和后向 LSTM 分别对输入的独热编码表示的字符向量进行处理，之后前向 LSTM 输出的隐状态和后向 LSTM 输出的隐状态进行拼接，得到最终的单词的字符向量表示。

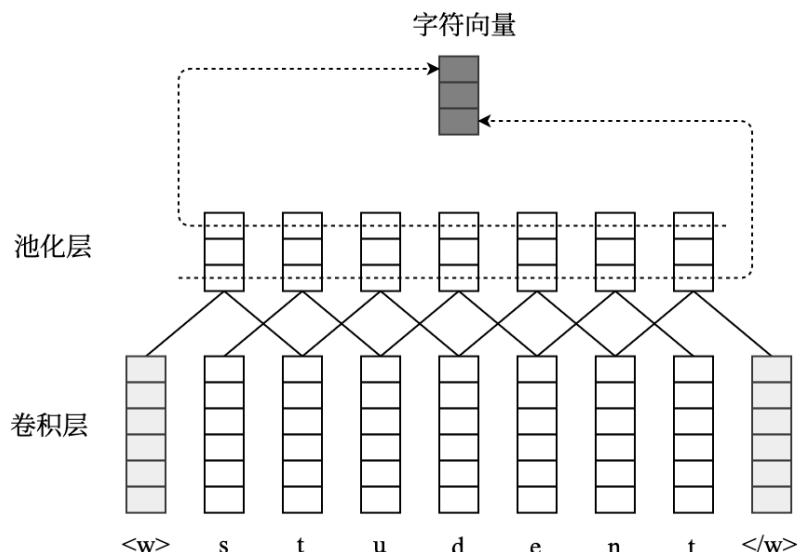


图 3.5 CNN 字符向量学习模型结构图

Fig. 3.5 The schematic diagram of CNN-based character embedding model

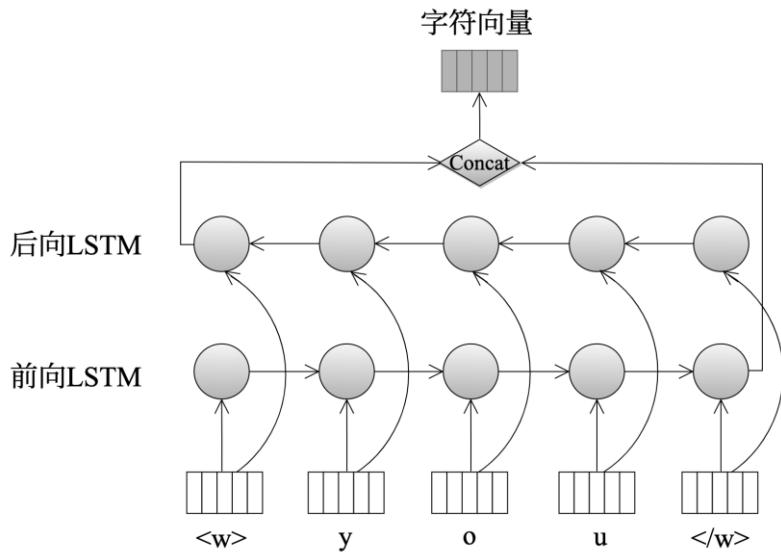


图 3.6 BiLSTM 字符向量学习模型结构图

Fig. 3.6 The schematic diagram of BiLSTM-based character embedding model

3.3 基于词素单元的蒙古文韵律建模方法

蒙古文的形态学构词特点造成严重的数据稀疏问题,为了缓解数据稀疏问题对蒙古文韵律预测模型精度的影响,本文对蒙古文单词进行拆分解构,将原先的单词级别建模单元转换为词素级别建模单元,并结合 LSTM 的序列建模能力,对 CRF 模型进行替换,提出了基于词素单元的蒙古文韵律建模方法。图 3.7 所示为蒙古文单词序列转换为蒙古文词素序列后对应的蒙古文韵律短语标签序列。

Mongolian English Trans:

ᠮᠱᠳ

Most importantly, it is good for human health.

Latin:

ᠮᠱᠳ
ᠶ
ᠳ

neN qihvla ni homun-u bey_e-yin erekul qihirag-tv tvsalan_a.

Segmentation:

ᠮ
ᠱ
ᠳ

neN qihvla ni homun -u bey_e -yin erekul qihirag -tv tvsalan_a.

Phrase Break Label:

ᠮ
ᠱ
ᠳ

neN [NB] qihvla [NB] ni [B] homun [NB] -u [NB] bey_e [NB]

-yin [B] erekul [NB] qihirag [NB] -tv [NB] tvsalan_a [B].

图 3.7 蒙古文词素单元的韵律短语标签示例

Fig. 3.7 A prosody phrase break label example for Mongolian morpheme units

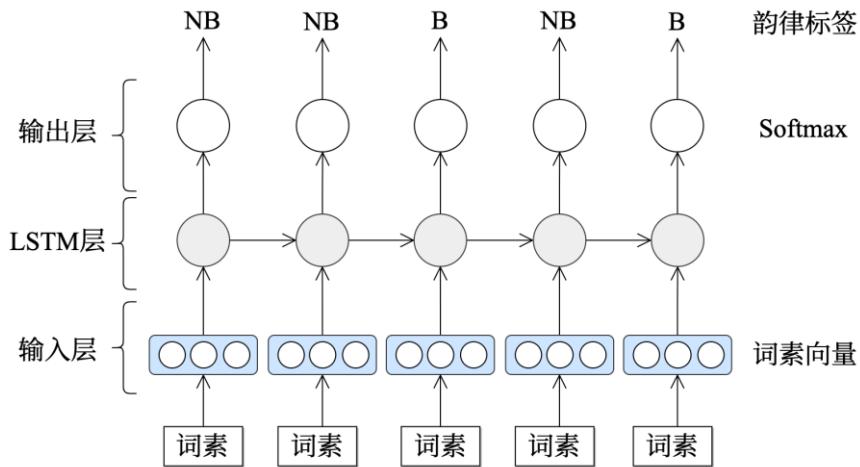


图 3.8 基于词素单元的蒙古文韵律建模框架

Fig. 3.8 The block diagram of subword-based Mongolian prosody model

图 3.8 所示为基于词素单元的蒙古文韵律建模方法的基本框架。该模型以蒙古文词素为建模单元，通过 LSTM 神经网络对蒙古文韵律短语进行建模。以下首先介绍蒙古文词素向量的训练方法，之后介绍基于词素单元和 LSTM 网络的蒙古文韵律建模方法的具体结构。

3.3.1 蒙古文词素向量

模型训练前，首先需要获得蒙古文词素的语义向量表示，即词素向量。本文借鉴 3.2.2 节中词向量的训练方法来对蒙古文词素向量进行训练，考虑到蒙古文语料中严重的数据稀疏问题，由于 Skip-Gram 模型与 CBOW 模型相比可以更好地处理低频词，因此我们选择 Skip-Gram 模型^[95]进行蒙古文词素向量的训练。

具体地，我们使用 Skip-Gram 向量模型来根据当前词素单元预测它在文本序列中周围的词素单元。假设给定蒙古文文本序列 “homun-u bey_e-yin erekul qihirag-tv tvsalan_a”，通过词素切分，得到其词素序列 “homun -u bey_e -yin erekul qihirag -tv tvsalan_a”。如果将 “bey_e” 看作当前中心词素单元，时间窗口大小为 2 的话，那么 Skip-Gram 的目标就是最大化给定 “bey_e” 生成时间窗口内邻近词素单元 “homun”、“-u”、“-yin”、“erekul”的条件概率。在这个例子中，“bey_e” 叫做中心词素，“homun”、“-u”、“-yin” 和 “erekul” 叫做背景词素。条件概率的公式化表达为公式（3-4）：

$$\prod_{t=1}^T \prod_{-m \leq j \leq m, j \neq 0} P(w^{(t+j)} | w^{(t)}) \quad (3-4)$$

其中 T 为词素序列的长度，词素序列中第 t 个词素为 $w^{(t)}$ 。 m 表示时间窗口的大小。上式的大似然估计可以等价表示为最小化以下损失函数，如公式 (3-5) 所示。

$$-\frac{1}{T} \prod_{t=1}^T \prod_{-m \leq j \leq m, j \neq 0} P(w^{(t+j)} | w^{(t)}) \quad (3-5)$$

损失函数中的给定中心词素预测邻近背景词素的条件概率可以通过 Softmax 函数定义为公式 (3-3)：

$$P(w^{(t+j)} | w^{(t)}) = \frac{\exp(u_o^\top v_c)}{\sum_{i \in M} \exp(u_i^\top v_c)} \quad (3-6)$$

其中， o 是输出词素的编号， c 是当前中心词素的编号。 u 和 v 分别是输出词素和输入词素的向量表示， M 是词素集合的大小。

在使用随机梯度下降法训练模型时，为了降低计算复杂度，如果序列长度 T 较大，可以采用随机采样的方法降低计算量，具体的，每次迭代时随机采样一个较短的子序列进行损失计算。然后，根据该损失计算词素向量的梯度并迭代词素向量。随机采样的子序列的损失实际上对子序列中给定中心词素生成背景词素的条件概率的对数求平均。通过微分的形式进行梯度计算，如公式 (3-7) 所示。

$$\frac{\partial P(w_o | w_c)}{\partial v_c} = u_o - \sum_{j \in v} \frac{\exp(u_o^\top v_c)}{\sum_{i \in v} \exp(u_i^\top v_c)} u_j \quad (3-7)$$

该式也可以写作 (3-8) 的形式：

$$\frac{\partial P(w_o | w_c)}{\partial v_c} = u_o - \sum_{j \in v} P(w_j | w_c) u_j \quad (3-8)$$

由于迭代过程中每一步梯度计算的开销与蒙古文词素词典 V 的大小有关，蒙古文词素词典 V 的大小决定了 Skip-Gram 向量模型权重矩阵的规模，蒙古文词素词典规模庞大，在计算 Softmax 时需要考虑词素词典中的所有可能性，因此会极大地加大计算开销。本文使用负采样和层序 Softmax 两种近似梯度计算方法^[95,96]来有效减小计算开销。

负采样主要对于高频词素进行抽样，我们以一定的丢弃概率 $p(w_i)$ 将高频词素从训练文本中删除。词素出现频率越高，被删除的概率越大。负采样技术使网络每次只更新

一小部分权重，这样就会降低梯度下降过程中的计算量。丢弃概率 $p(w_i)$ 的公式化表达为公式（3-9）：

$$p(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}} \quad (3-9)$$

公式中 $f(w_i)$ 是蒙古文词素 w_i 的出现频率， t 是控制采样操作的阈值参数，本文设置为 10^{-4} 。该参数数值越小，意味着该词素被删除的概率越大。

层序 Softmax 利用了二叉树的数据结构，它使用哈夫曼树替代原来的 Softmax 层以及对应的权重系数矩阵，哈夫曼树的结构是基于词素出现频率构建的，高频词素位于哈夫曼树的上端，低频词素位于哈夫曼树的下端。训练进行更新时只需要更新从根节点到当前叶节点路径上所有节点的参数矩阵即可，同时由于高频词素在离根节点较近的地方，因此大部分词素都不需要走完哈夫曼树的最大深度，只需要更新浅层的权重即可。

训练结束后，对于词素词典中的任一索引为 i 的词，一共可以生成两种向量表示，即中心词素和背景词素的两组词素向量。最终，我们使用中心词素向量作为最终的蒙古文词素向量表示。

以时间窗口等于 2 为例，以蒙古文单词为建模单元的 Skip-Gram 向量模型与以蒙古文词素为建模单元的 Skip-Gram 向量模型有很大差别。假设单词 Skip-Gram 向量模型中的中心词为“bey_e-yin”，背景词为“homum-u”、“eregul”和“qihirag-tv”。而词素 Skip-Gram 向量模型的单元由单词转换为词素，中心词素为“bey_e”，背景词素“homun”、“-u”、“-yin”和“eregul”。可以看出词素向量模型的背景词素中将词干与结尾后缀作为独立的语义单元，可以更显著的学习到蒙古文的形态学知识，得到有意义的蒙古文词素向量。下一节我们将蒙古文词素向量作为蒙古文词素单元的特征表示，送入基于 LSTM 网络的蒙古文韵律模型进行蒙古文韵律短语停顿的预测。

3.3.2 LSTM 韵律模型

本文基于 LSTM 网络搭建了基于词素单元的蒙古文韵律模型。如图 3.2 所示，基于词素单元的蒙古文韵律模型包括输入层、LSTM 层和输出层。

输入层即将输入的蒙古文词素单元转换为上节获得的对应的蒙古文词素向量作为其特征表示。假设输入长度为 T 的蒙古文词素序列 $X = [x_1, x_2, \dots, x_T]$ ，根据 3.3.1 节中预

先训练好的蒙古文词素向量，通过查找词表（LookUp Table）将其转换为对应的特征向量 $\mathbf{V} = [v_1, v_2, \dots, v_T]$ ，之后将该特征向量 \mathbf{V} 送入 LSTM 层进行下一步处理。

LSTM 层读取输入的词素特征向量 \mathbf{V} 将其转换为高层特征表示。首先从左到右读取特征向量 \mathbf{V} ，依次得到隐状态输出 $\mathbf{H} = [h_1, h_2, \dots, h_T]$ 。其中 LSTM 隐层单元在任一时刻 t 的输入来自两部分，第一部分来自于当前时刻输入层的输入 v_t ，第二部分来自于前一时刻隐层单元的输出 h_{t-1} ，用公式描述为（3-10）：

$$h_t = f(\mathbf{W}v_t + \mathbf{U}h_{t-1} + \mathbf{b}) \quad (3-10)$$

其中， f 是 Sigmoid 非线性激活函数， \mathbf{W} 是输入与隐层单元之间的权重矩阵，隐层单元间的权重矩阵用 \mathbf{U} 表示， \mathbf{b} 是偏置向量。

特征向量 \mathbf{V} 与隐状态向量 \mathbf{H} 具有相同的时间长度，每一个时间步的特征向量 v_t 都有一个隐状态向量 h_t 与其对应。LSTM 层得到的隐状态输出 \mathbf{H} 送入输出层解码得到最终的词素对应韵律标签。

输出层将 LSTM 层输出的隐状态 $\mathbf{H} = [h_1, h_2, \dots, h_T]$ 进行解码，最终输出每个词素对应的韵律标签 $\mathbf{Y} = [y_1, y_2, \dots, y_T]$ 。用公式描述为（3-11）和（3-12）：

$$y_t = g(\mathbf{V}h_t + \mathbf{c}) \quad (3-11)$$

$$y_t^* = \text{argmax}(y_t / x_1, \dots, x_t) \quad (3-12)$$

其中 g 是输出层函数， \mathbf{V} 是隐层单元和输出单元间的权重矩阵， \mathbf{c} 是偏置向量。

输出层采用 Softmax 函数，将输出的向量转化为 0 到 1 之间的概率值，然后对概率值进行归一化，从而找到概率值最大的韵律标签 y_t 即为最终的韵律标签。

模型训练使用交叉熵损失函数作为优化目标。公式如（3-13）所示：

$$\text{Loss} = -\sum_{t=1}^T \log(P(y_t / d_t)) \quad (3-13)$$

3.4 融合形态向量和音系向量的蒙古文韵律建模方法

如 3.3 节所述，传统蒙古文韵律建模方法以蒙古文单词为建模单元，而单词的向量表示是从蒙古文文本语料资源中学习得到的，但是规模有限的蒙古文文本资源很难学习到意义充分的蒙古文词向量。而且，蒙古文复杂的构词特点导致集外词问题明显，也给蒙古文韵律建模带来很大挑战。另外，LSTM 的单方向性对于基于时间序列的预测来说是非常有意义的，但是对于时序序列来说，更重要的上下文语境信息没有被充分建模。所以 BiLSTM 用两个独立的 LSTM 层对输入序列进行双向处理，从而消除输入方向之间的不对称性，它结合了长时依赖和上下文建模的优点，表现出更好的性能。

综合考虑以上两个因素，为了更进一步充分利用蒙古文形态学知识和音系学知识提升蒙古文韵律建模方法的性能，本节基于 BiLSTM 神经网络提出了融合形态向量和音系向量的蒙古文韵律建模方法。图 3.9 所示为融合形态向量和音系向量的蒙古文韵律建模方法的基本框架。

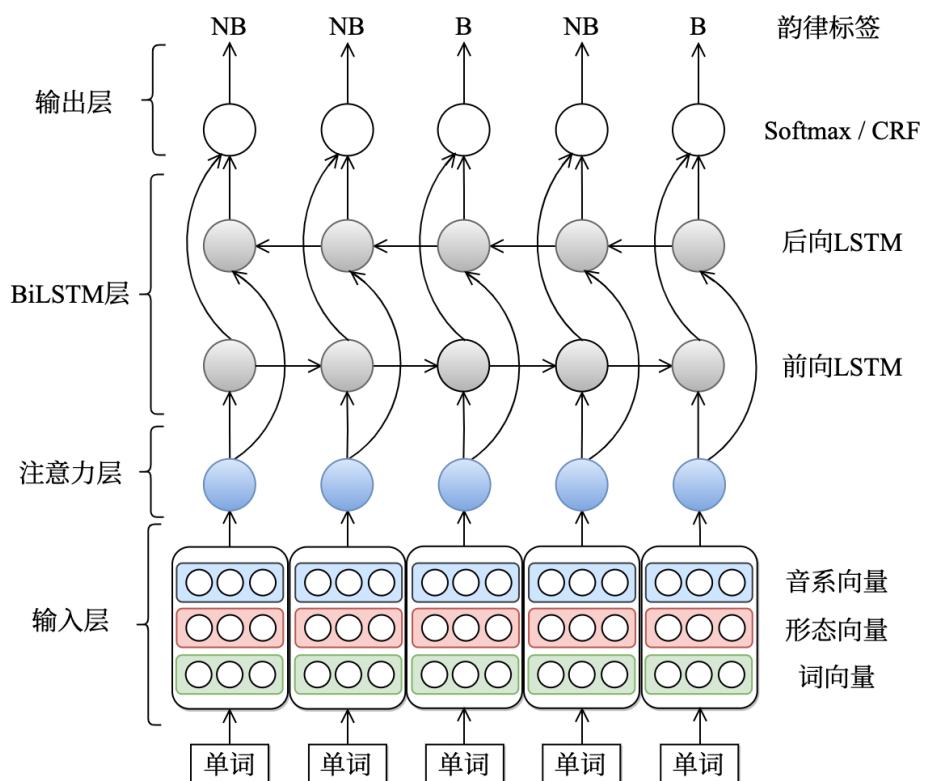


图 3.9 融合形态向量和音系向量的蒙古文韵律建模框架

Fig. 3.9 The block diagram of morphological and phonological embeddings of Mongolian prosody model

该模型使用 3.2.3 节中子词向量表示的学习方法，学习蒙古文单词的形态向量与音系向量，之后将它们作为蒙古文单词向量的辅助，将三种向量融合在一起作为蒙古文单词的特征向量表示。该融合后特征向量输入基于 BiLSTM 的蒙古文韵律模型来计算最终的蒙古文韵律标签。以下首先介绍蒙古文形态向量与音系向量的训练方法，之后介绍融合形态向量与音系向量的蒙古文韵律建模方法的具体结构。

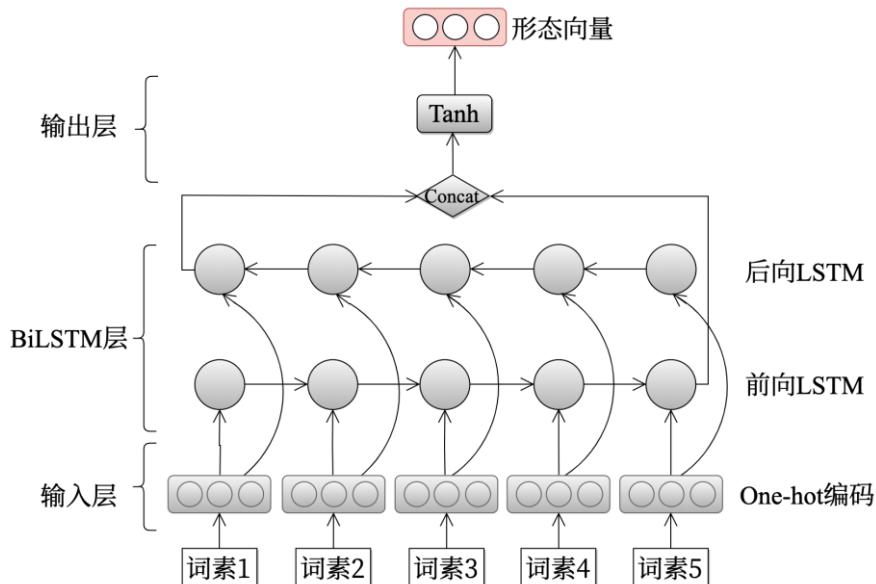


图 3.10 基于 BiLSTM 的蒙古文形态向量训练框架

Fig. 3.10 The block diagram of BiLSTM-based Mongolian morphological embedding model

3.4.1 蒙古文形态向量与音系向量

为了利用 BiLSTM 模型强大的上下文建模能力，本节我们使用 BiLSTM 模型来自由地学习蒙古文单词不用层面的最优向量表征方式^[103]。通过借鉴 3.2.3 节中的子词向量表示学习方法，本节对蒙古文形态向量和音系向量进行学习。

对于蒙古文形态向量，以蒙古文独特的形态学特点为依据，我们先将蒙古文单词转换为其词素序列，与 3.2.3 节不同的是，我们将得到的单词词素序列输入 BiLSTM 模型来抽取更深层次的隐含特征表示，最终输出的向量表示即为“蒙古文形态向量”。如图 3.10 所示为基于 BiLSTM 的蒙古文形态向量抽取模型，一共包含输入层、BiLSTM 层和输出层三部分。

输入层主要对给定的蒙古文单词 W 进行词素切分，得到其词素序列表示 $M = [m_1, m_2, \dots, m_N]$ ，并将其转换为独热向量表示 $X = [x_1, x_2, \dots, x_N]$ 。之后 BiLSTM 层对独

热向量 \mathbf{X} 进行信息提取，BiLSTM 使用前向 LSTM 和后向 LSTM 读取独热向量序列后分别输出隐状态向量 $\vec{\mathbf{H}}$ 和 $\bar{\mathbf{H}}$ ，公式化表示为（3-14）和（3-15）：

$$\vec{\mathbf{h}}_t = f(\vec{\mathbf{W}} \mathbf{x}_t + \vec{\mathbf{U}} \vec{\mathbf{h}}_{t-1} + \vec{\mathbf{b}}) \quad (3-14)$$

$$\bar{\mathbf{h}}_t = f(\bar{\mathbf{W}} \mathbf{x}_t + \bar{\mathbf{U}} \bar{\mathbf{h}}_{t-1} + \bar{\mathbf{b}}) \quad (3-15)$$

其中 $\vec{\mathbf{W}}$ 、 $\vec{\mathbf{U}}$ 、 $\vec{\mathbf{b}}$ 分别是前向 LSTM 的权重矩阵和偏置向量。 $\bar{\mathbf{W}}$ 、 $\bar{\mathbf{U}}$ 、 $\bar{\mathbf{b}}$ 分别代表后向 LSTM 的权重矩阵和偏置向量。之后对前向隐状态和后向隐状态向量进行拼接，得到最终的序列隐状态 \mathbf{H} 送入输出层。公式化表示为（3-16）：

$$\mathbf{H} = [\vec{\mathbf{H}}_t; \bar{\mathbf{H}}_t] \quad (3-16)$$

输出层读取 BiLSTM 层输出的隐状态向量 \mathbf{H} 得到最终的蒙古文形态向量。具体地，输出层使用 Tanh 非线性函数对隐状态向量进行变换降维，得到固定维度的蒙古文形态向量 \mathbf{ME} 。公式化表示为（3-17）：

$$\mathbf{ME} = \text{Tanh}(\mathbf{W}_m \mathbf{H}) \quad (3-17)$$

其中 \mathbf{W} 是输出层的权重矩阵。

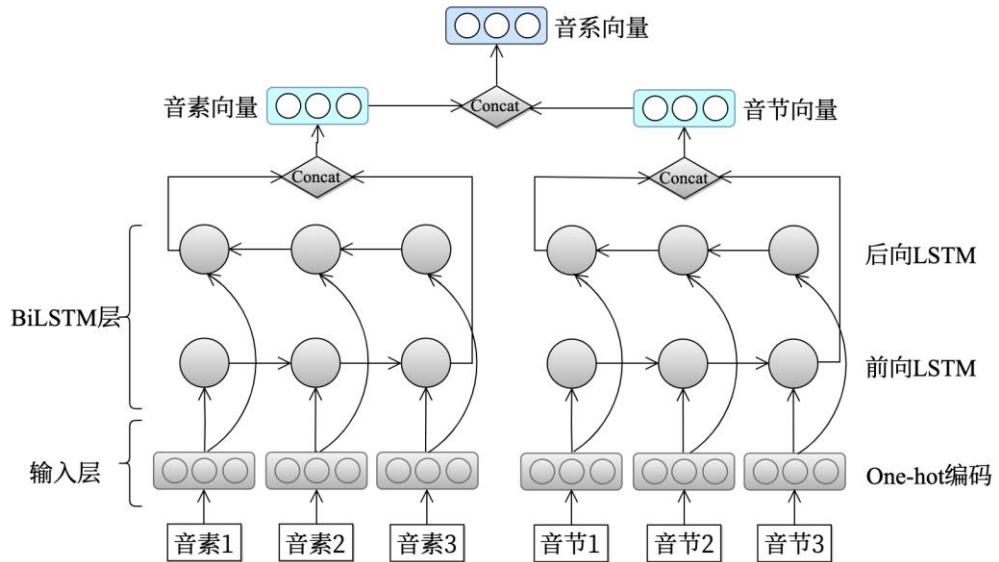


图 3.11 基于 BiLSTM 的蒙古文音系向量训练框架

Fig. 3.11 The block diagram of BiLSTM-based Mongolian phonological embedding model

