评委一评分，签名及备注	队号： 1190	评委三评分，签名及备注
评委二评分，签名及备注	选题： C	评委四评分，签名及备注

题目：手机语音识别技术的设计与实现

摘要

语音识别技术是一门新兴计算机智能技术。本文在参阅相关语音识别技术文献及书籍的基础上，综述了当前语音技术的进展，进而重点研究了手机语音识别技术的关键环节，提出了一种新的语音识别模型，并从实验仿真角度出发验证了改进模型的优势。

首先，建立了语音识别系统模型，并对模型的各个关键环节可采用的方法进行了的分析与比较，并选择现阶段较权威的方法进行重点研究，从理论研究与算法实现两个方面进行了详细的论述。其次，将 BP 神经网络网络和遗传算法综合起来应用于手机语音识别的研究中。文中详细论述了 BP 神经网络和遗传算法的原理，MATLAB 实现及各自的优缺点，深入分析了用遗传算法改进 BP 神经网络识别模型的优势，并用语音实验进行仿真，进一步验证了理论分析的正确性。通过仿真实验建立了相应的语音识别系统，并将其与普通的 BP 网络语音识别系统进行了比较分析，证明了该识别算法的高效性和方案的可行性。然后，考虑所建模型的假设条件及各关键环节存在的缺陷，本文给出了基于手机语音识别用户的用户手册指南，旨在让客户最有效地使用本文所建识别系统，减少识别故障或识别错误率。

本文的完成的主要任务及创新点可以概括为以下四个方面：

1. 将改进的 BP 神经网络算法应用于手机语音识别技术，解决了传统 BP 网络识别模型收敛速度慢，易陷入局部收敛的缺陷。较明显地提高了识别效率；

2. 全面考虑手机语音识别各环节的关键技术，建立参阅相当量的重要文献的基础上，通过比较选择了权威的方案，并给出理论推导及实现方案。而不是单单应用某成熟理论或者传统经验；

3. 建立语音实验仿真平台，通过实验数据及识别效率证明了本文模型的高效性和正确性；

4. 总结了本文模型的不足，并给出了具体的尚待完善的环节及可以进一步发展的方向。

关键字：手机语音识别　BP 神经网络　遗传算法　语音实验

手机语音识别技术的设计与实现

目　录

1.假设和符号说明

1.1 模型假设

假设 1：识别模型的处理对象为录制的一段语音，不考虑断句不考虑返回给用户的形式，仅仅考虑语音内容识别的实现问题。

假设 2：语音实验过程，假设我们的同学发音完全正确，即为标准的普通话，同时忽略录音仪器，录音环境噪声过大等的影响。

假设 3：为方便模型验证，仅使用手机语音常用的口令训练和测试模型。

假设 4：鉴于模型的简化，仅考虑特定人发音的识别，非特定人发音的识别在本文中不做深入讨论。

1.2 符号说明

符号	符号含义
ARS	语音识别技术
MFCC	Mel 倒谱系数
MLP	多层感知网络
BP	神经网络
GA	遗传算法

2.问题背景与分析

2.1 问题背景

语音识别技术(Automatic Speech Recognition，ASR)是一门起源于 20 世纪 50 年代的新兴计算机智能技术，其核心思想是让机器能够识别和理解人类口述语言，并予以正确的响应或恰当的反馈，最终实现人机语言沟通的高智能化目标。

从语音识别技术的发展历程来看，早期的语音识别技术着眼于如何将人类口述语言转化为可读、可存储的机器语言，即“语音辨识”领域。而经过专家学者们的努力，语音识别的破解之法如傅立叶转换、倒频谱参数等已逐渐应用于该领域，使语音辨识系统已达到一个可接受的程度，并且辨识度愈来愈高。随着计算机功能、信号处理、软件编程等一系列高端技术领域的深入发展，目前语音识别技术的研究重点已从最基本的“辨识”功能转向“人机交流”功能，即机器能够针对用户的口述请求或命令做出正确的响应、反馈。例如移动终端上的火热应用——语音对话机器人、语音助手；智能手机上的广泛应用——语音拨号、语音导航、“微信”、娱乐小程序“小黄鸡”等等。更有研究人员在开发将语音识别技术应用于实现手机解锁功能[1]。

从应用的范围来看，语音识别是一门涉及面很广的交叉学科，与计算机、通信、语音语言学、数理统计、信号处理、神经心理学和人工智能等学科都有密切的关系。语音识别的最大优势在于使得人机用户界面更加自然和容易使用。随看计算机技术、模式识别和信号处理技术及声学技术等的发展，使得能满足各种需

要的语音识别系统实现成为可能。近二三十年来，语音识别在工业、军事、交通、医学、民用诸方面，特别是在计算机、信息处理、通信与电子系统、自动控制等领域中有着越来越广泛的应用。

2.2 语音识别技术研究现状及本文的研究思路

语音识别技术具有强大的应用前景。试想一下，未来的公司会议不再需要人工记录员；大学课堂老师不必再辛苦板书，学生也不必繁忙做笔记；电影后期制作的配音人员和字幕人员也不必再摧残耳朵……很多事情都将因语音识别技术的应用变得简单高效。而为了让人们能够享受到更多的诸如此类的语音识别技术的便利性，很多学者和研究人员在这方面做出了贡献。

在语音识别的传统技术研究方面，房安栋等[1]研究了语音识别中一种说话人的声纹识别，通过利用正交小波滤波器组来对信号进行预滤波，将基音周期参数和Mel倒谱系数(MFCC)两者组合，得到新的声纹特征。Kajarekar[2]提出一种基于手机应用的多项式支持向量机(SVM)方法，解决了手机上的语音识别功能。李曜等[3]针对传统的隐含马尔可夫模型(HMM)在语音识别方面存在的缺陷，提出了一种在识别的后处理阶段使用段长模型的方法，并应用在汉语识别系统上。田莎莎等[4]提出一种改进的MFCC特征参数，即BMFCC特征参数的方法，并证明该方法可以提高语音识别时的识别率和运算速度。

在语音识别的现实生活应用方面，Eklund等[5]通过调查在瑞典的电话语音识别和语音合成技术的应用范围及人们的感受，探讨了电话语音功能的开发规模为多大时才能得到最佳的使用效果。徐子豪等[6]通过研究语音识别技术和无线传感网络，设计了一套能够通过远程语音遥控进行便捷控制的智能家居系统，并通过测试证明该系统的识别率可以达到98%。Ayres等[7]通过比较手机、电脑上的语音识别技术，并对特定的场景下的语音识别系统进行比对和分析，创建了语音识别技术的语法扩展框架。苏征远等[8]设计了以ARM处理器为核心，Linux为操作系统的嵌入式语音识别设备，并证明该语音识别设别具有通用性好、拓展能力强等特点。郭超等[9]使用支持向量机作为分类算法，构建了低信噪比环境下的孤立词非特定人语音识别系统，并证明该系统具有较好的识别率。

在语音识别技术的研究方面，Salmela等[10]通过将多层感知网络(MLP)和隐马尔可夫模型相结合，创建了一种混合语音识别系统，并通过对网络的训练和测试得到较好的识别效果。吴炜烨[11]以神经网络的语音识别应用为基础，提出一种改进的BP神经网络结构，并通过参数比较证明改进后的神经网络具有更好的语音识别效果。Howell等[12]设计了基于语音识别的移动电话服务系统，并通过使用一个适当的空间隐喻提高可视化水平，使参与者更有效地在分层服务体系结构中实现语音导航功能。宋清昆等[13]提出了一种基于改进遗传算法的小波神经网络控制器，并研究证明此方法可以克服基本遗传算法收敛速度慢，容易陷入“早熟”收敛，计算稳定性不好等一系列问题，进一步提高了小波神经网络控制器的性能。宋亚男等[14]利用MATLAB软件编程工具，以凌阳SPCE061A为基础，结合机器人语音识别的需求，实现了机器人语音识别系统演示实验和半开放实验。余华等[15]通过改进径向基神经网络，并将其运用于语音识别系统，从而证明改进型的神经网络在针对非特定人的孤立词识别上效果很好。Linder等[16]以神经网络技术为基础了，开发一个声学语音分析系统，并验证该系统可以作为筛选设备监控、记录和诊断的专业化语音识别系统。周琍[17]以BP神经网路作为语音识别的基础，并利用MATLAB封装友好便捷的图形界面，实现了较好的人机交互语音识别功能。

刘纪平[18]研究了遗传算法和神经网络相结合在语音识别应用中的设计，并通过仿真实验建立了相关的语音识别系统。Melin等[19]提出了一种将遗传优化神经网络的模块化与模糊响应进行集成的新方法，并验证了此方法可广泛用于生物基因识别或语音识别等领域。

以这些前人的相关研究为基础，通过详细的比对分析，本文选取神经网络和遗传优化算法相结合的方式，并利用 MATLAB 强大的语音处理功能，分析并建立了手机语音识别模型系统。在考虑单音，句式等不同情形下的语音识别需求的可能，从模型收敛速度，和识别率来论证所提模型的合理性。最后，文章运用仿真实例，对该模型的识别率、正确率作出仿真评价，验证了改进后的 BP 神经网络语音识别模型具有很高的可靠性。

2.3 问题重述与分析

2.3.1 问题重述

本文的研究重点是为手机运营商设计和构建一套语音识别模型(即语音机器人系统)。即手机用户通过微信公共账号等形式将已录制的语音文件发送给客服机器人，而客服机器人通过该语音识别模型可以正确识别用户的需求(例如查询话费余额、查询套餐余量、查询最新的优惠活动等)。

为简化构建语音识别模型的过程，在以一段录制的语音作为一个识别单位，不需要考虑断句，不需要考虑返回给用户的形式，只要求能够识别出语音内容的前提下，依次解决以下三个问题：

问题 1：通过建立模型来说明语音识别技术的各个环节；

问题 2：根据已建立的语音识别模型为手机运营商制定一个可行的用户操作规则；

问题 3：根据已制定的用户操作规则，以一个实际例子来验证“客服机器人”语音识别模型的有效性和可行性，例如：查询话费功能的实现。

2.3.2 问题分析

根据应用程序开发理论，一个新程序的开发一般包括五个环节：需求分析——可行性分析——框架设计——逻辑实现——测试调试。本文在“问题背景”部分已详细陈述手机运营商的设计要求以及手机用户的实际需求，因此接下来进入可行性分析环节，即分析语音识别模型在解决上述三个问题时所运用的理论、技术知识以及实现难点。

问题 1 的分析

语音识别模型的构建是本文的核心。在构建模型前，首先需从逻辑上确定一个完整的语音识别系统具体包括哪几个环节。

综合文献[14][20][31]，本文总结完整的语音识别系统的构建大致可以分为四部分：

一是语音采集及预处理部分：通过采集环节将人类口述语言以波形文件的格式储存在计算机的记录存储结构中；然后经过预处理，将已存储的语音波形文件转化为随时间变化的语音特征序列，这个过程也称为语音特征提取，通过预处理后的特征序列即可以体现原始语音信号的主要特征。

二是语音识别部分(即声学模型的匹配)：声学模型是指体现不同语音单元特征的统计分布参数模型。声学模型常采用包含状态转移的高斯混合模型来表示。语音识别部分的任务是为已预处理的语音特征序列寻找最匹配的声学模型。

三是语音处理部分(即语言模型)：语音单元之间通常存在着相关性及冗余，语言模型就是利用语音单元相关性以提高识别准确率的一种模型方法[2]。语言模型一般由识别语音命令、语法网络或统计方法等构成。

四是识别系统的测试与校正：任何一个系统构建后，都需要一个测试与调试环节来确保系统的有效性。因此语音识别系统还需要一个“自适应”的反馈模块，对“声学模型”和“语言模型”的处理结果进行必要的“校正”，以进一步提高语音识别系统的准确度。

问题 2 的分析

手机运营商之所以期望制定用户操作规则，是因为语音识别模型中的某些问题在现阶段的技术水平下是无法避免的，例如噪音过大识别效果就不好等等情形。同时也是为了使用户在使用客服机器人语音识别模型时更便捷、更高效，防止错误操作等。具体而言，本问题的解决思路考虑一下几点：

(1)用户操作规则应当简洁明了、通俗易懂；

(2)用户操作规则应当包含语音识别模型的功能及用途介绍；

(3)用户操作规则应当包含对语音识别模型的使用范围、使用方式、收费情况的清晰介绍；

(4)用户操作规则应当包含模型使用过程中异常情形的处理方法或温馨提示；

(5)用户操作规则应当表明语音识别模型对用户个人信息的安全性和保密性；

(6)用户操作规则的用语应当尽量礼貌、准确、人性化。

问题 3 的分析

在实例验证方面，建立语音实验仿真平台，通过实验数据及识别效率证明了本文模型的高效性和正确性。

3.模型的理论准备

3.0 本节总论

本节结构主要分为以下几个部分：

1.对语音识别的发展研究历史及现状做了相关概述，简单介绍了现有语音识别系统的模型及分类。

2.阐述了语音识别的基本原理和流程，重点介绍并分析了语音识别各种环节的关键方法及最新研究进展，并在论文下一节给出了相应的验证结果及具体分析。

3.针对本文具体要解决的问题，讨论并分析了几种常用的识别模型在本问题中的实现方法及优缺点，进而提出了自己的创新模型，并从理论上分析了所提模型的优势及可行性前景，并将在下一章节(模型建立及求解环节)用具体算例验证本文所提模型的合理与优势。

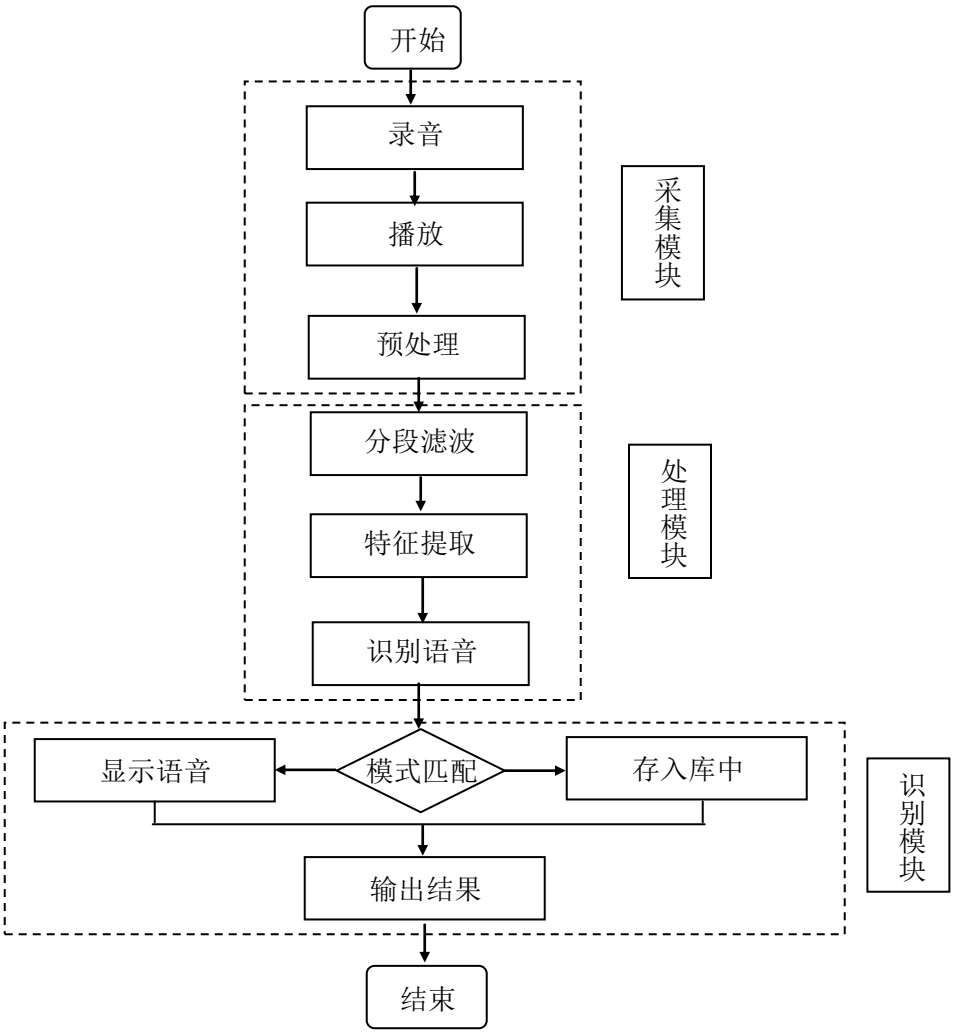


图 3.1 语音识别系统流程图[31]

3.1 模型前期准备

3.1.1 语音采集模块

MATLAB(Matrix Laboratory)是目前非常流行的科学计算和工程计算软件工具，提供了一个高性能的数值计算、信号处理和可视化开发环境以及大量具有特殊用途的函数和工具箱，其中包括基本的语音信号处理函数。基本语音处理函数包括wav文件的读写函数，声卡的录音和放音函数及波形的显示函数等，还有一些第三方开发的语音处理工具(如VoiceBox等)。利用这些函数和工具，我们就可以方便地进行一些语音的处理工作。例如：

1. 语音采样可以用以下命令实现：

x=wavrecord(k,fs,'dtype')

其中x为语音采样信号，fs为采样率，k为采样秒数，'dtype'为采样数据类型。

2. 语音数据也可以用以下命令从语音文件中读取：

x=wavread(filename')

3.1.2 语音处理模块

1. 预处理模块[21]

在语音信号进入语音信号处理系统之前，很多因素都能够引起语音信号出现失真现象，比如发声器官自身原因或者是采集语音信号的各种设备的因素，所以在语音信号真正处理之前就必须对语音信号进行预处理工作，或叫做前端处理。进行语音信号预处理的最终目的是尽可能的确保处理后所得到的语音信号更加平滑、更加均匀，并且最好能够提高语音信号的质量。语音信号的预处理有很多方式，本文主要采用预加重、分帧、加窗等预处理方式。

2. 语音端点检测[1]

在许多包含语音识别的系统中，如语声应答系统、说话人识别系统和语音识别系统等，都要求首先对系统的输入信号进行判断，准确找出语音段的起始点和终止点，这就是语音端点检测。端点检测作为语音识别系统预处理阶段遇到的第一个关键技术，其准确性在某种程度上直接决定了整个语音识别系统的成败。端点检测的目的就是从包含语音的信号中确定出语音的起点以及终点，使采集的数据真正是语音信号的数据，从而减少数据量、运算量和减少处理时间。

