

班 级：计科 1603 班
学 号：2016014344



北京化工大学

毕业设计(论文)

题 目 移动机器人的语音聊天功能实现

专 业 自动化

学 生 吴灏兵

指导教师 胡标

2020 年 6 月 2 日

移动机器人的语音聊天功能实现

摘要:由于当今科学技术的飞速进步,人工智能的也得到了很大的发展,语音聊天机器人早已在我们日常的衣食出行中扮演着不可或缺的角色。目前的市场上已经出现出了各种各样的用来解决不同领域的问题的智能语音聊天机器人。聊天机器人需要实现的基础功能就是要实现用户和聊天机器人之间的聊天就像人与人之间的聊天。近年来,自然语言处理技术得到了发展进步,现在的聊天机器人已经能够比较智能地“领悟”用户说的话所想要表达的含义,但要实现语音聊天机器人市场化还需要走很长一段路,语音聊天机器人显露的只是冰山一角。机器人聊天在未来有着巨大无比的潜力和广阔的市场,所以谁能先在技术上取得突破实现语音聊天机器人市场化,谁就能掌握市场。

本毕设的目的是基于树莓派做一个能和人进行简单语音聊天的机器人,该系统具有关键字唤醒功能,并且能对一些简单的问题进行语音回答。该语音聊天机器人系统大致概括为以下几个部分:第一步:用户通过需要输入设备(麦克风)录音,第二步:百度语音将用户输入的语音转换为相应的文字(语音识别),第三步:图灵机器人对已转换的文字进行相应的回复,第四步:百度语音将图灵机器人的回复转换为语音(语音合成),第五步:最后通过音响输出语音(播放),除此之外,在语音聊天的基础上添加了语音唤醒。

关键词: 聊天机器人, 树莓派

Realization of voice chat function of mobile robot

ABSTRACT: Due to the continuous development and progress of artificial intelligence, voice chat robots have become an indispensable part of our daily lives. At present, various intelligent voice chat robots for solving problems in different fields have appeared on the market. The basic function that chatbots need to implement is to implement chatting between users and chatbots just like chatting between people. Due to the continuous development and progress of natural language processing technology, current chat robots have more intelligently "understood" the meaning of what users say, but the current marketization of voice chat robots still needs to go a long way. Voice chat robots reveal only icebergs A corner. Robot chat has huge potential and a vast market in the future, so whoever can make a breakthrough in technology to realize the marketization of voice chat robots, who can master the market.

The purpose of this design is to make a simple voice chat robot with people based on the Raspberry Pi. The system has a keyword wake-up function and can answer some simple questions by voice. The chat robot system generally includes the following parts: the user records through the input device (microphone), Baidu voice converts the sound to text (speech recognition), the Turing robot responds to the text, and Baidu voice converts the text to speech (speech synthesis), And finally output voice (play) through the audio, and add voice wake-up on the basis of voice chat.

KEY WORDS: Chatbot, Raspberry Pi

目 录

前

言. (1)

第 1 章 概论.....(2)

第 1.1 节 语音聊天机器人的发展概况.....(2)

第 1.2 节 语音聊天机器人关键技术的发展概况.....(3)

第 1.3 节 语音聊天机器人操作系统简介.....(5)

第 1.4 节 语音聊天机器人简介.....(9)

第 1.5 节 主要研究内容.....(9)

第 1.6 节 本章小结.....(9)

第 2 章 语音聊天功能的实现与原理.....(10)

第 2.1 节 树莓派的基本配置.....(10)

2.1.1 烧录系统.....(10)

2.1.2 WiFi 网络配置.....(11)

2.1.3 开启 SSH 服务.....(12)

2.1.4 查看树莓派的 IP 地址.....(12)

2.1.5 使用 putty 登陆系统.....(12)

2.1.6 开启 vnc 远程.....(13)

第 2.2 节 语音聊天功能的实现.....(14)

2.2.1 测试录音.....(14)

2.2.2 下载相应的 API 和软件.....	(14)
2.2.3 测试百度语音识别的 API 是否正常使用.....	(16)
2.2.4 测试图灵机器人是否能正常使用.....	(17)
第 2.3 节 聊天功能的实现.....	(18)
2.3.1 录音.....	(19)
2.3.2 语音识别.....	(19)
2.3.3 图灵回复.....	(20)
2.3.4 语音合成.....	(21)
2.3.5 播放.....	(21)
第 2.4 节 本章小结.....	(21)
第 3 章 离线语音唤醒实现.....	(22)
第 3.1 节 离线语音唤醒.....	(22)
3.1.1 snowboy 介绍.....	(22)
3.1.2 snowboy 安装.....	(22)
3.1.3 自定义唤醒词.....	(23)
第 3.2 节 语音唤醒和语音识别结合.....	(26)
3.2.1 语音唤醒和语音识别结合的原理.....	(26)
3.2.2 语音唤醒和语音识别结合的实现.....	(27)
第 3.3 节 本章小结.....	(29)
结论.....	(30)
参考文献.....	(31)