对于蒙古文音系向量：结合蒙古文独特的音系学特点，我们将蒙古文单词转换为其音素序列和音节序列，从而学习蒙古文单词的音系向量。蒙古文音系向量包括蒙古文音素向量和音节向量，它们分别获取之后再拼接在一起作为蒙古文音系向量。我们使用与上一节相同的 BiLSTM 模型对音素序列和音节序列进行处理。如图 3.11 所示为基于 BiLSTM 的蒙古文音系向量抽取模型。给定蒙古文单词 W，按照 2.3.1 节中的方法获得单词的音素序列 $P=[p_1, p_2, \dots, p_k]$ 和音节序列 $S=[s_1, s_2, \dots, s_k]$ 。需要注意的是，为了充分考虑蒙古文不同位置的音素、音节对蒙古文单词发音的不同影响，我们使用阿拉伯数字对输入的音素序列和音节序列添加了位置标记（Location_Tag），以期更加精细的刻画音系向量。之后使用前向 LSTM 和后向 LSTM 对音系、音节序列的带有位置标签的独热表示向量进行处理，同样将隐状态拼接后通过 Tanh 函数输出音素向量 PhoE 和音节向量 SylE。最终将音素向量和音节向量进行拼接得到蒙古文单词的音系向量。公式化表示为（3-18）、（3-19）和（3-20）：

$$\text{PhoE} = \text{BiLSTM}([p_1; \text{location_tag}], \dots, [p_k; \text{location_tag}]) \quad (3-18)$$

$$\text{SylE} = \text{BiLSTM}([s_1; \text{location_tag}], \dots, [s_k; \text{location_tag}]) \quad (3-19)$$

$$\text{PE} = \text{Tanh}([\text{PhoE}; \text{SylE}]) \quad (3-20)$$

综上所述，结合蒙古语的形态学和音系学特点，利用 BiLSTM 模型的全局上下文建模能力，得到了蒙古文单词的形态向量和音系向量。下一节，我们将充分融合蒙古文形态向量和音系向量，使用基于 BiLSTM 网络的蒙古文韵律建模方法进行蒙古文韵律短语停顿的预测。

3.4.2 BiLSTM 韵律模型

本文基于 BiLSTM 网络搭建了融合形态向量和音系向量的蒙古文韵律模型。如图 3.9 所示，融合形态向量和音系向量的蒙古文韵律模型包括输入层，注意力层，BiLSTM 层和输出层。

输入层即将输入的蒙古文单词通过查找词表找到其对应的词向量 WE、形态向量和音系向量。蒙古文词向量使用 3.2.2 节中的 Skip-Gram 向量模型预先训练得到。之后将这三种蒙古文词级别的特征向量表示送入注意力层进行信息融合。

注意力层输入三种蒙古文单词特征向量，通过加权求和的方式将三种特征向量整合在一起得到新的蒙古文词向量。具体地，给定蒙古文单词的词向量 WE、形态向量 ME、音系向量 PE，它们的权重分别通过两个两层全连接神经网络进行预测，之后将三种向量乘以各自的权重 w_1, w_2, w_3 后拼接在一起即为最终的蒙古文词向量表示 WE^* 。公式化表示为（3-21）公式组：

$$\begin{aligned} w_1 &= 1 - w_2 + w_3 \\ w_2 &= 1 - \sigma(M_1 \text{Tanh}(M_2 \text{WE} + M_3 \text{PE})) \\ w_3 &= 1 - \sigma(M'_1 \text{Tanh}(M'_2 \text{WE} + M'_3 \text{ME})) \\ \text{WE}^* &= w_1 \cdot \text{WE} + w_2 \cdot \text{PE} + w_3 \cdot \text{ME} \end{aligned} \quad (3-21)$$

其中 M_1, M_2, M_3 和 M'_1, M'_2, M'_3 是权重矩阵， $\sigma()$ 是 Logistic 函数，它将计算的结果规范在 0 到 1 之间。通过注意力层的信息融合，可以使最终的蒙古文词向量 WE^* 从不同的信息来源获取最大的信息收益，增强蒙古文词向量表示的鲁棒性。

BiLSTM 层读取输入的特征向量 WE^* 来提取更丰富的高层语义特征。首先使用前向 LSTM 从左到右读取特征向量 WE^* ，依次得到隐状态 \bar{H} 。之后使用后向 LSTM 同样得到隐状态 \bar{H} ，隐状态 \bar{H} 与 H 具有相同的时间长度，最后将对应时间步的隐状态求和得到每一时间步的最终隐状态输出 H 。BiLSTM 层得到的隐状态输出 H 送入输出层解码得到最终的蒙古文单词对应韵律标签。

输出层将 BiLSTM 层输出的隐状态 H 进行解码，本文选择两种输出层函数进行解码。第一种是 Softmax 函数，与上一节方法相同，它可以将输出的向量转化为 0 到 1 之间的概率值，然后对概率值进行归一化，从而找到概率值最大的韵律标签即为最终的韵律标签。但是 Softmax 函数在计算时假设输出的标签之间互相独立，相邻标签的依赖关系没有考虑，计算结果也很容易陷入局部最优解。另一种是条件随机场，对于序列标注问题，标签之间有着很强烈的依赖关系。例如“停顿”标签后一定跟随“非停顿”标签，不可能继续出现“停顿”标签。为了更好的对相邻标签的依赖关系进行描述并且获得全局最优解，CRF 输出层以最大化正确的完整标签序列 $Y = [y_1, y_2, \dots, y_T]$ 为优化目标，公式化表达为（3-22）：

$$E = -s_{(y)} + \log \sum_{y \in Y} e^{s(y)} \quad (3-22)$$

其中, s_y 是输出序列 y 的分数, \tilde{Y} 是输入序列对应的所有可能的输出序列。模型训练使用交叉熵损失函数作为优化目标。

3.5 实验

3.5.1 实验配置

本节实验数据分为蒙古文词(词素)向量训练数据和蒙古文韵律模型训练数据两部分。

蒙古文词(词素)向量训练数据是从蒙古文各大主流网站中爬虫得到。首先对蒙古文网页进行爬取,之后对爬取到的 HTML 文件进行正则化,将其中的网页标签代码删除。之后得到可用的蒙古文词向量训练数据,经统计,该数据含有 2 亿个蒙古文单词,词汇量为 300 万。为了训练词素向量,我们对其进行词干后缀切分,最终得到蒙古文词素向量训练数据,经统计,共包含 3 亿个蒙古文词素,词素级别的词汇量为 250 万。

我们使用 2.3 节搜集整理的蒙古语语音合成语音库中对应的文本语料库以及另外收集扩充的文本数据作为蒙古文韵律模型训练数据。最终整理得到约 5 万 9 千句蒙古文文本,其中包含蒙古文单词约 45 万个,单词词汇量约 22 万个,蒙古文韵律短语约 11 万个,蒙古文韵律短语的平均长度为 3 个蒙古文单词。经音节划分、字母转音素、词干后缀切分后,共包含音节数量约 106 万、音素数量约 188 万、词素数量约 50 万。80% 作为实验训练集,10% 作为实验验证集,另外 10% 作为测试集。

在模型训练时,需要将其转换为序列标注问题标准的标注格式。本文采用“BES”标注格式^[107,108]对标注数据进行处理,其中“B”表示当前单元是一个单词且其处于所在韵律短语的非末尾位置;“E”表示当前单元是一个单词且其处于所在韵律短语的末尾位置;“S”表示当前单元是位于句中或句尾的标点符号,其不属于任何韵律短语。标注示例如图 3.12 所示。

neN	qihvla	ni	,	homun-u	bey_e-yin	eregul	qihirag-tv	tvsalan_a	.
B	B	E	S	B	E	B	B	E	S

图 3.12 采用“BES”格式的标注示例示意图

Fig. 3.12 The schematic diagram of annotated example in "BES" format

实验评价分为客观评价和主观评价。客观评价使用 F 值 (F-Score)。F 值是准确率和召回率的调和平均值，用来平衡这两项指标。其中，准确率 (Precision, P) 表示模型输出的韵律标签中预测正确的标签个数占全部输出标签数量的百分比；召回率 (Recall, R) 表示模型输出的韵律标签中预测正确的标签个数占目标训练样本中全部标签数量的百分比。F 值的数值范围是 0 到 1，F 值越大，表明实验结果越好。它们分别表示为公式 (3-23)、(3-24) 和 (3-25)：

$$P = \frac{Y_c}{Y_a} \times 100\% \quad (3-23)$$

$$R = \frac{Y_c}{Y_{Ta}} \times 100\% \quad (3-24)$$

$$F = \frac{2 \times P \times R}{P + R} \times 100\% \quad (3-25)$$

其中 Y_c 表示实验输出的正确韵律标签个数， Y_a 表示实验得到的所有韵律标签个数；

Y_{Ta} 表示训练数据中的所有韵律标签个数。

主观评价指标包括 2.4.1 节中的 MOS 评测和 A/B 倾向性测试。A/B 倾向性测试要求测听者对听到的两个合成语音文件进行判断，并选择出其中合成质量较好的一个。

3.5.2 建模单元的比较

为了验证基于词素单元的蒙古文韵律建模方法的有效性，本文构建了 9 个系统：

(1) CPw：基线系统，2.3.1.2 小节中以蒙古文单词为建模单元的基于 CRF 的蒙古文韵律建模方法。其中 CRF 的特征模板中包括上下文窗口内邻近的蒙古文单词和单词词性。

(2) CPs：以蒙古文词干为建模单元的基于条件随机场的蒙古文韵律建模方法。该系统将训练数据中的蒙古文单词进行词干后缀切分后，只保留词干作为建模单元。其中条件随机场的特征模板中包括上下文窗口内邻近的蒙古文单词的词干和单词词性。

(3) CPB：以蒙古文词素为建模单元的基于条件随机场的蒙古文韵律建模方法。该系统将训练数据中的蒙古文单词进行词干后缀切分后，将词干和后缀等词素单元全部保留作为独立的建模单元。其中条件随机场的特征模板中包括上下文窗口内邻近的蒙古文

单词词素和所属单词的词性。

(4) **CEw**: 以蒙古文单词为建模单元、以蒙古文单词词向量为特征的基于 CRF 的蒙古文韵律建模方法。该系统在 CPw 系统的基础上，使用蒙古文单词词向量特征替换蒙古文单词词性特征。

(5) **CEs**: 以蒙古文词干为建模单元、以蒙古文词干向量为特征的基于 CRF 的蒙古文韵律建模方法。该系统在 CPs 系统的基础上，使用蒙古文单词词干的对应词干向量特征替换词性特征。

(6) **CEB**: 以蒙古文词素为建模单元、以蒙古文词素向量为特征的基于 CRF 的蒙古文韵律建模方法。该系统在 CPB 系统的基础上，使用蒙古文词素向量特征替换词性特征。

(7) **LEw**: 以蒙古文单词为建模单元的基于 LSTM 的蒙古文韵律建模方法。其中系统输入为蒙古文单词词向量。

(8) **LEs**: 以蒙古文单词词干为建模单元的基于 LSTM 的蒙古文韵律建模方法。其中系统输入为蒙古文词干向量。

(9) **LEB**: 本文提出的基于词素单元蒙古文韵律建模方法。即以蒙古文词素为建模单元的基于 LSTM 的蒙古文韵律建模方法。其中系统输入为蒙古文词素向量。

以上系统中，条件随机场的上下文特征模板均选择 Unigram 和 Bigram 两种特征模板进行比较。建模单元的向量表示均选择五种向量维度进行比较，包括 50、100、150、200 和 300 维。LSTM 的结构均包含一个隐含层，隐含层单元个数是 512 个，初始学习率为 0.01，采用 Glorot Uniform 进行权重的初始化。LSTM 训练 50 步作为最终的模型进行测试。

实验结果如表 3.1、3.2 和 3.3 所示，从表 3.1 可以看出，CPs 系统相比基线系统，在两种特征模板下，性能均略微下降。而 CPB 系统超越另外两个系统，在 Bigram 特征模板下获得了最高的 F 值。可以得知，词干和后缀单元作为独立的建模单元可以更好的学习蒙古文词汇的语义知识，并且一定程度上缓解有限语料规模下数据稀疏问题带来的影响，可以很好地提升蒙古文韵律建模的精度。从表 3.2 可以看出，CEB 系统在所有向量维度中均取得最好的效果，CEB 系统的表现随着向量维度从 50 增加到 300 而先上升后下降，其中向量维度为 150 时表现最好。表明向量维度太小不足以提供丰富的上下文

信息，而向量维度太大又会产生过多的冗余信息。另外，与表 3.1 相比，基于词素向量的方法表现更好。从表 3.3 可以得出与之前一致的结论，LEB 系统在词素向量维度为 150 时取得了最高的 F 值。与基于条件随机场的蒙古文韵律建模方法相比，LSTM 模型拥有更强大的建模能力，更适用于韵律建模任务。

通过分析实验结果得知，由于蒙古文文本数据中严重的数据稀疏问题，导致现有数据中的单词级别的建模单元不能很好的表达蒙古文单词的语义信息，通过将单词级别建模单元转换为粒度更加精细的词素单元，可以很好的缓解数据稀疏对于蒙古文韵律建模的负面影响，对于现有蒙古文韵律建模数据来说，词素单元是更加合适的建模单元。

综上所述，基于词素单元的蒙古文韵律建模方法很好的学习到蒙古文词素的语义知识，提升了蒙古文韵律建模的精度。

表 3.1 不同条件下基于 CRF 的蒙古文韵律建模的 F 值比较

Table 3.1 The experiment results of CRF-based Mongolian prosody model with different condition

系统名称	F 值 (Unigram 模板)	F 值 (Bigram 模板)
CPw	82.23	82.40
CPs	82.12	82.34
CPB	82.52	82.96

表 3.2 不同条件下基于 CRF 且以向量为输入的蒙古文韵律建模的 F 值比较

Table 3.2 The experiment results of CRF-based Mongolian prosody model taking embedding as input in different conditions

系统名称	F 值 (Unigram 模板)					F 值 (Bigram 模板)				
	50	100	150	200	300	50	100	150	200	300
CEw	82.67	82.89	82.85	82.81	82.73	82.86	82.92	82.98	82.87	82.78
CEs	82.67	82.70	82.73	82.68	82.59	82.65	82.73	82.83	82.80	82.79
CEB	83.45	83.59	83.68	83.44	83.53	83.66	83.72	83.79	83.68	83.55

表 3.3 不同条件下基于 LSTM 的蒙古文韵律建模的 F 值比较

Table 3.3 The experiment results of LSTM-based Mongolian prosody model in different conditions

系统名称	F 值				
	50	100	150	200	300
LEw	85.63	85.47	85.89	85.78	85.64
LEs	84.78	84.83	85.01	84.88	84.78
LEB	89.51	89.77	89.89	89.79	88.93

3.5.3 不同向量融合效果的比较

为了验证融合形态向量和音系向量的蒙古文韵律建模方法的有效性。本节构建了 4 个系统：

- (1) WE：以蒙古文单词词向量为输入，基于 BiLSTM 的蒙古文韵律建模方法。
- (2) WE+ME：以蒙古文单词词向量和蒙古文形态向量为输入，基于 BiLSTM 的蒙古文韵律建模方法。
- (3) WE+PE：以蒙古文单词词向量和蒙古文音系向量为输入，基于 BiLSTM 的蒙古文韵律建模方法。
- (4) WE+ME+PE：本文提出的融合形态向量和音系向量的蒙古文韵律建模方法。即以蒙古文单词词向量、蒙古文单词形态向量和蒙古文单词音系向量为输入，基于 BiLSTM 的蒙古文韵律建模方法。

以上系统中，基于 BiLSTM 的蒙古文韵律模型均使用 2 层 LSTM 结构，每个 LSTM 层包含 160 个节点，采用丢弃率为 0.5 的 dropout 技术用来防止模型过拟合。初始学习率设置为 1.0，训练批次大小设置为 64。采用 AdaDelta 优化器训练参数。输出层激活函数采用 softmax 函数。对于形态向量和音系向量训练的 BiLSTM 网络均使用节点数为 200 的 LSTM 层，隐层节点个数为 50。WE 系统中词向量维度为 100；WE+ME 系统中词向量维度和形态向量维度均为 100；WE+PE 系统中词向量维度和音系向量维度均为 100；WE+ME+PE 系统中词向量维度为 100，形态向量和音系向量维度均为 50。

本节采用以上 4 个系统，实验结果如表 3.4 所示。从表 3.4 可知，WE+ME、WE+PE 与 WE 系统相比，均由于融合了蒙古文的语言知识而得到了性能的提升，而 WE+ME+PE 由于对形态向量和音系向量充分的融合，达到了最优的性能表现。

通过分析实验结果得知，由于蒙古文文本资源与英语、汉语等主流语种的文本资源相比资源稀少、获取难度大，另外基于蒙古语复杂的语言特点，使得在此低资源文本数据中训练得到的蒙古文词向量难以学习到丰富的上下文语义知识，因此，使用蒙古文形态向量和音系向量作为蒙古文词向量的辅助，多个信息联合输入可以增强蒙古文单词级别的语义信息，从而提升模型建模精度。

表 3.4 不同向量融合效果的比较结果

Table 3.4 The experiment results of Mongolian prosody model in different conditions

系统名称	F 值
WE	88.58
WE+ME	90.21
WE+PE	90.39
WE+ME+PE	91.05

3.5.4 注意力层有效性的验证

为了验证融合形态向量和音系向量的蒙古文韵律建模方法中，注意力层的有效性。本节基于 WE+ME+PE 系统，对注意力层的注意力机制，进行消融实验。实验结果如表 3.5 所示。

表 3.5 注意力层有效性的验证结果

Table 3.5 The ablation experiment results of attention layers

系统名称	是否添加注意力层	F 值
WE+ME+PE	否	90.41
WE+ME+PE	是	91.05

从表 3.5 中可以看出，注意力层中注意力机制的使用可以很好的使得模型动态学习到不同输入来源的知识，使得模型从多个输入来源中获得最大的信息收益，从而提升模型表现。

3.5.5 不同输出层的比较

为了比较不同输出层对融合形态向量和音系向量的蒙古文韵律建模方法的影响。本节基于 WE+ME+PE 系统，对不同输出层的表现进行实验比较。本节将比较 3.3.2 节中使用 Softmax 和 CRF 作为输出层的两种实现方式，实验结果如表 3.6 所示。

表 3.6 不同输出层的比较结果

Table 3.6 The experiment results of different output layers

系统名称	输出层	F 值
WE+ME+PE	CRF	90.27
WE+ME+PE	Softmax	91.05

从表 3.6 的结果可知, Softmax 输出层与 CRF 输出层相比取得了更高的 F 值, 这与其他序列标注任务中(如词性标注、命名实体识别等)的实验结果并不一致。究其原因, 我们认为这与韵律建模任务的本质特点有关, 以命名实体识别任务为例, 单词的标签包括时间、地点、组织机构、人名等, 由于语法规则的约束, 这些标签在训练数据中分布很均匀, 每个标签的数量也不会有很大差别。这样的情况下, CRF 输出层可以很好的学习到上下文语境中的命名实体标签对当前标签的影响。但是对于韵律建模任务, 单词的韵律标签只有“非停顿”和“停顿”两类。经统计发现, 我们的训练数据中, “非停顿”标签的数量占全部标签的 85%, 因此, 这对 CRF 输出层的学习能力造成很大的负面影响。因此本文实验结果中 Softmax 输出层优于 CRF 输出层, 我们认为是合理的。

3.5.6 对集外词鲁棒性效果的验证

为了验证融合形态向量和音系向量的蒙古文韵律建模方法对集外词具有很好的鲁棒性。本节构建了两个测试集 Test-A 和 Test-B 进行验证。测试集 Test-A 中, 包含 50 句蒙古文文本, 其中没有集外词, 称为“集内测试集”。测试集 Test-B 共有 50 句蒙古文文本, 其中包含 30% 的集外词, 称为“集外测试集”。

表 3.7 对集外词鲁棒性效果的验证结果

Table 3.7 The experiment results of out-of-vocabulary (OOV) problem

系统名称	Test-A	Test-B
WE	88.81	85.96
WE+ME+PE	90.22	89.73

本节对 WE 系统和 WE+ME+PE 系统在两个测试集上的表现进行验证。各个系统的 F 值结果如表 3.7 所示。从表 3.7 的结果可知, WE+ME+PE 系统对于集内测试集 Test-A 和集外测试集 Test-B 的表现均优于 WE 基线系统。另外, 分别对两个测试集的效果比较发现, WE+ME+PE 系统相比 WE 系统在测试集 Test-A 上的 F 值提升了 1.41%, 而在测试集 Test-B 上的 F 值提升了 3.77%, F 值提升幅度要高于 Test-A 测试集, 具有更大的性能提升。原因在于, Test-A 测试集为集内测试集, Test-B 为集外测试集, 本文提出的融合形态向量与音系向量的韵律建模方法可以增强蒙古文词向量的鲁棒性。数据稀疏情况下训练得到的蒙古文单词词向量学习不到充分的语义信息, 形态向量和音系向

量可以作为词向量的辅助信息增强单词词向量的语义信息表达能力。尤其对于集外词的词向量特征来说，形态向量和音系向量提供的辅助信息尤为重要，可以显著增强集外词词向量特征表示的鲁棒性，从而在集外测试集表现出更好的性能提升。因此，我们相信该方法可以很好的缓解集外词问题带来的负面影响。

为了更直观的对模型的学习效果进行说明，我们从 Test-B 测试集中选择出蒙古文句子（拉丁表示）“toro-yin yabvdal-vn hwriyan-v baigvlvmji-yin ogereqilelte-yin tosul-i hinan batvlagsan yabvdal bwl”为例进行具体说明，表 3.8 所示为文本的不同表示形式以及 WE、WE+ME、WE+PE 和 WE+ME+PE 四个系统不同的预测结果。其中灰色背景用来突出显示预测结果中的错误标签，“*”号用来表示单词之间的界限。

表 3.8 不同系统预测效果的举例说明

Table 3.8 An example comparing predicted labels produced by different systems

输入文本	toro-yin yabvdal-vn hwriyan-v baigvlvmji-yin ogereqilelte-yin tosul-i hinan batvlagsan yabvdal bwl
词素序列	toro * yin * yabvdal * -vn * hwriyan * -v * baigvlvmji * -yin * ogereqilelte * -yin * tosul * -i * hinan * batvlagsan * yabvdal * bwl
音素序列	t o r I n * y a b d a s l l i l n * h o r o l n E l * b a e l g l a s l m j I l n * o g o s r q l o s l t t I l n * t o s o s l l I l * h a n a s l n * b a t a s l l s a s l n * y a b d a s l l * b w l t o r o -yin * ya b v d a l -vn * h w r i y a n -v * b a i g v l v m j i -yin * o g e r e q i l e l t e -yin * t o s u l -i * h i n a n * b a t v l a g s a n * y a b v d a l * b w l
音节序列	
真实韵律标签	NB NB NB B NB B NB NB NB B
WE 预测标签	toro-yin [NB] yabvdal-vn [NB] hwriyan-v [B] baigvlvmji-yin [NB] ogereqilelte-yin [B] tosul-i [B] hinan [B] batvlagsan [B] yabvdal [NB] bwl [B]
WE+ME 预测标签	toro-yin [NB] yabvdal-vn [NB] hwriyan-v [NB] baigvlvmji-yin [B] ogereqilelte-yin [NB] tosul-i [B] hinan [B] batvlagsan [B] yabvdal [NB] bwl [B]
WE+PE 预测标签	toro-yin [NB] yabvdal-vn [NB] hwriyan-v [NB] baigvlvmji-yin [B] ogereqilelte-yin [NB] tosul-I [B] hinan [B] batvlagsan [NB] yabvdal [NB] bwl [B]
WE+ME+PE 预测标签	toro-yin [NB] yabvdal-vn [NB] hwriyan-v [NB] baigvlvmji-yin [B] ogereqilelte-yin [NB] tosul-I [B] hinan [NB] batvlagsan [NB] yabvdal [NB] bwl [B]