在语音端点检测中目前比较常用的方法有以下几种[18]：

1. 基于短时能量和短时过零率的双门限端点检测方法

想要通过检测区分语音、噪音、静音三种语音状态可以采用的一种方式是通过能量区分。当信噪比较高的时候仅仅需要通过计算输入语音信号的短时能量或者短时平均幅值就能够很好的把语音信号从噪音背景信号中区分开来。具体来说是通过比较输入语音信号的能量与语音阈值的大小来判断是语音信号还是噪声信号，从而得出输入信号的语音端点。

2. 基于频带方差的端点检测方法

语音处理系统是随时间变化的，实际进行计算的时候是利用短时频带方差的方法，这种语音端点检测方法实质就是计算某一帧信号内各频带能量之间的方差。这种方法可以利用短时频带方差判断语音的起止点。如果是语音信号能量越大时起伏越激烈，频带方差越大；相反对于噪音信号，能量越小，起伏就越平缓，频带方差的值越小。由于清音和噪声段的能量很相近所以双门限方法就可能出现一些错误的划分，然而，采用频带方差的方法就会使频谱分布出比较均匀的噪声，例如白噪声，其频谱方差就比较小，而对于清音和浊音，其频谱方差就都比较大，从而可以更好的检测出语音信号的端点。

3. 基于倒谱的端点检测方法

功率谱的对数值的逆傅氏变换称为倒谱。信号的倒谱能很好表示语音信号的的特征，在较强的噪声环境下，经常采用倒谱系数作为端点检测的特征值。基于倒谱的语音端点检测方法类似基于能量的端点检测方法。利用倒谱距离轨迹可以检测语音信号的端点。但是，如果信号严重失真，那么就会给端点检测带来困难，因为这种情况下难以选择合适的门限值，实验表明基于倒谱的语音端点检测在某些方面还是有很多可取之处的，具体的介绍本文不再过多涉及，有兴趣的读者可以查阅相关的文献。

本文主要介绍的方法就为基于短时能量和短时过零率的双门限端点检测方法。

3.特征参数提取[27]

根据人的听觉机理的研究发现，人耳对不同频率的声波有不同的听觉灵敏

度。从 200Hz 到 5KHz 之间的语音信号对语音的清晰度影响最大。低音掩蔽高音容易，反之则困难。在低频处的声音掩蔽的临界带宽较高频端小。据此，人们从低频到高频这一段频带内按临界带宽的大小由密到稀安排一组带通滤波器，对输入信号进行滤波。将每个带通滤波器输出的信号能量作为信号的基本特征，对此特征经过进一步处理后就可作为语音的输入特征。由于这种特征不依赖于信号的性质，对输入信号不做任何的假设和限制，又利用了听觉模型的研究成果，因此，这种参数与基于声道模型的 LPCC 相比具有较好的鲁棒性，更符合人耳的听觉特性，而且当信噪比降低时仍然具有较好的识别性能。

MFCC 是在 Mel 标度频率域提取出来的倒谱参数，Mel 标度描述了人耳频率的非线性特性，它与频率的关系可用下式近似表示：

$$Mel(f) = 2595 \times \lg(1 + f/700)$$

求 Mel 倒谱系数的方法是将时域信号做时/频变换后，对其对数能量谱用依照 Mel 刻度分布的三角滤波器组做卷积，再对滤波器组的输出向量做离散余弦变换(DCT)，这样得到的前 N 维向量称为 MFCC。

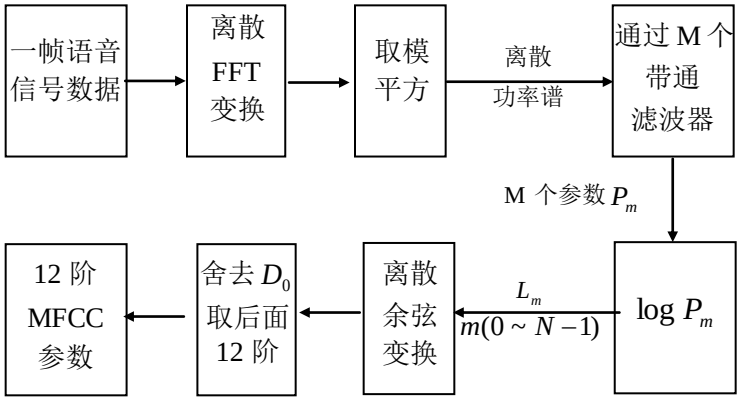


图 3.2　MFCC 参数提取流程

1. 语音信号进行预处理

输入的语音信号含有外界的噪音，因此需要通过预处理进行滤波过程。这里可以直接调用 MATLAB 中的 Filter 函数进行滤波，其使用形式为

$$y = filter([1 - 0.9375], 1, x)$$

其中 x 为输入信号， y 为过滤后的信号，$[1-0.9375]$ 为与滤波相关的参数。

2. 对语音信号的分帧过程

对于录制的语音信号的需要进行分频，同时为了使得帧与帧之间能够平滑过渡，保持其连续性，采用重叠分帧的方式。一般分帧长度为 256、512 或其他个数的数据点，而帧移设置为分帧长度的 0 ~ 0.5 。这里直接利用 MATLAB VoiceBox 工具箱中的 enframe 函数完成。其使用形式为

$$S = enframe(y, framelen, frameinc)$$

其中，x 为待分帧信号，S 为已完成分帧的各组信号； $framelen$ 为分帧的长度， $frameinc$ 为帧移量。

3. 对已分帧信号进行加入 Hamming 窗

为了加强音框左端和右端的连续性，还需要用窗函数 $\omega(n)$ 乘以一分帧的语

音信号 $S(n)$，得到

$$S_w(n) = \omega(n) \cdot S(n)$$

采用 Hamming 窗则

$$\omega(n) = \begin{cases} 0.54 - 0.46\cos(\dfrac{2\pi n}{N-1}) & 0 \le n \le N \\ 0 & \text{其他} \end{cases}$$

也可以直接调用 MATLAB 中 hamming 函数，其使用形式为

$$\omega = \text{hamming}(N)$$

4. 对各分帧信号进行离散傅立叶变换

$$X_a(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi k}{N}}, \quad 0 \le k \le N$$

式中 $x(n)$ 为输入的语音信号，N 表示傅立叶变换的点数。这里直接采用 MATLAB 中的快速傅立叶变换函数 fft，其调用形式为

$$X_a(k) = fft(x)$$

5. 对步骤四得到的频谱幅度进行取模平方，得到其能量谱

$$E_a(k) = conj(X_a(k)).* X_a(k)$$

6. 将各帧语音信号的能量谱通过一组 Mel 尺度的三角滤波器

定义一个有 M 个滤波器的滤波器组(滤波器的个数和临界带的个数相近)，采用的滤波器为三角滤波器，中心频率为 $f(m), m = 1,2,\cdots M$，这里取 $M = 20$。各 $f(m)$ 之间的间隔随着 m 值的减小而缩小，随着 m 值的增大而增宽。

三角滤波器的频率响应定义为：

$$H_m(k) = \begin{cases} 0, k < f(m-1) \\ \dfrac{2(k - f(m-1))}{(f(m+1) - f(m-1))(f(m) - f(m-1))}, f(m-1) \le k \le f(m) \\ \dfrac{2(f(m+1) - k)}{(f(m+1) - f(m-1))(f(m+1) - f(m))}, f(m) \le k \le f(m+1) \\ 0, k \ge f(m+1) \end{cases}$$

其中 $\sum_{m=0}^{M-1} H_m(k) = 1$；

计算每个滤波器组输出的对数能量为：

$$S(m) = \ln\left(\sum_{k=0}^{N-1} E_a(k) H_m(k)\right), \quad 0 \le m \le M$$

7．经离散余弦变换(DCT)得到 MFCC 系数：

$$C(n)=\sum_{m=0}^{N-1}S(m)\cos\left(\frac{\pi n(m-0.5)}{M}\right)，\quad 0\le n\le M$$

也可以直接调用 MATLAB 中的 dct 函数得到。

3.2 语音识别模块准备

3.2.1 语音识别模型框架[31]

不同的语音识别系统，虽然具体实现细节有所不同，但所采用的基本原理与模型都是相似的。一个完整的语音识别系统可大致分为三部分：

(l) 语音特征提取：其目的是从语音波形中提取出随时间变化的语音特征序列。

(2) 声学模型与模式匹配(识别算法)：声学模型通常将获取的语音特征通过学习算法产生。在识别时将输入的语音特征同声学模型(模式)进行匹配与比较，得到最佳的识别结果。

(3) 语言模型与语言处理：语言模型包括由识别语音命令构成的语法网络或由统计方法构成的语言模型，语言处理可以进行语法、语义分析。对小词表语音识别系统，往往不需要语言处理部分。

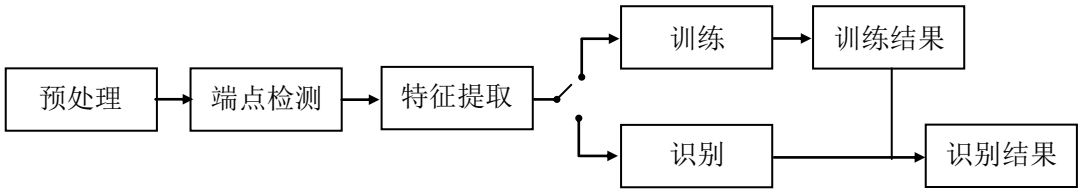


图 3.3 语音识别基本原理图

3.2.2 语音识别系统的分类

语音识别系统可以根据对输入语音的限制加以分类[14][27].

如果从说话者与识别系统的相关性考虑，可以将识别系统分为三类：(1)特定人语音识别系统；仅考虑对于专人的话音进行识别。(2)非特定人语音系统；识别的语音与人无关，通常要用大量不同人的语音数据库对识别系统进行学习。(3)多人的识别系统；通常能识别一组人的语音或者成为特定组语音识别系统。该系统仅要求对要识别的那组人的语音进行训练。

如果从说话的方式考虑，也可以将识别系统分为 3 类：(1)孤立词语音识别系统。孤立词识别系统要求输入每个词后要停顿；(2)连接词语音识别系统。连接词输入系统要求对每个词都清楚发音。一些连音现象开始出现；(3)连续语音识别系统。连续语音输入是自然流利的连续语音输入。大量连音和变音会出现。

如果从识别系统的词汇量大小考虑．也可以将识别系统分为三类：(1)小词汇量语音识别系统。通常包括几十个词的语音识别系统；(2)中等词汇量的语音识别系统。通常包括几百个词到上千个词的识别系统；(3)大词汇量语音识别系统。通常包括几千到几万个词的语音识别系统。随着计算机与数字信号处理器运算能力以及识别系统精度的提高，识别系统根据词汇量大小进行分类也不断进行变化。目前是中等词汇量的识别系统。将来可能就是小词汇量的语音识别系统。这些不同的限制也确定了语音识别系统的困难度[3]。

本文将综合上述情况考察所建模型的自适应性。

3.2.3 目前几种主要的语音识别技术[18]

(1) 线性预测编码(Linear Predictive Coding，LPC)技术和动态时间规整(Dynamic Time Warping，DTW)算法：这是语音识别技术发展早期的两种技术，主要用于研究孤立语音、小词汇量的孤立数字等。基于模板匹配原则，LPC 技术有效地解决了语音特征提取问题，DTW 则有效地解决了说话人语速不均匀造成的时间伸缩变化问题。这两种技术主要适合于对特定人的语音识别系统，而针对非特定人识别时则效果不佳。

(2) 矢量量化(Vector Quantization，VQ)和隐马尔可夫模型(Hidden Markov Model，HMM)算法：这两种语音识别技术是在动态时间规整技术的基础上发展起来的，但不再以模板匹配为支撑，而是基于概率统计原则，主要用于中、大词汇量以及连续语音识别。这两种技术代表了语音识别技术的飞速发展。

HMM 技术的一个基本假设是：语音信号是准平稳的，并且其中平稳部分可以由 HMM 中的状态来表示。因而传统的 HMM 算法并不能很好的表现语音信号的时域结构，分辨能力较弱，需要对其进行相关的改进和处理。

(3) 人工神经网络(ANN)算法：人工神经网络是一个自适应非线性动力学系统，通过模拟人脑神经细胞的结构和功能来进行处理工作。随着对神经网络应用领域的不断深入研究，人们发现多层前馈神经网络具有优异的分类特性，并在模式识别领域表现出巨大的潜力[3]。而语音识别技术作为模式识别领域的一个重要分支，也开始得到学者们的广泛重视。

(4)以 HMM 和 ANN 相结合的语音识别方法已被研究，这种结合方法充分发挥了 HMM 时间规整能力强和 ANN 分辨能力强的特点，因而可以得到较好的时间匹配和模式分类。而其他的一些模式识别、机器学习方法如支持向量机(Support Vector Machine)技术、进化计算(Evolutionary Computation)技术等在语音识别领域的应用也逐渐被开发。

3.2.4 本文在阅读前人研究文献基础上提出的新识别技术

在传统的语音识别算法中，模式匹配法是在对语音做过预处理之后，通过特征参数的提取及模式匹配完成识别。由于语音信号的高度多变性，输入模式要与标准模式完全匹配是几乎不可能的。因此，识别时要预先制定好计算输入的语音特征模式与模板语音特征模式的类似或距离的规则，距离最小者就是最类似的模式。神经网络的语音识别算法与传统方法的差异在于提取了语音的特征参数后，不像传统方法那样有输入模式与标准模式的比较匹配，而是靠神经网络中大量的连接权对输入模式进行非线性运算，产生最大兴奋的输入点就代表了输入模式对应的分类。比较起来，神经网络识别系统更接近人类的感知过程。

但是，神经网络语音识别算法仍然具有一些明显的不足。具体体现在以下几个方面：

(l) 训练算法收敛速度慢。无论是前馈神经网络，径向基神经网络，还是自适应神经网络，由于神经网络的训练算法基本都采用线性反馈调整权值的原理，因此有着收敛速度慢的缺陷。

(2) 容易陷入局部极小值。BP 算法是以梯度下降法为基础的非线性优化方法，因而可能产生一个局部最小值。且实际问题的求解空间往往存在着许多局部极小点，更使这种陷于局部最小值的可能性大大增加。通常，在 BP 算法中随机设置初始权值，初始权值的大小对是否会陷入这种局部最小有很大的影响。如果

这些权值太大，则网络一开始就会陷入这种局部最小值。网络的训练一般较难达到全局最优。

(3) 网络结构不易确定。由于人工神经网络的结构具有极大的灵活性，因此到目前还没有具有普遍意义的定义网络结构的方法，有待进一步研究。

本文在阅读大量优化算法[4][11][13][15][17][18][22]的基础上，发现遗传算法具有很好的处理非线性问题的能力，可以实现全局寻优。遗传算法(GeneticAlgorithm，简称 GA)是从于自然选择和遗传规律的并行全局搜索算法，具有较强的宏观搜索能力既算法具有寻优的全局性，克服了 BP 算法中容易出现的局部极小问题。这启发本文将遗传算法和 BP 算法相结合，希望得到一种更高效的算法，这将在下面论述中进行具体的讨论。

3.3 神经网络语音识别模型

3.3.1 BP 神经网络基本原理[17]

BP 神经元与其它神经元类似，不同的是 BP 神经元的传输函数为非线性函数，最常用的函数是 109519 和 tansig 函数，有的输出层也采用线性函数。BP 网络一般为多层神经网络。信息从输入层流向输出层，因此是一种多层前馈神经网络。如果多层 BP 网络的输出层采用 S 形传输函数(如 109519)，其输出值将会限制在一个较小的范围内(0，l)；而采用线性传输函数则可以取任意值。举例来说，三层的 BP 网络的结构如下图所示[33].

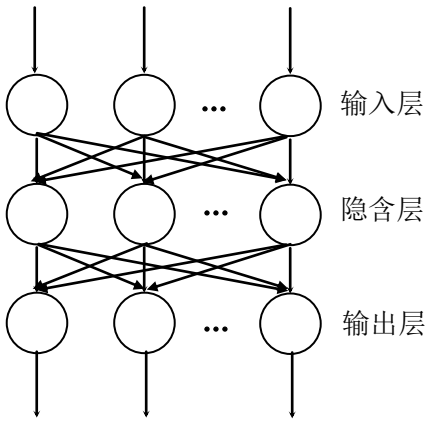


图 3.4 三层的 BP 网络结构图

在确定了 BP 网络的结构后，要通过输入和输出样本集对网络进行训练，即网络的阈值和权值进行学习和修正，以使网络实现给定的输入输出映射关系。BP 网络的学习过程分为两个阶段：

第一个阶段是输入已知学习样本，通过设置的网络结构和前一次迭代的权值和阈值，从网络的第一层向后计算各神经元的输出。

第二个阶段是根据网络的输出误差对网络连接权值和闭值进行修改，从最后一层向前计算各权值和阈值对总误差的影响(梯度)，据此对各权值和闭值进行修改。以上两个过程反复交替，直到达到收敛为止。由于误差逐层往回传递，以修正层与层之间的权值和阈值，所以称该算法为误差反向传播算法。

神经网络通过这种自适应的学习功能，手机用户录制语音时的物理特征(如用户声源和声道结构的差别)、环境特征(如背景噪音、传输通道等)的限制，同时不拘泥于特殊的语音参数和输入模式，实现对不同用户在不同时间、不同地点的语音信号的快速训练和精确识别。同时神经网络的可训练性使其能够根据不同语音

特征自主改变网络性能，而且能够进行快速判断并具有容错性，特别适合于解决无固定算法而又样本量大变化多的语音识别领域。

针对本文研究的具体问题而言，基于神经网络的语音识别原理图大致如下所示：

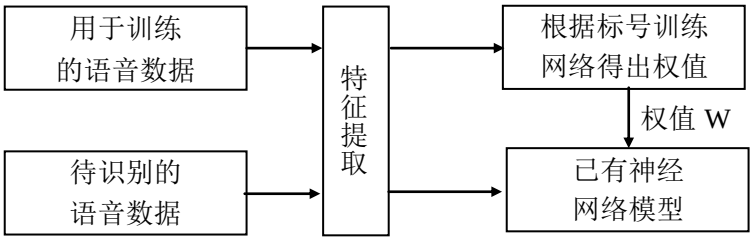


图 3.5 基于神经网络的语音识别原理图

3.3.2 BP 神经网络算法[33]

本部分论述神经网络的基本算法，在模型求解部分将根据此算法编写神经网络 MATLAB 程序。基本 BP 算法包括两个方面：信号的前向传播和误差的反向传播。即计算实际输出时按从输入到输出的方向进行，而权值和阈值的修正从输出到输入的方向进行。