致谢.....(33)

前 言

现在的聊天机器人种类繁多，如果从应用目的来看，大致可以分为两大类，第一类是显示目的驱动型机器人，显示目的驱动顾名思义就是聊天机器人的用途是比较明确的，一般是指在具体相对明确的目标和限定的知识范围内，并且用户输入和机器人输出都是有一定范围的，该类型的机器人适用于某一个特定的场景，比如商品善良，更加容易实现市场化。第二类隐式目的驱动型，隐式目的驱动则是指开放域问题，聊天的范围没有太多的限制，只要在合法内容范围，用户可以和机器人进行自由的对话，即聊天内容的范围没有太多的局限，这类机器人在实际的开发应用中一般多用于娱乐型机器人，商业价值比较低。语音聊天机器人从应用的表现形式来看，主要有客服，私人助理等等，其中客服是聊天机器人的主要用途。

目前已开发智能聊天机器人大多数都具有相关的信息查询服务和智能聊天的功能，围绕着这两大功能又逐渐扩展出了六大功能模块：商品买卖、考试问答、日程安排、学习咨询、工作、交友。随着这些功能不断的发展完善逐渐形成了以智能聊天机器人为主要角色的智能虚拟社区

在聊天机器人的众多发展功能中，客服是聊天机器人的一个主要发展功能。当客户打进电话之后，聊天机器人会自动分析客户的讲话内容，发现相应的问题，然后自动判断用户需要的是什么服务。在实际情况下，客服处理解决的问题很多都是一些简单重复，机械性的问题，这些问题完全可以交给机器人解决。企业可以通过使用聊天机器人来代替人，从而减少人工服务的投入，进而降低服务成本，提高公司的利润。私人助理是聊天机器人的另一大趋势，在日常生活中像查询时间、预订酒店、订购机票、拨打电话等比较简单的、不需要太多决策的事情，聊天机器人完全可以帮我们代劳，比如 iPhone 的 Siri，小米公司的小爱这些都是比较热门的私人助理。

一个比较理想的语音聊天机器人，应当让使用者与机器人之间对话像人和人之间的对话一样自然。我们日常聊天的过程具有以下两大特点，第一个特点是对于历史信息的记忆，第二个特点是在当前信息基础上的联想，聊天机器人可以在聊天的过程中及时记录和分析最近的对话信息来进行回答，并且可以根据当前对话的信息进行联想来转移话题，从而使得整个人机对话的过程更逼近人和人之间的聊天。

第 1 章 概论

第 1.1 节 语音聊天机器人的发展概况

国外发达国家对于语音聊天机器人上的研究起步比较早，并且取得了许多不错的成果，截至目前为止，国外已经建立了比较科学完善的语音聊天机器人评价机制，拥有了相对比较丰富的技术文献，有的相关公司和研究机构已经研发出了一些相对比较完备的聊天机器人系统，比如我们比较熟悉的 Siri。

和国外相对比较成熟的研究成果相比，我国对于语音聊天机器人的研究还有很长一段路需要走，这主要原因有三个：第一个原因，我国对于语音聊天机器人的研究起步相对比较晚，除此之外，中国是一个发展中国家，对于语音聊天机器人研究这方面投入的人力、物力以及重视的程度都不能和国外发达国家相提并论。因此我国对于语音聊天机器人无论是在研究水平的深度还是在已有的聊天机器人规模上都和国外存在着很大的差距。第二个原因，对中文进行自然语言处理的难度比对英文进行处理的难度要大得多。中文的自然语言处理不仅比较特殊而且更加的复杂，因为中文的自然语言处理需要使用计算机对中文进行音、形、义等各方面进行相应的处理和加工，比如对中文进行相应的词法、语法、语义分析等，这样才能将我们说的话转换为计算机能识别的信息，计算机才能像人一样理解中文的含义并且生成相应的回复信息。但是中文博大精深，即使同一个字，在不同声调的读音下，都有着不同的含义，除此之外，中文还有着各种各样的修辞，双关，夸张，比喻等等表现手法，英文没有这些复杂的表现手法，因此国外的一些相对比较成熟的技术和研究成果我们都无法直接使用，这些都大大增加了对于中文进行自然语言处理的难度。第三个原因：国内对于聊天机器人的体制制度还不够完善，关于自然语言处理技术的文献大多都是关于英文的，而关于中文的自然语言处理的文献寥寥无几。除此之外，国内还缺乏中文自然语言处理的相应的基础资源，比如关于中文自然语言处理的书籍和文献、中文知识库，语料库不够丰富，对于中文自然语言处理技术的评价机制不够科学完善等等。