从表中可以看出 WE 方法对于形态表示和音系表示丰富的单词比较敏感，例如对于单词“baigvlvmji-yin”，“baigvlvmji-yin”和“batvlagsan”等都作出了错误的判断。通过形

态向量和音系向量的融入，WE+ME 系统和 WE+PE 系统都一定程度提升了韵律预测的精度。例如，对于单词“baigvlvmji-yin”和“ogereqilelte-yin”，由于它们具有丰富的形态信息和音系表示信息，可以拆分为多个子词单元的表示，WE+ME 和 WE+PE 正是利用了这样单词内部的丰富信息从而做出了正确的预测。更进一步，WE+ME+PE 综合了形态信息和音系表示信息二者的知识，更加全面的对单词的语义信息进行表示，最终实现了最好的预测效果。通过该例子可以说明，形态向量和音系向量可以为集外词的语义表示提供有价值的信息。正如之前提到，单词“ogereqilelte-yin”的向量表示可以通过借助其单词内部丰富的词素、音素和音节等子词单元来形成更加鲁棒、健壮的语义表示。综上所示，形态向量和音系向量可以很好的为单词向量表示提供辅助信息，从而得到更好的向量表示，这对于蒙古语等形态学特点丰富的低资源语言来说尤为重要。

3.5.7 主观测听实验

为了评价本文提出方法对于提升基于深度神经网络的蒙古语语音合成系统整体质量的效果，我们使用改进的蒙古文韵律模型替换基线深度学习蒙古语语音合成系统中的韵律模型，最终构建了 3 个系统对语音合成的整体质量进行比较：

- (1) **DNN**: 基线系统，即 2.3 节中的基于深度神经网络的蒙古语语音合成系统。其韵律模型采用 CRF 模型，声学模型采用 DNN 模型。
- (2) **DNN-LEB**: 本文改进的基于深度神经网络的蒙古语语音合成系统。其韵律模型采用 3.4 节提出的基于词素单元的蒙古文韵律建模方法，选择其中效果最优的 LEB 系统作为韵律模型具体实现；声学模型采用 DNN 模型。
- (3) **DNN-WMP**: 本文改进的基于深度神经网络的蒙古语语音合成系统。其韵律模型采用 3.5 节提出的融合形态向量和音系向量的蒙古文韵律建模方法，选择其中效果最优的 WE+ME+PE 系统作为韵律模型具体实现；声学模型采用 DNN 模型。

我们从测试集中随机选取 60 句蒙古文文本，选取 10 位以蒙古语为母语的测听者进行语音评价，他们的年龄在 20 岁到 30 岁之间。MOS 评测和 A/B 倾向性测试结果如图 3.11 和 3.12 所示。

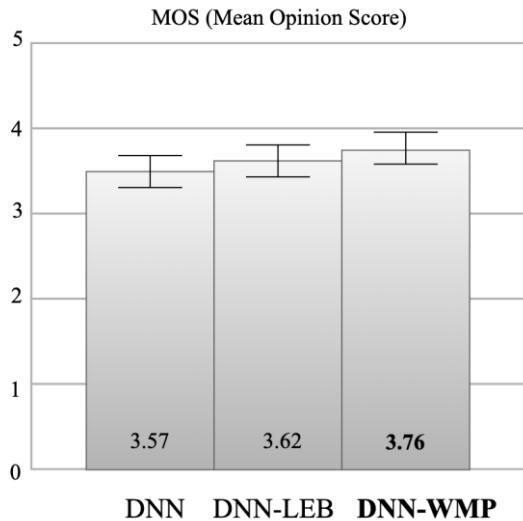


图 3.13 蒙古语 DNN、DNN-LEB 与 DNN-WMP 系统主观 MOS 评测结果（置信度 95%）

Fig. 3.13 MOS scores of speech quality among DNN, DNN-LEB and DNN-WMP for Mongolian language,
with confidence level of 95%

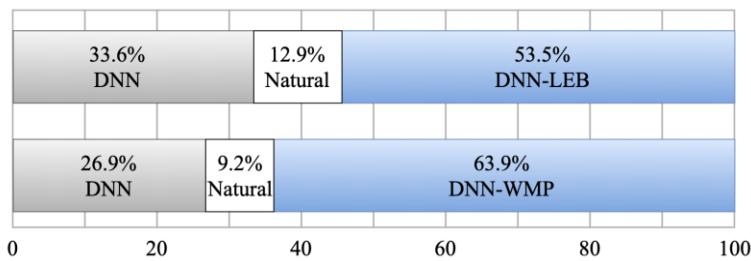


图 3.14 蒙古语 DNN、DNN-LEB 与 DNN-WMP 系统 A/B 倾向性测试结果（置信度 95%）

Fig. 3.14 The results of A/B preference test among DNN, DNN-LEB and DNN-WMP for Mongolian language,
with confidence level of 95%

从主观实验结果图 3.11 可以看出, DNN-WMP 和 DNN-LEB 与 DNN 相比获得了更高的 MOS 分数, 更加被测听者所认可, 其中 DNN-WMP 的表现最好, MOS 分数达到 3.76。图 3.14 可以看出 DNN-WMP 和 DNN-LEB 更加受到测听者的青睐并且 DNN-WMP 占有更大的比重。总结原因在于, (1) DNN-LEB 系统中使用蒙古文词素单元进行韵律建模, 通过转换为粒度更小的语言单元来提升蒙古文韵律建模的精度, 从而提升最终蒙古语语音合成系统的自然度, 与 DNN 系统相比获得了更高的 MOS 分数; (2) DNN-LEB 系统虽然转换了韵律建模粒度, 但是其信息来源仍然比较单一。DNN-WMP 系统吸收了单词级别的形态向量和音系向量提供的辅助语义知识, 进一步增强了蒙古文单词

的语义表达能力，从而提升了蒙古文韵律建模的精度。并且，最终成功在 MOS 分数这一指标中成功超越 DNN-LEB 和 DNN 系统。实验证明，本文提出的融合蒙古文形态学和音系学知识的蒙古文韵律建模方法对于提升基于深度神经网络的蒙古语语音合成系统的合成自然度起到积极的促进作用。

3.6 本章小结

本章深入结合蒙古文的语言文字特点，提出了基于词素单元的蒙古文韵律建模方法和融合形态向量和音系向量的蒙古文韵律建模方法，通过主观和客观实验充分证明了方法的有效性。客观实验中，通过建模单元的比较、不同向量融合效果的比较、不同输出层的比较以及对注意力层有效性和集外词问题的解决效果方面进行了详细的实验比较。主观实验中，将提出的蒙古文韵律模型与基于深度神经网络的蒙古语语音合成模型结合，最终通过实验证明了其对于提升基于深度神经网络的蒙古语语音合成自然度的积极影响。

本章的相关研究工作已经发表在 27th International Conference on Computational Linguistics (COLING2018), 15th Conference of the International Speech Communication Association (InterSpeech2018) 和 15th Pacific Rim International Conference on Artificial Intelligence (PRICAI2018) 国际会议，且其扩展工作已投稿于 SCI 期刊 IEEE Transactions on Audio, Speech and Language Processing (TASLP)。

第四章 基于多任务学习的蒙古文韵律建模方法

4.1 引言

上一章中,我们充分结合蒙古文语言特点和循环神经网络的强大建模能力对基于深度神经网络的蒙古语语音合成系统进行改进,提升了其合成语音的韵律表现。随着多任务学习技术的提出,基于神经网络的多任务学习,尤其是基于深度神经网络的多任务学习在很多自然语言处理领域问题中取得了很大的成功并获得了广泛的应用^[108],比如把词性标注、句子句法成分分析、命名实体识别、语义角色标注等任务进行联合建模。这些任务互相关联,通过联合训练的方式来达到性能提升。

因此,本章将关注重点放在如何引入相关联任务的知识提升蒙古文韵律模型的预测精度,进而提升基于深度神经网络的蒙古语语音合成系统的合成质量。基于此,本章充分利用蒙古文字母音素转换与蒙古文韵律建模两个相关性很强的任务的各自优势,并将“编码器-解码器”网络引入韵律预测建模,将蒙古文韵律建模和蒙古文字母转音素整合到同一个训练框架,提出了基于多任务学习的蒙古文韵律建模方法。

本章首先对相关技术进行介绍,包括“编码器-解码器”网络和多任务学习的理论基础,然后对基于多任务学习的蒙古文韵律建模方法进行具体的方法描述。最后通过实验比较分析验证了方法的有效性。

4.2 相关技术

4.2.1 “编码器-解码器” 网络

传统的单一神经网络中,输入和输出通常都表示为固定长度的向量,如果输入输出长度不相等会使用补零等操作进行对齐。然而在许多重要任务,例如机器翻译、语音识别、对话系统中,输入和输入转换为序列表示后,其序列长度往往不是固定维度的。因此,“编码器-解码器”网络被提出用来突破先前神经网络的局限,可以处理可变长序列问题。“编码器-解码器”网络可以是多种网络结构的组合。它是一种包含两个神经网络的模型,两个神经网络分别扮演编码器(Encoder)和解码器(Decoder)的角色。

Cho 等人提出了一种基于 RNN 的编码器-解码器神经网络用于机器翻译^[110]。编码器和解码器共同实现了整个模型输入序列到输出序列的端到端训练。其网络结构如图 4.1 所示：

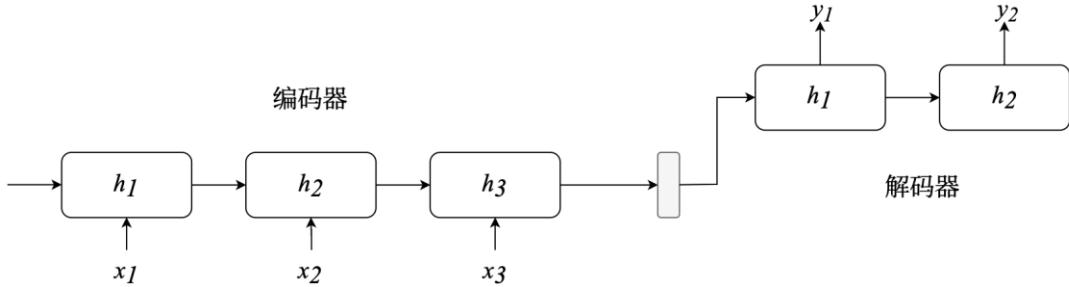


图 4.1 “编码器-解码器”网络结构示意图

Fig. 4.1 The block diagram of “Encoder-Decoder” network

整个模型包含编码器和解码器两部分：编码器将一个可变长度的序列转换成为一个固定长度的向量表示，解码器再将这个固定长度的向量表示转换为一个可变长度的序列。这使得模型可以从一个可变长度序列到另一个可变长度序列的转换。即学习到对应的条件概率 $P(y_1, \dots, y_T / x_1, \dots, x_T)$ ，其中 T 和 T' 的数值可以不相等，即输入序列的长度和输出序列的长度不一定相同。

最简单的编码器实现为循环神经网络，逐次读入输入序列 x 中的每一个元素，其中循环神经网络的隐状态的更新方式如公式（4-1）所示：

$$h_{(t)} = f(h_{(t-1)}, x_t) \quad (4-1)$$

其中， $h_{(t-1)}$ 为前一时刻的隐状态， x_t 为当前时刻的特征输入， h_t 为当前时刻的隐状态。

在读入序列的最后一个元素（通常为结束标记）后，基于循环神经网络的编码器隐状态则为整个输入序列的概括信息 c 。接下来解码器的作用就是将该概括信息解码得到目标序列，一般使用与编码器同样的结构——循环神经网络。基于循环神经网络的解码器根据隐状态 $h'_{(t)}$ 预测下一个元素 y_t ，从而生成整个输出序列。不同于编码器中的循环神经网络，解码器中的循环神经网络的隐状态 $h'_{(t)}$ 除了依赖于上一个隐含层的状态 $h'_{(t-1)}$ 和之前的输出 y_t 外，还依赖整个输入序列的概括信息 c ，如公式（4-2）所示：

$$h'_{(t)} = f(h'_{(t-1)}, y_t, c) \quad (4-2)$$

类似的，下一个输出元素的条件分布如公式（4-3）所示：

$$p(y_t | y_{t-1}, y_{t-2}, \dots, y_1, c) = g(h(t), y_{t-1}, c) \quad (4-3)$$

“编码器-解码器”网络的两部分循环神经网络通过最大化对数似然函数进行联合训练优化，如公式（4-4）所示。

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^N \log p_{\theta}(y_n | x_n) \quad (4-4)$$

其中， θ 为模型的参数， x_n 和 y_n 分别为输入和输出序列的匹配样本。当模型训练完毕后，我们可以利用模型根据给定的输入序列生成相应的输出序列，或是根据给定的输入和输出序列匹配样本计算概率得分 $p_{\theta}(y/x)$ 。

如上所述，“编码器-解码器”网络将输入序列压缩成一个固定长度的向量 c ，但当输入序列长度很长，尤其是明显长于训练集的数据长度时，模型的效果就会显著下降。针对这个问题，Bahdanau 等人提出了一种结合注意力机制的“编码器-解码器”网络结构^[11]。图 4.2 展示了基于注意力机制的“编码器-解码器”网络的工作细节：

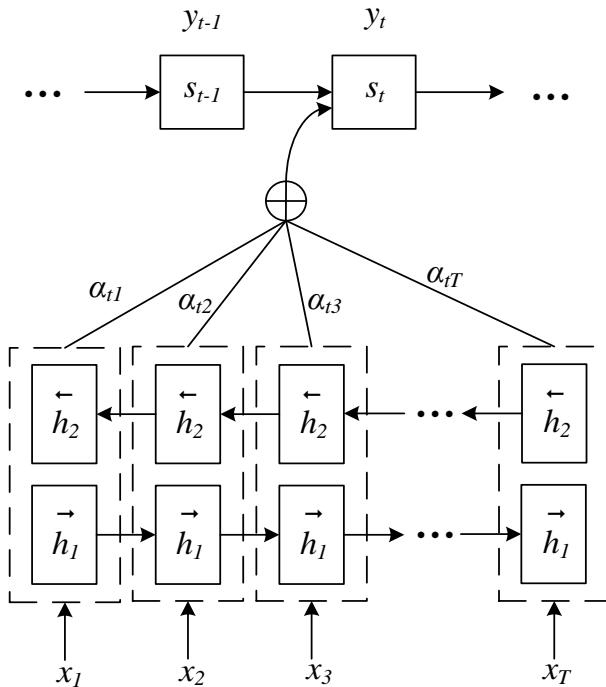


图 4.2 基于注意力机制的“编码器-解码器”网络结构图

Fig. 4.2 The block diagram of attention-based “Encoder-Decoder”network

4.2.2 多任务学习

多任务学习属于迁移学习方法的一种具体形式，Rich Caruana 在文献[111]中总结到：“多任务学习可以利用相关任务的领域知识来实现改善泛化性能的目的”。

多任务学习的目标是充分借助当前任务之外的信息知识来提升当前任务的性能和表现，最终实现泛化准确率的提高、学习速度的加快以及模型可理解性的加强等。如果按照机器学习的观点，多任务学习可以归属于归纳迁移（Inductive Transfer）的一种形式。归纳迁移是一种改进模型的机器学习方法，其原理是将归纳偏置（Inductive Bias）引入模型训练，使得模型更倾向于某些假设。例如，在机器学习领域较为常见的 L1 正则化其实正是一种归纳偏置方法，通过 L1 正则化可以使模型更容易计算得到稀疏的解。而针对具体的多任务学习场景，归纳偏置的实现是通过添加与主任务相关联的辅助任务来实现的。主任务将相关辅助任务的训练数据所包含的潜在的领域知识作为一种已知推导偏差，来实现主任务泛化效果的提升。这样的训练方式可以使得模型较好的计算得到可以同时解释多个任务的解，因此可以提升模型的泛化性能。

综上所述，多任务学习过程中涉及若干个相互关联的任务，这几个任务可以同时并行学习。训练过程中梯度同时进行反向传播，通过底层的共享知识表示，多个任务可以实现互相学习。具体的，可以将底层知识共享表示分为两类：第一种是参数硬共享机制，第二种是参数软共享机制。下面将分别介绍。

(1) 参数硬共享机制：该机制是多任务学习中知识共享的一种常见方式。图 4.3 所示为硬共享机制的示意图。具体的，该机制将若干任务之间的隐藏层参数共享，但是这些任务都保留各自相关的输出层，其中共享的隐含层被称为“共享层”。当同时学习和训练的任务数量越多时，模型就能学习到越多个关联任务之间的同一个隐含知识表示，通过这样的共享方式可以有效降低模型训练过拟合的风险。

(2) 参数软共享机制：软共享机制中每个任务都有属于自己的模型参数，图 4.4 所示为硬共享机制的示意图。为了保证相关联任务之间模型参数的相似性，可以对多个任务中模型参数的距离进行正则化，其中带有正则化操作的网络层被称为“约束层”。

值得注意的是，多任务学习中的主任务和辅助任务必须具有一定的关联性^[113-115]。如 2.2 节所述，音素是蒙古语发音的基本单元，蒙古语发音是由音素决定的，音素序列相比于字符序列能够更准确地表征发音信息。因此，蒙古文字母转音素任务与蒙古文韵

律建模任务具有天然的相关性，基于此，本文选择相关任务，将蒙古文韵律模型任务和蒙古文字母转音素任务进行同时训练，并采用硬共享机制，提出了基于多任务学习的蒙古文韵律建模方法。下一节我们将详细介绍基于多任务学习的蒙古文韵律建模方法的具体内容。

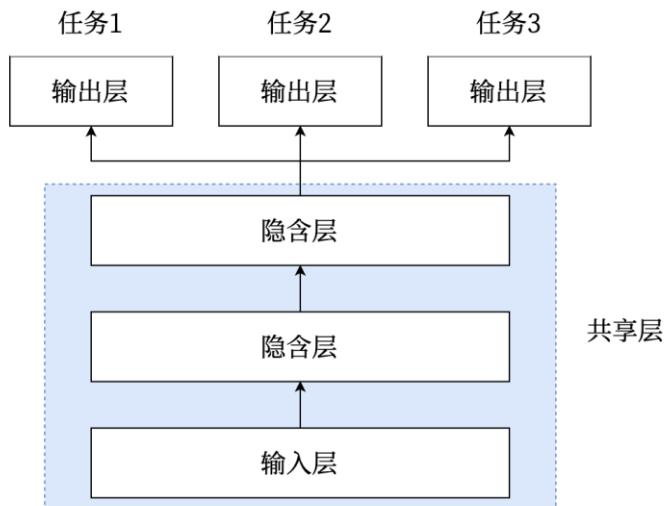


图 4.3 参数硬共享机制示意图

Fig. 4.3 The schematic diagram of hard parameter sharing for multi-task learning

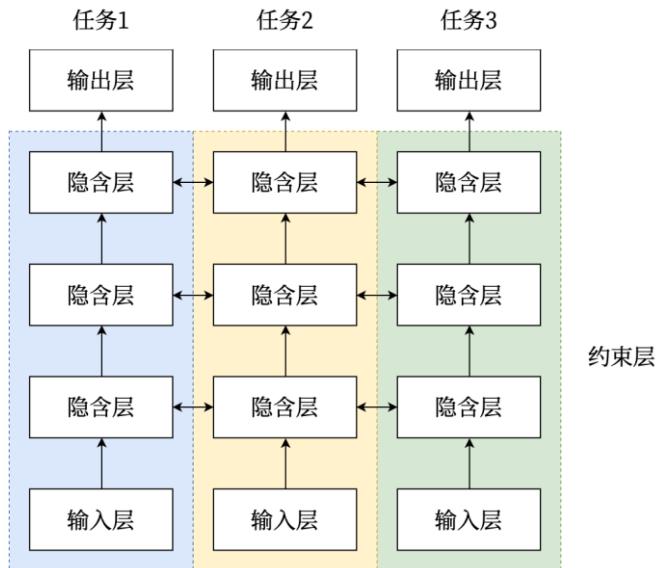


图 4.4 参数软共享机制示意图

Fig. 4.4 The schematic diagram of soft parameter sharing for multi-task learning

4.3 基于多任务学习的蒙古文韵律建模方法

本节我们提出基于多任务学习的蒙古文韵律建模方法，蒙古文韵律模型与蒙古文字母转音素两个任务联合训练。整个模型采用“编码器-解码器”框架，由文本编码器、韵律解码器和音素解码器三部分组成。其中两个任务共享的文本编码器由一层 BiLSTM 网络组成，对输入的蒙古文单词序列进行处理；韵律解码器由一层 LSTM 网络组成，对输入单词序列的韵律标签序列进行解码，输出每个单词对应的韵律标签；音素解码器同样使用一层 LSTM 网络对文本编码器输出信息进行处理，对输入单词序列对应的音素序列进行预测。图 4.5 所示为模型的整体框架图，下面将按顺序介绍模型的内部结构。

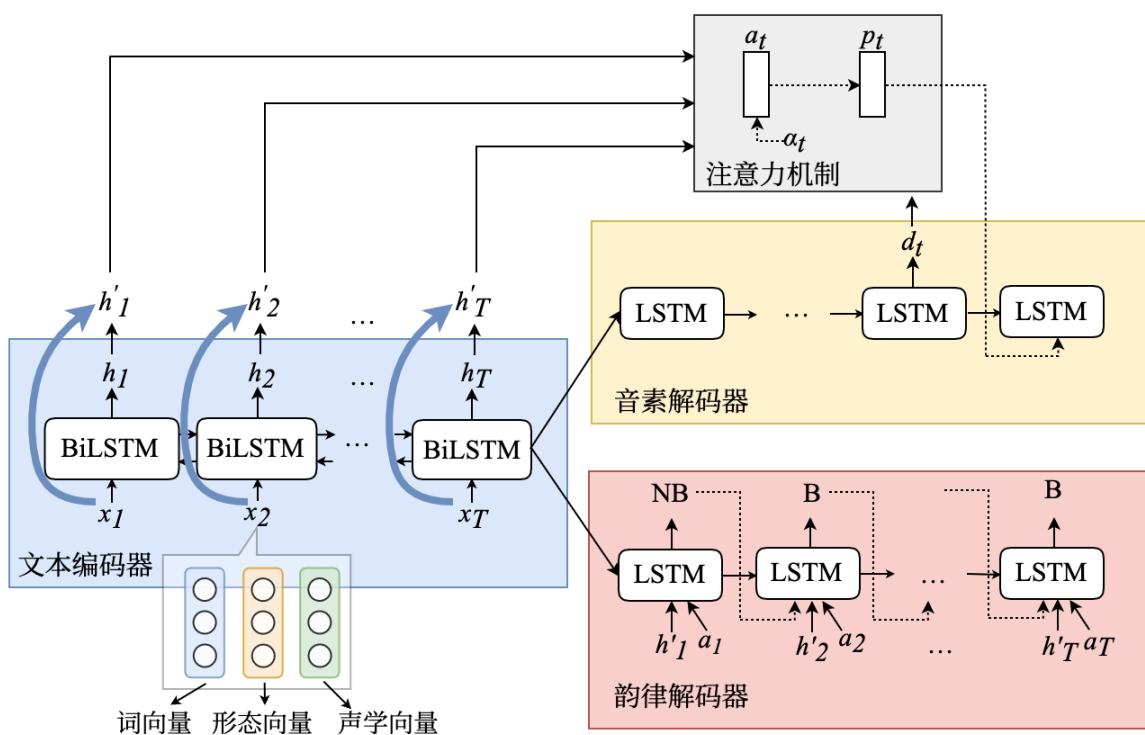


图 4.5 基于多任务学习的蒙古文韵律建模框架

Fig. 4.5 The block diagram of multi-task learning based Mongolian prosody model

4.3.1 文本编码器

文本编码器读取单词序列对应的特征向量进行信息编码，输出编码器隐含向量，之后韵律解码器和声学解码器读取编码器隐含向量解码输出韵律标签序列和音素序列。给

定输入蒙古文单词序列 $\mathbf{W}=[w_1, w_2, \dots, w_T]$, 单词序列对应的韵律标签序列 $\mathbf{Y}=[y_1, y_2, \dots, y_T]$ 和单词序列的目标音素序列 $\mathbf{P}=[p_1, p_2, \dots, p_T]$ 。