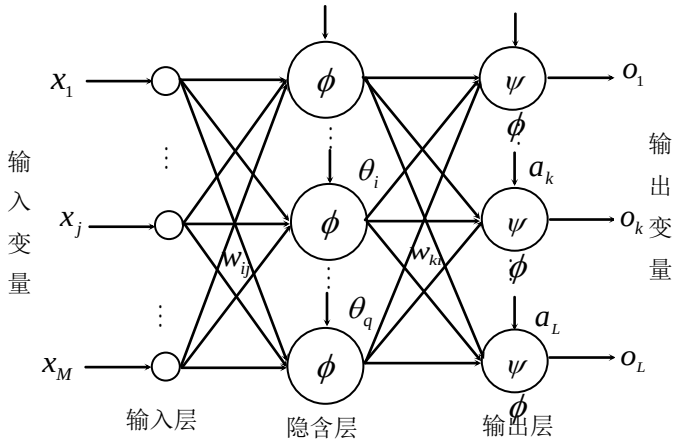


图 3.6 BP 网络结构

Fig.3.6 Structure of BP network

图中：x_j 表示输入层第 j 个节点的输入，$j=1,\cdots M$；

w_{ij} 表示隐含层第 i 个节点到输入层第 j 个节点之间的权值；

θ_i 表示隐含层第 i 个节点的阈值；

$\phi(x)$ 表示隐含层的激励函数；

w_{kj} 表示输出层第 k 个节点到隐含层第 i 个节点之间的权值，$i=1,\cdots q$；

a_k 表示输出层第 k 个节点的阈值，$k=1,\cdots L$；

$\psi(x)$ 表示输出层的激励函数；

o_k 表示输出层第 k 个节点的输出。

(1) 信号的前向传播过程

隐含层第 i 个节点的输入 net_i：

$$net_i = \sum_{j=1}^{M} w_{ij} x_j + \theta_i \tag{3-1}$$

隐含层第 i 个节点的输出 y_i：

$$y_i = \phi(net_i) = \phi(\sum_{j=1}^{M} w_{ij} x_j + \theta_i) \tag{3-2}$$

输出层第 k 个节点的输入 net_k：

$$net_k = \sum_{i=1}^{q} w_{ki} y_i + a_k = \sum_{i=1}^{q} w_{ki} \phi(\sum_{j=1}^{M} w_{ij} x_j + \theta_i) + a_k \tag{3-3}$$

输出层第 k 个节点的输出 o_k：

$$o_k = \psi(net_k) = \psi(\sum_{i=1}^{q} w_{ki} y_i + a_k) = \psi\left(\sum_{i=1}^{q} w_{ki} \phi(\sum_{j=1}^{M} w_{ij} x_j + \theta_i) + a_k\right) \tag{3-4}$$

(2) 误差的反向传播过程

误差的反向传播，即首先由输出层开始逐层计算各层神经元的输出误差，然后根据误差梯度下降法来调节各层的权值和阈值，使修改后的网络的最终输出能接近期望值。

对于每一个样本 p 的二次型误差准则函数为 E_p：

$$E_p = \frac{1}{2} \sum_{k=1}^{L} (T_k - o_k)^2 \tag{3-5}$$

系统对 P 个训练样本的总误差准则函数为：

$$E = \frac{1}{2} \sum_{p=1}^{P} \sum_{k=1}^{L} (T_k^p - o_k^p)^2 \tag{3-6}$$

根据误差梯度下降法依次修正输出层权值的修正量 Δw_{ki}，输出层阈值的修正量 Δa_k，隐含层权值的修正量 Δw_{ij}，隐含层阈值的修正量 $\Delta\theta_i$。

$$\Delta w_{ki} = -\eta \frac{\partial E}{\partial w_{ki}}; \quad \Delta a_k = -\eta \frac{\partial E}{\partial a_k}; \quad \Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}}; \quad \Delta\theta_i = -\eta \frac{\partial E}{\partial \theta_i} \tag{3-7}$$

输出层权值调整公式：

$$\Delta w_{ki} = -\eta \frac{\partial E}{\partial w_{ki}} = -\eta \frac{\partial E}{\partial net_k} \frac{\partial net_k}{\partial w_{ki}} = -\eta \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial net_k} \frac{\partial net_k}{\partial w_{ki}} \tag{3-8}$$

输出层阈值调整公式：

$$\Delta a_k = -\eta \frac{\partial E}{\partial a_k} = -\eta \frac{\partial E}{\partial net_k} \frac{\partial net_k}{\partial a_k} = -\eta \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial net_k} \frac{\partial net_k}{\partial a_k} \tag{3-9}$$

隐含层权值调整公式：

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} = -\eta \frac{\partial E}{\partial net_i} \frac{\partial net_i}{\partial w_{ij}} = -\eta \frac{\partial E}{\partial y_i} \frac{\partial y_i}{\partial net_i} \frac{\partial net_i}{\partial w_{ij}} \tag{3-10}$$

隐含层阈值调整公式：

$$\Delta\theta_i = -\eta\frac{\partial E}{\partial \theta_i} = -\eta\frac{\partial E}{\partial net_i}\frac{\partial net_i}{\partial \theta_i} = -\eta\frac{\partial E}{\partial y_i}\frac{\partial y_i}{\partial net_i}\frac{\partial net_i}{\partial \theta_i} \tag{3-11}$$

又因为：

$$\frac{\partial E}{\partial o_k} = -\sum_{p=1}^{P}\sum_{k=1}^{L}(T_k{}^p - o_k{}^p) \tag{3-12}$$

$$\frac{\partial net_k}{\partial w_{ki}} = y_i \quad , \quad \frac{\partial net_k}{\partial a_k} = 1 \quad , \quad \frac{\partial net_i}{\partial w_{ij}} = x_j \quad , \quad \frac{\partial net_i}{\partial \theta_i} = 1 \tag{3-13}$$

$$\frac{\partial E}{\partial y_i} = -\sum_{p=1}^{P}\sum_{k=1}^{L}(T_k^p - o_k^p)\cdot\psi'(net_k)\cdot w_{ki} \tag{3-14}$$

$$\frac{\partial y_i}{\partial net_i} = \phi'(net_i) \tag{3-15}$$

$$\frac{\partial o_k}{\partial net_k} = \psi'(net_k) \tag{3-16}$$

所以最后得到以下公式：

$$\Delta w_{ki} = \eta\sum_{p=1}^{P}\sum_{k=1}^{L}(T_k{}^p - o_k{}^p)\cdot\psi'\left(net_k\right)\cdot y_i \tag{3-17}$$

$$\Delta a_k = \eta\sum_{p=1}^{P}\sum_{k=1}^{L}(T_k{}^p - o_k{}^p)\cdot\psi'\left(net_k\right) \tag{3-18}$$

$$\Delta w_{ij} = \eta\sum_{p=1}^{P}\sum_{k=1}^{L}(T_k^p - o_k^p)\cdot\psi'\left(net_k\right)\cdot w_{ki}\cdot\phi'(net_i)\cdot x_j \tag{3-19}$$

$$\Delta\theta_i = \eta\sum_{p=1}^{P}\sum_{k=1}^{L}(T_k^p - o_k^p)\cdot\psi'\left(net_k\right)\cdot w_{ki}\cdot\phi'(net_i) \tag{3-20}$$

基于上述公式推导[11]，我们可以归纳为以下几个步骤：

第一步，网络初始化；给各连接权值分别赋一个区间(-1，1)内的随机数，设定误差函数 e，给定计算精度值 ε 和最大学习次数 M；

第二步，随机选取第 k 个输入样本及对应期望输出

$$d_o(k) = \left(d_1(k), d_2(k), \cdots, d_q(k)\right)$$

$$x(k) = \left(x_1(k), x_2(k), \cdots, x_n(k)\right)$$

第三步，计算隐含层各神经元的输入和输出；

第四步，利用网络期望输出和实际输出，计算误差函数对输出层的各神经元的偏导数 $\delta_o(k)a$；

第五步，利用隐含层到输出层的连接权值、输出层的 $\delta_o(k)$ 和隐含层的输出计

算误差函数对隐含层各神经元的偏导数$\delta_h(k)$；

第六步，利用输出层各神经元的$\delta_o(k)$和隐含层各神经元的输出来修正连接权值$w_{ho}(k)$；

第七步，利用隐含层各神经元的$\delta_h(k)$和输入层各神经元的输入修正连接权。

第八步，计算全局误差$E=\frac{1}{2m}\sum_{k=1}^{m}\sum_{o=1}^{q}(d_o(k)-y_o(k))^2$；

第九步，判断网络误差是否满足要求。当误差达到预设精度或学习次数大于设定的最大次数，则结束算法。否则，选取下一个学习样本及对应的期望输出，返回到第三步，进入下一轮学习。

3.4 遗传算法

3.4.1 遗传算法基本原理[34]

遗传算法(Genetic Algorithms)是1976年由美国Michigan大学Holland教授正式提出的模拟生物在自然环境中的遗传和进化过程而形成的一种自适应全局优化概率搜索算法。它把“生存竞争、优胜劣汰、适者生存”的生物竞争机制引入优化参数而形成的编码串联群体中，将搜索空间(欲求取解空间)映射为遗传空间，即把每一个可能的解编码为一个向量，称为一个染色体或个体(Individual)。向量的每个元素称为基因。所有染色体组成种群(population)，并按预定的目标函数对每个染色体进行评价，根据其结果给出一个适应度(FitnesS)的值。算法开始时先随机产生一些染色体，计算其适应度。根据适应度对各染色体进行选择(selection)、交叉(erossover)、变异(mutation)等遗传操作 (geneticoperators)，剔除适应度低的染色体，留下适应度高的染色体，从而得到新的种群，这样的过程称为一次进化(Evolution)。遗传算法就这样反复迭代，向着更优解的方向进化，直至满足某种预定的优化指标。

3.4.2 遗传算法基本步骤[34]

生物体每一代的进化过程主要是通过染色体之间的交叉和染色体的变异来完成的。与此相对应，遗传算法中最优解的搜索过程也模仿生物的这个进化过程，使用所谓的遗传操作作用于群体中，从而得到新一代群体。

使用选择、交叉、变异三种遗传算子的进化过程的主要运算如下：

Step1：初始运行参数。设置进化代数计数器 t=0;设置最大进化代数 T；随机生成 M 个个体作为初始群体 $P(0)$；

Step2：通过某种编码方法(如二进制编码、实值编码)随机产生一组初始个体，构成初始种群，其中每个初始个体代表了问题的一个初始解；

Step3：个体评价。计算群体 $P(t)$中各个个体的适应度；

Step3：选择运算。将选择算子作用于群体；

Step4：交叉运算。将交叉算子作用于群体；

Step5：变异运算。将变异算子作用于群体。群体 $P(t)$经过选择、交叉、变异运算后得到下一代群体 $P(t+1)$；

StePG：终止条件判断。若 $t<T$，则 $t=t+1$，转到步骤 2；若 t 全 T，则以进化过程中所得到的具有最大适应度的个体作为最优解输出，终止运算。

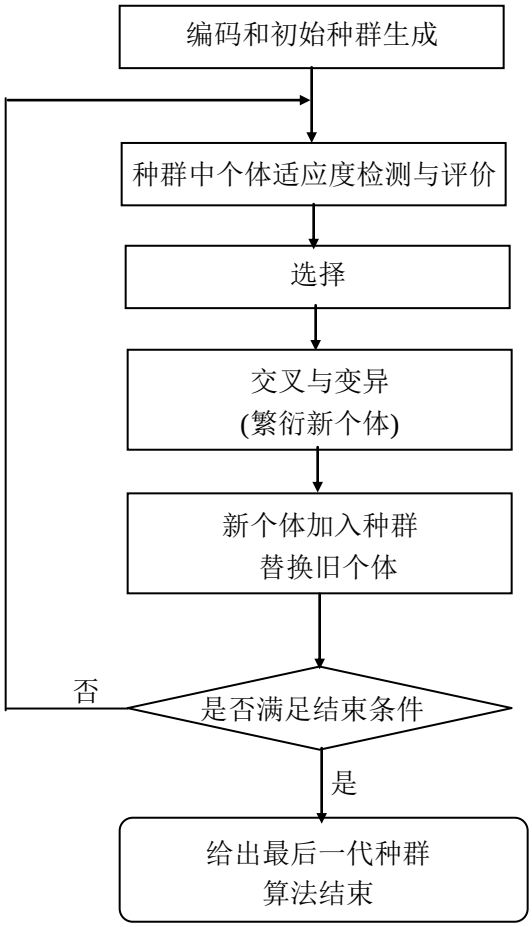


图 3.7 遗传算法流程

3.5 基于遗传算法优化的神经网络模型的建立

BP 算法虽然具有简单和可塑的优点，但它本质上是基于梯度下降的算法，因而不可避免地具有收敛速度慢、易陷入局部极小值、全局搜索能力弱、受网络结构限制等缺点，而遗传算法具有很强的全局搜索能力，可以很好地解决 BP 算法所遇到的问题。而另一方面，遗传算法易出现过早的不成熟的收敛，虽然相应的改进方法很多，但主要目的是维持算法进行中的个体多样性，如调整操作参数，增加种群规模等，这些方法都没有考虑到使改进后的算法具有学习能力和鲁棒性，这两点也正好是神经网络的优势所在。

由上可知，若能把神经网络和遗传算法结合起来，则可以充分利用两者的优点，使新算法既有神经网络的学习能力和鲁棒性，又有遗传算法的很强的全局随机搜索能力。

这启发本文将遗传算法和 BP 算法相结合，得到一种更高效的算法，这将在下面论述中进行具体的讨论。

3.5.1 改进算法的基本思想

首先用遗传算法对网络隐层节点数、初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子进行优化设计，在解空间中定位出较好的搜索空间，然后用神经网络算法在这些小的解空间中对网络的隐层节点数、初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子再次寻优，搜索出最优解[34]。

3.5.2 算法实现的关键步骤[34]

基于遗传算法优化的BP神经网络学习算法，具体步骤如下：

Step1：随机产生一组实值串种群，每一个个体表示网络隐层节点数、初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子的一个集合；

Step2：对实值串中的隐层节点数部分进行解码，生成相应的网络结构；

Step3：对实值串中的其余部分分别进行解码，生成网络的初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子；

Step4：正向运行网络，计算适应度值,评价网络性能；

Step5：通过选择、交叉和变异等遗传操作，产生下一代种群，形成下一代网络；

Step6：重复Step2至Step5，直到R≤Rmax。或达到进化代数g_max。此时，将最终种群中的个体进行解码，从而得到通过GA优化后的值。其中Rmax为遗传算法所要达到的性能指标；

Step7：将GA优化后的网络隐层节点数、初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子作为BP神经网络使用的参数；

Step8：双向运行网络，调节网络权值(包括阈值)和伸缩平移因子，评价网络性能。保存网络隐层节点数、初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子，学习过程结束。

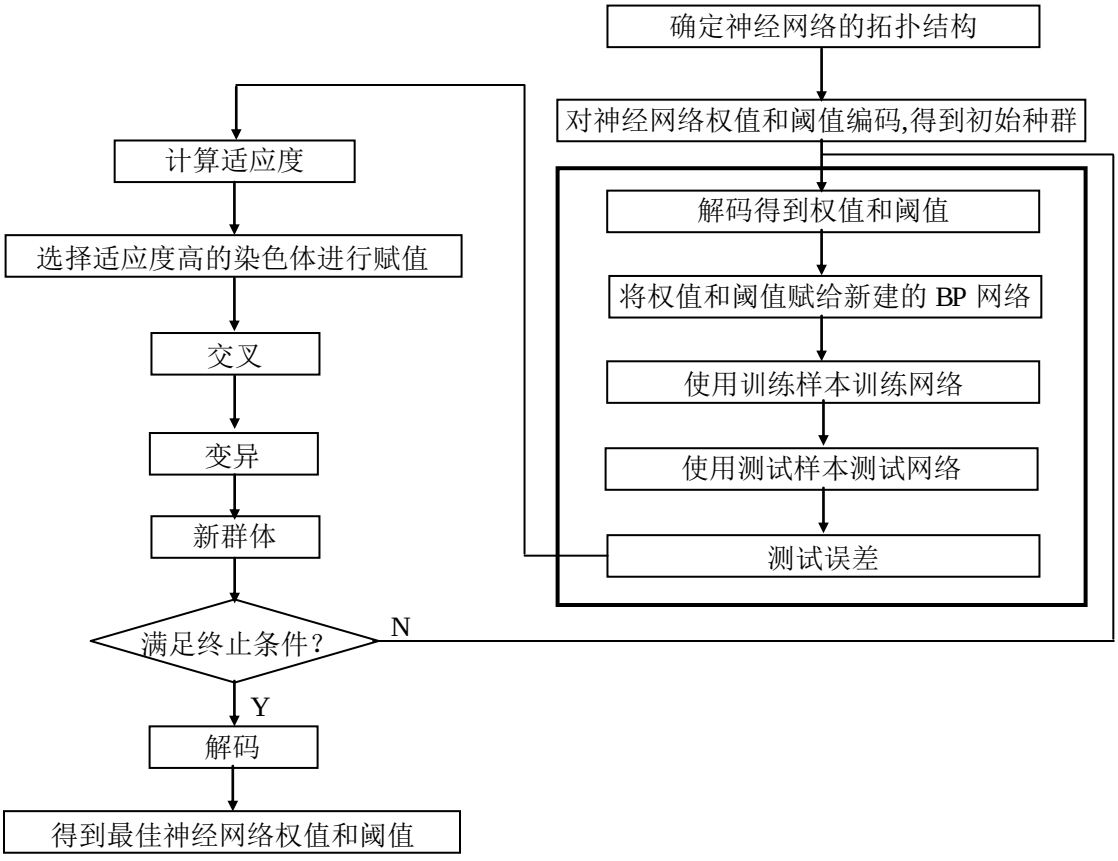


图 3.8 改进的 BP 神经网络的语音识别流程图

3.6 用户规则制定

本节假设已用上述理论建立好某手机的语音识别功能，为方便个性化处理，

本文为语音识别系统取名为“大芒助手”，并模拟用户使用情况，给出相应的用户指南及注意事项。

“大芒助手”语音识别系统用户指南

感谢您对“大芒助手”语音识别系统的选择和钟爱。本系统依托于改进后的神经网络以及 MATLAB 编程语言平台，专门为 3G 移动通信设施用户研发设计，特别适合对语音功能需求较多的手机用户，大幅度提高了用户手机的利用价值，备受手机用户青睐。