随着聊天机器人在国外的迅速升温，国内各大信息技术公司也开始着手对聊天机器人进行了研究，并且也取得了一些不错的成绩。在 2012 年，京东自主研发智

能客户机器人 JIMI，能够向客户 24 小时提供所需的服务，服务的范围包括了网上购物的各个环节，比如关于产品功能、价格的介绍，产品购买过程的咨询，以及产品的售后服务等等。到了 2015 年，阿里巴巴也成功研发出了聊天机器人阿里小蜜，阿里小蜜与 JIMI 一样是一个智能客服机器人，都只能处理一些纯业务问题，比如向用户提供网上购物涉及的各类相关问题的咨询服务，当客户的咨询问题超出了业务范围时，则需要进行人工服务处理。聊天机器人在智能客服机器人方向的应用有着广阔的市场前景，首先从工作效率来看，传统的人工客服在不用休息的情况下一直拨打电话，每天最多也能打 300 多通电话，在这些电话中有相当的一部分是无法接通或者没有意向的，而智能语音机器人只要有电的情况下每天至少能拨打 1000 通电话，并且智能语音机器人能通过智能系统轻松的过滤掉那些无效的对话，不仅大大的节省了时间同时还提高了收益效果。然后从成本上来看，销售客服的人员流动性比较大，很多人很难长时间稳定工作。销售客服人员在人才市场上招聘是比较困难的，而且招聘上来人员还需要进行一段时间的相关业务和技能培训，才能正式的上岗工作。然后在实际的招聘和培训过程中，有相当一部分人员因为受不了各方面压力，或者觉得薪资太低而离职，这样就会让公司之前所做的工作付诸东流，造成许多损失。对于销售客服人员不仅需要支付一定的薪资、提供各种保险保障还需要对其进行相应的培训，而智能语音机器人就不必要担心招聘人数少、需要进行相关技能培训、各种原因离职这些问题，只需要对智能机器人进行引进，然后进行简单的设置即可直接上岗进行工作。与人工相比智能语音机器人还有一个很明显的优点，智能语音机器人不管面对什么样的客户，都能保持一致的工作热情，保证工作效率的同时还能提高客户的体验。当然智能语音聊天机器人也不是完全能取代人工客服的，智能语音机器人一般只能处理一些比较简单、具有普遍性的问题，当遇见比较特殊的问题还是需要人工客服处理。

第 1.2 节 语音聊天机器人关键技术的发展概况

一个新技术的出现往往是在多种技术的不断发展共同作用下产生的。语音聊天机器人也是如此，是众多技术不断发展的产物。在实现语音聊天机器人的众多技术中，自然语言处理技术的地位是其他技术无法撼动的。

自然语言处理，英文名：Natural Language Processing，简称 NLP，被人们称作

是人工智能浩瀚星河中最璀璨的那一颗明星。自然语言处理这个概念本身其实是很庞大的，简短的几句定义很难明白它是什么。为了方便介绍，我在这里把它分成“自然语言”和“处理”两个部分。先来看自然语言，和计算机语言有所区别，自然语言是人类在不断的发展过程中逐渐形成的一种专递信息的方式，包括口语和书面语，这些反映了我们的所想所思。平时常用的一句简单问候“你最近过得怎么样？”，以及各类文章，这些都是自然语言的表达形式。目前世界上所有的语种语言，包括英语、德语、西班牙语以及我们的汉语，这些都属于自然语言的范畴。

当我们和美国人聊天时，我们的对话过程能顺利的进行，这是因为我们听到英语后，大脑会对英语进行一系列的加工处理，然后在我们大脑里寻找和这个英语单词相匹配的中文词语，最后将这句英语全部转化为我们熟悉的中文，这样我们才能了解美国人想要表达的含义。和机器人进行聊天也是如此，需要将我们说的话转化为计算机能够识别的计算机语言，但计算机毕竟不是人，没有办法像我们一样对文字进行处理，需要有一套自己的处理方式。因此自然语言处理，简单来说就是以用户的自然语言形式作为输入，计算机在其内部通过已经定义的算法对用户的输入进行相应的加工、处理、计算等一系列操作后（模拟人类对自然语言的理解），最终对用户的问题做出相应的回答。