具体地, 文本编码器读取单词序列 $\mathbf{W}=[w_1, w_2, \dots, w_T]$ 后, 通过查找词表将其转换为对应的单词特征向量表示 $\mathbf{X}=[x_1, x_2, \dots, x_T]$, 与上一章不同的是, 为了更充分考虑声学参数、蒙古文语法特点与其韵律表现的关系, 这里的单词特征向量表示由三部分组成, 如公式(4-5)所示:

$$\mathbf{X}=[\mathbf{WE}; \mathbf{ME}; \mathbf{AE}] \quad (4-5)$$

其中蒙古文词向量 \mathbf{WE} 、蒙古文形态向量 \mathbf{ME} 和蒙古文单词声学参数向量 \mathbf{AE} 。蒙古文声学参数向量包括蒙古文单词时长特征 dur 、蒙古文单词频谱特征 spe 和蒙古文单词基频特征 fre , 如公式(4-6)所示。

$$\mathbf{AE}=[\text{dur}; \text{spe}; \text{fre}] \quad (4-6)$$

我们将每一个蒙古文单词对应的所有单词语音片段的声学参数进行平均值计算, 这些声学参数都是依照单词的时间边界从真实的标注语音数据中提取得到。之后存为声学参数映射词典, 用于训练阶段和推理阶段声学参数的提取。对于集外词的声学参数, 我们采用随机初始化的方式进行赋值。

文本编码器使用一层 BiLSTM 网络对输入蒙古文单词的特征向量 \mathbf{X} 进行信息编码后输出编码器隐状态向量 \mathbf{H} , 如公式(4-7)所示。

$$\mathbf{H}=\text{Encoder}(\mathbf{X})=\text{BiLSTM}(x_1, \dots, x_T) \quad (4-7)$$

为了缩短编码器与解码器的距离以更好的学习输入序列与输出序列的对齐关系, 我们采用“桥接”(Bridge method)方法^[116]。如图 4.5 中文本编码器中的蓝色箭头所示, 将文本编码器的输入特征向量序列 \mathbf{X} 与隐状态向量 \mathbf{H} 进行拼接得到拼接隐状态向量 \mathbf{H}' , 如公式(4-8)所示。

$$\mathbf{H}'=[\mathbf{X}; \mathbf{H}] \quad (4-8)$$

最后拼接后的向量 \mathbf{H}' 作为最终的文本编码器输出送入韵律解码器和音素解码器进行两个任务的同时解码。

4.3.2 韵律解码器

韵律解码器采用一层 LSTM 网络，利用编码器输出的隐状态向量为输入单词序列预测其正确的韵律标签。在每一个时间步解码时，使用上一个时间步解码器隐状态 s_{t-1} ，上一时间步解码输出的韵律标签 y_{t-1} ，当前时间步编码器隐状态向量 h'_t 以及注意力向量 a_t 进行计算得到当前时间步的解码器隐状态 s_t ，如公式（4-9）所示。

$$s_t = \text{Decoder_PB}(s_{t-1}, y_{t-1}, h'_t, a_t) \quad (4-9)$$

其中注意力向量 a_t 是通过对编码器全部时刻的隐状态向量 h' 进行加权求和计算得到的。由于韵律短语标签的长度与输入单词序列的长度相同，因此这里的注意力向量是一种显式对齐关系，每一个时间步注意力向量 a_t 提供输入序列中当前位置的上下文信息。计算过程如公式组（4-10）所示。

$$a_t = \sum_{j=1}^T \alpha_{i,j} h'_j \quad (4-10)$$

其中分配给编码器输出的全部隐状态的对应权重 $\alpha_{i,j}$ 的计算方式为公式（4-11）。

$$\alpha_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k=1}^T \exp(e_{i,k})} \quad (4-11)$$

$e_{i,k}$ 是某时刻的编码器隐状态与解码器隐状态的相似度得分，其计算方式如公式（4-12）所。

$$e_{i,k} = g(s_{t-1}, h'_k) \quad (4-12)$$

其中函数 g 是一层前馈神经网络。

4.3.3 音素解码器

音素解码器与韵律解码器共享相同的文本编码器，同样以文本编码器输出的拼接隐状态向量 h' 作为输入。在每一个解码时间步，使用上一时间步输出的音素 p_{t-1} 、上一

间步音素解码器的隐状态 s_{t-1} 、以及注意力向量 a_t 进行计算得到当前时间步的输出解码器隐状态 s_t ，如公式（4-13）所示。

$$s_t = \text{Decoder_G2P}(p_{t-1}, s_{t-1}, a_t) \quad (4-13)$$

需要注意的是，这里的注意力向量与韵律解码器中的注意力向量不同，由于字母转音素任务中输入的单词序列与输出的音素序列长度不匹配，因此该注意力向量除了提供每一时间步的上下文信息，还需要学习到单词与音素两种序列的隐式对齐关系^[116-119]。其计算方式如公式（4-14）所示。

$$a_t = \sum_{i=1}^T \alpha_{i,t} h'_i \quad (4-14)$$

其中， $\alpha_{i,t}$ 是一个权重系数，它表示音素解码器在某一时刻进行解码时，文本编码器输出的全部隐状态 h' 中每一时刻对当前解码的重要程度。其计算方式为公式（4-15）和（4-16）所示。

$$\alpha_t = \text{softmax}(u_t) \quad (4-15)$$

$$u_{i,t} = v^T \tanh(\mathbf{W}_1 h_i + \mathbf{W}_2 d_t + b_a) \quad (4-16)$$

其中 v ， b_a 和矩阵 \mathbf{W}_1 、 \mathbf{W}_2 随着其他网络参数一起更新。

最终通过 Softmax 函数对两个解码器输出的隐状态进行解码得到最终各自的输出序列。模型使用两个任务的交叉熵损失函数的和进行联合训练。

4.4 实验

4.4.1 实验配置

本节实验数据分为蒙古文词向量训练数据、蒙古文韵律模型训练数据、蒙古文字母转音素训练数据、蒙古语声学参数映射词典和蒙古语语音训练数据五部分。

蒙古文词向量训练数据、蒙古文韵律模型训练数据来源与 3.4.1 节的训练数据相同。蒙古文字母转音素数据采用蒙古语正字法词典，通过转换蒙古文韵律模型训练数据而得到，蒙古语正字法词典包含词条约 4 万个。蒙古语声学参数映射词典包含词条约 10 万

个，存储单词声学参数维度为 37 维。蒙古语语音数据与 3.5.1 节中的蒙古语语音合成语音库相同。所有训练数据划分为：80% 作为实验训练集，10% 作为实验验证集，另外 10% 作为测试集。

实验评价分为客观评价和主观评价。客观评价指标包括蒙古文韵律建模的评价指标和蒙古文字母转音素的评价指标，韵律评价指标即 3.5.1 节中的 F 值。音素评价指标是词错误率（Word Error Rate，WER）。其公式如（4-17）所示。

$$WER = (1 - \frac{W_c}{W_t}) \times 100\% \quad (4-17)$$

其中 W_c 代表被正确解码输出的蒙古文单词个数， W_t 代表数据中蒙古文单词的总数。

主观评价指标与 2.4.1 节中的 MOS 评测和 A/B 倾向性测试相同。

4.4.2 实验设计

为了验证基于多任务学习的蒙古文韵律建模方法的有效性，本文构建了三个系统：

(1) **G2P**：蒙古文字母转音素模型。该系统只训练蒙古文字母转音素模型，包括文本编码器和音素解码器。

(2) **PB**：蒙古文韵律模型。该系统只训练蒙古文韵律模型，包括文本编码器和韵律解码器。

(3) **G2P&PB**：本文提出的基于多任务学习的蒙古文韵律模型。该系统将两个任务同时训练，包括文本编码器、韵律解码器和音素解码器。

以上系统中，蒙古文词向量的维度为 100 维，形态向量的维度是 100 维。集内词的声学特征从声学参数映射词典中获得，集外词的声学特征通过随机初始化的方式赋值。声学特征一共 37 维，包含 35 维的谱参数 MGC，1 维的对数基频参数 LogF0 和单词时长参数。编码器、解码器中的 LSTM 层单元数为 128，初始化学习率为 1.0，训练批次大小为 64，使用 Adam 优化器进行参数优化。如果模型损失函数在训练步数 10 步之内不下降则停止模型训练。

4.4.3 实验结果与分析

4.4.3.1 客观实验

为了验证本文提出的基于多任务学习的蒙古文韵律建模方法的有效性，我们对上一小节中的四个系统进行测试，实验结果如表 4.1 所示。

如表 4.1 所示，G2P&PB 系统与其他两个独立任务相比，在词错误率和 F 值两个指标上均取得了很好的表现。原因在于，G2P 任务将蒙古文单词转换为音素序列表示，而 PB 任务对蒙古文单词序列的韵律结构进行预测，两者具有天然的相关性，多任务学习很自然的可以对二者任务进行联合建模，通过联合训练，可以充分挖掘 G2P 任务和 PB 任务中隐藏的蒙古语语义知识，使用隐藏的语义知识可以同时促进蒙古文字母转音素和韵律结构预测任务的效果。实验结果证明本文提出的基于多任务学习的蒙古文韵律预测方法可以更好的学习两个相关任务的知识，使得蒙古文字母转音素任务和蒙古文韵律建模任务可以互相促进，共同学习，达到共同的性能提升。

表 4.1 基于多任务学习的蒙古文韵律建模方法的有效性验证结果

Table 4.1 The experiment results of multi-task learning based Mongolian prosody model

系统名称	词错误率 (WER%)	F 值
G2P	19.46	-
PB	-	88.50
G2P&PB	19.32	89.15

4.4.3.2 消融实验

本节采用多种技术以期提升蒙古文韵律建模的表现，为了进一步验证模型内部结构对模型整体性能的影响，本节以 G2P 系统和 PB 系统为基础，对模型内部的桥接方法、形态向量、三种声学特征等因素进行了消融实验。一共产生 32 组系统分别编号为 1 到 32，所有系统的实验结果如表 4.2 所示。

通过观察实验结果，我们得出以下几点结论：

(1) 桥接方法可以将输入信息与模型的编码器隐藏信息结合从而为解码器提供更加丰富的语义信息。我们将编号 1 中只有蒙古文词向量作为输入的系统称为“基线系统”，其词错误率为 24.53%，其 F 值为 83.26%，而系统 17 与系统 1 相比，加入了桥接方法，

表 4.2 基于多任务学习的蒙古文韵律建模方法的消融实验结果

Table 4.2 The ablation experiment results of multi-task learning based Mongolian prosody model

系统 编号	桥接 方法	蒙古文 词向量	蒙古文 形态向量	时长 特征	声学特征		词错误率 (WER%)	F 值
					谱 参数	基频 参数		
1	0	1	0	0	0	0	24.53	83.26
2	0	1	1	0	0	0	24.01	84.13
3	0	1	0	1	0	0	23.85	83.98
4	0	1	0	0	1	0	23.79	84.02
5	0	1	0	0	0	1	23.80	84.13
6	0	1	1	1	0	0	22.98	84.52
7	0	1	1	0	1	0	22.93	84.56
8	0	1	1	0	0	1	22.91	84.61
9	0	1	0	1	1	0	23.71	84.65
10	0	1	0	1	0	1	23.67	84.72
11	0	1	0	0	1	1	23.59	84.76
12	0	1	1	1	1	0	22.58	85.12
13	0	1	1	1	0	1	22.51	85.19
14	0	1	1	0	1	1	22.47	85.24
15	0	1	0	1	1	1	23.35	83.54
16	0	1	1	1	1	1	21.27	86.13
17	1	1	0	0	0	0	22.37	85.32
18	1	1	1	0	0	0	21.98	85.95
19	1	1	0	1	0	0	21.76	85.89
20	1	1	0	0	1	0	21.70	85.86
21	1	1	0	0	0	1	21.63	85.90
22	1	1	1	1	0	0	20.62	86.35
23	1	1	1	0	1	0	20.35	86.39
24	1	1	1	0	0	1	20.29	86.41
25	1	1	0	1	1	0	21.11	86.32
26	1	1	0	1	0	1	21.03	86.43
27	1	1	0	0	1	1	21.15	86.45
28	1	1	1	1	1	0	20.31	87.15
29	1	1	1	1	0	1	20.25	87.21
30	1	1	1	0	1	1	20.29	87.19
31	1	1	0	1	1	1	20.92	87.33
32	1	1	1	1	1	1	19.46	88.50

其词错误率为 22.37%，F 值为 85.32%，在两个指标上分别提升了 2.16% 和 2.06%，相似的，系统 18 与系统 2、系统 19 与系统 3、系统 20 与系统 4、系统 21 与系统 5 和系统 22 和系统 6 等，均由于桥接方法的加入而获得了性能提升。原因在于，桥接方法将输入的蒙古文单词级别特征表示与系统编码器输出的高层次语义特征相融合，形成更丰富的蒙古文单词级别特征向量后送入解码器，在蒙古文单词向量信息不充分的情况下很好的缓解了模型学习压力，使得模型在多任务学习框架下更好的学习到输入序列与输出序列的映射关系，最终提升了韵律建模的性能表现。

(2) 形态向量和声学特征对于两个任务的性能提升均起到了重要的作用。系统 2 到系统 16 等均在系统 1 中仅有词向量输入的基础上加入其他单词级别特征表示，使得模型在词错误率 WER 和 F 值两个指标上均得到了提升。另外，使用桥接方法后，系统 18 到 32 也在系统 17 的基础上加入了形态向量和声学特征，发现系统表现也取得了提升。总结原因在于，单词级别的形态向量和声学特征作为补充特征输入，增强了蒙古文单词表示的鲁棒性，使得模型有更加全面和可靠的信息来源支撑，并最终提升蒙古文韵律模型和蒙古文字母转音素模型的表现。其中值得注意的是，对于蒙古文字母转音素任务来说，声学特征对于模型表现的提升要强于蒙古文形态向量。而对于蒙古文韵律建模任务，声学特征和蒙古文形态特征向量的贡献几乎相同。分析其原因，我们认为这样的现象与两个任务各自的属性相对应，蒙古文字母转音素是将蒙古文单词转换为其音素列表，该任务本来就是更多的关注单词的发音信息。而对于蒙古文韵律建模任务，正如上一章的研究结论所言，蒙古文单词的形态特点和发音特点都对于蒙古语这一天然的黏着语起到重要的作用。

4.4.3.3 主观测听实验

为了验证基于多任务学习的蒙古文韵律建模方法对于提升蒙古语语音合成自然度的有效性。我们使用改进的蒙古文韵律模型替换基线深度学习蒙古语语音合成系统中的韵律模型，最终构建了 2 个深度学习蒙古语语音合成系统对合成结果进行比较：

- (1) **DNN**: 基线系统，即 2.3 节中的基于深度神经网络的蒙古语语音合成系统。其韵律模型采用 CRF 模型，声学模型采用 DNN 模型。
- (2) **DNN-WMP**: 该蒙古语语音合成系统采用 3.5 节提出的融合形态向量和音系向

量的蒙古文韵律建模方法构建韵律模型，采用 DNN 模型构建声学模型。

(3) DNN-MTL：该蒙古语语音合成系统采用 4.3 节提出的 G2P&PB 模型构建韵律模型，采用 DNN 模型构建声学模型。

我们从测试集中随机选取 60 句蒙古文文本，选取 10 位以蒙古语为母语的测听者进行语音评价，他们的年龄在 20 岁到 30 岁之间。MOS 评测和 A/B 倾向性测试结果如图 4.6 和 4.7 所示。

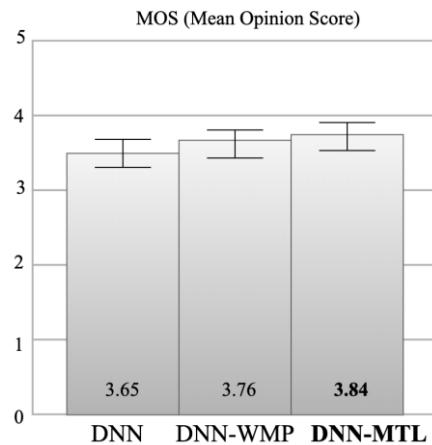


图 4.6 蒙古语 DNN、DNN-WMP 与 DNN-MTL 系统主观 MOS 评测结果（置信度 95%）

Fig. 4.6 MOS scores of speech quality between DNN, DNN-WMP and DNN-MTL for Mongolian language,
with confidence level of 95%

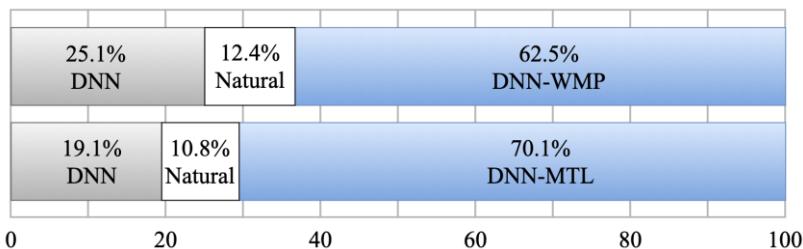


图 4.7 蒙古语 DNN、DNN-WMP 与 DNN-MTL 系统 A/B 倾向性测试结果（置信度 95%）

Fig. 4.7 The results of A/B preference test between DNN, DNN-WMP and DNN-MTL for Mongolian language, with confidence level of 95%

通过主观实验结果可以看出，DNN-MTL 的 MOS 分数达到 3.84，DNN 系统的分数略低，达到 3.65，DNN-MTL 的合成表现明显优于 DNN 系统；另外将本章提出的 DNN-MTL 与第三章的最优系统 DNN-WMP 比较发现，DNN-MTL 仍然在 MOS 上表现出了较优的合成表现。在 A/B 倾向性测试中，先对 DNN 与 DNN-WMP 系统进行比较，DNN 系统占 25.1% 百分比，DNN-WMP 系统占 62.5% 百分比，中立选项占比 12.4%；之后比

较 DNN 系统与 DNN-MTL 系统，发现 DNN-MTL 比重占到 70.1%，DNN 占比 19.1%，中立选项占比 10.8%，这个比重差距也体现出 DNN-MTL 系统明显更加受到测听者的认可。分析原因在于，DNN-MTL 以基于多任务学习的蒙古文韵律建模为基础，使用多任务学习的机制将蒙古文字母转音素任务和蒙古文韵律建模任务联合训练，使得蒙古文韵律建模任务获得了更高的精度，因此蒙古文文本特征表示包含了更加准确的韵律信息，使得模型生成出更加富有表现力的语音参数，并最终产生自然度更高的合成语音。这验证了本文提出的基于多任务学习的蒙古文韵律建模方法，可以很好的提升基于深度神经网络的蒙古语语音合成系统的合成自然度。

4.5 本章小结

本章提出了基于多任务学习的蒙古文韵律建模方法，将蒙古文韵律建模与蒙古文字母转音素两个任务联合建模，通过联合训练的方式提升模型的整体性能，并最终提升基于深度神经网络的蒙古语语音合成系统的合成自然度。分别从客观实验、消融实验、主观实验三个方面对模型效果进行了详细分析。实验结果表明，基于多任务学习的蒙古文韵律建模方法可以充分利用相关任务的隐含知识来提升模型预测的精度，最终使得基于深度神经网络的蒙古语语音合成自然度有了进一步提升。

本章的相关研究工作已经发表在 33rd International Conference on Neural Information Processing（ICONIP2019）国际会议。

第五章 基于知识蒸馏的端到端声学建模方法

5.1 引言

在前两章中，我们融合蒙古语语言特点并利用多任务机制，提升了蒙古文韵律建模的精度，提升了基于深度神经网络的蒙古语语音合成系统的韵律表现。但是这些研究都只属于神经网络语音合成框架的韵律模型，基于深度神经网络的语音合成方法与传统语音合成方法都是由一系列复杂的子模块串联构成，前端模块和后端模块的串联导致前期工作的误差会随着流水线不断积累而放大，导致信息损失从而限制了合成语音质量的进一步提升。因此，第五章和第六章将关注重点放在基于端到端声学建模的蒙古语语音合成，以期提升蒙古语语音合成系统的整体表现。端到端声学建模方法凭借其简洁的模型结构、语言学知识的弱依赖性以及不需要人工干预的优势，可以生成接近真人的合成语音质。如 2.5 节所述，基于端到端声学建模的蒙古语语音合成方法采用基于“编码器-解码器”的端到端声学模型实现了从蒙古文文本字符到语音频谱参数的端到端训练。其中端到端声学模型的声学解码器采用逐帧预测的方式预测频谱参数，这样的自回归解码方式会造成严重的曝光偏差问题^[120,121]，从而影响端到端蒙古语语音合成系统的鲁棒性。具体地，在模型训练阶段，声学解码器以真实频谱参数作为输入预测下一时间步的频谱参数，我们称之为“教师激励（Teacher-Forcing）”解码模式。而在模型推理阶段，声学解码器没有真实频谱参数作为参考，只能以上一时间步预测得到的频谱参数作为输入来计算当前时间步的频谱参数，我们称之为“自由运行（Free-Running）”解码模式。

图 5.1 展示了不同解码模式的具体示意图。

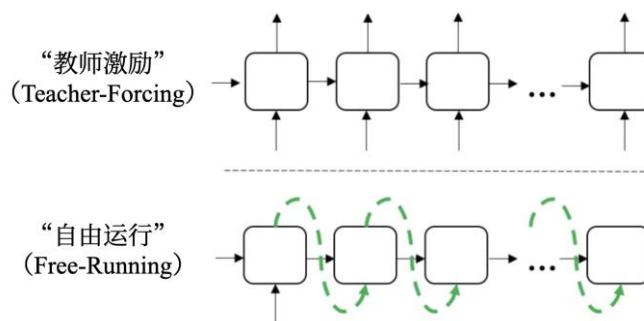


图 5.1 “教师激励”与“自由运行”解码模式示意图

Fig. 5.1 The schematic diagram of “Teacher-Forcing” and “Free-Running” decode method

不同的解码模式导致模型训练阶段与推理阶段存在严重的不匹配问题,这一问题导致模型在对输入字符序列进行解码时出现不可预测的错误发生^[121-123]。尤其是在处理长文本数据或领域外文本数据时,会造成合成语音中出现跳词、漏词、重复等现象。严重影响了端到端声学模型的鲁棒性,影响合成语音的自然度。本章从端到端蒙古语语音合成模型的端到端声学模型出发,以提升端到端蒙古语语音合成的鲁棒性和自然度为目标,提出了基于知识蒸馏策略的端到端声学建模方法。

本章首先介绍知识蒸馏策略的理论基础,然后对基线声学建模方法进行介绍,之后对基于知识蒸馏策略的端到端声学建模方法进行具体的方法描述。最后通过实验比较和具体的实验分析验证了方法的有效性。

下一节我们将详细介绍基于知识蒸馏的端到端声学建模方法的具体内容。

5.2 端到端声学建模方法

随着深度学习技术的不断发展,基于“编码器-解码器”的端到端语音合成方法逐渐成为主流,端到端语音合成方法使得合成系统尽量简化,减少了人工干预和对语言学相关背景知识的要求。端到端声学建模方法直接输入字符或者音素序列,系统输出语音波形。前端文本分析模块得到了极大简化,甚至可以直接忽略,因此无需对语音数据进行详细标注。借助深度学习模型强大的表达能力,端到端语音合成系统表现出令人惊艳的合成效果,其中 Google 发布的基于“编码器-解码器”的语音合成模型 Tacotron、Tacotron2^[51,52]及在此基础上的很多变形结构均以其直观简洁的模型框架获得了广泛的应用。

本节采用基于 Tacotron2 模型的端到端声学建模方法搭建基于端到端声学建模的蒙古语语音合成系统,作为本章的基线系统。基于端到端声学建模的蒙古语语音合成系统延续了 Tacotron2 模型的优点,无需复杂的文本模块,通过“编码器-解码器”网络直接学习蒙古文字符到蒙古语声学参数的映射关系,实现声学模型的端到端训练。端到端蒙古语语音合成模型以蒙古文字符的拉丁表示作输入,以梅尔频谱参数作输出,其模型整体框架如图 5.2 所示,包括“端到端声学模型”和“声码器”两部分。