✧ 功能介绍

“大芒助手”是一款功能出色的语音助手，其界面精美时尚，操作简洁，语音识别率高，能够为用户提供及时、贴身的语音服务，给用户的日常通信和生活帮大忙，大大提供手机的利用价值，满足用户多样化的需求。

除一般语音助手提供的语音拨号、语音读短信功能以外，“大芒助手”还为用户提供包括套餐余量查询、话费余额查询、最新优惠活动查询、语音搜索、语音定位、语音回复电子邮件等在内的一系列实用功能和服务。此外，开放的“大芒助手”是一个典型的中文版语音应用程序，不受限于手机系统，可在任何智能手机上运行，便捷实用。

✧ 安装说明

点击“大芒助手”安装程序：DaMang.exe 即可完成安装。

✧ 操作说明

(1) 单击“大芒助手”程序图标，进入语音助手主界面；

(2)在主界面菜单选择您所需服务功能，单击即可进入具体功能界面。如“查询”页面；

(3)在“查询”页面，长按麦克风图标，开始进行语音录制，直至语句说完后松开，然后单击选项“发送”，即可完成语音发送。若要放弃已录制语音，单击“取消”即可；

(4)返回语音助手主界面，选择“未读信息”服务功能，单击进入其界面。即可看到刚刚语音查询的具体结果——“话费清单”。然后单击该清单，在弹出选项(“文本阅读”、“语音阅读”、“删除”)中选择您所需操作单击即可。操作完成之后，单击“返回”，即可回到“未读信息”的界面。

(5)返回语音助手主界面，单击“退出”服务功能，即可退出本语音助手程序。

✧ 注意事项

为方便您正确使用“大芒助手”语音程序，请认真阅读以下简明规则。

(1) 录制语音时，请尽量语句清晰，停顿有序，语速不易过快，以保证大芒助手可以更加快速的识别您的语音需求，进而缩短对您响应的时间；

(2) 录制语音时，请尽量选择安静场所，噪音过大或过多都会加重大芒助手的负担，造成你使用过程中的不便；

(3) 大芒语音助手的语音识别功能尚未完全开发，目前仅设置有普通话、粤语两种语音的识读，特向您致歉，但我们的开发团队仍在不懈努力，以为您提供更多的地方语言识读功能；

(4) 当手机电池不足时，请尽量避免使用“大芒助手”，这样既可减少您与“大芒助手”对话过程中手机突然关机的尴尬，也可为您节约宝贵的电量以应对紧急来电；

(5) 在公共场所或不便语音录制的场所，您可选择不与大芒助手语音对话，其程序中有自带专门的键盘操作供您使用，同样实现“大芒助手”语音带给你的一切功能；

(6) “大芒”助手依赖数据流量运行，因此在手机网络信号较差的场所，大芒助手的语音发送功能可能会受影响，敬请您谅解；

(7) 在使用“大芒”助手的各项功能时，请遵守各项法律法规并尊重您所在地区的风俗习惯、他人的隐私等合法权益，避免干扰他人生活。

✧ **特别声明：**

(1) “大芒”助手对于用户的个人信息等一系列隐私内容严格保密，并根据用户使用协议采取相关措施谨慎保护；

(2) “大芒”助手是某某服务供应商的专利产品，未经授权或特别说明，不得用于非法信息传递等场合；

(3) “大芒助手”以方便用户为宗旨，内容健康向上，乐观积极，请您放心使用；

(4)为保证您的安全，请从正规网址下载大芒助手语音识别程序，对于不法途径下载引起的相关经济法律问题，我们不承担责任，并保留相关追责权。

感谢您的阅读。您的支持和信赖是“大芒助手”不断进步的动力！

4.模型建立及算法实现

4.0 本节总论

上文主要论述了语音识别技术各环节的理论和相应算法的具体思路及详细的推导过程。而在本节中，重点就手机语音识别的具体问题进行展开，并验证理论建模的正确性，但不再赘述第3节中具体的理论过程及公式推导。

4.1 模型识别前处理

4.1.1 模型训练数据生成

1. 特定人单个词语音数据提取及预处理

运用本文所建系统的采集模块分别录制本队某同学“话”、“费”、“查”、“询”4 个语音，并命名为 1a.wav、2a.wav、3a.wav、4a.wav，作为实验 a 的模型训练和检验数据，并重新录制该同学一组“话”语音作为实验 a 的检测数据。为方便模型这里记“话”对应数字 1，费对应数字 2，“查”对应数字 3，“询”对应数字 4。

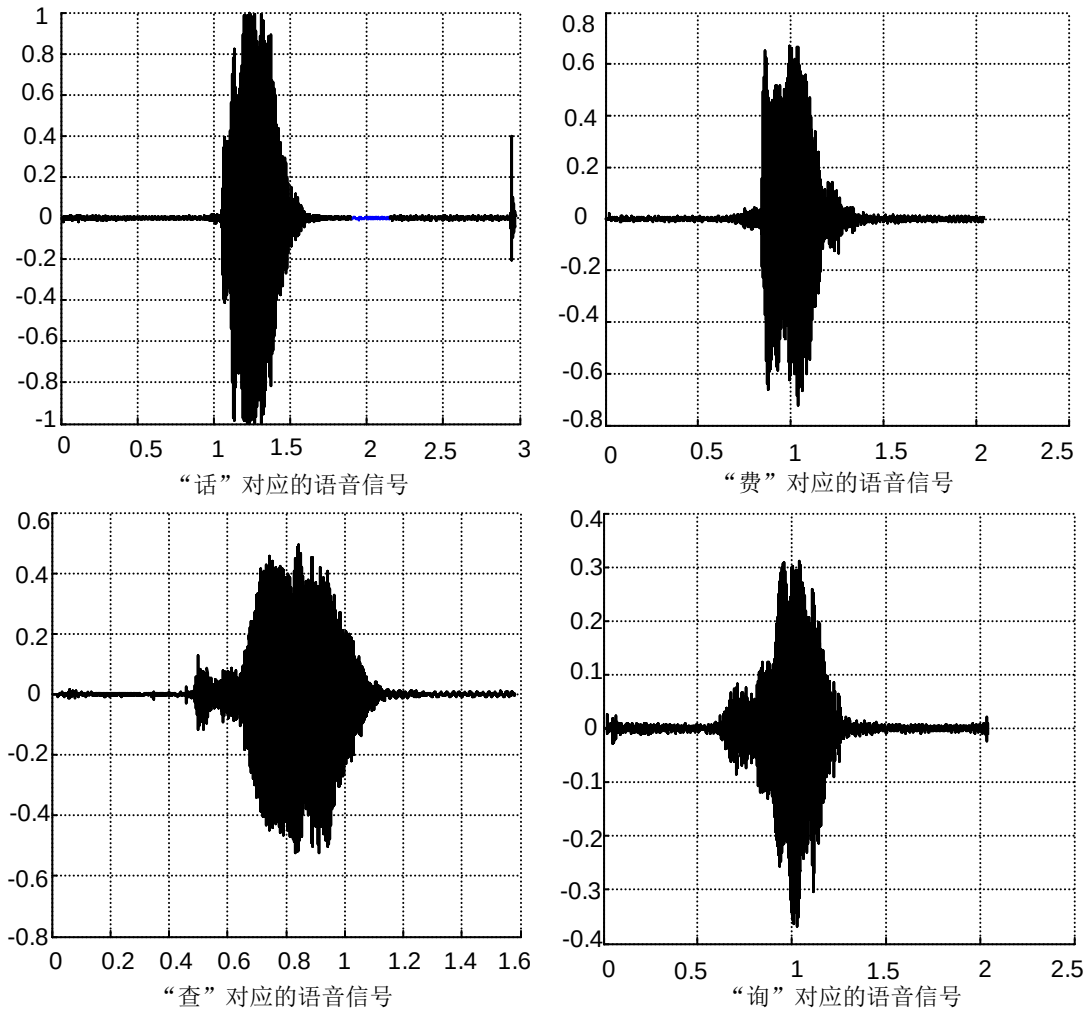


图 4.1 “话费查询”四个字对应的语音信号

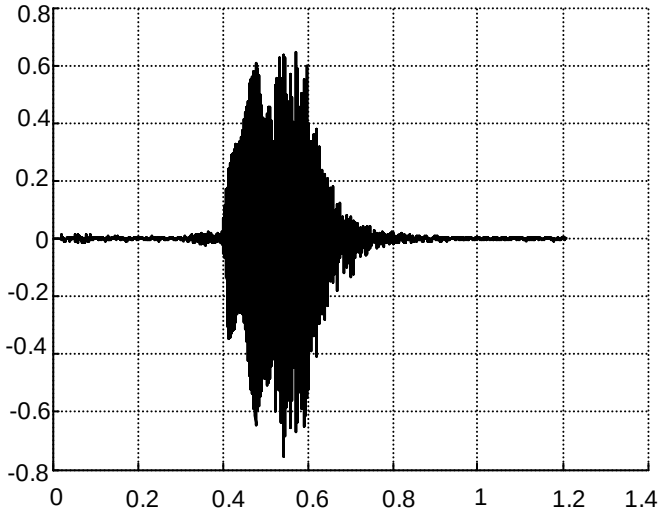
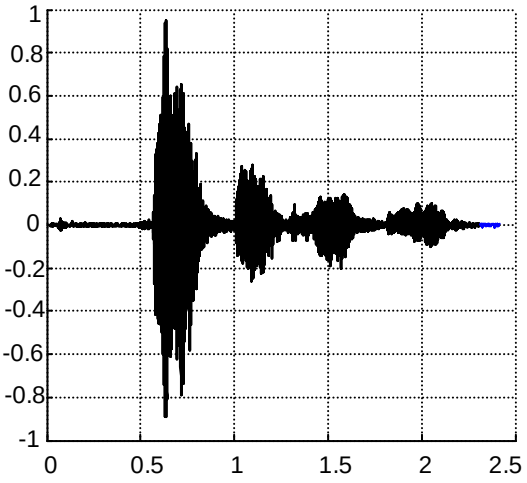


图 4.2 “话”待检测的语音信号

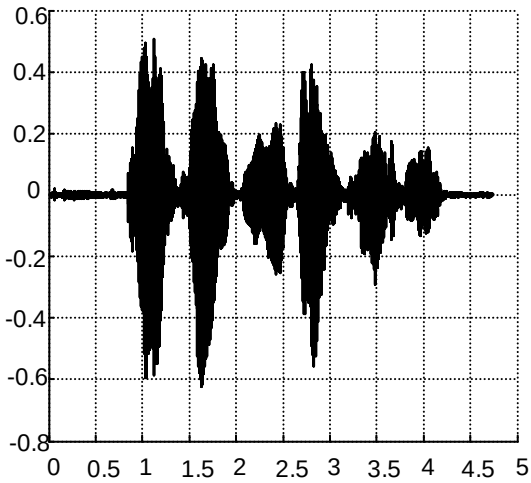
2. 特定人词组语音数据提取

同理，利用本文所建系统的采集模块分别录制本队某同学“话费查询”、“套餐余量查询”、“业务办理”3 个语音，并命名为 1b.wav、2b.wav、3b.wav，作为实验 b 的语言库模型训练和检验数据，并重新录制该同学一组“话费查询”的语

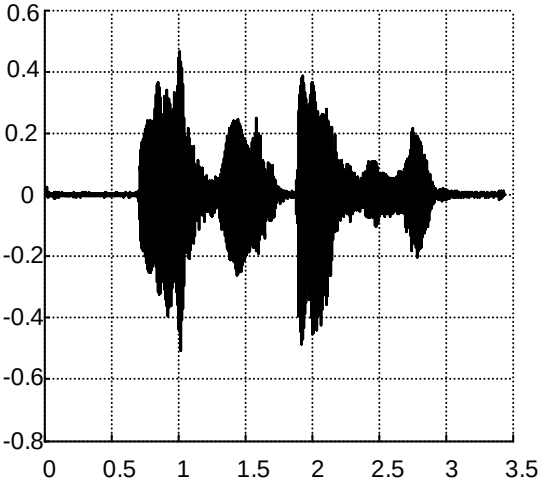
音作为实验 b 的检测数据。为方便模型这里记“话费查询”对应数字 1，“套餐余量查询”对应数字 2，“业务办理”对应数字 3。利用 MATLAB 提取相应的语音信号见下图。