随着计算机处理速度不断的提高，互联网技术和人工智能技术得到了飞速发展，自然语言处理技术已经在各个领域得到了广泛的应用，比如我们平时常用的搜索引擎，新闻推荐，智能音箱等产品，都是以自然语言处理技术为核心的互联网和人工智能产品。在过去的几个世纪中，工业革命通过机械使我们的双手不断得到解放，在如今的人工智能革命中，计算机将代替人工对大规模的自然语言信息进行处理。由于语言是人类思想的证明，因此自然语言处理是人工智能的至高境界，处于人工智能这座金字塔的顶端。

此外，自然语言处理技术的研究也在日新月异变化，学者和科学家们每年投向 ACL（计算语言学年会，自然语言处理领域的顶级会议）的有关论文数爆炸式增长，为自然语言处理提供了许多理论研究，自然语言处理的应用效果和应用范围被不断的刷新，有趣的任务和更先进的算法更是层出不穷。

从 2006 年深度神经网络反向传播算法的提出开始，伴随着互联网的爆炸式发展

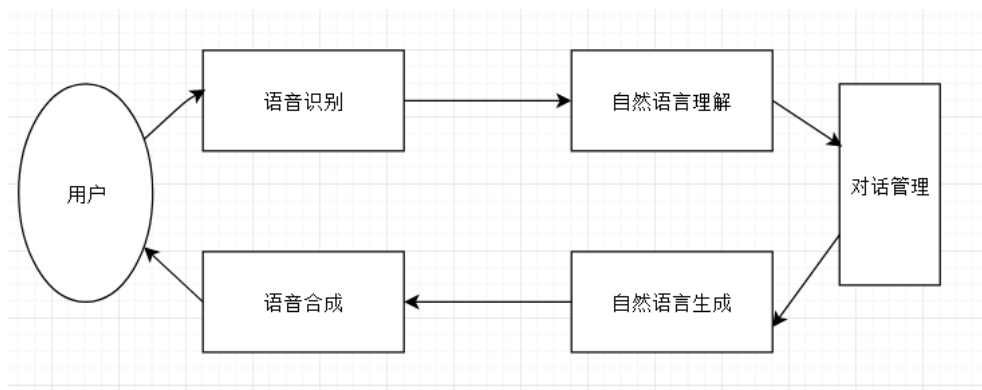
和计算机（特别是 GPU）处理能力的进一步提高，人们不再依赖语言学知识和有限的标注数据，自然语言处理领域迈入了深度学习时代。

基于互联网海量数据，并结合深度神经网络的强大拟合能力，人们可以非常轻松地应对各种自然语言处理问题。越来越多的自然语言处理技术趋于成熟并显现出了巨大的商业价值，自然语言处理的发展经历了多个历史阶段的不断演进，不同的学派之间相互补充促进，共同推动了自然语言处理技术的快速发展。自然语言处理和人工智能领域的发展进入了鼎盛时期。

第 1.3 节 聊天机器人操作系统简介

以我们比较熟悉的苹果公司的 Siri 为例，它实际上是一个非常复杂的智能系统，而我们常常使用的聊天功能只是它众多功能里的一个。Siri 是成百上千名工程师、研究人员在巨大的物力、财力的支持下，将几十个、几百个、甚至更多的工程技术相互结合的产物。本文不是探讨如何研究一个像 Siri 一样功能强大的聊天机器人系统，而是介绍简单的对话聊天机器人技术。一般情况下，聊天机器人系统都应当包含语音识别模块、自然语言处理、对话管理模块、自然语言生成模块、语音合成模块五大功能模块，除了这五个比较重要的功能模块，还需要有相应的输入设备（比如麦克风）和输出设备（音响），这样才能保证机器人能进行简单的语言聊天。