基于端到端模型的声学建模方法直接以蒙古文字符的拉丁表示作为输入,对输入蒙

古文文本进行 2.4.1.1 小节所述的正则化处理后，输入端到端声学模型。

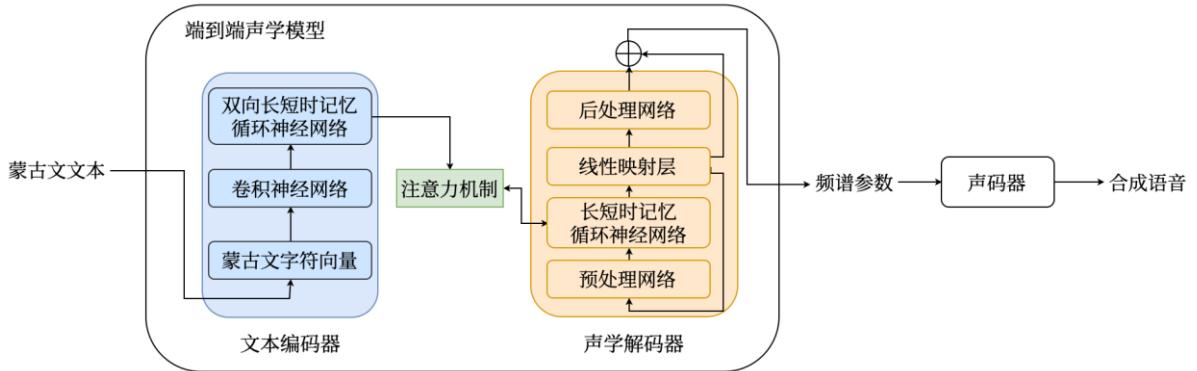


图 5.2 基于端到端声学建模的蒙古语语音合成模型框架图

Fig. 5.2 The block diagram of end-to-end Mongolian TTS model

端到端声学模型是整个端到端蒙古语语音合成模型的核心。端到端声学模型采用“编码器-解码器”框架，也叫做频谱特征预测模型，主要包括文本编码器和包含注意力机制的声学解码器。其中文本编码器包括 3 层卷积神经网络与 1 层双向长短时记忆循环神经网络。文本编码器输入给定的蒙古文文本，文本中的字符序列首先转换为独热编码（One-hot Vector），之后通过查找词表将其转换为浮点数向量 $\mathbf{X} = [x_1, x_2, \dots, x_T]$ 。最后将输入字符向量 \mathbf{X} 通过 3 层卷积神经网络提取上下文信息，接着送入 1 层双向的长短时记忆网络 BiLSTM 生成文本编码器的隐状态，即公式（5-1）和（5-2）：

$$f_e = \text{ReLU}(F_3 * \text{ReLU}(F_2 * \text{ReLU}(F_1 * x))) \quad (5-1)$$

$$\mathbf{H} = \text{BiLSTM}(f_e) \quad (5-2)$$

其中， F_1 、 F_2 、 F_3 为三个卷积核，ReLU 为每一个卷积核上的非线性激活函数。 $*$ 号表示卷积运算操作。

借助于卷积神经网络强大的上下文感知能力以及循环神经网络捕获长时依赖的优秀表现，文本编码器的隐状态 $\mathbf{H} = [h_1, h_2, \dots, h_T]$ 中具有更高级别的上下文隐含表示信息。我们将该隐状态 \mathbf{H} 作为文本编码器的输出送入到基于注意力机制的声学解码器来对频谱声学参数进行预测。

基于注意力机制的声学解码器是整个端到端蒙古语语音合成模型的关键模块，基于注意力机制的声学解码器主要包括注意力机制、预处理网络、长短时记忆神经网络、线

性映射层和后处理网络等。注意力机制主要用作编码器和解码器的桥接，在端到端语音合成系统中，声学解码器采用位置敏感注意力机制^[124,125]。与 4.2.1 小节中注意力计算方式不同，传统的注意力机制计算方式只利用了解码器在 $j-1$ 时刻的目标隐状态 s_{i-1} 和编码器的输出隐状态 h_j ，而在位置敏感注意力机制中计算方式如公式（5-3）所示。

$$e_{ij} = \text{score}(s_i, c\alpha_{i-1}, h_j) = v_a^T \tanh(\mathbf{W} s_i + \mathbf{V} h_j + \mathbf{U} f_{ij} + \mathbf{b}) \quad (5-3)$$

其中， v_a 、 \mathbf{W} 、 \mathbf{U} 、 \mathbf{V} 和 \mathbf{b} 为带训练权重参数， s_i 为当前解码器的隐状态， h_j 是当前解码器的隐状态，偏置值 \mathbf{b} 被初始化为 0， f_{ij} 是之前的注意力权重 α_{i-1} 经过卷积而得到的位置特征（Location Feature），如公式（5-4）和（5-5）所示：

$$f = \mathbf{F}^* c\alpha_{i-1} \quad (5-4)$$

$$c\alpha_i = \sum_{j=1}^{i-1} \alpha_j \quad (5-5)$$

最后使用公式（5-6）得到注意力向量 c ：

$$c_i = \sum_{j=1}^T \alpha_{ij} h_j \quad (5-6)$$

由于引入了这样的注意力机制，使得模型在计算当前适合的注意力权重时，会更加有效地计算出权重分布。这样的注意力机制可以同时考虑内容和输入元素的位置信息，可以更好的约束输入字符与输出频谱参数的对应关系，减少由于对齐出错产生的发音错误，另外还可以加快模型收敛。

预处理网络包含两个全连接隐含层，均使用 ReLU 函数作为激活函数。为了缓解自回归声学解码器的曝光偏差问题，预处理网络输入频谱参数时，采用计划采样（Scheduled Sampling, SS）方法^[127]，每个时间步都以一定的概率来决定当前时间步的输入是上一个解码时间步的估计频谱参数还是上一时间步的真实语音频谱参数。之后其输出与上一个解码时间步计算得到的注意力向量 c 做拼接，然后送入长短时记忆网络。预处理网络可以提升模型的泛化能力以及收敛速度。

长短时记忆神经网络包括 2 层 LSTM，它以预处理网络输出的拼接向量作输入，其输出用来计算新的注意力向量，最后新计算出的注意力向量与长短时记忆神经网络的输

出通过残差连接拼接到一起送入线性映射层以预测下一解码时间步的频谱参数输出。同时线性映射层还会输出该解码步是否结束的概率来决定是否停止解码过程。

后处理网络包含 5 层卷积神经网络，它用来进一步提升频谱参数的生成质量。后处理网络通过预测一个残差信号，然后与卷积前的频谱参数做叠加，得到最终预测的频谱参数。该频谱参数通过声码器还原为语音波形信号。

最终模型使用后处理网络处理前后梅尔频谱预测的均方误差 MSE、结束标签的交叉熵以及正则项作为损失函数，如公式（5-7）所示。

$$Loss = \frac{1}{n} \sum_{i=1}^n (y_{real,i} - y_i)^2 + \frac{1}{n} \sum_{i=1}^n (y_{real,i} - y_{final,i})^2 + \lambda \sum_{i=1}^p w_j^2 \quad (5-7)$$

其中， $y_{real,i}$ 是目标语音的真实频谱参数， y_i 、 $y_{final,i}$ 是分别进入后处理网络前、后的频谱参数， n 为 batch 中的样本数， λ 为正则化参数， p 为参数总数， w 为神经网络中的参数。

声码器通过 Griffin-Lim 语音重建算法实现^[128]。由于梅尔频谱相比其他声学参数包含了更多的语音信号低频细节，所以梅尔频谱属于较低层次的声学特征。而频谱幅度与梅尔频谱相比包含了更多的高频信息，与原始语音波形更为接近。如果直接使用声学模型预测得到的梅尔频谱进行语音波形的恢复很难实现与真实语音接近的合成结果。因此 Griffin-Lim 算法首先将梅尔频谱转换为频谱幅度，然后根据得到的频谱幅度进行相位预测，最终对语音波形进行恢复输出最终的合成蒙古语语音。

传统的端到端语音合成方法采用计划采样解码方法对端到端声学模型进行训练，它是“教师激励”与“自由运行”解码方式的一种折中解决方案，可以一定程度上缓解曝光偏差问题。但是语音信号是一种连续的时序信号，这样的解码方式会破坏输出声学参数的连续性，会对合成语音的质量带来负面影响。因此，我们提出了基于知识蒸馏的端到端声学建模方法。

5.3 基于知识蒸馏的端到端声学建模方法

本节我们提出基于知识蒸馏的端到端声学建模方法。2014 年，Hinton 在其文章中首次提出了知识蒸馏的概念^[129]。知识蒸馏的主要思想是训练一个较小的网络模型来模仿

一个预先训练好的大型网络或者集成的网络。这种训练模式又被称为“教师-学生”训练。它将复杂的、学习能力强的教师模型学到的特征表示“知识”蒸馏出来，传递给参数量小、学习能力弱的学生模型。具体地，先预训练一个从真实数据中学习参数分布的教师模型，之后在训练学生模型的过程中添加一个以最小化教师模型预测结果的概率分布为目标的损失函数，这个损失函数被称为“蒸馏损失（Distillation Loss）”。最终实现学生模型对教师模型学到的知识进行学习，以提升学生模型的建模能力。

借鉴“知识蒸馏”的核心思想，本节提出了基于知识蒸馏的端到端声学建模方法。图 5.3 所示为该方法的模型框架图。该方法采用“教师-学生”训练框架。教师模型和学生模型具有与 5.3 节相同的模型结构，但是在声学解码器中使用了完全不同的解码模式。其中教师模型采用基于“教师激励”解码模式的声学解码器，学生模型采用基于“自由运行”解码模式的声学解码器。推理阶段只需要使用训练好的学生模型进行合成。下面将对教师声学模型、学生声学模型以及基于知识蒸馏的端到端声学模型训练方法进行详细介绍。

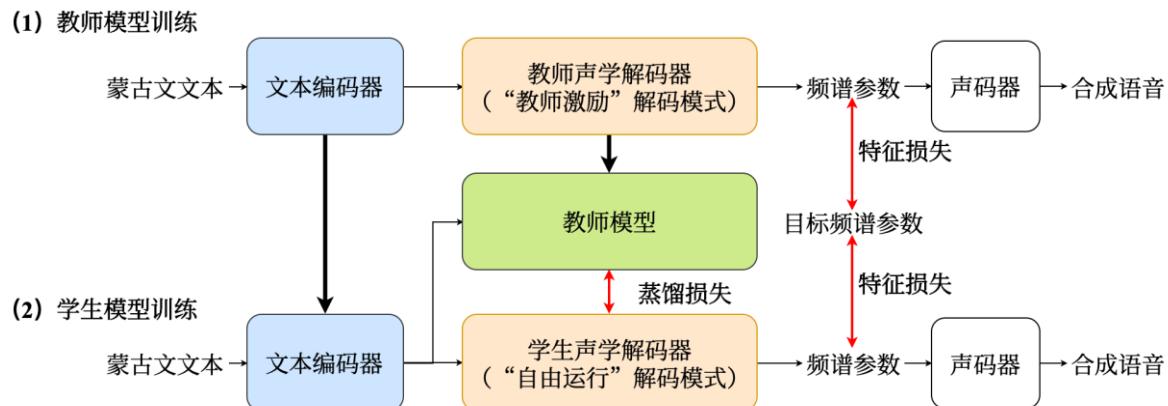


图 5.3 基于知识蒸馏的端到端声学建模框架图

Fig. 5.3 The block diagram of knowledge distillation based end-to-end Mongolian TTS model

5.3.1 教师模型

教师模型采用与 5.3 节相同的文本编码器和基于“教师激励”解码模式的声学解码器。文本编码器读取蒙古文输入字符序列，之后声学解码器在进行每一步解码时，都以

前一时间步的真实频谱参数作为输入来预测当前时间步的频谱参数。假设 $P(y/x, \theta)$ 是教师模型的参数，其中 x 是输入的蒙古文字符序列， y 是目标的频谱参数序列， \hat{y} 是预测得到的频谱参数序列。那么整个模型的优化过程可以公式描述为（5-8）：

$$P(\hat{y}/x, \theta) = \prod_{t=1}^{T'} P(\hat{y}_t / y < t, x, \theta) \quad (5-8)$$

教师模型采用“教师激励”解码模式，将真实语音参数作为解码器的输入进行解码，可以从目标语音数据中学习到更真实的声学参数分布，使得教师模型声学解码器隐状态具有很强的表征能力。

5.3.2 学生模型

学生模型的文本编码器与教师模型的文本编码器保持一致，声学解码器的解码方式采用“自由运行”模式。文本编码器读取蒙古文输入字符序列，之后声学解码器按照模型在推理阶段的解码方式，使用上一时间步预测得到的频谱参数，输入到当前时间步进行当前时间步频谱参数的预测，每一时间步的频谱参数预测结果都直接依赖于上一时间步生成的频谱参数。假设 $P(y/x, \theta)$ 是教师模型的参数，其中 x 是输入的蒙古文字符序列， \hat{y} 是预测得到的频谱参数序列。那么整个模型的优化过程可以公式化表示为（5-9）：

$$P(\hat{y}/x, \theta) = \prod_{t=1}^{T'} P(\hat{y}_t / \hat{y} < t, x, \theta) \quad (5-9)$$

学生模型的解码过程只依赖于每一时间步的频谱参数预测结果，因此其声学解码器隐状态的表征能力明显不足，我们相信，对于使用“自由运行”解码模式的学生模型来说，表征能力更强的基于“教师激励”解码模式的模型可以充当很好的教师模型角色进行知识蒸馏，下一节我们将详细介绍“教师-学生模型”的知识蒸馏训练具体过程。

5.3.3 模型训练

模型训练过程一共包括两个步骤，第一步是教师模型训练，第二步是学生模型训练。第一步，先对教师模型的文本编码器和声学解码器进行预训练，训练的损失函数与5.2节的损失函数保持一致，本节中记作 $Loss_f$ 。

第二步，使用教师模型中训练好的文本编码器的参数对学生模型中文本编码器的参数进行初始化，之后以教师模型声学解码器的隐状态输出为目标，在原有损失函数的基础上添加一个“蒸馏损失”使得学生模型的声学解码器隐状态输出逼近教师模型声学解码器的隐状态输出，使得学生模型的声学解码器可以输出更符合真实语音参数分布的隐状态向量，从而得到更真实的频谱参数。

具体的，给定蒙古文字符输入序列 $\mathbf{X} = [x_1, x_2, \dots, x_T]$ 和目标语音频谱参数序列 $\mathbf{Y} = [y_1, y_2, \dots, y_T]$ 。对于教师模型文本编码器，编码器对输入序列 \mathbf{X} 的上下文信息进行处理，得到编码器输出的隐状态向量 $\mathbf{H} = [h_1, h_2, \dots, h_T]$ ，公式表示为（5-10）：

$$h_t = \text{Encoder}(h_{t-1}, x_t) \quad (5-10)$$

之后将编码器输出的隐状态 h_t 同时送入教师模型声学解码器（Decoder_T）和学生模型（Decoder_S）声学解码器。在教师模型声学解码器中，在每一个时间步解码时，解码器都使用上一个时间步的解码器隐状态 s_{t-1} 、上一时间步的真实频谱参数 y_{t-1} 和当前时间步的隐状态 h_t 作为输入计算当前时间步的解码器隐状态 s_t ，公式表示为（5-11）：

$$s_t = \text{Decoder_T}(s_{t-1}, y_{t-1}, \sigma(h_t)) \quad (5-11)$$

而在学生模型声学解码器中，每一个时间步解码的输入包括上一个时间步的解码器隐状态 s_{t-1} 、上一时间步输出的估计频谱参数 \hat{y}_{t-1} 和当前时间步的隐状态 h_t ，来计算得到 s_t ，公式表示为（5-12）：

$$\hat{s}_t = \text{Decoder_S}(\hat{s}_{t-1}, \hat{y}_{t-1}, \sigma(h_t)) \quad (5-12)$$

为了实现学生模型声学解码器的隐状态逼近教师模型声学解码器的隐状态以实现知识蒸馏。我们在原来 $Loss_f$ 的基础上，添加了一个蒸馏损失函数 $Loss_d$ ，公式表达如（5-13）：

$$Loss_d = \frac{1}{T} \sum_{t=1}^T \| s_t - \hat{s}_t \|^2 \quad (5-13)$$

通过对两个代价函数进行加权求和得到总代价函数 $Loss$ ，公式表示为（5-14）：

$$Loss = Loss_f + \lambda \cdot Loss_d \quad (5-14)$$

其中权重 λ 是一个超参数，用来平衡两个代价函数的数值差异。

最终，我们使用总代价函数进行学生模型的训练。学生模型训练好之后，在推理阶段，可以直接使用学生模型进行合成。

5.4 实验

本节将对本文提出的基于知识蒸馏的端到端蒙古语语音合成建模方法的合成语音表现进行评估，由于该方法与语种并没有直接的关系，为了证明该方法在多个语种的通用性，本节最终在蒙古语、汉语、英语三个语种数据上进行测试。合成语音样例在网页展示²。

5.4.1 实验配置

本节一共选择蒙古语、汉语、英语三种语言的语音数据进行实验，下面详细介绍三种语音数据的具体情况：

蒙古语语音数据：由于端到端声学模型需要依赖大量的无标注数据进行训练，因此，本节对蒙古语语音合成语音库进行扩展。首先收集整理了约 15000 句蒙古语文本作为发音人朗读的依据，之后邀请内蒙古大学蒙古学学院蒙古语播音主持专业的女大学生进行录制。录音时，对照蒙古语语音合成语料库，按照新闻播音标准风格进行朗读，语句之间要有明显的停顿。整个录音过程在内蒙古大学计算机学院标准录音室完成，确保蒙古语语音合成语音库的音质清晰。另外，录音过程中有监督人员进行监督，如果发生错读、漏读等现象则该蒙古文文本重新录制，在录音过程中进行监督校正可以很大程度减少蒙古语语音合成语音库的发音错误产生，从而减轻后期人工校正的负担。录音保存格式为：采样率 44.1KHz，采样精度 16-bit，单声道，wav 格式。最终录制完成时长约 20 小时的标准蒙古语语音合成语音库用于端到端声学模型的训练。

汉语语音数据：采用标贝公司公开发布的中文标准女声音库数据集³。该数据共约 12 小时，语音格式为单声道录音，48KHz 采样率、采样精度 16-bit、PCM WAV 格式。

² <https://ttslr.github.io/ICASSP2020/>

³ https://www.data-baker.com/open_source.html

英语语音数据：采用公开的英语语音合成数据集 LJSpeech⁴。该数据共 24 小时，其中包括 13100 句英语文本，语音格式为 22.05kHz 采样率，采样精度 16-bit。

训练过程中数据划分为：90%作为实验训练集，5%作为实验验证集，另外 5%作为测试集。实验评价分为主观评价和客观评价。主观评价指标与 2.4.1 节中的 MOS 评测和 A/B 倾向性测试相同。客观评价指标包括词错误率 WER，但是该词错误率与 4.4.1 中的词错误率不同，本节中，我们通过对合成的语音进行人工测听来统计合成结果中出现的跳词、漏词和重复单词发音的个数，将其求和后除以待合成文本的单词总数，得到最终的词错误率结果。

5.4.2 实验设计

为了验证基于知识蒸馏的端到端蒙古语语音合成方法的有效性，本文分别为每种语言构建了 2 个系统：

(1) End2End：基线系统，采用 5.2 节中的基线端到端声学建模方法搭建蒙古语语音合成系统，其中计划采样 SS 方法作为声学解码器的解码方式，记作 End2End 系统。

(2) End2End-KD：本文提出的基于知识蒸馏的端到端声学建模方法。教师模型采用“教师激励”解码模式的声学解码器，学生模型的声学解码器采用“自由运行”解码模式。

以上系统中，对于蒙古语语音合成系统，采用蒙古文拉丁字符作为输入，字符向量维度为 256 维，模型输出为 80 维的梅尔频谱特征。对于汉语语音合成系统，使用汉字拼音序列作为输入，拼音序列对应的向量维度为 256 维，模型输出为 160 维的梅尔频谱特征。对于英语语音合成系统，使用英文字母序列作为输入，字母序列对应的向量维度为 256 维，模型输出为 80 维的梅尔频谱特征。所有系统中，每一时间步输出 2 帧频谱参数，代价函数平衡参数设置为 1。训练 batch_size 设置为 32，使用带有学习率衰减的 Adam 优化器训练模型，其中学习率自 50k 步开始从 10^{-3} 动态衰减到 10^{-5} 。教师模型训练步数 150k 步，之后使用该教师模型训练学生模型 150k 步。

⁴ <https://keithito.com/LJ-Speech-Dataset/>

5.4.3 实验结果与分析

5.4.3.1 主观测听实验

为了验证本文提出的基于知识蒸馏的端到端蒙古语语音合成方法的有效性, 我们对 5.4.2 节中的 2 个系统进行主观测试。我们从测试集中随机选取 60 句蒙古文文本, 另外构造 40 句包含重复单词、数字等挑战性蒙古文文本, 一共 100 句蒙古文文本作为测试数据。选取 10 位以蒙古语为母语的测听者进行语音评价, 他们的年龄在 20 岁到 30 岁之间。MOS 评测和 A/B 倾向性测试结果如图 5.4 和 5.5 所示。

通过实验结果可以发现, End2End-KD 系统相比于 End2End 系统取得了更高的 MOS 分数, 知识蒸馏策略的使用使得 MOS 分数从基线系统的 3.31 大幅提高到 3.94, 在 A/B 倾向性测试中, End2End-KD 占比 80.1%, End2End 系统占比 16.2%, 中间的中立占比只占有很少的 3.7%, End2End-KD 获得了更明显的领先优势。实验证明基于知识蒸馏的端到端声学建模方法可以生成更加自然、音质更好的蒙古语语音。

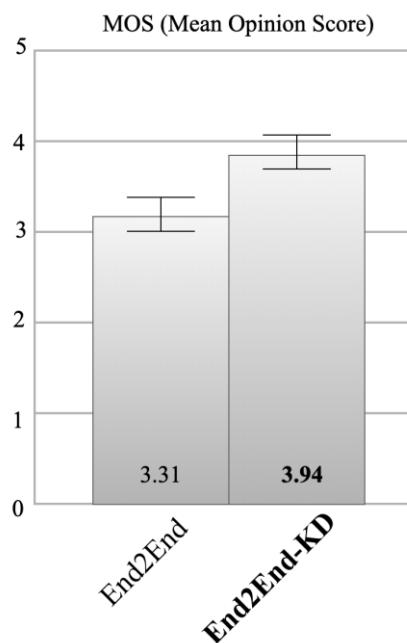


图 5.4 蒙古语 End2End 与 End2End-KD 系统主观 MOS 评测结果 (置信度 95%)

Fig. 5.4 MOS scores of speech quality between End2End and End2End-KD for Mongolian language, with confidence level of 95%

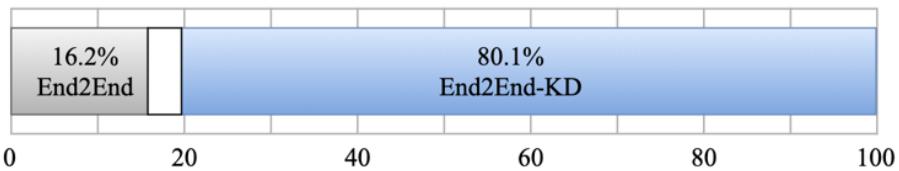


图 5.5 蒙古语 End2End 与 End2End-KD 系统 A/B 倾向性测试结果（置信度 95%）

Fig. 5.5 The results of A/B preference test between End2End and End2End-KD for Mongolian language, with confidence level of 95%

5.4.3.2 客观实验

客观实验同样以上一小节中得到的合成语音为依据, 对合成语音的词错误率进行统计。实验结果如表 5.1 所示。

从表 5.1 可以发现, End2End 系统的合成语音中词错误率为 15.34%, 而 End2End-KD 系统大幅降低了词错误率, 将其提升到 3.42%。漏词、跳词现象得到很好的缓解。实验证明, 基于知识蒸馏的端到端声学建模方法可以更加稳定的合成语音, 很好的缓解自回归解码器的曝光偏差问题。

表 5.1 蒙古语 End2End 与 End2End-KD 系统词错误率比较结果

Table 5.1 Comparison of Word Error Rate (WER%) between End2End and the proposed End2End-KD for Mongolian language

系统名称	词错误率 (WER%)
End2End	15.34
End2End-KD	3.42

5.4.3.3 其他语种扩展

为了测试该方法在其他语种的扩展性, 本节使用汉语数据和英语数据分别搭建了 5.4.2 节中的系统。为了保证测试结果的可靠性, 我们选择一些对于语音合成任务很有挑战性的文本。对于汉语, 我们从国际语音合成大赛 Blizzard Challenge 2019^[129]中官方提供的测试集中随机选择 500 句作为测试样例; 对于英语, 我们首先选择文献^[131]提供的 50 句测试样例, 另外我们又搜集了 50 句挑战性测试样例, 其中包含单英文字母、英文单词拼写、连续数字和长句 (平均包含英文字母 128 个) 等。最终, 使用 500 句和

100 句挑战性测试样例分别对汉语和英语语音合成系统进行测试。我们选取 20 位英语母语发音者和 15 位汉语母语发音者作为测听者。主观实验和客观实验结果如图 5.6、图 5.7 和表 5.2 所示。

通过本节的实验结果可以发现，End2End-KD 系统在汉语和英语数据上，主观实验和客观实验的表现均明显优于 End2End 系统，这与上一节中的蒙古语实验结果一致。这证明了该方法具有很好的扩展性和通用性，可以很好的扩展到其他语种。