“话费查询”对应的语音信号



“套餐余量查询”对应的语音信号



“业务办理”对应的语音信号

图 4.3　特定短句对应的各语音信号对比

4.1.2 语音数据端点检测

利用第 3 节语言数据端点检测技术，编写 MATLAB 程序，提取有效语音信息，这里只展示“话”的有效语音信息提取图像，其他端点检测图见附件。

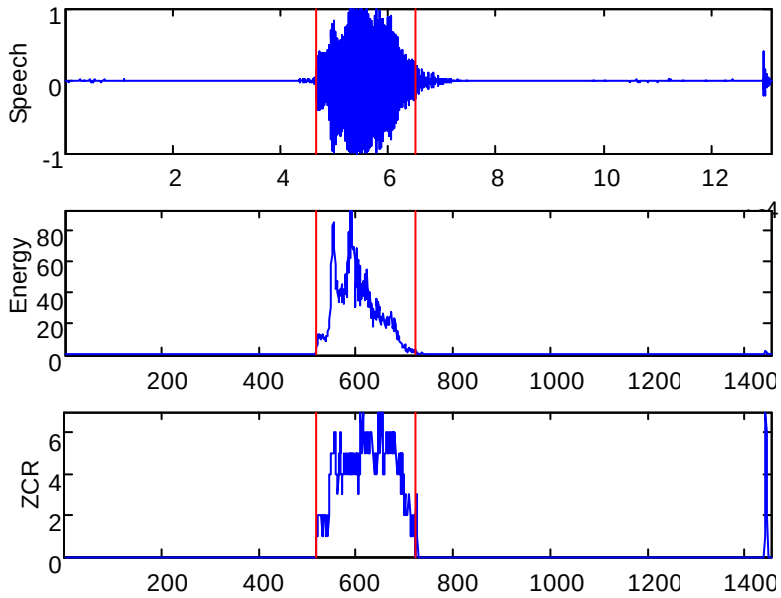


图 4.5 “话”的有效端点检测

4.1.3 语音数据特征参数提取

具体提取方法及步骤说明在第三节中已详细说明，MATLAB 程序详见附件，这里不再赘述，仅提供语音 MFCC 曲面图。

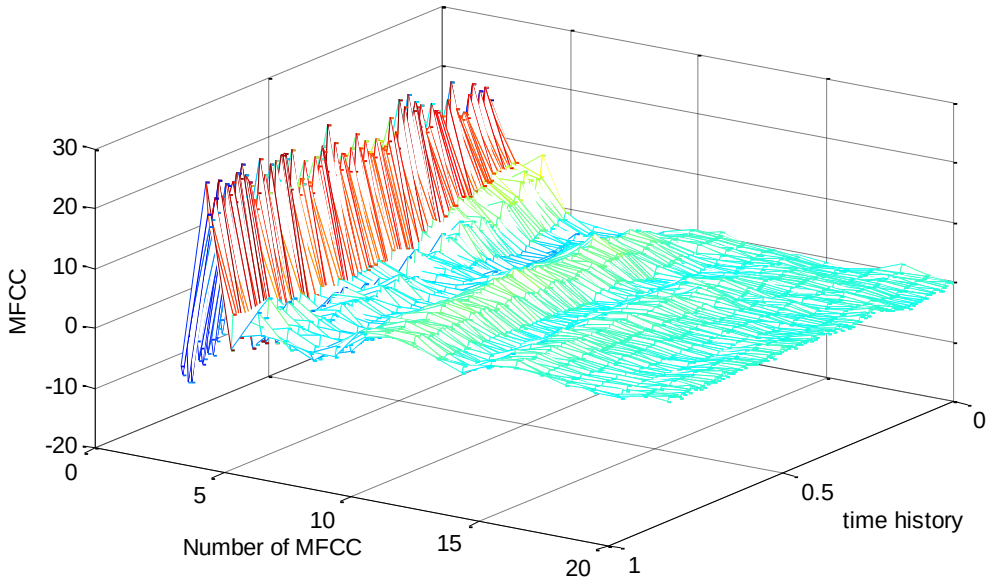


图 4.6“话”的 MFCC 参数

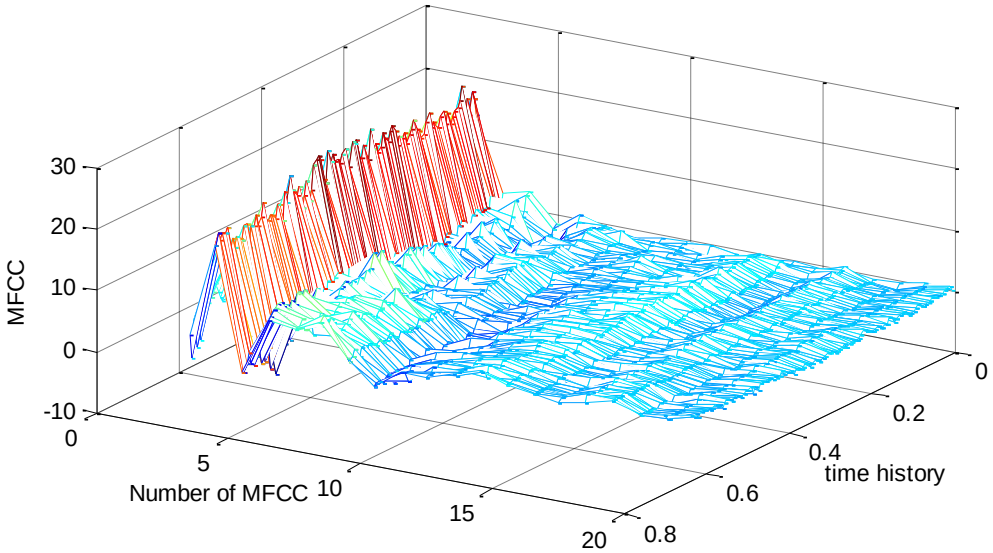


图 4.7“费”的 MFCC 参数

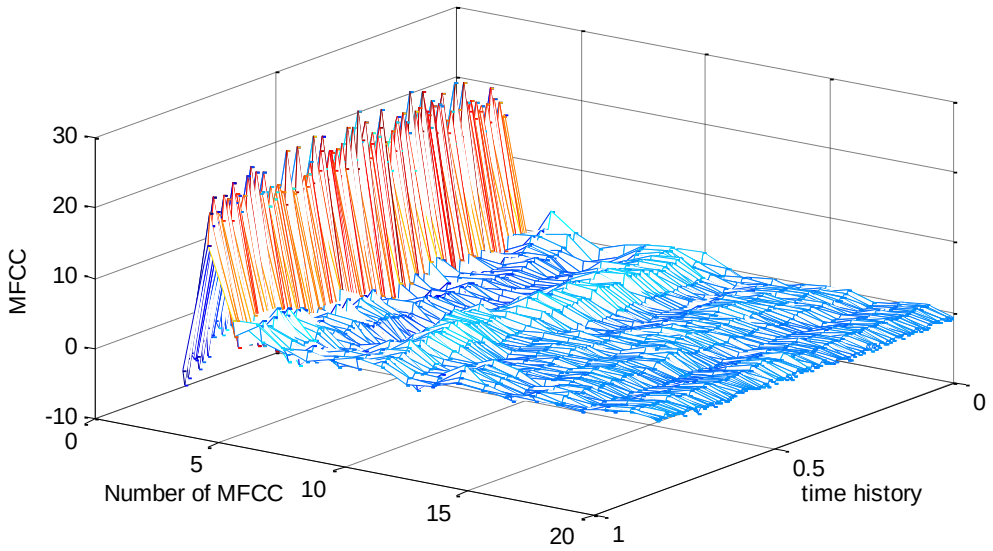


图 4.8“查”的 MFCC 参数

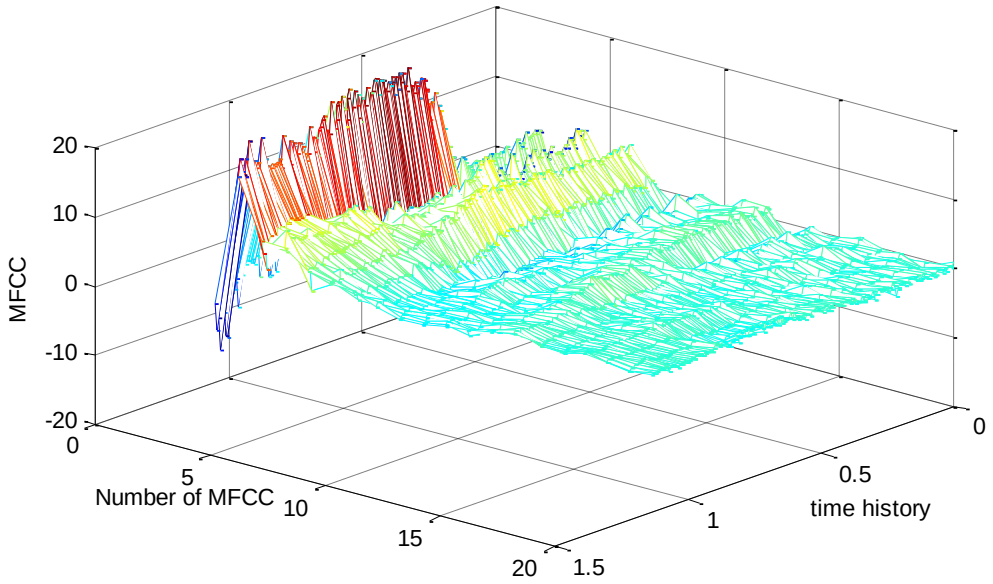


图 4.9“询”的 MFCC 参数

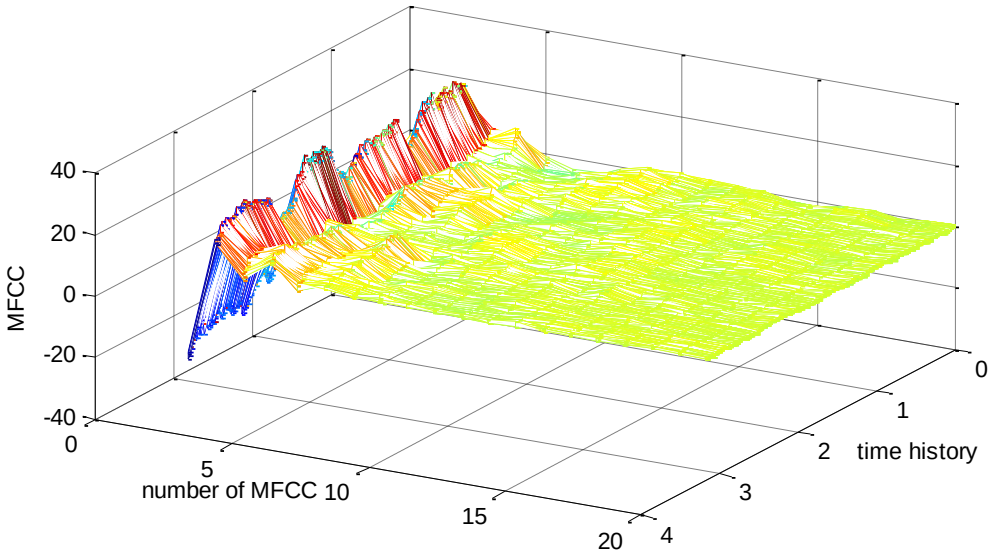


图 4.10“话费查询”的 MFCC 参数

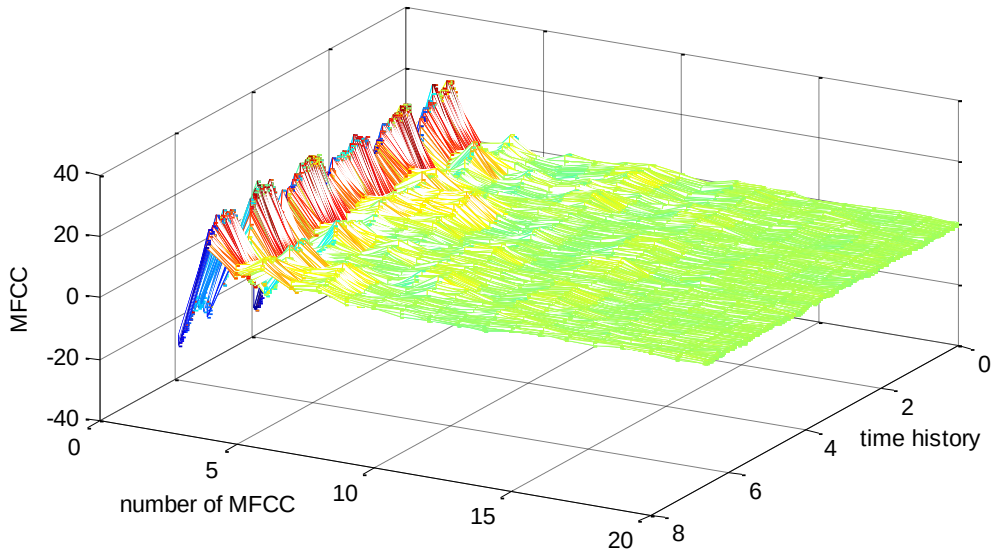


图 4.11 “业务办理”的 MFCC 参数

4.2 BP 神经网络识别

4.2.1 BP 模型建立

基于 BP 神经网络语言特征信号分类算法建模包括 BP 神经网络构造，BP 网络训练和 BP 神经网络分类，具体算法见 3.2 节的推导过程。

4.2.2 MATLAB 实现

根据本文 3.3 节 BP 神经网络理论，在 MATLAB 软件中编程实现基于 BP 神经网络的语音识别算法。

1. 归一化方法及 MATLAB 函数

数据归一化方法在本案例中选取最大最小法。函数形式如下：

$$x_k = (x_k - x_{\min}) / (x_{\max} - x_{\min})$$

式中， $x_{\max}$ 为序列中的最小数， $x_{\min}$ 为序列中的最大数。

MATLAB 实现采用 MATLAB 自带函数 mapminmax，具体程序见附件。

2. 数据选择及相关处理

不同的语音的 MFCC 分别用 1，2，3 等数字标识，一个语音单元的数据为 21 维，第 1 维为类别标识(作为网络的输出)，后 20 维为语音特征参数即 MFCC（作为网络的输入）。训练数据与测试数据的具体选择见下文具体的实验算法实现。根据语音类别标识设定每组语言信号的期望输出值，详见下文实验算法实现中的参数设置。

3. BP 神经网络训练与测试

下文将针对不同的实验进行网络训练和测试，为方便与改进后的 BP 网络进行比较，本文两种算法训练参数设置为:最大迭代次数 ePochs=100，目标误差 goal=0.0001，最小下降梯度 mi_grad=le-005，最大失败次数 maxfai1=5，最后可

得到相应的识别测试精度。

4.2.3 实验 a 算法实现

BP 神经网络构建根据系统输入输出数据特点确定 BP 神经网络的结构，前文语音处理得到特征参数 MFCC（网络输入）有 20 维，待分类的语音信号共有 4 类，所以 BP 神经网络的结构为 20-40-4，即输入层有 20 个节点，隐含层有 40 个节点(隐含层的确定没有唯一的标准，这里采用尝试法找到较合适的层数为 40)，输出层有 4 个节点。每一个语音类型从语音波形中提取的 MFCC 有 87 组，取前 4×87 组(第一批录音数据)作为训练数据，后 1×87 组(第二批录音)作为测试数据。根据本实验 4 个类别，每个类别(1,2,3,4)的期望输出值分别为[1,0,0,0]、[0,1,0,0], [0,0,1,0], [0,0,0,1].用训练好的神经网络对测试语音进行识别测试。

Table4.1 BP 神经网络识别率(平均识别率 0.92115)

语音信号分类	“话”—1	“费”—2	“查”—3	“询”—4
识别正确率	0.7705	1	0.9298	0.9843

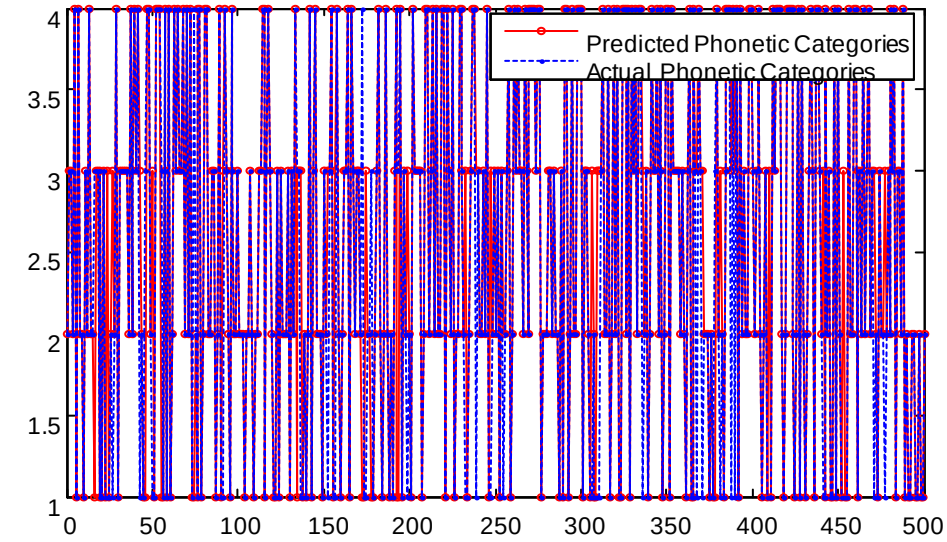


图 4.12　实验 a 检测的语音类别与实际语音匹配(BP 模型)

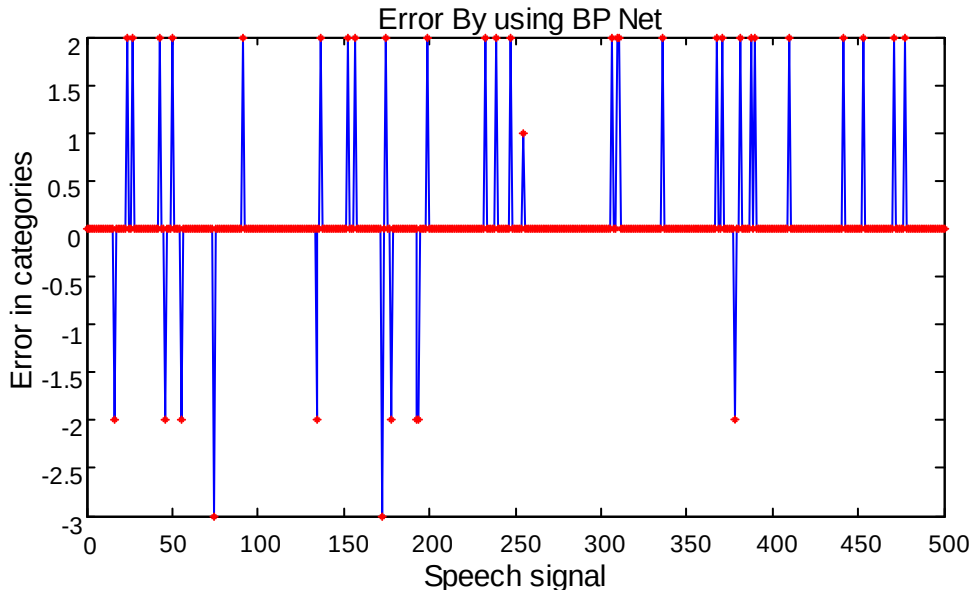


图 4.13　实验 a 识别误差(BP 模型)

4.2.4 实验 b 算法实现

BP 神经网络构建根据系统输入输出数据特点确定 BP 神经网络的结构，由于语音特征输入信号有 20 维，待分类的语音信号共有 3 类，所以 BP 神经网络的结构为 20-40-3，即输入层有 20 个节点，隐含层有 40 个节点(隐含层的确定没有唯一的标准，这里采用尝试法找到较合适的层数为 40)，输出层有 3 个节点。每一个语音类型从语音波形中提取的 MFCC 有 507 组，取前 3×507 组(第一批录音)作为训练数据，后 1×507(第二批录音)组作为测试数据。根据本实验 3 个类别，每个类别(1,2,3)的期望输出值分别为[1,0,0]、[0,1,0]，[0,0,1]。用训练好的神经网络对测试语音进行识别。

Table 4.2　BP 神经网络识别率(平均识别率 0.853867)

语音信号分类	“话费查询”—1	“套餐余量查询”—2	“业务办理”—3
识别正确率	0.9592	0.8919	0.7105

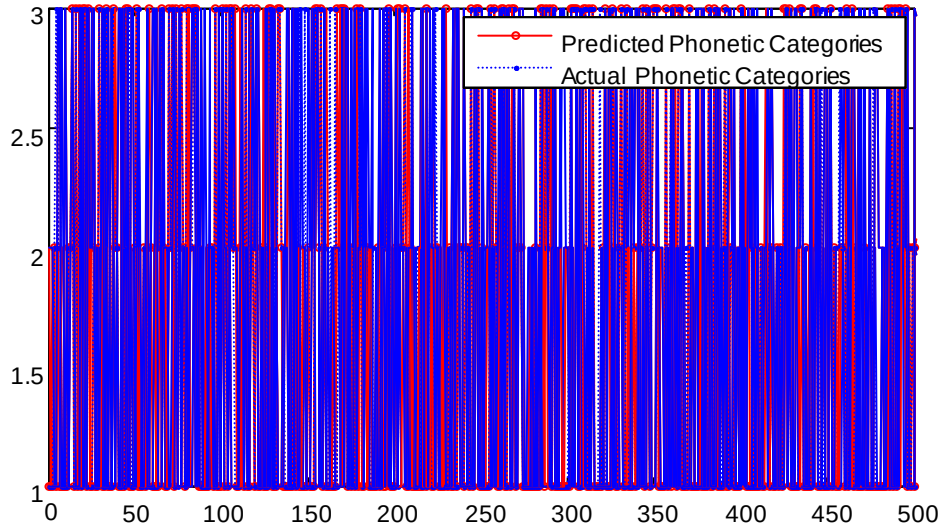


图 4.14 实验 b 检测的语音类别与实际语音匹配(BP 模型)

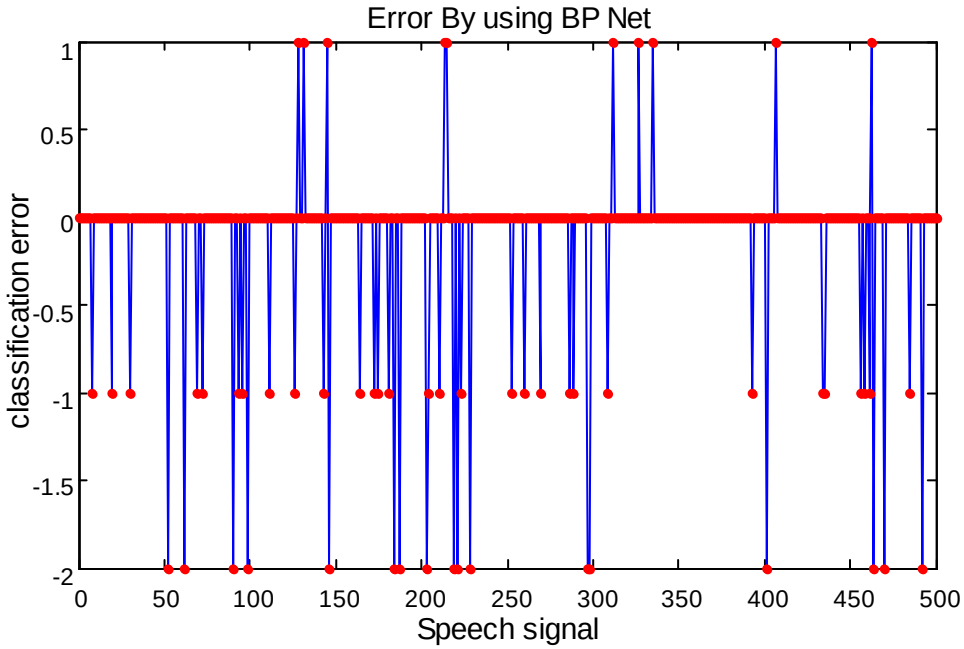


图 4.15　实验 b 识别误差(BP 模型)

4.3 基于遗传算法优化的 BP 神经网络识别

遗传算法优化的具体理论在 3.4 节模型中已论述，这里不再赘述。遗传算法优化 BP 神经网络分为 BP 神经网络结构的确定，遗传算法优化和 BP 神经网络预测 3 个部分。遗传算法优化中，每个种群中的个体都包含了一个网络所有的权值和阀值，个体通过适应度函数计算个体适应度值，遗传算法通过选择，交叉和变异操作找到最优适应度值对应个体。

为此，在 3.3 节中 BP 网络语音识别的步骤和结论的基础上，引入遗传算法对网络的初始连接权值进行优化处理，利用得到的遗传算法优化的神经网络进行语音识别。

改进后的神经网络语音识别的总体算法步骤见如下流程图。

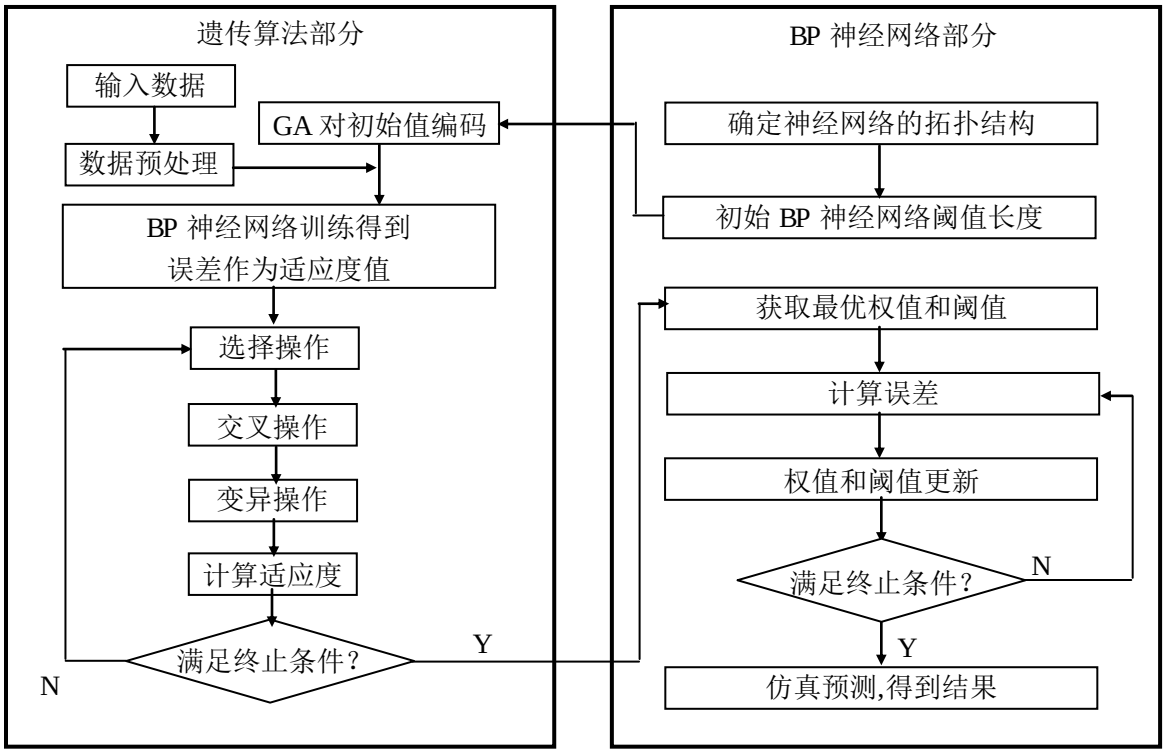


图 4.16　改进的 BP 神经网络的语音识别流程图

4.3.1 遗传算法优化的 MATLAB 实现

根据上文编写的算法步骤，利用 MATLAB 强大的函数库及工具箱编写相应的程序，并就实验 a 和实验 b 进行仿真，具体的参数设置见下节实验改进算法的实现。

4.3.2 实验 a 改进算法实现

改进后的 BP 神经网络构建同样根据系统输入输出数据特点确定 BP 神经网络的结构，前文语音处理得到特征参数 MFCC（网络输入）有 20 维，待分类的语音信号共有 4 类，所以 BP 神经网络的结构为 20-40-4，即输入层有 20 个节点，隐含层有 40 个节点(隐含层的确定没有唯一的标准，这里采用尝试法找到较合适的层数为 40)，输出层有 4 个节点。每一个语音类型从语音波形中提取的 MFCC 有 87 组，取前 4×87 组(第一批录音数据)作为训练数据，后 1×87 组(第二批录音)作为测试数据。根据本实验 4 个类别，每个类别(1,2,3,4)的期望输出值分别为 [1,0,0,0]、[0,1,0,0], [0,0,1,0], [0,0,0,1].用训练好的改进 BP 神经网络模型对测试语

音进行识别。

Table 4.3　遗传算法改进的 BP 神经网络识别率(平均识别率 0.97657)

语音信号分类	“话”—1	“费”—2	“查”—3	“询”—4
识别正确率	0.9063	1	1	1

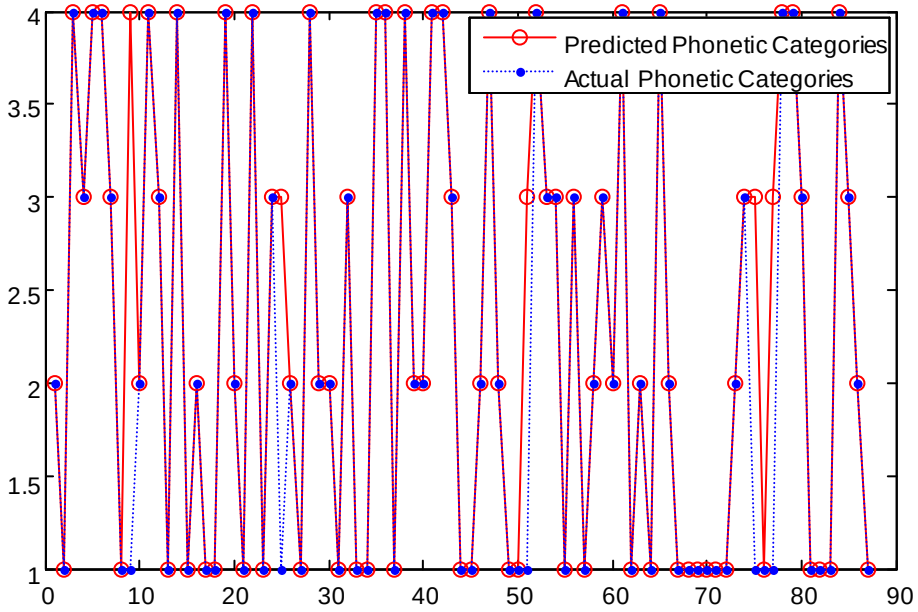


图 4.17 实验 a 检测的语音类别与实际语音匹配(GA&BP 模型)

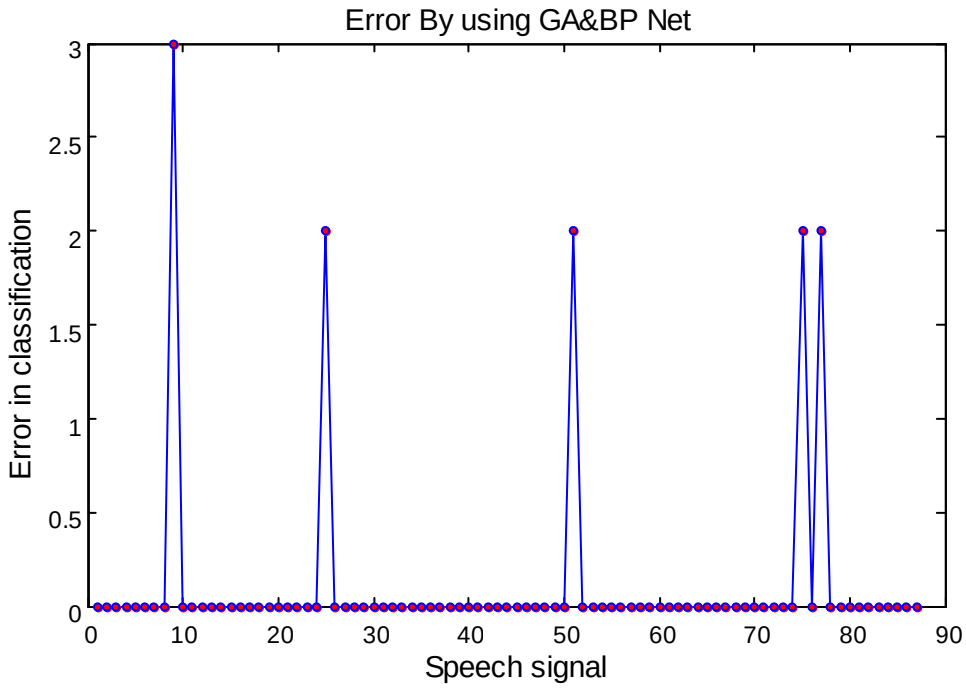


图 4.18　实验 a 识别误差(GA&BP 模型)

4.3.3 实验 b 改进算法实现

改进后的 BP 神经网络构同样条件下，根据系统输入输出数据特点确定 BP 神经网络的结构，由于语音特征输入信号有 20 维，待分类的语音信号共有 3 类，所以 BP 神经网络的结构为 20-30-3，即输入层有 20 个节点，隐含层有 30 个节点(隐含层的确定没有唯一的标准，这里采用遗传算法优化找到合适的层数)，输

出层有 3 个节点。每一个语音类型从语音波形中提取的 MFCC 有 507 组，取前 3×507 组(第一批录音)作为训练数据，后 1×507(第二批录音)组作为测试数据。根据本实验 3 个类别，每个类别(1,2,3)的期望输出值分别为[1,0,0]、[0,1,0], [0,0,1]。用训练好的神经网络对测试语音进行识别。

Table4.4 遗传算法改进的 BP 神经网络识别率(平均识别率 0.8866)

语音信号分类	“话费查询”—“1”	“套餐余量查询”—“2”	“业务办理”—“3”
识别正确率	0.902	0.8244	0.9333

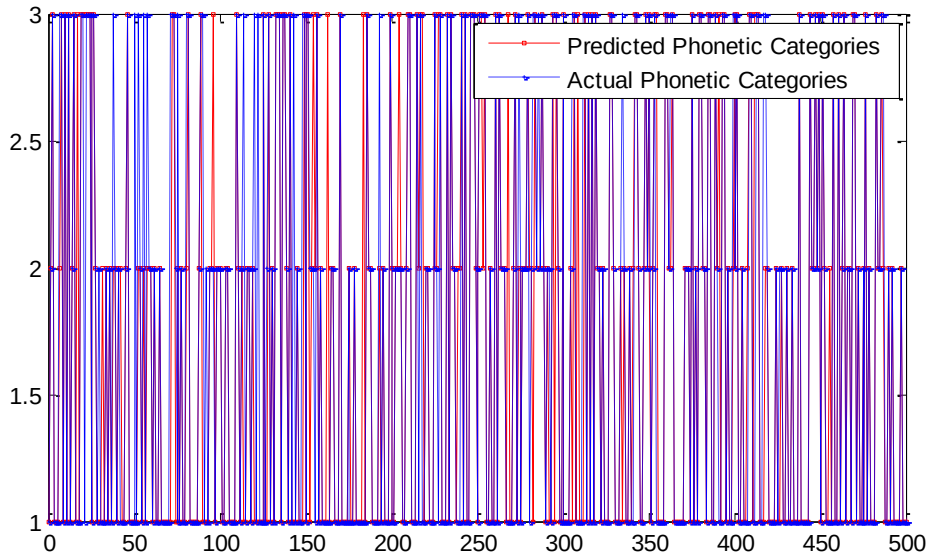


图 4.19 实验 b 检测的语音类别与实际语音匹配(GA&BP 模型)

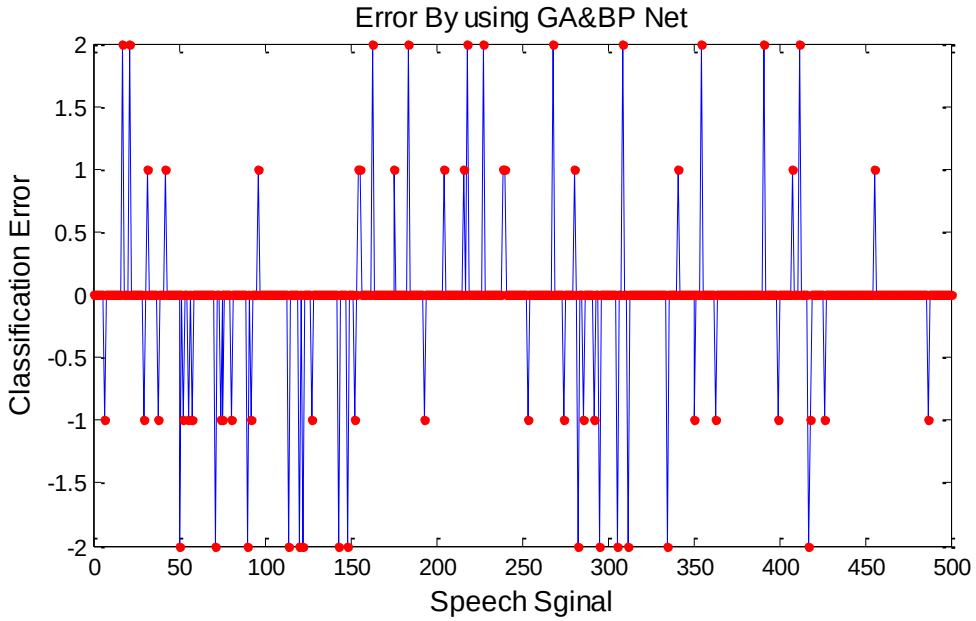


图 4.20 实验 b 识别误差(GA&BP 模型)

4.4 模型改进前后对比

从模型改进前后及不同实验条件下的的识别情况可以得出以下两个方面的结论：

1)实验a中的模型识别率要比实验b中模型的识别率高，即说明单个字的识别精度高与连续字符的识别精度。其中原因可以总结为：字数增多识别单元特征参数MFCC差别将弱化；而发音者对语速的控制和连贯性也难以保持一致；同一识别单元同一人发音差异更大。

2)两种实验情况下，模型改进后的识别率明显高于改进前的识别率，这说明基于遗传改进后的BP神经网络的算法比现有的基本BP算法取得了更好的效果。基于遗传神经网络的算法避免了局部极小，从而快速地找到最优解，降低了学习时间，提高了识别率。

为此，可以论证，利用遗传算法强大的全局搜索能力，对BP神经网络的隐层节点数、初始权值(包括阈值)、伸缩和平移因子以及学习率和动量因子进行全面优化，在解空间中定位出较好的搜索空间，然后用BP神经网络算法在这些小的解空间中搜索出最优解。形成了一种遗传BP神经网络学习算法。实验算例表明了该算法的收敛速度和精度优于BP神经网络算法，它是可行的、有效的。这是由于遗传算法能以较快的速度减少搜索空间范围。而且不易陷入局部极小点，有很强的全局搜索能力，而BP神经网络则具有局部搜索效率高的特点，将两者结合起来便可提高算法的整体性能。

5.模型总结与改进

5.1 模型总结

本文在参阅大量文献[18][22][31][32]的基础上，发现传统的语音识别系统对识别对象的内在机理要求必须清晰明确且容易控制，而在实际应用中，语音信号是受说话人物理特性、环境特性等因素影响的复杂集合，本文直接略过对人类的听觉模型的考虑，建立通用的语音辨识模型。我们知道，传统的语音识别算法的运算量很大，相应的语音识别系统实现起来较为复杂。而具有很强的黑盒特性的 BP 神经网络则很适合应用于语音识别。同时，该识别算法也具有容易实现的特点。BP(误差反向传播网络)算法是目前网络训练最常用的学习方法之一，但是它存在着两个突出的弱点，即收敛速度慢和可能收敛到局部极小点。为了克服这些弱点，引入遗传算法 (GeneticAlgorithms)来搜索神经网络连接权值的全局最优解，并将得到的网络应用到语音识别系统中，以使识别效果相较于单纯的 BP 网络识别系统有一定的提高。

本文完成的主要任务是针对语音识别的特点，结合人工智能领域中两种较为有效的方法—BP 网络和 GA 算法，构建了一种新的遗传算法优化的神经网络的语音识别算法。首先对语音识别的基本原理和流程做了简要介绍，概括了语音识别过程中预处理和特征参数提取的方法，详细描述了人工神经网络尤其是 BP 网络语音识别系统的原理与算法，并总结归纳了 BP 网络学习算法存在的一些问题。继而根据这些问题分析了引入遗传算法产生新型的改进 BP 神经网路的必要性和可行性。最后结合遗传算法和 BP 网络两者的优点，提出了改进的 BP 神经网络的语音识别算法，即先利用遗传算法搜索出近似最优解，然后把它作为 BP 学习算法的初始值，再混合训练神经网络进行语音识别。还给出了该算法的实现方案，并根据这个方案实现了两种语音实验的辨识任务，并和传统的 BP 神经网络识别模型进行对比。通过试验表明：改进的语音识别系统具有较普通的 BP 网络语音识别系统更好的识别性能和效率。主要研究成果和创新点如下：

l) 在参阅大量的文献的基础上，对语音识别技术的当前研究进展做了概述，在此基础上，提出一种新的改进算法即基于遗传算法优化的 BP 神经网络语音识别模型。

2) 在分析语音识别技术的各环节方面，也是建立在综述当前研究的基础上，选择一种更具通用性和自适应性的方法。在语音信号。

3) 自适应特征重组方法，能够有效地将那些对建模作用很小的冗余的特征分量抛弃，从而可以节约大量的存储开销，同时也可加快系统的识别速度，更加适用于实时性要求高但是资源有限的应用背景。同时，利用 ANN 学习能力的自适应性来重组特征，这要比通过人为的方法进行特征重组更有利于提高系统的性能。

4) 基于提出的多重演化神经网络模型实现了一个语音识别原型系统，为今后将仿生学理论运用于语音识别领域的同类研究提供理论与实践的基础。

5.2 模型尚待改进的地方

5.2.1 模型忽略的因素

由于比赛时间有限，而本队成员均为非语音识别专业的学生，语音识别技术本身是一门博大精深的学问，所以我们很难在短时间内将模型继续可能存在的问题都解决，故在此提出未解决或在模型中简化的方面，供大家参考。本文的模型是建立在一定的假设中建立的，而在实际环境中，很多因素是无法直接忽略的，以下几个方面是尚待考虑的因素[4][9][13] [18][23][24]：

l) 不同人发同一个音的语音信号差别很大，而在本文中，为方便起见，直接设定特定人发音作为语音库的神经网络训练和测试，难免存在误差。

为了解决不同口音的语音识别，人们采集了不同口音的普通话的语音库，如广东口音、上海口音、福建口音、四川口音等；也有的以说标准普通话训练处的非特定人语音识别系统基础上作口音修正的。实际上即使是标准普通话发音差别也很大，例如新闻联播播音员的语音信号互相之间也有明显的差别。即使是同一个人在不同时间说同一句话也是有较大差别的。

2) 话筒和语音通道对语音信号的影响较大，话筒的型号、位置和方向对语音信号都有影响。用同一话筒在计算机上演示的语音识别系统性能很好，改到不同的线路中使用，性能可能就明显下降。

3) 连续语音的发音随语境而变化，将连续语音切成单个音节时，与单个音节发音相比，语音发生很大变化，而且随上下文不同而变化。因此根据不同上下文用不同识别基元，可以提高识别率。从而出现连续语音识别、词识别和单音节识别。连续语音识别，识别基元可以是音节、声韵母、音素等。词识别识别基元可以是词、音节、声韵母或音素，以词为识别基元识别率高，以音节、声韵母或音素为识别基元组词灵活性大。

4) 环境噪声使语音信号产生畸变

平稳噪声相对比较好处理，因为可以预先测量，比较容易去除。非平稳噪声比较难办，特别是噪声与语音强度可比时，识别率大幅度下降。人具有将注意力集中在要听的声音上，计算机就难于做到这一点。

5.2.2 遗传算法优化的 BP 神经网络识别模型有待继续深化

1) 利用闭值和初始连接权的进化虽然能克服 BP 网络的一些不足，但是仍然不够完善，如网络需要大量的训练样本、较大的时间开销;网络较差的外延性能等等；

2) 本文只是针对孤立词语音识别进行了一些基础性实验，得到一些了结论，但这些结论在大词汇量连续语音识别方面尚不具有说服力，需要更深入地研究和探讨如何将改进后的 BP 神经网络进一步应用到更广泛的语音识别领域，使之具有更强的可实现性和更高的实际应用价值这一命题；

3) 如何利用遗传算法来优化神经网络的结构及其学习规则并将其应用于模式识别领域，目前尚未有成熟的理论与方法，因此，这个课题还值得进一步研究。

6.参考文献

[1] 房安栋，刘军万，杂背景下声纹识别系统的研究方法综述，电子世界，2013.