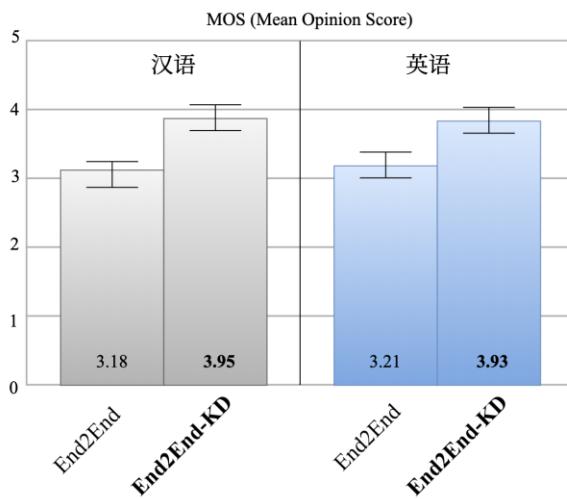


图 5.6 汉语和英语 End2End 与 End2End-KD 系统主观 MOS 评测结果 (置信度 95%)

Fig. 5.6 MOS scores of speech quality between End2End and the proposed End2End-KD for Chinese and English language, with confidence level of 95%

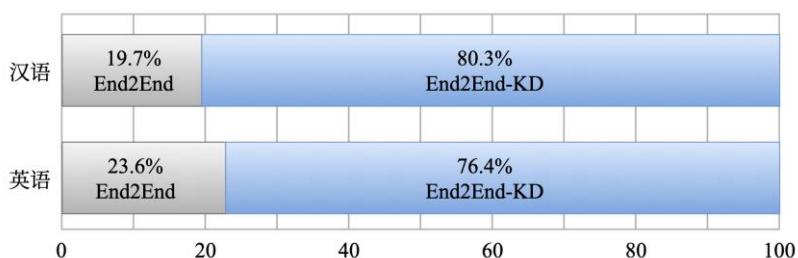


图 5.7 汉语和英语 End2End 与 End2End-KD 系统 A/B 倾向性测试结果 (置信度 95%)

Fig. 5.7 The results of A/B preference test between End2End and the proposed End2End-KD for Chinese and English language, with confidence level of 95%

表 5.2 汉语和英语 End2End 与 End2End-KD 系统词错误率比较结果

Table 5.2 Comparison of Word Error Rate (WER%) between End2End and the proposed End2End-KD for

Chinese and English language		
系统名称	语种	词错误率 (WER%)
End2End	汉语	9.44
	英语	23.82
End2End-KD	汉语	0.67
	英语	2.17

5.5 本章小结

本章提出了基于知识蒸馏的端到端蒙古语语音合成建模方法,对其中的教师模型和学生模型做了详细介绍。之后通过详细的主观实验、客观实验分析了模型的实际表现,另外通过汉语、英语等主流语种扩展测试,证明了该方法具有很好的扩展性。最终实验结果表明,本文提出的基于知识蒸馏的端到端蒙古语语音合成建模方法可以很好的解决端到端声学模型的跳词、漏词、重复的现象,可以合成自然度更高的合成语音。进一步提升了端到端蒙古语语音合成系统的自然度。

本章的相关研究工作已经发表在 45th International Conference on Acoustics, Speech, and Signal Processing (ICASSP2020) 国际会议。

第六章 融合显式韵律信息的端到端声学建模方法

6.1 引言

上一章中，我们采用知识蒸馏策略，针对端到端声学建模的曝光偏差问题提出了解决方案。采用“教师-学生”训练框架提高了端到端声学模型的鲁棒性，很好的缓解了基于端到端声学模型的合成语音中的跳词、漏词、重复等现象，提升了基于端到端模型的蒙古语语音合成系统的自然度。端到端声学建模方法虽然具有模型结构简洁、不依赖语言学知识、不需要人工干预等优点，凭借这些优点可以生成高质量的合成语音。但是正因为其结构简洁，其训练过程只简单学习<字符，语音>对的映射关系，导致模型输入信息来源单一。如2.2节所述，蒙古语与汉语等语种一样，有着复杂的韵律结构，若干蒙古文字符组成蒙古文韵律词，若干蒙古文韵律词组成蒙古文韵律短语，若干蒙古文韵律短语组成语调短语，最终蒙古文句子由若干蒙古文语调短语组成。这样的韵律层级结构包含的丰富的语法和语义知识。本文第三章和第四章的实验也证明将语言的语法语义知识引入语音合成系统可以提升合成语音的自然度。因此，简单的字符输入并不能将复杂的语法语义信息包含在内，其具有很弱的泛化性能，并限制了基于端到端声学模型的蒙古语语音合成系统自然度的进一步提升。本章从打破简单字符输入的限制出发，充分利用韵律结构的语法语义知识以进一步提升基于端到端声学模型的蒙古语语音合成方法的合成自然度为目标，将蒙古文字符的韵律知识作为原字符向量表示的增强，并提出了两种融合显式韵律信息的端到端声学建模方法，分别是特征级别的韵律信息融合方法和模型级别的韵律信息融合方法。

本章首先详细介绍特征级别的韵律信息融合方法，然后对模型级别的韵律信息融合方法进行具体介绍。最后通过实验比较分析验证了方法的有效性。

6.2 融合显式韵律信息的端到端声学建模方法

本文提出了两种韵律信息融合方法，第一是在特征级别进行韵律信息融合，第二是在模型级别进行韵律信息融合，以确保端到端声学模型在生成高质量合成语音的同时具

有更富有表现力的韵律节奏表现。下面两小节将分别介绍两种韵律信息融合方法。

6.2.1 特征级别融合方法

本节我们提出特征级别韵律信息融合的端到端声学建模方法，在特征表示层面，将韵律信息融入端到端蒙古语语音合成系统。图 6.1 所示为该方法的结构框架图。

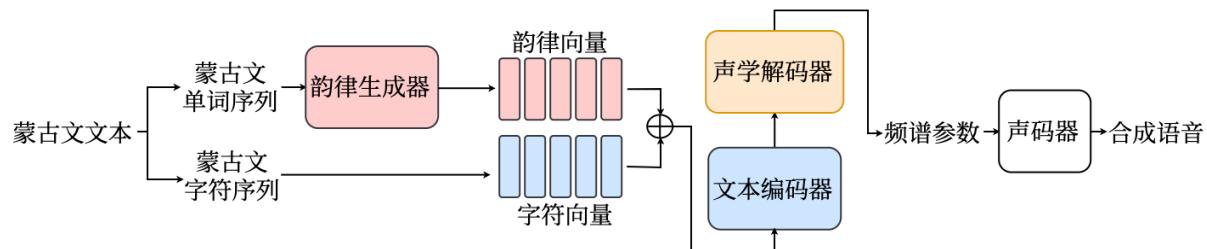


图 6.1 特征级别韵律信息融合方法示意图

Fig. 6.1 The schematic diagram of feature-level prosodic information fusion method

该方法以端到端蒙古语语音合成模型为基本框架，同样包括文本编码器和基于注意力机制的声学解码器。不同的是，在文本编码器读取给定的蒙古文字符序列之前，对字符的向量表示特征进行了韵律信息增强。即对蒙古文文本中的单词序列和字符序列分开处理，文本中的单词序列经过韵律生成器预测得到韵律向量，文本中的字符序列转换为浮点数字符向量，最后字符向量与该字符对应的韵律向量进行拼接得到联合向量，联合向量送入文本编码器和声学解码器进行模型的训练。

具体地，假设输入蒙古文文本的单词序列为 $W=[w_1, w_2, \dots, w_n]$ ，其字符序列为 $C=[c_1, c_2, \dots, c_m]$ 。韵律生成器读取单词序列 W 并通过下采样得到字符级别的韵律向量 $PE=[pe_1, pe_2, \dots, pe_m]$ 。字符序列的向量表示为 $CE=[ce_1, ce_2, \dots, ce_m]$ 。

为了充分挖掘字符级别的韵律知识，本节首先构建了“韵律生成器”来得到端到端声学模型中输入字符对应的韵律向量。该韵律生成器是基于 BiLSTM 网络搭建而成，韵律生成器包括输入层，BiLSTM 层和输出层。

输入层将输入的蒙古文单词序列转换为词向量表示作为其特征表示。假设输入长度为 T 的蒙古文单词序列 $X=[x_1, x_2, \dots, x_T]$ ，根据预先训练好的蒙古文词向量，通过查找词表将其转换为对应的特征向量 $V=[v_1, v_2, \dots, v_T]$ ，之后将该特征向量 V 送入 BiLSTM

层进行下一步处理。

BiLSTM 层读取输入的单词词向量 V 将其转换为高层特征表示。使用前向 LSTM 和后向 LSTM 读取单词词向量序列后分别输出隐状态向量。之后对前向隐状态和后向隐状态向量进行拼接，得到最终的隐状态 H 送入输出层解码得到最终的单词对应的韵律标签。

输出层使用 Softmax 函数将 BiLSTM 层输出的隐状态 H 进行解码，输出每个单词对应的韵律标签 $\mathbf{Y} = [y_1, y_2, \dots, y_T]$ 。

最终我们将韵律标签转换为独特形式的向量表示，作为韵律生成器得到的单词级别韵律向量。需要注意的是，单词序列和字符序列长度不相等，他们具有不同的时间维度，为了将韵律信息充分融入序列中的每个字符，我们将韵律向量进行上采样，将单词的韵律向量分配到该单词内的所有字符。上采样操作示意图如图 6.2 所示，其中“<sil>”表示标点符号、韵律短语边界等停顿。

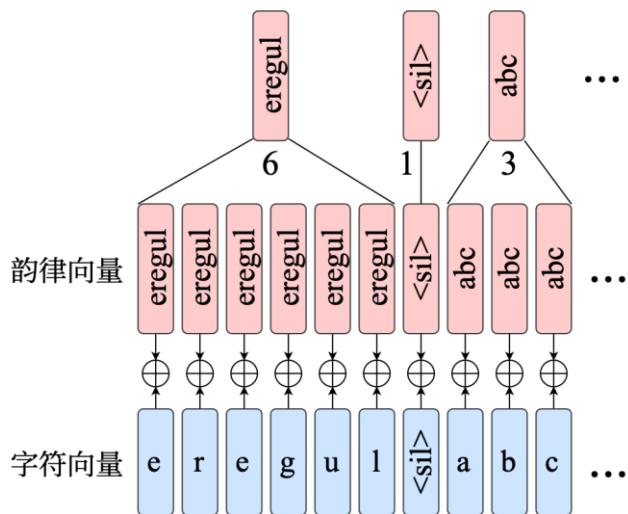


图 6.2 韵律向量上采样操作示意图

Fig. 6.2 The schematic diagram of upsampling operation of prosody embedding

以蒙古文单词“eregul”为例，其字符序列为“e”、“r”、“e”、“g”、“u”、“l”，“*l*”，每个字符对应的字符向量用蓝色矩形表示。文本中单词序列经韵律生成器得到的对应韵律向量用粉色矩形表示。如图所示，单词 *eregul* 的字符个数为 6，我们将单词 *eregul* 对应的单词级别韵律向量复制 6 次得到 6 个相同的字符级别的韵律向量。

最终我们为输入文本的所有字符均分配到对应的韵律向量，这个字符级别的韵律向

量将被充分融入到端到端声学模型中。具体的，将字符向量 \mathbf{CE} 与字符级别韵律向量 \mathbf{PE} 进行拼接得到最终的联合向量 \mathbf{JE} ，公式表示为（6-1）。

$$\mathbf{JE} = [\mathbf{PE}; \mathbf{CE}] \quad (6-1)$$

之后将韵律信息增强后的联合向量通过文本编码器生成隐状态序列 \mathbf{H} ，即公式（6-2）。

$$\mathbf{H} = \text{Encoder}(\mathbf{JE}) \quad (6-2)$$

隐状态 \mathbf{H} 作为文本编码器的输出送入到基于注意力机制的声学解码器来对频谱声学参数 \mathbf{Y} 进行预测。即公式（6-3）所示：

$$\mathbf{Y} = \text{Decoder}(\mathbf{H}) \quad (6-3)$$

需要注意的是，该方法中的韵律生成器是在语音合成模型训练之前提前训练的，并不参与语音合成模型的训练，它的作用仅仅是生成韵律向量并与字符向量一起得到联合向量。该联合向量包含有丰富的韵律信息，对原始的字符向量表示特征进行了韵律信息增强，使得端到端声学模型在特征表示级别充分融合语言的韵律知识，加入韵律信息对端到端声学模型的训练进行指导并最终提升合成语音的韵律表现和自然度。

6.2.2 模型级别融合方法

本节我们提出模型级别韵律信息融合的端到端声学建模方法，在模型训练层面，将韵律信息融入端到端蒙古语语音合成系统。该方法对端到端声学建模方法进行扩展，采用多任务学习框架将韵律信息融入端到端声学建模过程。图 6.3 所示为该方法的结构框架图。

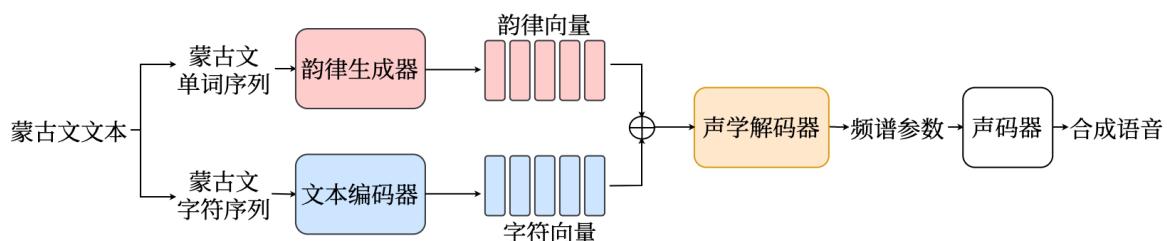


图 6.3 模型级别韵律信息融合方法示意图

Fig. 6.3 The schematic diagram of model-level prosodic information fusion method

该模型将基于“编码器-解码器”的端到端声学模型与韵律生成器整合在同一个端到端训练框架。待合成蒙古文文本中的字符序列通过文本编码器输出高层次的字符级向量表示，与此同时，待合成蒙古文文本中的单词序列通过韵律生成器输出字符级韵律向量，之后对文本编码器输出的字符级向量与韵律生成器输出的字符级韵律向量进行拼接，拼接后得到韵律信息增强后的联合向量表示，该联合向量送入带有注意力机制的声学解码器，预测最终的声学参数。

具体地，假设输入蒙古文文本的单词序列为 $W=[w_1, w_2, \dots, w_n]$ ，其字符序列为 $C=[c_1, c_2, \dots, c_m]$ 。首先，主任务的文本编码器读取字符序列 C 得到编码器输出的隐状态序列 H ，该隐状态序列 H 作为高层次的字符向量表示，记作 $CE'=[ce'_1, ce'_2, \dots, ce'_m]$ 。公式表示为（6-4）。

$$CE' = H = \text{Encoder}(C) \quad (6-4)$$

与此同时，韵律生成器读取单词序列 W 并得到字符级别的韵律向量 $PE=[pe_1, pe_2, \dots, pe_m]$ 。我们采用与 6.2.1 相同的上采样操作将高层次字符向量 CE' 与字符级别韵律向量 PE 进行拼接得到最终的联合向量 JE' ，即公式（6-5）所示。

$$JE' = [PE'; CE'] \quad (6-5)$$

最后将该联合向量 JE' 送入声学解码器进行频谱声学参数 Y 的预测，即公式（6-6）所示。

$$Y = \text{Decoder}(JE') \quad (6-6)$$

需要注意的是，该方法中的端到端声学模型和韵律生成器是联合训练的，文本编码器、韵律生成器和声学解码器的参数是同时更新的，韵律生成器在训练过程中对文本编码器和声学解码器的训练起到很好的辅助作用。该模型级别的韵律信息融合方法相比特征级别的韵律信息融合方法，更充分的将韵律信息融入模型的训练过程，模型训练过程中可以学习到更加显式的韵律知识。韵律生成器促进了端到端声学模型的训练并对合成语音自然度和表现力的提升起到积极作用。下一节将对两种显式韵律信息融合方法进行实验比较和分析。

6.3 实验

本节将对本文提出的两种融合显式韵律信息的端到端声学建模方法的合成语音表现进行评估，由于该方法与语种并没有直接的关系，为了证明该方法在多个语种的通用性，本节最终在蒙古语、汉语两个语种数据上进行测试。合成语音样例在网页展示⁵。

6.3.1 实验配置

本节一共选择蒙古语、汉语两种语言的语音数据进行实验，下面详细介绍两种语音数据的具体情况。

蒙古语语音数据采用 5.4.1 节中的 20 小时的蒙古语语音合成语音库。其中包括约 15000 句蒙古语语音。汉语语音数据采用清华大学人机交互与媒体集成研究所录制制作的汉语普通话语音合成语料库（TsingHua-Corpus of Speech Synthesis, TH-CoSS）^[132]。我们选择其中一个说话人 03FR00 的数据作为训练数据，共约 9 小时，语音格式为单声道录音，16KHz 采样率、采样精度 16-bit。

蒙古文词向量训练数据与 3.6.1 节的训练数据相同。蒙古文韵律生成器训练数据来源于蒙古语语音库对应的文本数据，其中包括蒙古文文本约 15000 句，蒙古文韵律短语约 30 万个，蒙古文韵律短语的平均长度为 3.5 个蒙古文单词。中文词向量数据来源于腾讯人工智能实验室公开发布的词向量数据^[133]，该中文词向量在 800 万中文文本数据中训练得到，维度为 200 维。中文韵律生成器训练数据同样来自 TH-CoSS 的文本语料库，一共包含约 5 万 6 千句文本，汉字数量约 10 万个。

训练过程中数据划分为：90% 作为实验训练集，5% 作为实验验证集，另外 5% 作为测试集。实验评价指标与 3.5.1 节中的相同，分为主观评价和客观评价。主观评价指标包括 MOS 评测和 A/B 倾向性测试。客观评价包括韵律预测 F 值。

6.3.2 实验设计

为了验证两种融合显式韵律信息的端到端声学建模方法的有效性，本文分别为每种语言构建了 3 个系统：

⁵ <https://ttslr.github.io/MTL-Tacotron/>

(1) End2End: 基线系统，采用 5.2 节中的端到端声学建模方法构建端到端语音合成系统。

(2) End2End-PE: 本文提出的特征级别韵律信息融合的端到端声学建模方法。将韵律向量与字符向量拼接得到韵律增强后的字符向量表示后，送入文本编码器。

(3) End2End-MTL: 本文提出的模型级别韵律信息融合的端到端声学建模方法。将端到端声学模型与韵律生成模型联建模，韵律生成模型输出的韵律向量与文本编码器输出的字符向量拼接后得到联合向量，之后送入声学解码器，采用多任务学习的框架进行两个任务的同时训练。

以上系统中，对于韵律生成模型，蒙古文和中文词向量均采用 200 维词向量作为输入，BiLSTM 网络均使用 2 层 LSTM 结构，每个 LSTM 层包含 160 个节点，采用丢弃率为 0.5 的 dropout 技术用来防止模型过拟合。其中，End2End-PE 中的韵律生成模型初始学习率设置为 1.0，训练批次大小设置为 64，采用 AdaDelta 优化器训练参数，训练到收敛后停止训练作为预训练模型。End2End-MTL 系统中的韵律生成模型随模型其他部分一起训练更新。

对于端到端声学模型，蒙古语语音合成采用蒙古文拉丁字符作为输入，字符向量维度为 256 维，韵律向量维度为 5 维。模型输出为 80 维的梅尔频谱特征；汉语语音合成使用带声调的拼音序列作为输入，将拼音序列看作字符序列，其对应的向量维度为 256 维，模型输出为 80 维的梅尔频谱特征。所有系统中，每一时间步输出 2 帧频谱参数，代价函数平衡参数设置为 1。训练 batch_size 设置为 32，使用带有学习率衰减的 Adam 优化器训练模型，其中学习率自 50k 步开始从 10^{-3} 动态衰减到 10^{-5} 。End2End-PE 中的端到端声学模型训练 150k 步；End2End-MTL 中声学模型和韵律生成模型联合训练 150k 步作为最终模型。

6.3.3 实验结果与分析

6.3.3.1 主观测听实验

为了验证本文提出的特征级别韵律信息融合方法和模型级别韵律融合方法的有效性，我们对 6.3.2 节中的 3 个系统进行主观测试。我们从测试集中随机选取 60 句蒙古文

文本，选取 10 位以蒙古语为母语的测听者进行语音评价，他们的年龄在 20 岁到 30 岁之间。MOS 评测和 A/B 倾向性测试结果如图 6.4 和 6.5 所示。

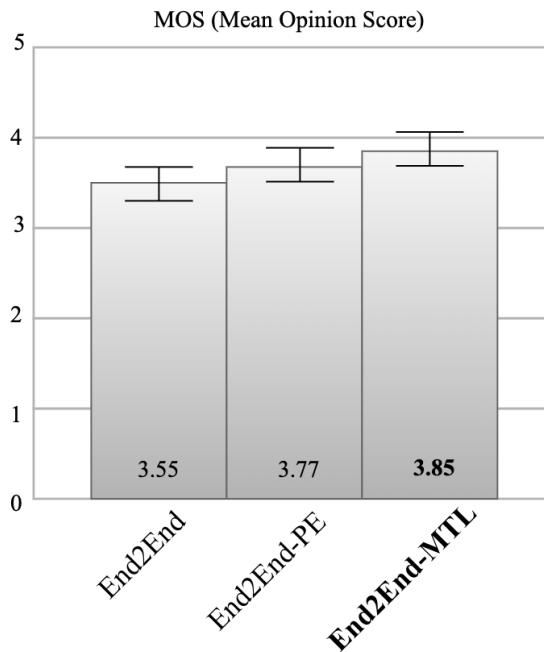


图 6.4 蒙古语 End2End、End2End-PE 与 End2End-MTL 系统主观 MOS 评测结果（置信度 95%）

Fig. 6.4 MOS scores of speech quality among End2End, End2End-PE and End2End-MTL for Mongolian language, with confidence level of 95%

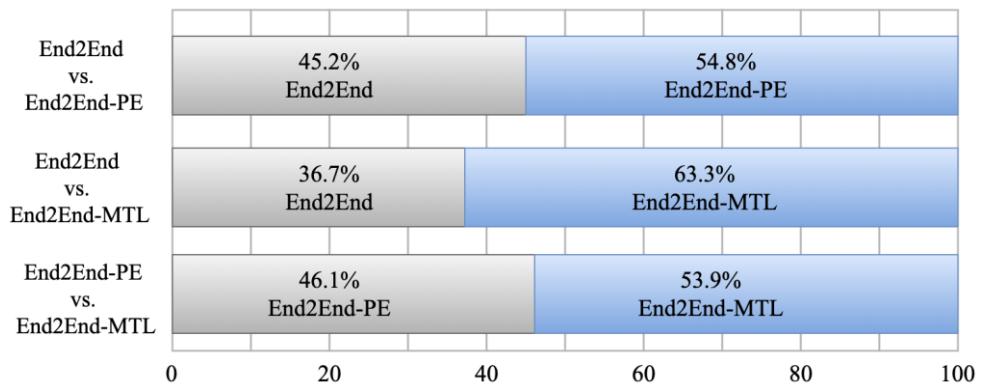


图 6.5 蒙古语 End2End、End2End-PE 与 End2End-MTL 系统 A/B 倾向性测试结果（置信度 95%）

Fig. 6.5 The results of A/B preference test among End2End, End2End-PE and End2End-MTL for Mongolian language, with confidence level of 95%

通过实验结果可以发现, MOS 评测中, End2End-MTL 的 MOS 分数为 3.85, End2End-PE 的 MOS 分数为 3.77, 而 End2End 基线系统的 MOS 分数仅有 3.55, End2End-MTL 和 End2End-PE 两个系统均明显超越基线系统 End2End, 并且 End2End-MTL 系统相比于 End2End-PE 系统取得了更高的分数, 实现了最好的合成效果; 在 A/B 倾向性测试中, End2End-PE 以 54.8% 的比重超过 End2End 系统的 45.2% 获得了更高的倾向性得分, End2End-MTL 以 63.3% 的比重明显超过 End2End 系统的 36.7%, 最后 End2End-MTL 以 53.9% 的比重又超过了 End2End-PE 系统的 46.1% 占比, 表现出最好的合成表现。原因在于, End2End-PE 系统将韵律向量与字符向量在模型输入端进行融合, 将信息更加丰富的特征向量输入到模型进行训练, 使得模型表现出更好的合成语音自然度; 进一步, End2End-MTL 系统在模型级别对韵律信息进行融合, 将编码器输出的字符向量与联合训练机制下得到的韵律向量融合, 最终送入解码器对声学参数进行预测, 与简单的增强输入相比, 字符的韵律信息得到更好的利用, 系统合成语音获得了最好的评测表现。

实验证明两种融合显式韵律信息的端到端声学建模方法可以有效提升端到端蒙古语语音合成系统的自然度, 其中, 在模型级别融合韵律信息的方法更加有效, 可以通过使用韵律知识指导端到端模型训练过程来生成表现更加自然的蒙古语语音。