[2] Kajarekar， Phone-based cepstral polynomial SVM system for speaker recognition， Proceedings of Interspeech，2008.

[3] 李曜，刘加，段长在汉语语音识别系统后处理阶段的应用，清华大学学报(自然科学版)，49：1270-1273，2009.

[4] 田莎莎，唐菀，佘纬，改进MFCC参数在非特定人语音识别中的研究，科技通报，3(29)：140-142，2013.

[5] Robert Eklund，Anders Lindström，Xenophones：An investigation of phone set expansion in Swedish and implications for speech recognition and speech synthesis，Speech Communication， 35(2)：81-102， 2001.

[6] 徐子豪，张腾飞，基于语音识别和无线传感网络的智能家居系统设计，计算机测量与控制，20(1)：181-182，2012.

[7] Tony Ayres，Brain Nolan，Voice activated command and control with speech recognition over WiFi， Science of Computer HYPERLINK “http：//www.sciencedirect.com/science/journal/01676423”Programming ， 59(1)：109-126，2006

[8] 苏征远，易燕，解永刚，戴祖诚，基于ARM的语音识别系统设计，价值工程，4：126，2012.

[9] 郭超，张雪英，刘晓峰，支持向量机在低信噪比语音识别中的应用，计算机工程与应用，49(5)：213-215，2013.

[10] Petri Salmela，Mikko Lehtokangas，Jukka Saarinen，Neural network based digit recognition system for voice dialling in noisy environments，Information Sciences，121(3)：171-199， 1999.

[11] 吴炜烨，基于神经网络语音识别算法的研究，学位论文，中南大学，2009.

[12] Mark Howell，Steve Love，Mark Turner，Visualisation improves the usability of voice-operated mobile phone services，International Journal of Human-Computer Studies，64(8)：754-769，2006.

[13] 宋清昆，高健凯，丁然，王宏伟，基于改进遗传算法的小波神经网络控制器的研究，自动化技术与应用，2(31)：1-5，2012.

[14] 宋亚男，林锡海，徐荣华，宋子寅，机器人语音识别实验设计与实现，实验技术与管理，2(30)：36-38，2013

[15] 余华，杨露菁，李启元，基于径向基神经网络的语音识别技术，控制工程，16：90-93，2009.

[16] Roland Linder，Andreas E. Albers，Markus Hess，Siegfried J. Pöppl， Rainer Schönweiler， Artificial Neural Network-based Classification to Screen for Dysphonia Using Psychoacoustic Scaling of Acoustic Voice Features ，Journal of Voice， 22(2)： 155-163， 2008.

[17] 周琍，BP网络在语音识别中的应用，科技信息，1：101-102，2013.

[18] 刘纪平，多重演化神经网络在语音识别中的应用，学位论文，武汉大学，2011.

[19] Patricia Melin，Daniela Sánchez，Oscar Castillo ，GeneticHYPERLINK “http：//www.sciencedirect.com/science/article/pii/S0020025512001430” HYPERLINK

"http://www.sciencedirect.com/science/article/pii/S0020025512001430"optimization of modular neural networks with fuzzy response integration for human recognition，Information Sciences， 197：1-19，2012.
[20] 姜奇平，Sensory语音识别技术，互联网周刊[J]，10：67-68，2012.
[21] 韩志艳，语音信号鲁棒特征提取及可视化技术研究，博士论文，东北大学，2009.
[22] 曾玉娟，神经网络在模式识别应用中的模型与算法研究，博士论文，中国科学院半导体研究所，1998.
[23] Jing Li,Thomas Fang Zheng,William Byrne,Dan Jurafsky. A Dialectal Chinese Speech Recognition Framework[J] ,2006
[24] Chao Huang,Tao Chen,Eric Chang. Accent Issues in Large Vocabulary Continuous Speech Recognition[J] ,2004.
[25] Pan S T，Wu C H，Lai CC. The application of Improved Genetic Algofitm on the Training of Neural Network for Speech Reedgnition[qI} The Second International Conference on Innovative Computing， Information and Control(ICICIC'2007). 2007：168-168
[26] 刘萍,廖广锐.高噪声背景下的语音识别系统设计[J]. 计算机与数字工程. 2009(07)
[27] 国玉晶,刘刚,刘健,郭军. 基于环境特征的语音识别置信度研究[J]. 清华大学学报(自然科学版). 2009(S1)
[28] 吕勇,吴镇扬.基于最大似然多项式回归的鲁棒语音识别[J]. 声学学报(中文版). 2010(01)
[29] 梁兴隆,张歆奕.基于能量竞争学习算法的特定人数字语音识别[J]. 电子世界. 2013(04)
[30] 胡光锐,韦晓东.基于倒谱特征的带噪语音端点检测[J]. 电子学报. 2000(10)
[31] 任杰. 语音识别技术概述[J]. 大众科技. 2010(08)
[32] 王瑕瑚等.嵌入式盲人手机语音识别与控制系统设计.控制技术.2009.17(10)
[33] 史峰等，MATLAB神经网络30个案例分析，北京，北京航空航天大学出版社，2010.4 起始页：1-4页
[34] 史峰，王辉等，MATLAB智能算法30个案例分析，北京：北京航空航天出版社，2011.7 起始页：29-36页

7.附 件

7.1 部分程序代码

7.1.1 录音及数据处理程序

```
clear all;
fs=44100;
nbits=16; %8,16 or 24
nchans=1; %1 or 2(mono or stereo)
adformat='double'; % 'double', 'single', 'int16', 'uint8', or 'int8'.
addir='myrecorder/xb/xb.wav';
rdata = audiorecorder(fs, nbits, nchans);
disp('按下任何按键开启录音');pause;
record(rdata);
disp('正在录音...');disp('按下任何按键停止录音');pause;
stop(rdata);
disp('这是您的录音');
pdata=play(rdata);
myrcd = getaudiodata(rdata,adformat);
audiowrite(addir,myrcd,fs);
tt=(0:1:length(myrcd)-1)/fs;
f1=figure(1);
clf,hold on;
plot(tt,myrcd);
grid;
hold off;
saveas(f1,[addir,'.fig']);
save addir.mat addir;

function f=enframe(x,win,inc)
    nx=length(x);
    nwin=length(win);
    if (nwin == 1)
        len = win;
    else
        len = nwin;
    end
    if (nargin < 3)
        inc = len;
    end
    nf = fix((nx-len+inc)/inc);
    f=zeros(nf,len);
    indf= inc*(0:(nf-1)).';
    inds = (1:len);
    f(:) = x(indf(:,ones(1,len))+inds(ones(nf,1),:));
    if (nwin > 1)
        w = win(:)';
        f = f .* w(ones(nf,1),:);
```

```
    end
end
clear;
close all;
% addir='myrecorder/1a.wav';
load addir.mat;

[x,fs,nbits]=wavread(addir);%首先打开经录好的信号,一段口哨声
x = x / max(abs(x));%幅度归一化到[-1,1]
%参数设置
FrameLen = 256;        %帧长
inc = 90;              %未重叠部分，这里涉及到信号分帧的问题，在后边再解释。
amp1 = 10;             %短时能量阈值
amp2 = 2;              %即设定能量的两个阈值。
zcr1 = 10;             %过零率阈值
zcr2 = 5;              %过零率的两个阈值，感觉第一个没有用到。

minsilence = 6;    %用无声的长度来判断语音是否结束
minlen  = 15;      %判断是语音的最小长度
status  = 0;       %记录语音段的状态
count   = 0;       %语音序列的长度
silence = 0;       %无声的长度

%计算过零率
tmp1  = enframe(x(1:end-1), FrameLen,inc);
tmp2  = enframe(x(2:end)  , FrameLen,inc);
signs = (tmp1.*tmp2)<0;
diffs = (tmp1 - tmp2)>0.02;
zcr   = sum(signs.*diffs,2);%虽然没搞懂上边的原理，但是可以推测存的是各桢的过零率。上边计算过零率的放到后边分析，这里只要了解通过这几句得到了信号各帧的过零率值，放到 zcr 矩阵中。

%计算短时能量
%amp = sum((abs(enframe(filter([1 -0.9375], 1, x), FrameLen, inc))).^2, 2);%不知道这里的 filter 是干啥的？但的出来的是各贞的能量了。
amp = sum((abs(enframe( x, FrameLen, inc))).^2, 2);%通过把 filter 给去掉，发现结果差不多，所以个人感觉没必要加一个滤波器，上边出现的 enframe 函数放到后边分析。这里知道是求出 x 各帧的能量值就行。

%调整能量门限
amp1 = min(amp1, max(amp)/4);
amp2 = min(amp2, max(amp)/8);%min 函数是求最小值的，没必要说了。

%开始端点检测
for n=1:length(zcr)%从这里开始才是整个程序的思路。Length（zcr）得到的
```

是整个信号的帧数。

```
    goto = 0;
    switch status
    case {0,1}                          % 0 = 静音, 1 = 可能开始
        if amp(n) > amp1                % 确信进入语音段
            x1 = max(n-count-1,1); % 记录语音段的起始点
            status  = 2;
            silence = 0;
            count   = count + 1;
        elseif amp(n) > amp2 || zcr(n) > zcr2 % 可能处于语音段
            status = 1;
            count  = count + 1;
        else                                % 静音状态
            status  = 0;
            count   = 0;
        end
    case 2                                  % 2 = 语音段
        if amp(n) > amp2 ||zcr(n) > zcr2        % 保持在语音段

            count = count + 1;
        else                                % 语音将结束
            silence = silence+1;
            if silence < minsilence % 静音还不够长，尚未结束
                count  = count + 1;
            elseif count < minlen      % 语音长度太短，认为是噪声
                status  = 0;
                silence = 0;
                count   = 0;
            else                            % 语音结束
                status  = 3;
            end
        end
    case 3
        break;
    end
end

count = count-silence/2;
x2 = x1 + count -1;                     %记录语音段结束点
f2=figure(2);
clf,hold on;
subplot(3,1,1)
plot(x)
axis([1 length(x) -1 1])%限制 x 轴与 y 轴的范围。
ylabel('Speech');
line([x1*inc x1*inc], [-1 1], 'Color', 'red');
line([x2*inc x2*inc], [-1 1], 'Color', 'red');%注意下 line 函数的用法：基于两点连成一条直线，就清楚了。
```

```
subplot(3,1,2)
plot(amp);
axis([1 length(amp) 0 max(amp)])
ylabel('Energy');
line([x1 x1], [min(amp),max(amp)], 'Color', 'red');
line([x2 x2], [min(amp),max(amp)], 'Color', 'red');

subplot(3,1,3)
plot(zcr);
axis([1 length(zcr) 0 max(zcr)])
ylabel('ZCR');
line([x1 x1], [min(zcr),max(zcr)], 'Color', 'red');
line([x2 x2], [min(zcr),max(zcr)], 'Color', 'red');
hold off;
saveas(f2,[addir,'dd.fig']);

mydat=x(x1*inc:x2*inc);
f3=figure(3);
clf,hold on;
plot((1:length(mydat))/fs,mydat);
grid;
hold off;

saveas(f3,[addir,'jd.fig']);
audiowrite([addir,'jd.wav'],mydat,fs);

nn=256;%帧数
mm=128;%帧移

mfccdat=mfcc(mydat, fs,mm,nn);
[mmm,nnn]=size(mfccdat);
ttt=(1:nnn)*nn/fs;
f4=figure(4);
clf,hold on;
mesh(ttt,(1:mmm),mfccdat);
xlabel('时间序列');
ylabel('特征参数维数');
zlabel('MFCC');
grid;
view(115,30);
hold off;
saveas(f4,[addir,'mfcc.fig']);

save([addir,'mfcc.mat'],'mfccdat');

function rmfcc = mfcc(myrcd, fs,m,n)
    len = length(myrcd);
    nbFrame = floor((len - n) / m) + 1;      %沿-∞方向取整,音框个数
```

```
M=zeros(n,nbFrame);
for ii = 1:1:n
    for jj = 1:1:nbFrame
        M(ii, jj) = myrcd(((jj - 1) * m) + ii);    %对矩阵 M 赋值
    end
end

hh = hamming(n);        %加 hamming 窗,以增加音框左端和右端的连续性
M2 = diag(hh) * M;

frame=zeros(n,nbFrame);
for ii = 1:1:nbFrame
    frame(:,ii) = fft(M2(:, ii));    %对信号进行快速傅里叶变换 FFT
end

m2 = melfb(20, n, fs); %将上述线性频谱通过 Mel 频率滤波器组得到 Mel 频谱,下面在将其转化成对数频谱
n2 = floor(n/2)+1;

zz = m2*abs(frame(1:n2,:)).^2;
rmfcc = dct(log(zz));%将上述对数频谱，经过离散余弦变换(DCT)变换到倒谱域，即可得到 Mel 倒谱系数(MFCC 参数)
end

function m = melfb(p, n, fs)
f0 = 700 / fs;
fn2 = floor(n/2);
lr = log(1 + 0.5/f0) / (p+1);

bl = n * (f0 * (exp([0 1 p p+1] * lr) - 1));
b1 = floor(bl(1)) + 1;
b2 = ceil(bl(2));
b3 = floor(bl(3));
b4 = min(fn2, ceil(bl(4))) - 1;
pf = log(1 + (b1:b4)/n/f0) / lr;
fp = floor(pf);
pm = pf - fp;

r = [fp(b2:b4) 1+fp(1:b3)];
c = [b2:b4 1:b3] + 1;
v = 2 * [1-pm(b2:b4) pm(1:b3)];
m = sparse(r, c, v, p, 1+fn2);
end
```

7.1.2 BP 神经网络部分程序

```
%% 清空环境变量
clc
clear
```

```
%%下载语音的 MFCC 数据
load a1.mat
load a2.mat
load a3.mat
load a4.mat
load ax.mat
%数据变换及处理
a1=a1';a2=a2';
a3=a3';a4=a4';
ax=ax';x=1;%表示为“话”的语音信息
%有效语音信息截取
[m1,n1]=size(a1);
[m2,n2]=size(a2);
[m3,n3]=size(a3);
[m4,n4]=size(a4);
[mx,nx]=size(ax);
mm=min([m1,m2,m3,m4,mx]);
nn=min([n1,n2,n3,n4,nx]);
%用数字标记语音类别并合并数据
aa1=[ones(m1,1),a1];
aa2=[2*ones(m2,1),a2];
aa3=[3*ones(m3,1),a3];
aa4=[4*ones(m4,1),a4];
aax=[x*ones(mx,1),ax];

dataa=[aa1(1:mm,:);aa2(1:mm,:)];
dataa=[dataa;aa3(1:mm,:)];
dataa=[dataa;aa4(1:mm,:)];
dataa=[dataa;aax(1:mm,:)];

%随机排序
k=rand(1,5*mm);
[m,n]=sort(k);

%输入输出数据
input=dataa(:,2:nn+1);
output1  =dataa(:,1);

%把输出从 1 维变成 4 维
for i=1:5*mm
    switch  output1(i)
        case 1
            output(i,:)=[1  0 0 0];
        case 2
            output(i,:)=[0  1 0 0];
        case 3
            output(i,:)=[0  0 1 0];
        case 4
            output(i,:)=[0  0 0 1];
```

```
    end
end

%随机提取 T4 个样本为训练样本，T1 个样本为预测样本
input_train=input(n(1:4*mm),:)';
output_train=output(n(1:4*mm),:)';
input_test=input(n(4*mm+1:5*mm),:)';
output_test=output(n(4*mm+1:5*mm),:)';

%输入数据归一化
[inputn,inputps]=mapminmax(input_train);

%% 网络结构初始化
innum=20;
midnum=40;
outnum=4;


%权值初始化
w1=rands(midnum,innum);
b1=rands(midnum,1);
w2=rands(midnum,outnum);
b2=rands(outnum,1);

w2_1=w2;w2_2=w2_1;
w1_1=w1;w1_2=w1_1;
b1_1=b1;b1_2=b1_1;
b2_1=b2;b2_2=b2_1;

%学习率
xite=0.1
alfa=0.01;

%% 网络训练
for ii=1:10
    E(ii)=0;
    for i=1:1:4*mm
        %% 网络预测输出
        x=inputn(:,i);
        % 隐含层输出
        for j=1:1:midnum
            I(j)=inputn(:,i)'*w1(j,:)'+b1(j);
            Iout(j)=1/(1+exp(-I(j)));
        end
        % 输出层输出
        yn=w2'*Iout'+b2;

        %% 权值阀值修正
        %计算误差
```

```
            e=output_train(:,i)-yn;
            E(ii)=E(ii)+sum(abs(e));

            %计算权值变化率
            dw2=e*Iout;
            db2=e';

            for j=1:1:midnum
                S=1/(1+exp(-I(j)));
                FI(j)=S*(1-S);
            end
            for k=1:1:innum
                for j=1:1:midnum

            dw1(k,j)=FI(j)*x(k)*(e(1)*w2(j,1)+e(2)*w2(j,2)+e(3)*w2(j,3)+e(4)*
            w2(j,4));

           db1(j)=FI(j)*(e(1)*w2(j,1)+e(2)*w2(j,2)+e(3)*w2(j,3)+e(4)*w2(j,4));
                end
            end

            w1=w1_1+xite*dw1';
            b1=b1_1+xite*db1';
            w2=w2_1+xite*dw2';
            b2=b2_1+xite*db2';
            w1_2=w1_1;w1_1=w1;
            w2_2=w2_1;w2_1=w2;
            b1_2=b1_1;b1_1=b1;
            b2_2=b2_1;b2_1=b2;
        end
end
%% 语音特征信号分类
inputn_test=mapminmax('apply',input_test,inputps);

for ii=1:1
        for i=1:mm%4*mm
            %隐含层输出
            for j=1:1:midnum
                I(j)=inputn_test(:,i)'*w1(j,:)'+b1(j);
                Iout(j)=1/(1+exp(-I(j)));
            end

            fore(:,i)=w2'*Iout'+b2;
        end
end
%% 结果分析
%根据网络输出找出数据属于哪类
for i=1:mm
        output_fore(i)=find(fore(:,i)==max(fore(:,i)));
```

```
end

%BP 网络预测误差
error=output_fore-output1(n((4*mm+1):5*mm))';

%画出预测语音种类和实际语音种类的分类图
figure(1)
plot(output_fore,'r')
hold on
plot(output1(n(4*mm+1:5*mm))','b')
legend('预测语音类别','实际语音类别')
%画出误差图
figure(2)
plot(error)
title('BP 网络分类误差','fontsize',12)
xlabel('语音信号','fontsize',12)
ylabel('分类误差','fontsize',12)

%print -dtiff -r600 1-4

k=zeros(1,4);
%找出判断错误的分类属于哪一类
for i=1:mm
    if error(i)~=0
        [b,c]=max(output_test(:,i));
        switch c
            case 1
                k(1)=k(1)+1;
            case 2
                k(2)=k(2)+1;
            case 3
                k(3)=k(3)+1;
            case 4
                k(4)=k(4)+1;
        end
    end
end

%找出每类的个体和
kk=zeros(1,4);
for i=1:mm
    [b,c]=max(output_test(:,i));
    switch c
        case 1
            kk(1)=kk(1)+1;
        case 2
            kk(2)=kk(2)+1;
        case 3
            kk(3)=kk(3)+1;
```

```
            case 4
                kk(4)=kk(4)+1;
        end
end

%正确率
rightridio=(kk-k)./kk
%下面使用遗传算法对网络进行优化
P=XX;
T=YY;
R=size(P,1);
S2=size(T,1);
S1=25;%隐含层节点数
S=R*S1+S1*S2+S1+S2;%遗传算法编码长度
aa=ones(S,1)*[-1,1];
popu=50;%种群规模
initPpp=initializega(popu,aa,'gabpEval');%初始化种群
gen=100;%遗传代数
%下面调用 gaot 工具箱，其中目标函数定义为 gabpEval
[x,endPop,bPop,trace]=ga(aa,'gabpEval',[],initPpp,[1e-6                    1
1],'maxGenTerm',gen,...
   'normGeomSelect',[0.09],['arithXover'],[2],'nonUnifMutation',[2  gen 3]);
%绘收敛曲线图
figure(1)
plot(trace(:,1),1./trace(:,3),'r-');
hold on
plot(trace(:,1),1./trace(:,2),'b-');
xlabel('Generation');
ylabel('Sum-Squared  Error');
figure(2)
plot(trace(:,1),trace(:,3),'r-');
hold on
plot(trace(:,1),trace(:,2),'b-');
xlabel('Generation');
ylabel('Fittness');
```

7.1.3 遗传算法改进的ＢＰ网络部分程序

```
%下面将初步得到的权值矩阵赋给尚未开始训练的 BP 网络
[W1,B1,W2,B2,P,T,A1,A2,SE,val]=gadecod(x);%工具箱可下载
net.LW{2,1}=W1;
net.LW{3,2}=W2;
net.b{2,1}=B1;
net.b{3,1}=B2;
XX=P;
YY=T;
%设置训练参数
net.trainParam.show=1;
net.trainParam.lr=1;
net.trainParam.epochs=100;
net.trainParam.goal=0.001;
%训练网络
net=newff(minmax(XX),[19,25,1],{'tansig','tansig','purelin'},'trainlm');
```

7.2 语音单元端点检测图

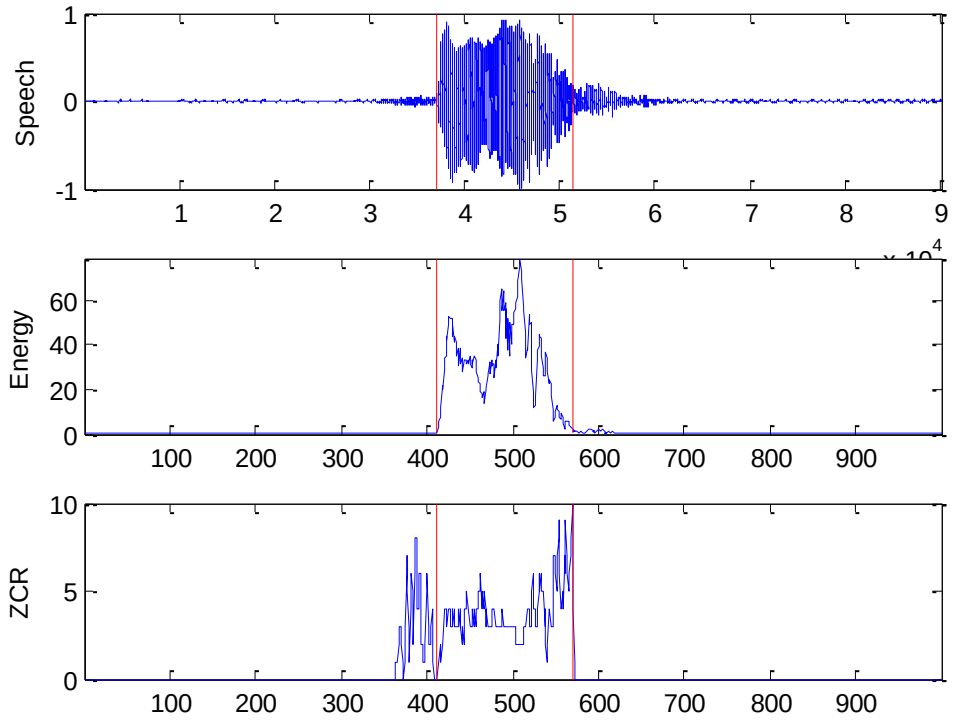


图 7.1 “费”的语音端点检测

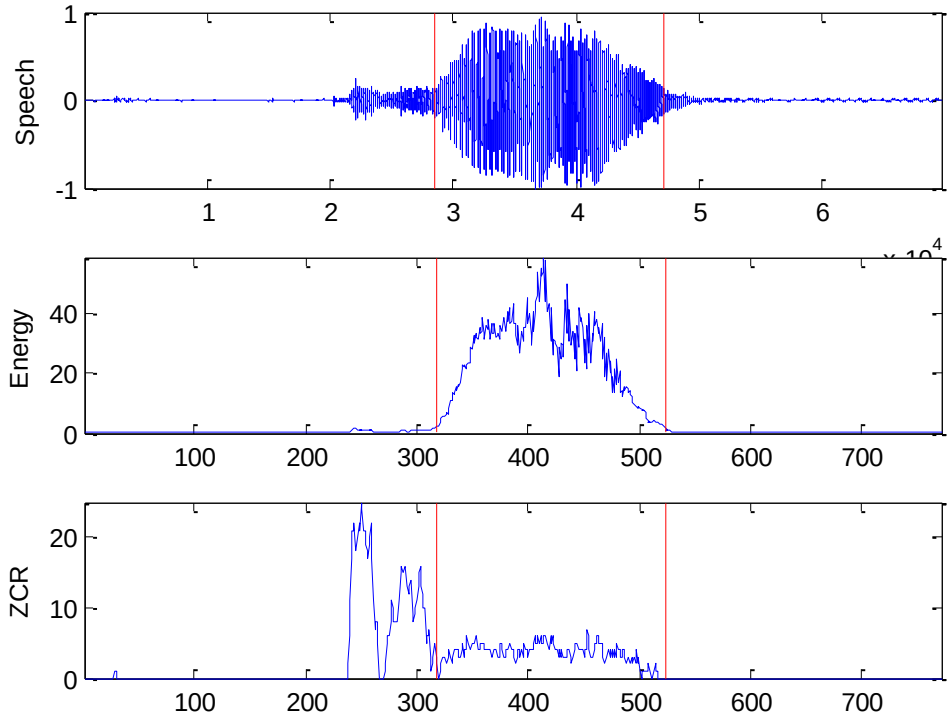


图 7.2 “查”的语音端点检测

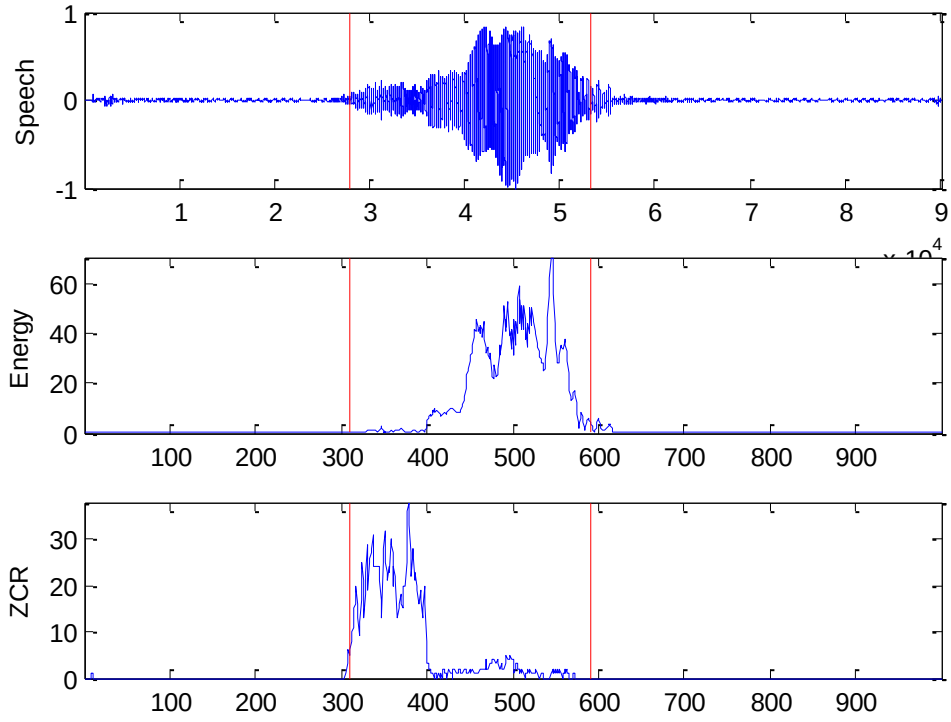


图 7.3 “询”的语音端点检测

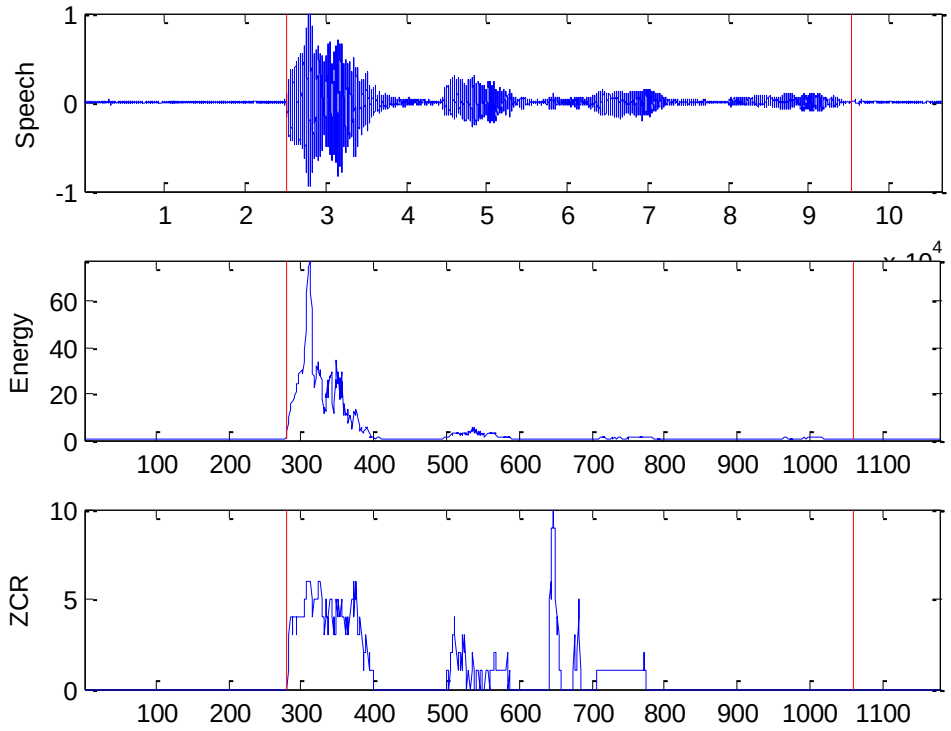


图 7.4 “话费查询”语音端点检测

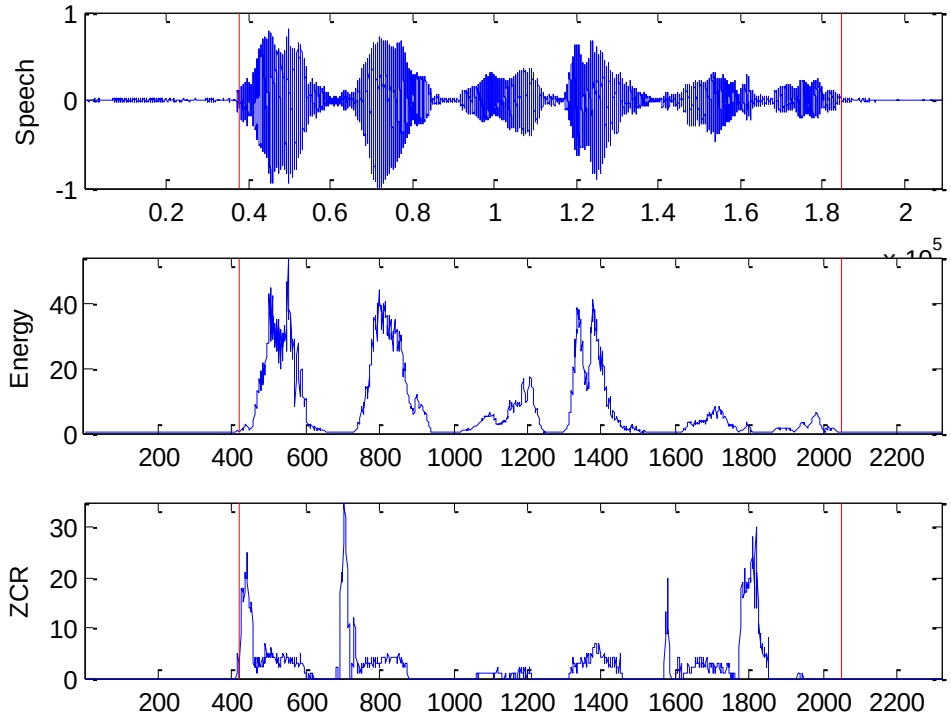


图 7.4 “套餐余量查询”语音端点检测

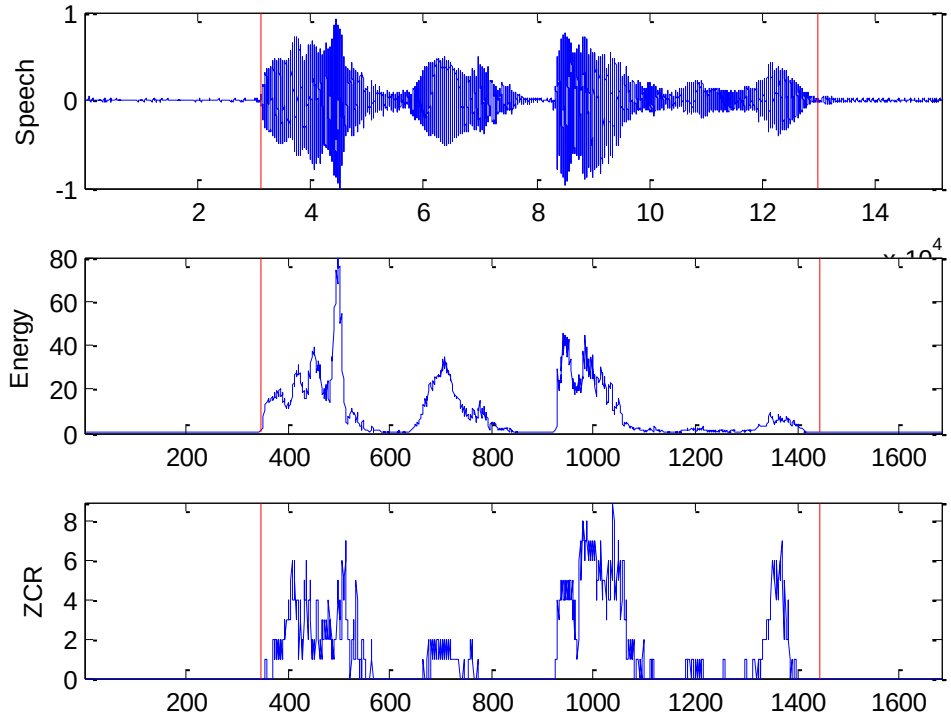


图 7.4 “业务办理”语音端点检测