6.3.3.2 客观实验

客观实验用来评价韵律生成模型在两种系统中的具体表现。实验结果如表 6.1 所示。从表 6.1 可以发现, End2End-MTL 系统中的韵律生成模型预测 F 值为 90.29, 与 End2End-PE 系统中的韵律生成模型相比取得了提升。分析其中原因, End2End-MTL 中的韵律生成模型采用多任务学习机制与端到端声学模型联合训练, 而 End2End-PE 系统中韵律生成模型是预先训练得到, 预先训练得到的韵律生成模型缺少端到端声学模型中隐含声学信息的指导, 而联合训练过程中, 韵律生成模型通过与端到端声学模型联合训练的方式, 在模型训练过程中相互优化, 在端到端声学模型的帮助下最终提升韵律生成模型的精度, 从而得到更加准确的韵律向量, 并最终提升语音合成的表现。实验证明, 通过模型级别的韵律信息融合, 可以充分发挥多任务学习的优势, 使得端到端声学模型和韵律生成模型同时提升建模精度, 在联合训练的过程中生成更加准确的韵律向量, 可以促使端到端声学模型输出更加自然的合成语音。

表 6.1 蒙古文韵律生成器在 End2End-PE 与 End2End-MTL 系统的表现比较

Table 6.1 Comparison of prosody generator between End2End-PE and the proposed End2End-MTL for

Mongolian language	
系统名称	F 值
End2End-PE	88.87
End2End-MTL	90.29

6.3.3.3 文本长度实验

如 5.4.2 所述，端到端声学模型对于长文本的韵律建模问题仍然不够鲁棒。为了进一步分析 End2End-MTL 系统即模型级别韵律信息融合方法的适用场景，本节对 End2End-MTL 处理不同长度输入文本时的表现进行了 A/B 倾向性测试。我们搜集不同长度的文本并构建三个不同的测试集：Test1：文本包含的字符个数在 0 到 50 之前；Test2：文本包含的字符个数在 51 到 100 之间；Test3：文本包含的字符个数在 101 到 200 之间。我们从每个测试集中选取 80 个句子，分别使用 End2End 基线系统和 End2End-MTL 系统进行测试，测试结果如图 6.6 所示。

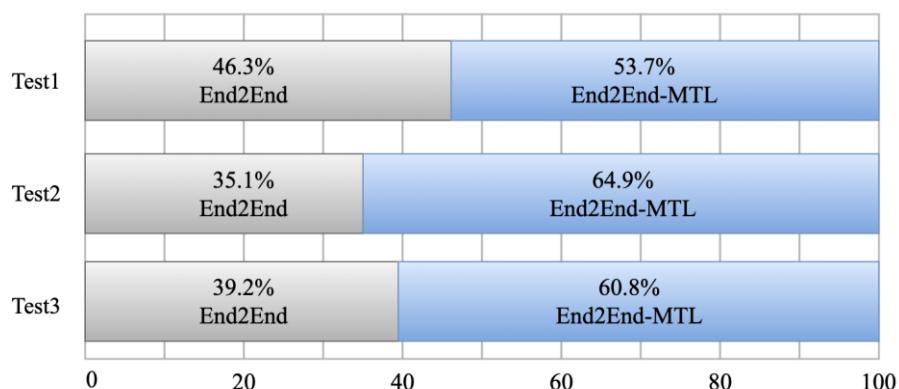


图 6.6 蒙古语 End2End、End2End-PE 与 End2End-MTL 系统对不同长度文本韵律建模的 A/B 倾向性测试实验结果（置信度 95%）

Fig. 6.6 The results of A/B preference test among End2End, End2End-PE and End2End-MTL for Mongolian language, with confidence level of 95%

从图 6.6 可以明显看出，对于所有测试集，End2End-MTL 系统的表现均优于基线系统，分别高于基线 End2End 系统 7.4%、29.8% 和 21.6% 占比。值得注意的是，在 Test2

和 Test3 测试集上, End2End-MTL 的优势更加明显, 与基线系统的差距均超过了 20%。分析其中原因, End2End-MTL 系统通过将韵律信息融入模型训练过程, 可以使用韵律信息对模型训练进行指导, 并最终生成更加稳定的声学参数, 使得合成的语音在面对不同长度文本时都具有更丰富的韵律表现。我们的方法可以为文本提供丰富和有力的监督信息, 从而提升对于长文本的建模能力。实验证明了该方法对于处理长文本的韵律建模问题具有更好的泛化性和鲁棒性。

6.3.3.4 其他语种扩展

为了测试该方法在其他语种的扩展性, 本节使用汉语数据分别搭建了 6.3.2 节中的系统。我们从测试集中随机挑选 80 句进行测试。我们选取 15 位汉语母语发音者作为测听者。主观实验和客观实验结果如图 6.7、图 6.8 和表 6.2 所示。

通过本节的实验结果可以发现, End2End-MTL 系统在汉语上的表现与蒙古语一致, 主观实验和客观实验的表现均明显优于 End2End 系统。这证明了该方法具有很好的扩展性和通用性, 可以很好的扩展到其他语种。

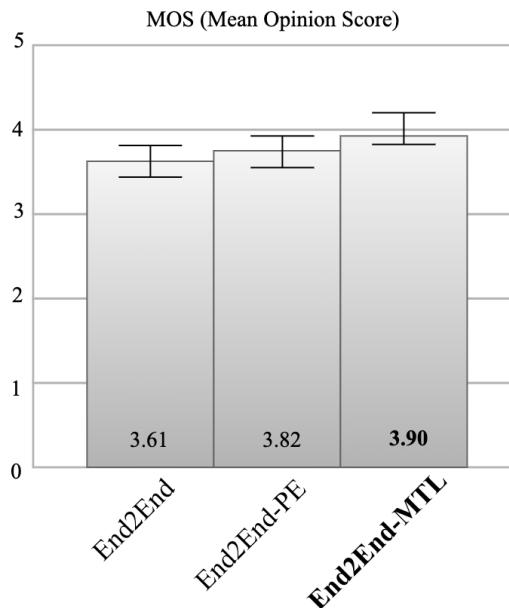


图 6.7 汉语 End2End、End2End-PE 与 End2End-MTL 系统主观 MOS 评测结果（置信度 95%）

Fig. 6.7 MOS scores of speech quality among End2End, End2End-PE and End2End-MTL for Chinese language, with confidence level of 95%

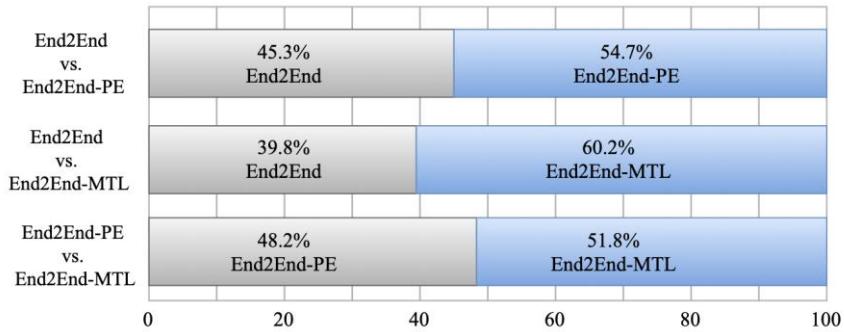


图 6.8 汉语 End2End、End2End-PE 与 End2End-MTL 系统 A/B 倾向性测试结果 (置信度 95%)

Fig. 6.8 The results of A/B preference test among End2End, End2End-PE and End2End-MTL for Chinese language, with confidence level of 95%

表 6.2 汉语韵律生成器在 End2End-PE 与 End2End-MTL 系统的表现比较

Table 6.2 Comparison of prosody generator between End2End-PE and the proposed End2End-MTL for

Chinese language	
系统名称	F 值
End2End-PE	87.90
End2End-MTL	90.31

另外, 我们同样对汉语数据进行了不同长度文本合成效果的比较。比较结果如图 6.9 所示。从图 6.9 可以看出, 对于汉语数据, End2End-MTL 同样拥有不错的长文本韵律建模能力, 在 Test2 和 Test3 数据集的表现同样具有不错的表现。该实验同样验证了在模型级别融合韵律信息可以有效增强端到端声学建模对不同长度文本韵律表现的泛化性能。

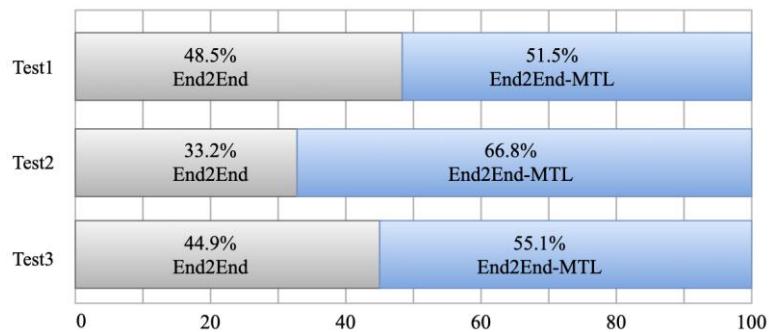


图 6.9 汉语 End2End、End2End-PE 与 End2End-MTL 系统 A/B 倾向性测试结果 (置信度 95%)

Fig. 6.9 The results of A/B preference test among End2End, End2End-PE and End2End-MTL for Chinese language, with confidence level of 95%

6.4 本章小结

本章提出了两种融合显式韵律信息的端到端蒙古语语音合成声学建模方法，对特征级别的韵律信息融合方法和模型级别的韵律信息融合方法做了详细介绍。之后通过详细的主观实验、客观实验分析了模型的实际表现，并关于模型对不同长度文本的合成表现进行了分析；另外通过汉语等主流语种扩展测试，证明了该方法具有很好的扩展性。最终实验结果表明，本文提出的融合韵律信息的端到端声学建模方法可以很好的将显式韵律信息融入模型训练，使用显式韵律信息指导端到端模型的训练过程，进一步提升了端到端蒙古语语音合成系统的自然度。

本章的相关研究工作已经投稿于 SCI 期刊 IEEE Signal Processing Letters (SPL)。

第七章 总结与展望

7.1 本文工作总结

蒙古语语音合成技术作为蒙古语智能信息处理的重要组成部分受到越来越多的关注并且取得了蓬勃的发展，但是相对于其它汉语、英语等主流语种，蒙古语语音合成技术还不够成熟，语音合成质量难以达到实用需求。

本文将深度学习技术全面引入蒙古语语音合成的前端模块和后端模块，针对现有语音合成中韵律建模和声学建模存在的问题，充分利用蒙古语语言特点、相关任务的隐含知识以及先进的模型训练方法，对蒙古语语音合成系统的整体表现进行进一步提升。总结起来，本文工作的主要贡献有：

(1) 充分结合蒙古语语言特点和深度学习技术提升基于深度神经网络的蒙古语语音合成的自然度。语音合成技术涉及语言学、计算机科学等多种学科。为了充分挖掘蒙古语的形态学知识和音系学知识来提升蒙古语语音合成的表现，提出了基于词素单元的蒙古文韵律预测方法与融合形态向量和音系向量的蒙古文韵律建模方法。实验证明改进的蒙古文韵律建模方法增强了蒙古文文本特征的特征描述能力，进一步增强了基于深度神经网络的蒙古语语音合成的自然度。

(2) 利用相关任务的隐含知识提升基于深度神经网络的蒙古语语音合成的自然度。蒙古文韵律建模和蒙古文字母转音素两个任务具有天然的相关性，将蒙古文韵律建模和蒙古文字母转音素通过“编码器-解码器”网络整合到同一个训练框架，实验证明两个任务联合训练进一步增强了蒙古文韵律模型的性能，从而提升了基于深度神经网络的蒙古语语音合成的自然度。

(3) 利用知识蒸馏技术提升基于端到端声学模型的蒙古语语音合成的自然度、表现力和鲁棒性。采用“教师-学生”训练框架，先训练更符合真实语音声学参数分布的教师模型，之后使用教师模型作为指导训练学生模型。使得学生模型可以生成更加可靠和稳定的合成语音。实验证明基于知识蒸馏的端到端蒙古语语音合成方法可以显著提升合成语音的整体表现，合成语音的自然度、韵律表现得到了显著提升，并且跳词、漏词、重复等问题得到了明显缓解。另外，还证明了该方法具有很好的语言扩展性和通用性。

(4) 充分将显式韵律信息融入到基于端到端声学模型的蒙古语语音合成模型中。提出特征级别的韵律信息融合方法和模型级别的韵律信息融合方法,实验证明融合显式韵律信息的端到端声学模型可以生成自然度更高的合成语音,其中模型级别的韵律信息融入方法更加有效。更进一步通过实验分析了该方法对于长文本的韵律建模能力并且验证了该方法的语言扩展性。

7.2 未来工作展望

虽然蒙古语语音合成研究已经取得了阶段性成果,然而蒙古语语音合成技术方兴未艾,随着深度学习技术的不断发展,蒙古语语音合成研究仍然具有广阔的发展空间。在本文研究内容的基础上,我们总结了以下几点后续的研究工作:

(1) 实时蒙古语语音合成系统:语音合成系统的主要任务就是将文本序列转化为接近真人发音的语音数据,而针对以语音合成服务为基础服务的上游语音交互系统来说,语音合成速度的快慢与否直接影响到上游语音交互系统的效率表现。如果合成速度太慢,就会出现语音播放卡顿或者持续等待播放等问题。因此,语音合成系统的实时性能直接关系到系统应用的效果。本文研究的基于深度神经网络声学模型的蒙古语语音合成系统和基于端到端声学模型的蒙古语语音合成系统都采用逐帧解码的方式进行语音生成,这样的解码方式大大增加了合成语音的生成时间,降低了合成效率,尤其当待合成本文长度太长时效率会更加低下,难以满足实时要求。因此,下一步工作中,可以摆脱自回归解码方式的限制为研究目标,将非自回归语音合成技术应用到蒙古语语音合成模型来显著提高其合成效率、满足实时性要求。

(2) 情感蒙古语语音合成系统:本文研究的基于深度学习的蒙古语语音合成方法主要关注在保证发音内容传达准确的前提下如何提升合成语音的整体自然度,而没有关注语音信号中体现的情感表现力。情感语音合成是近几年语音合成的研究热点,语音的韵律和声学特征是指导情感语音合成的主要因素。因此,下一步工作中,可以实现情感蒙古语语音合成系统为目标。首先构建情感蒙古语语音合成都语音语料库;其次,充分挖掘蒙古语语言特点,研究确定蒙古语语音的情感声学特征参数、确定蒙古语情感声学特征与情感状态的映射关系、确定蒙古文文本情感分析与场景因素结合的蒙古文语音情感预测机制等问题,实现情感蒙古语语音合成系统。

(3) 多模态蒙古语语音合成系统：语音合成广泛应用于智能家居、虚拟主播、语音导航、信息播报、阅读教育、泛娱乐等领域，是人机交互的重要组成部分。人机交互的方式走过了键盘交互、触摸交互、语音交互等，每一次变化的背后都是对人和机器之间交互的便利性、自然性以及准确性所提出的更高的要求。近几年，多模态交互作为一个非常重要的人机交互方向具有势不可挡的发展趋势。首先，多模态交互能够让人类在不同的场景下选择不同的模态组合进行交互，进而提升人机交互的整体自然度；其次，多模态技术下，多个模态可以互相补充，能够通过多个模态信息的融合获得更准确的用户、情感和场景估计；最后，多模态技术能够让人机交互过程中拥有视觉、听觉和触觉等多维感觉，全方位体会机器表达的情感和语义信息。因此，下一步工作中，可以以实现多模态蒙古语语音合成系统为目标。首先构建多模态蒙古语语音合成图像语音语料库；其次，充分挖掘蒙古语语言特点，研究确定多模态蒙古语语音合成系统的多模态输入、多模态输出和中间认知环节的多模态推理和决策等问题，实现多模态蒙古语语音合成系统。

参考文献

- [1] Taylor P. Text-to-speech synthesis[M]. Cambridge university press, 2009.
- [2] Kratzenstein C. Tentamen resolvendi problema[J]. 1781.
- [3] Dudley W H. The vocoder[J]. Bell. Labs. Rec., 1939, 18: 122.
- [4] Lawrence W. The synthesis of speech from signal which have a low information rate[J]. Communication Theory, 1953: 460-469.
- [5] Fant G. Speech communication research[J]. Royal Swedish Academy of Engineering Sciences, 1953, 2: 331-337.
- [6] Black A W. Optimising selection of unit from speech databases for concatenative synthesis[C]. In: Proceedings of Eurospeech1995. 1995: 581-584.
- [7] Moulines E, Charpentier F. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones[J]. Speech communication, 1990, 9(5-6): 453-467.
- [8] Hunt A J, Black A W. Unit selection in a concatenative speech synthesis system using a large speech database[C]. In: Proceedings of the 1996 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 1996. 373-376.
- [9] Senior A W, Fructuoso J G. Deep networks for unit selection speech synthesis: U.S. Patent 9,460,704[P]. 2016.
- [10] Black A W, Bennett C L, Blanchard B C, et al. CMU Blizzard 2007: A hybrid acoustic unit selection system from statistically predicted parameters[C]. In: Proceedings of 2007 Blizzard Challenge Workshop, Bonn, Germany. 2007. 1-5.
- [11] Wan V, Agiomyrgiannakis Y, Silen H, et al. Google's Next-Generation Real-Time Unit-Selection Synthesizer Using Sequence-to-Sequence LSTM-Based Autoencoders[C]. In: Proceedings of the 2017 Conference of the International Speech Communication Association (InterSpeech). 2017. 1143-1147.
- [12] Merritt T, Clark R A J, Wu Z, et al. Deep neural network-guided unit selection synthesis[C]. In: Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016. 5145-5149.
- [13] 凌震华,王仁华. 基于统计声学模型的单元挑选语音合成算法[J]. 模式识别与人工智能, 2008,

21(03): 280-284.

[14] Rabiner L, Juang B. An introduction to hidden Markov models[J]. IEEE ASSP Magazine, 1986, 3(1): 4-16.

[15] Rabiner L R. A tutorial on hidden Markov models and selected applications in speech recognition[J]. Proceedings of the IEEE, 1989, 77(2): 257-286.

[16] Donovan R E. Trainable speech synthesis[D]. University of Cambridge, 1996.

[17] Tokuda K, Nankaku Y, Toda T, et al. Speech synthesis based on hidden Markov models[C]. In: Proceedings of the IEEE, 2013, 101(5): 1234-1252.

[18] 曹剑芬. 汉语韵律切分的语音学和语言学线索[A]. 见: 第五届全国现代语音学学术会议论文集[C].中国中文信息学会:中国中文信息学会,2001.184-187.

[19] 曹剑芬, 陈方忻. 基于文本信息的韵律结构预测及其在合成系统中的应用[A]. 见: 第七届全国人机语音通讯学术会议[C]. 中国中文信息学会:中国中文信息学会, 2007. 162-167.

[20] 初敏. 韵律研究与合成语音的自然度[A]. 见: 中国中文信息学会.新世纪的现代语音学——第五届全国现代语音学学术会议论文集[C].中国中文信息学会:中国中文信息学会,2001.303-309.

[21] Tao J, Dong H, Zhao S. Rule learning based Chinese prosodic phrase prediction[C]. In: Proceedings of 2003 International Conference on Natural Language Processing and Knowledge Engineering. 2003. 425-432.

[22] 陶建华,蔡莲红. 基于音节韵律特征分类的汉语语音合成中韵律模型的研究[J]. 声学学报,2003(05):395-402.

[23] Parlkar A, Black A W. A Grammar based approach to style specific phrase prediction[C]. In: Proceedings of the 2011 Conference of the International Speech Communication Association (InterSpeech). 2011.2149-2152.

[24] Zheng Y, Lee GG, and Kim B. Using multiple linguistic features for Mandarin phrase break prediction in maximum-entropy classification framework[C]. In: Proceedings of the 2004 Conference of the International Speech Communication Association (InterSpeech). 2004. 737-740.

[25] Li J F, Hu G, Wang R. Chinese prosody phrase break prediction based on maximum entropy model[C]. In: Proceedings of the 2004 Conference of the International Speech Communication Association (InterSpeech). 2004. 729-732.

- [26] Qian Y, Wu Z, Ma X, et al. Automatic prosody prediction and detection with Conditional Random Field (CRF) models[C]. In: Proceedings of the 2010 International Symposium on Chinese Spoken Language Processing (ISCSLP). 2010. 135-138.
- [27] Lafferty J, McCallum A, Pereira F C N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data[C]. In: Proceedings of the 2001 International Conference on Machine Learning (ICML). 2001. 282-289.
- [28] Taylor P, Black A W. Assigning phrase breaks from part-of-speech sequences[J]. Computer Speech & Language, 1998, 12(2): 99-117.
- [29] Sloan R, Akhtar S S, Li B, et al. Prosody Prediction from Syntactic, Lexical, and Word Embedding Features[C]. In: Proceedings of the 2019 ISCA Speech Synthesis Workshop (SSW). 2019. 269-274.
- [30] 陶建华. 基于统计和规则相结合的汉语语音合成的韵律模型[J]. 声学技术, 2003 (z2): 359-361.
- [31] 陶建华, 蔡莲红, 吴志勇. 基于统计模型的韵律建模方法[A]. 见: 第六届全国人机语音通讯学术会议论文集[C]. 2001. 77-81.
- [32] 陶建华, 赵晟, 蔡莲红. 基于统计韵律模型的汉语语音合成系统的研究[J]. 中文信息学报, 2002, 16(1): 2-7.
- [33] Yamagishi J. An introduction to HMM-based speech synthesis[J]. Technical Report, 2006. 1-5.
- [34] Zen H, Tokuda K, Masuko T, et al. A hidden semi-Markov model-based speech synthesis system[J]. IEICE transactions on Information and Systems, 2007, 90(5): 825-834.
- [35] Yoshimura T. Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis[C]. In: Proceedings of the 1999 European Conference on Speech Communication and Technology (EuroSpeech). 1999. 2374-2350.
- [36] Tokuda K, Masuko T, Miyazaki N, et al. Multi-space probability distribution HMM[J]. IEICE transactions on Information and Systems, 2002, 85(3): 455-464.
- [37] 吴义坚, 王仁华. 基于 HMM 的可训练中文语音合成[J]. 中文信息学报, 2006, 20(4): 77-83.
- [38] Tokuda K, Yoshimura T, Masuko T, et al. Speech parameter generation algorithms for HMM-based speech synthesis[C]. In: Proceedings of the 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (ICASSP). IEEE, 2000. 1315-1318.
- [39] Grünwald P D, Grunwald A. The minimum description length principle[M]. MIT press, 2007.

- [40] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.
- [41] Watts O, Yamagishi J, King S. Unsupervised continuous-valued word features for phrase-break prediction without a part-of-speech tagger[C]. In: Proceedings of the 2011 Conference of the International Speech Communication Association (InterSpeech). 2011. 2157-2160.
- [42] Vadapalli A, Prahallad K. Learning continuous-valued word representations for phrase break prediction[C]. In: Proceedings of the 2014 Conference of the International Speech Communication Association (InterSpeech). 2014. 41-45.
- [43] Watts O, Gangireddy S, Yamagishi J, et al. Neural net word representations for phrase-break prediction without a part of speech tagger[C]. In: Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014. 2599-2603.
- [44] Vadapalli A, Gangashetty S V. An Investigation of Recurrent Neural Network Architectures Using Word Embeddings for Phrase Break Prediction[C]. In: Proceedings of the 2016 Conference of the International Speech Communication Association (InterSpeech). 2016. 2308-2312.
- [45] Ding C, Xie L, Yan J, et al. Automatic prosody prediction for Chinese speech synthesis using BLSTM-RNN and embedding features[C]. In: Proceedings of the 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, 2015. 98-102.
- [46] Zheng Y, Tao J, Wen Z, et al. Blstm-crf based end-to-end prosodic boundary prediction with context sensitive embeddings in a text-to-speech front-end[C]. In: Proceedings of the 2018 Conference of the International Speech Communication Association (InterSpeech). 2018. 47-51.
- [47] Klimkov V, Nadolski A, Moinet A, et al. Phrase Break Prediction for Long-Form Reading TTS: Exploiting Text Structure Information[C]. In: Proceedings of the 2017 Conference of the International Speech Communication Association (InterSpeech). 2017. 1064-1068.
- [48] Zen H, Senior A, Schuster M. Statistical parametric speech synthesis using deep neural networks[C]. In: Proceedings of the Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2013. 7962-7966.
- [49] Qian Y, Fan Y, Hu W, et al. On the training aspects of deep neural network (DNN) for parametric TTS synthesis[C]. In: Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014. 3829-3833.

- [50] Hashimoto K, Oura K, Nankaku Y, et al. The effect of neural networks in statistical parametric speech synthesis[C]. In: Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015. 4455-4459.
- [51] Wang Y, Skerry-Ryan R J, Stanton D, et al. Tacotron: Towards end-to-end speech synthesis[C]. In: Proceedings of the 2017 Conference of the International Speech Communication Association (InterSpeech). 2017. 4006-4010.
- [52] Shen J, Pang R, Weiss R J, et al. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions[C]. In: Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018. 4779-4783.
- [53] 刘知远, 孙茂松, 林衍凯, 等. 知识表示学习研究进展[J]. 计算机研究与发展, 2016, 53(2): 247-261.
- [54] Turian J, Ratinov L, Bengio Y. Word representations: a simple and general method for semi-supervised learning[C]. In: Proceedings of the 2010 annual meeting of the association for computational linguistics. Association for Computational Linguistics (ACL), 2010. 384-394.
- [55] Rendel A, Fernandez R, Hoory R, et al. Using continuous lexical embeddings to improve symbolic-prosody prediction in a text-to-speech front-end[C]. In: Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016. 5655-5659.
- [56] Pascual S, Bonafonte A. Prosodic break prediction with RNNs[C]. In: Proceedings of the 2016 International Conference on Advances in Speech and Language Technologies for Iberian Languages (IberSPEECH). 2016. 64-72.
- [57] Huang Y, Wu Z, Li R, et al. Multi-Task Learning for Prosodic Structure Generation Using BLSTM RNN with Structured Output Layer[C]. In: Proceedings of the 2017 Conference of the International Speech Communication Association (InterSpeech). 2017. 779-783.
- [58] Zen H. Deep learning in speech synthesis[J]. Technical Report, 2013. 1-5.
- [59] Fan Y, Qian Y, Xie F L, et al. TTS synthesis with bidirectional LSTM based recurrent neural networks[C]. In: Proceedings of the 2014 Conference of the International Speech Communication Association (InterSpeech). 2014. 1964-1968.
- [60] Achanta S, Godambe T, Gangashetty S V. An investigation of recurrent neural network architectures

- for statistical parametric speech synthesis[C]. In: Proceedings of the 2015 Conference of the International Speech Communication Association (InterSpeech). 2015. 859-863.
- [61] Zen H, Agiomyrgiannakis Y, Egberts N, et al. Fast, compact, and high quality LSTM-RNN based statistical parametric speech synthesizers for mobile devices[C]. In: Proceedings of the 2016 Conference of the International Speech Communication Association (InterSpeech). 2016. 2273-2277.
- [62] Zen H. Acoustic modeling in statistical parametric speech synthesis-from HMM to LSTM-RNN[C]. In: Proceedings of the 2015 Symposium on Machine Learning in Speech and Language Processing (MLSLP). 2015. 1-5.
- [63] Wang X, Lorenzo-Trueba J, Takaki S, et al. A comparison of recent waveform generation and acoustic modeling methods for neural-network-based speech synthesis[C]. In: Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018. 4804-4808.
- [64] Wang X, Takaki S, Yamagishi J. Investigating very deep highway networks for parametric speech synthesis[J]. Speech Communication, 2018, 96: 1-9.
- [65] Wang X, Takaki S, Yamagishi J. An autoregressive recurrent mixture density network for parametric speech synthesis[C]. In: Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017. 4895-4899.
- [66] Wang X, Takaki S, Yamagishi J. A Comparative Study of the Performance of HMM, DNN, and RNN based Speech Synthesis Systems Trained on Very Large Speaker-Dependent Corpora[C]. In: Proceedings of the 2016 ISCA Speech Synthesis Workshop (SSW). 2016. 118-121.
- [67] Wu Z, Watts O, King S. Merlin: An Open Source Neural Network Speech Synthesis System[C]. In: Proceedings of the 2016 ISCA Speech Synthesis Workshop (SSW). 2016. 202-207.
- [68] Van den OA, Dieleman S, Zen H, et al. WaveNet: A Generative Model for Raw Audio[C]. In: Proceedings of the 2016 ISCA Speech Synthesis Workshop (SSW). 2016. 125-125.
- [69] Kyle K Jose K, Sotelo S. Char2wav: End-to-end speech synthesis[C]. In: Proceedings of the 2017 International Conference on Learning Representations (ICLR). 2017. 1-5.
- [70] Mehri S, Kumar K, Gulrajani I, et al. SampleRNN: An unconditional end-to-end neural audio generation model[C]. In: Proceedings of the 2017 International Conference on Learning

- Representations (ICLR). 2017. 5-10.
- [71] Katzner K, Miller K. The languages of the world[M]. Routledge, 2002.
- [72] 呼和. 蒙古语和蒙古语的合成[A]. 见：第九届全国人机语音通讯学术会议论文集[C]. 2007. 322-331.
- [73] 敖其尔. 一种波形拼接的语音合成实验[A]. 见：第三届全国人机语音通讯学术会议（NCMMSC1994）论文集[C]. 1994. 408-412.
- [74] 高光来, 孟和, 吉雅. 基于词汇的蒙古语文语转换的实验[J]. 内蒙古大学学报: 自然科学版, 2000, 31(1): 121-124.
- [75] 萨其容贵. 蒙古语语音合成技术的研究[D]. 呼和浩特: 内蒙古大学, 2005.
- [76] 田会利. 基于词干词缀的有限词条的蒙古语语音合成系统的研究[D]. 呼和浩特: 内蒙古大学, 2007.
- [77] 孟和吉雅, 敖其尔. 基于词干词缀的蒙古语语音合成方法[J]. 内蒙古大学学报: 自然科学版, 2008, 39(6): 693-697.
- [78] 敖敏. 基于韵律的蒙古语语音合成研究[D]. 内蒙古大学, 2012.
- [79] 李婷会. 蒙古语的韵律预测方法研究[D]. 内蒙古大学, 2014.
- [80] 刘瑞. 基于条件随机场的蒙古语韵律短语预测方法[A]. 见：中国中文信息学会语音信息专业委员会.第十三届全国人机语音通讯学术会议(NCMMSC2015)论文集[C].中国中文信息学会语音信息专业委员会:清华信息科学与技术国家实验室(筹),2015. 552-556.
- [81] Liu R, Bao F, Gao G, et al. Mongolian prosodic phrase prediction using suffix segmentation[C]. In: Proceedings of the 2016 International Conference on Asian Language Processing (IALP). IEEE, 2016. 250-253.
- [82] 赵建东, 高光来, 飞龙. 基于 HMM 的蒙古语语音合成技术研究[J]. 计算机科学, 41(1): 80-82.
- [83] 赵建东. 基于隐马尔科夫模型的蒙古语语音合成技术研究[D]. 内蒙古大学, 2014.
- [84] Liu R, Bao F, Gao G, et al. Mongolian text-to-speech system based on deep neural network[C]. In: Proceedings of the 2017 National Conference on Man-Machine Speech Communication (NCMMSC). Springer, Singapore, 2017. 99-108.
- [85] Li J, Zhang H, Liu R, et al. End-to-End Mongolian Text-to-Speech System[C]. In: Proceedings of the 2018 International Symposium on Chinese Spoken Language Processing (ISCSLP). IEEE, 2018. 483-

487.

- [86] 刘郅楠. 基于端到端蒙古语语音合成方法的研究[D]. 内蒙古大学, 2019.
- [87] 清格尔泰. 蒙古语语法[M]. 内蒙古人民出版社, 1991.
- [88] Bao F, Gao G, Yan X, et al. Segmentation-based Mongolian LVCSR approach[C]. In: Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2013. 8136-8139.
- [89] 呼和. 蒙古语元音的声学分析[J]. 民族语文, 1999 (4): 58-60.
- [90] 飞龙,高光来,闫学亮.蒙古文字母到音素转换方法的研究[J].计算机应用研究,2013,30(06):1696-1700.
- [91] Liu Z, Bao F, Gao G. Mongolian Grapheme to Phoneme Conversion by Using Hybrid Approach[C]. In: Proceedings of the 2018 CCF International Conference on Natural Language Processing and Chinese Computing (NLPCC). Springer, Cham, 2018. 40-50.
- [92] Svantesson J O. Mongolian syllable structure[J]. Lund Working Papers in Linguistics, 2009, 42: 225-239.
- [93] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks[C]. In: Proceedings of the 2010 International Conference on Artificial Intelligence and Statistics (AISTATS). 2010. 249-256.
- [94] Kawahara H, Masuda-Katsuse I, De Cheveigne A. Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds[J]. Speech communication, 1999, 27(3-4): 187-207.
- [95] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[C]. In: Proceedings of the 2015 International Conference on Learning Representations (ICLR). 2015. 1-5.
- [96] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[C]. In: Proceedings of the 2013 Annual Conference on Neural Information Processing Systems (NeuIPS). 2013. 3111-3119.
- [97] Liu R, Bao F, Gao G, et al. A LSTM approach with sub-word embeddings for mongolian phrase break prediction[C]. In: Proceedings of the Proceedings of the 2018 International Conference on Computational Linguistics (COLING). 2018. 2448-2455.

- [98] Liu R, Bao F, Gao G, et al. Improving Mongolian Phrase Break Prediction by Using Syllable and Morphological Embeddings with BiLSTM Model[C]. In: Proceedings of the 2018 Conference of the International Speech Communication Association (InterSpeech). 2018. 57-61.
- [99] Liu R, Bao F L, Gao G, et al. Phonologically aware bilstm model for mongolian phrase break prediction with attention mechanism[C]. In: Proceedings of the 2018 Pacific Rim International Conference on Artificial Intelligence (PRICAI). Springer, Cham, 2018. 217-231.
- [100]Greff K, Srivastava R K, Koutník J, et al. LSTM: A search space odyssey[J]. IEEE transactions on Neural Networks and Learning Systems, 2016, 28(10): 2222-2232.
- [101]Schuster M, Paliwal K K. Bidirectional recurrent neural networks[J]. IEEE transactions on Signal Processing, 1997, 45(11): 2673-2681.
- [102]Santos C D, Zadrozny B. Learning character-level representations for part-of-speech tagging[C]. In: Proceedings of the 2014 International Conference on Machine Learning (ICML). 2014. 1818-1826.
- [103]Ling W, Dyer C, Black A W, et al. Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation[C]. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2015. 1520-1530.
- [104]Luong M T, Manning C D. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models[C]. In: Proceedings of the 2016 Annual Meeting of the Association for Computational Linguistics (ACL). 2016. 1054-1063.
- [105]Sennrich R, Haddow B, Birch A. Neural Machine Translation of Rare Words with Subword Units[C]. In: Proceedings of the 2016 Annual Meeting of the Association for Computational Linguistics (ACL). 2016. 1715-1725.
- [106]Bojanowski P, Grave E, Joulin A, et al. Enriching word vectors with subword information[J]. Transactions of the Association for Computational Linguistics, 2017, 5: 135-146.
- [107]Krishnan, Vijay and Vignesh Ganapathy. Named Entity Recognition [M]. 2005.
- [108]赵建东,高光来,飞龙.蒙古语语音合成语料库标注规则的设计[J].内蒙古大学学报(自然科学版),2013,44(03):324-328
- [109]Liu R, Bao F, Gao G. Building Mongolian TTS Front-End with Encoder-Decoder Model by Using Bridge Method and Multi-view Features[C]. In: Proceedings of the 2019 International Conference on

- Neural Information Processing (ICONIP2019). Springer, Cham, 2019. 642-651.
- [110]Cho K, van Merriënboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation[C]. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014. 1724-1734.
- [111]Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[C]. In: Proceedings of the 2015 International Conference on Learning Representations (ICLR). 2015. 23-27.
- [112]Caruana R. Multitask Learning: A Knowledge-based Source of Inductive Bias [C]. In: Proceedings of the 1993 International Conference on Machine learning (ICML), 1993. 41-48.
- [113]Caruana R. Multitask learning[J]. Machine learning, 1997, 28(1): 41-75.
- [114]Baxter J. A model of inductive bias learning[J]. Journal of Artificial Intelligence Research, 2000, 12: 149-198.
- [115]Xue Y, Liao X, Carin L, et al. Multi-task learning for classification with dirichlet process priors[J]. Journal of Machine Learning Research, 2007, 8(1): 35-63.
- [116]Kuang S, Li J, Branco A, et al. Attention Focusing for Neural Machine Translation by Bridging Source and Target Embeddings[C]. In: Proceedings of the 2018 Annual Meeting of the Association for Computational Linguistics (ACL). 2018. 1767-1776.
- [117]Bisani M, Ney H. Joint-sequence models for grapheme-to-phoneme conversion[J]. Speech Communication, 2008, 50(5): 434-451.
- [118]Rao K, Peng F, Sak H, et al. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks[C]. In: Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015. 4225-4229.
- [119]Toshniwal S, Livescu K. Jointly learning to align and convert graphemes to phonemes with neural attention models[C]. In: Proceedings of the 2016 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2016. 76-82.
- [120]Huszár F. How (not) to train your generative model: Scheduled sampling, likelihood, adversary? [J]. arXiv preprint arXiv:1511.05101, 2015. 1-5.
- [121]Schmidt F. Generalization in Generation: A closer look at Exposure Bias[C]. In: Proceedings of the

- 2019 Workshop on Neural Generation and Translation. 2019. 157-167.
- [122]Guo H, Soong F, He L, et al. A New GAN-Based End-to-End TTS Training Algorithm[C]. In: Proceedings of the 2019 Conference of the International Speech Communication Association (InterSpeech). 2019. 1288-1292.
- [123]Liu R, Sisman B, Li J, et al. Teacher-Student Training For Robust Tacotron-based TTS[C]. In: Proceedings of the 2020 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP2020). IEEE, 2020. 6274-6278.
- [124]Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]. In: Proceedings of the 2017 Advances in Neural Information Processing Systems (NeuIPS). 2017. 5998-6008.
- [125]He M, Deng Y, He L. Robust Sequence-to-Sequence Acoustic Modeling with Stepwise Monotonic Attention for Neural TTS [C]. In: Proceedings of the 2019 Conference of the International Speech Communication Association (InterSpeech). 2019.1293-1297.
- [126]Tachibana H, Uenoyama K, Aihara S. Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention[C]. In: Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018. 4784-4788.
- [127]Bengio S, Vinyals O, Jaitly N, et al. Scheduled sampling for sequence prediction with recurrent neural networks[C]. In: Proceedings of the 2015 Advances in Neural Information Processing Systems (NeuIPS). 2015. 1171-1179.
- [128]Griffin D, Lim J. Signal estimation from modified short-time Fourier transform[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1984, 32(2): 236-243.
- [129]Hinton G, Vinyals O, Dean J. Distilling the Knowledge in a Neural Network[J]. arXiv preprint arXiv:1503.02531, 2015. 1-5.
- [130]Liu R, Li J, Bao F, et al. The IMU speech synthesis entry for Blizzard Challenge 2019[C]. In: Proceedings of the 2019 Blizzard Challenge Workshop, 2019. 1-5.
- [131]Ren Y, Ruan Y, Tan X, et al. Fastspeech: Fast, robust and controllable text to speech[C]. In: Proceedings of the 2019 Advances in Neural Information Processing Systems (NeuIPS). 2019. 3165-3174.
- [132]Cai L, Cui D, Cai R. TH-CoSS, a Mandarin speech corpus for TTS[J]. Journal of Chinese Information Processing, 2007, 21(2): 94-99.

- [133]Song Y, Shi S, Li J, et al. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings[C]. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (ACL). 2018. 175-180.

致谢

行文至此，致谢的开始就是博士研究生学习生涯的结束。回顾六年的匆匆时光，百感交集。这一路努力奋斗、拼搏进取的青春岁月将成为我一生所珍视的纪念。在此，我要向所有曾陪伴、关心、照顾过我的人道一声感谢，让我能够在这一路上勇往直前！

首先，感谢我的导师高光来教授。高老师专业学识广博、治学态度严谨。该论文的完成，倾注了高老师大量的心血，从论文的选题、论文的执行到最终论文的定稿，高老师给予了悉心的指导和全方面的支持。“德高为师，身正为范”，高老师一丝不苟的科研精神将使我受益终生。借此机会，对高老师表达最诚挚的敬意和衷心的感谢！

感谢飞龙教授。飞龙老师对我的课题研究计划、研究工作的具体实施、实验过程中碰到的困难，以及论文的撰写提出了许多宝贵的建议，这些为我论文的顺利开展和完成打下了良好的基础。回想第一篇学术论文的写作投稿过程就是在飞龙老师的大力帮助下完成的，他对科研工作以及论文写作的独到见解都对我产生了潜移默化的影响，特此表达衷心的感谢！

感谢计算机学院给予我诸多关心和帮助的所有老师。感谢周建涛老师、侯宏旭老师、张学良老师、魏宏喜老师、马颖东老师、孙涛老师、安春燕老师、卢慧老师、刘实老师。学校的许多老师对我博士期间的学习、工作和生活同样给予了热心的帮助和支持，研究生院的姚淑艳老师、学生工作处的敖文格日乐老师、国际交流合作处的孙桂玲老师、外国语学院魏莉老师等。回想刚入校时，姚老师和敖老师带领我们校研究生会成员举办各种丰富的校园活动，教会我们许多书本里学不到的知识。孙老师在我出国访学的申请过程中不厌其烦的解答我的各种问题，为我顺利访学保驾护航。魏老师在博士英语课的课上都对我关照有加，她的学识和情怀同样是我学习的榜样。在此谢谢你们一路的关心、爱护和鼓励，祝你们工作顺利！万事顺遂！

感谢实验室所有同窗的帮助和关心。感谢张晖、苏向东、王炜华等师兄的大力帮助和支持。感谢路敏、李号、李劲东、王勇和师弟在科研上的帮助。路敏精通蒙古文，对实验过程中涉及到的蒙古文数据处理工作提供了无私帮助。李号思维敏捷，和他交流总能激发很多灵感。李劲东师弟基础知识扎实、动手能力强，与他的合作经历令人受益匪浅。王勇和师弟踏实认真，和他在科研过程中的讨论也使我受益颇多。感谢实验室的娜木汗、苏乙拉琪琪格、其乐木格、刘致楠、李庆龙、史霖炎、高伟、李敏、苏峰、赵飞、刘允、温子潇、高耀文、庞倩、牛米佳、王士杰以及其他没有在此列举出名字的师弟师

妹共同为实验室营造出轻松愉快的学习氛围，使原本枯燥的科研生活变得多姿多彩。感谢我的博士舍友顾瑞春，陪我度过一个又一个挑灯夜战的夜晚。感谢我的研究生好友，他们是郭广丰、郑伟、杨振华、张红伟、祁瑞东、史锦山、张新、杜健、姚志鹏、郭星、申志鹏、冯凯鹰、郑田玉、杨富强、徐业东等，我们互相鼓励、共同进步，怀念和你们一起把酒言欢的时光。祝你们前程似锦！

感谢我在新加坡国立大学访学期间的导师李海洲教授。李老师国际化的视野，前沿而精髓的学术造诣都让我永志不忘，也必将深刻影响我日后的工作和生活。感谢 Berrak Sisman 教授向我无私传授科研经验，在我科研工作中提出很多宝贵意见。感谢新加坡国立大学 HLT 实验室同学们在科研上的指点和生活上的关照，他们是张明阳、周坤、岳祥虎、罗兆杰、潘泽煦、陶睿杰、周一、高晓雪、周学浩、杜洪强、程禹、张马路等。感谢国家留学基金管理委员会在我访学期间提供奖学金资助。

最后，感谢我挚爱的家人。感谢我的父母，父亲沉稳寡言，母亲乐观坚强，你们用浓烈的爱为我构建起坚实而温馨的避风港。你们教会我责任和担当，你们告诉我仁爱和善良，你们是我值得用一生去感恩和报答的人，希望你们身体健康！感谢哥哥和嫂子一直以来对我无微不至的关怀与支持！感谢两位舅舅在我成长道路上对我的教导和关怀，我将永远心存感激！

衷心感谢参与评阅、评审本论文和出席答辩会的各位专家！

2020 年是特殊的一年，新型冠状病毒肆虐全球，同样要感谢在这场无声的战“疫”中无私奉献的英雄们，愿山河无恙！

雄关漫道真如铁，而今迈步重头越。博士论文即将完成，但科研之路才刚刚开始。

刘瑞

2020 年 4 月 27 日

攻读学位期间发表的学术论文

已发表论文（一作）

- [1] **Rui Liu**, Berrak Sisman, Jingdong Li, Feilong Bao, Guanglai Gao and Haizhou Li. Teacher-Student Training For Robust Tacotron-based TTS. In: 45th International Conference on Acoustics, Speech, and Signal Processing (ICASSP2020), pp 6274-6278, 2020.
- [2] **Rui Liu**, Feilong Bao, Guanglai Gao, Hui Zhang and Yonghe Wang. Improving Mongolian Phrase Break Prediction by Using Syllable and Morphological Embeddings with BiLSTM Model. In: 15th Conference of the International Speech Communication Association (InterSpeech2018), pp 57-61, 2018.
- [3] **Rui Liu**, Feilong Bao, Guanglai Gao, Hui Zhang and Yonghe Wang. A LSTM Approach with Sub-Word Embeddings for Mongolian Phrase Break Prediction. In: 27th International Conference on Computational Linguistics (COLING 2018), pp 2448-2455, 2018.
- [4] **Rui Liu**, Feilong Bao, Guanglai Gao, Hui Zhang and Yonghe Wang. Phonologically Aware BiLSTM Model for Mongolian Phrase Break Prediction with Attention Mechanism. In: 15th Pacific Rim International Conference on Artificial Intelligence (PRICAI2018), pp 217-231, 2018.
- [5] **Rui Liu**, Feilong Bao, and Guanglai Gao. Building Mongolian TTS Front-End with Encoder-Decoder Model by Using Bridge Method and Multi-view Features. In: 33rd International Conference on Neural Information Processing (ICONIP2019), pp 642-651, 2019.
- [6] **Rui Liu**, Jingdong Li, Feilong Bao and Guanglai Gao. The IMU speech synthesis entry for Blizzard Challenge 2019. In: Blizzard Challenge 2019.
- [7] **Rui Liu**, Feilong Bao, Guanglai Gao and Yonghe Wang. Mongolian Text-to-Speech System Based on Deep Neural Network. In: 14th National Conference on Man-Machine Speech Communication (NCMMSC2017), pp 99-108 , 2017.

- [8] **Rui Liu**, Feilong Bao, Guanglai Gao and Weihua Wang. Mongolian prosodic phrase prediction using suffix segmentation. In: 20th International Conference on Asian Language Processing (IALP2016), pp 250-253, 2016.
- [9] 刘瑞, 飞龙, 高光来, 张红伟, “基于条件随机场的蒙古语韵律短语预测方法”。第 13 届全国人机语音通讯学术会议 (NCMMSC2015), 天津, 中国, 2015.
- [10] **Rui Liu**, Berrak Sisman, Feilong Bao, Guanglai Gao and Haizhou Li. WaveTTS: Tacotron-based TTS with Joint Time-Frequency Domain Loss. In: The Speaker and Language Recognition Workshop (Odyssey 2020), pp 245-251, 2020.

已发表论文（非一作）

- [11] Jingdong Li, Hui Zhang, **Rui Liu**, Xueliang Zhang and Feilong Bao. End-to-End Mongolian Text-to-Speech System. In: 11th International Symposium on Chinese Spoken Language Processing (ISCSLP2018), pp 483-487, 2018.
- [12] Yonghe Wang, Feilong Bao, Guanglai Gao and **Rui Liu**. Research on Mongolian speech recognition based on TDNN-LSTM. In: 14th National Conference on Man-Machine Speech Communication (NCMMSC2017), pp 383-391, 2017.

已投稿论文（一作）

- [13] **Rui Liu**, Feilong Bao, Jichen Yang and Guanglai Gao. Exploiting Morphological and Phonological Features to Improve Prosodic Phrasing for Mongolian Speech Synthesis. In: IEEE Transactions on Audio, Speech and Language Processing (TASLP), 2020.

(二审阶段)

- [14] **Rui Liu**, Berrak Sisman, Feilong Bao, Guanglai Gao and Haizhou Li. Modeling Prosodic Phrasing with Multi-Task Learning in Tacotron-based TTS. In: IEEE Signal Processing Letters (SPL), 2020.

攻读学位期间参加的科研项目

- [1] 国家自然基金项目《基于深度学习的蒙古语语音问答技术研究》，项目号 61773224。
- [2] 国家自然基金项目《面向蒙古语新闻语音的新事件检测方法研究》，项目号 61563040。
- [3] 国家自然基金项目《面向电话语音的蒙古语关键词检测技术的研究》，项目号 61263037。
- [4] 内蒙古自然基金重大项目《规则与统计方法相结合的西里尔蒙古文与传统蒙古文相互转换系统研究》，项目号 2016ZD06。
- [5] 工信部电子信息产业基金项目《智能语音技术及产品研发与产业化——面向少数民族语言的智能语音技术及系统研发》（2014-425）。
- [6] （主持）内蒙古大学研究生科研创新项目《基于深度学习的蒙古语语音合成技术研究》，项目号 14020202-0123。