语音识别模块的主要功能是接收到用户通过输入设备（麦克风）录制的语音后，对其进行相应的处理、加工转化为相应的文本传送到自然语言处理进行下一步的操作。自然语言理解是实现聊天机器人的关键技术，它的核心功能是通过计算机将语音识别模块生成的文本进行相应的处理，然后生成相应的语义表达式，使计算机理解用户输入的话，了解用户的需求，并且将这些语义表达式送到对话管理模块。对话管理模块则主要协调聊天机器人系统的各个功能模块，避免各个功能模块间出现冲突，从而使聊天机器人的各个功能模块能够相互协作，除此之外，还需要把自然语言理解产生的语义表达式传送到自然语言生成模块进行下一步处理。自然语言生成模块主要用于根据自然语言理解产生的语义表达式生成相应的回复文本并且将该文本传输给语音合成模块，语音合成模块将回复文本最终转换成相应的语音通过输出设备输出。语音识别和语音合成在第二章会有详细介绍，这里将不做过多的介绍。



图（1）：语音聊天机器人框架

自然语言理解:

日常生活中，我们能和别人正常交流是因为彼此双方都能理解对方说的话的含义，但计算机没有我们聪明，不能直接识别我们说的话，了解我们想表达的什么。因此要实现用户和机器人之间的聊天能正常进行，这就需要机器人理解用户所说的话的含义。简单的说，自然语言理解就是为了使计算机能够理解用户的意图而对用户的输入进行各种相关处理技术。一般情况下，自然语言理解包括了用户需求判断，用户情感判断，相应的代词解释，省略恢复以及一定的拒绝回答技术。

（1）用户需求判断：计算机需要根据用户输入的语音对用户的需求进行相应的识别判断。从用户需求表现的明显程度强弱来看：第一种显示需求，用户的目的性比较强，计算机能比较明确的知道用户的需求，比如用户输入“给我定一张下个星期天早上八点去北京的飞机票”计算机就能做出准确的回复。第二种是隐式需求，用户的需求具有一定的模糊性，聊天机器人比较难判断用户的真实需求，比如用户输入“我的这件衣服已经穿了五年了”，计算机这时候就比较难判断用户的需求，因为用户有可能是想换新衣服，也有可能是想表达这件衣服质量很好。

（2）用户情感判断：察言观色是我们人类独有的，计算机无法像我们一样能够通过别人说话的语气、观察别人的脸色了解别人此时的心情是好还是坏。为了能让计算机的回复能更加的真实，因此需要对用户当前的情感进行判断。用户的情感也有显示和隐式之分。显示：比如用户输入“我多年的梦想今天终于成真了”，明确表明了用户梦想成真之后十分的开心。隐式：如果用户说“我今天考试刚刚60分”，这时候聊天机

器人就比较难判断用户是高兴还是不高兴。

(3) 代词解释和省略恢复：在我们日常的交流过程，我们经常为了表达方便，就会使用相应的代词（他，她，它）来指代聊天过程中的某个人、物品、事件，有时候甚至直接省略掉一部分句子成分，即使这样人与人之间也能进行正常的交流，但是聊天机器人没有那么的聪明，不能直接知道代词指代的是什么和句子中那些成分被省略掉了，这些代词的使用和句子成分省略使计算机更难识别用户的需求。因此聊天机器人对用户回复前需要进行相应的代词解释和省略恢复，使计算机明确代词的指代以及句子中省略的成分是什么，计算机才能更好的明白用户的需求，给出更加合乎当下聊天场景的回复。

(4) 回复确认：有时候因为各种原因导致用户输入的意图比较模糊，比如环境杂音，用户没表述清楚等，这时如果聊天机器人无法判断用户的意图，就需要聊天机器人系统具有主动询问的功能，即进一步询问用户的意图，这样计算机就能更好的明确用户的意图，这就是回复确认。

(5) 拒绝回答：对于用户的输入，聊天机器人并不是全部都需要要进行回答。聊天机器人在回答用户问题前应该对用户的问题进行一些相应的判断，如果用户的问题不在聊天机器人系统自身的回复范围内或者涉及到了一些敏感、违法的问题时，聊天机器人应该主动拒绝回答问题，并且提示用户该输入不在回复的范围之内，需要进行重新输入。

对话管理：

对话管理是聊天机器人系统的控制中枢，主要功能是对聊天机器人系统的各个功能模块进行协调，保障人机对话能够正常进行，对话管理能够根据机器人与用户对话的历史信息，对用户的问题作出更加符合场景的回答。对话管理大致可以概括为对用户的对话行为分析、对话状态判断以及对话策略学习等等。

(1) 对话行为识别：对话行为是指人机对话的过程中，计算机为了能更好的了解用户的意图，需要对用户的对话内容进行相应的识别，比如将用户的对话行为进行处理转化为计算机能够识别的信息。主要可以分为静态对话行为识别和动态行为识别两种，所谓的静态识别，即计算机会对用户常见的一些需求在系统内部提前定义好，当用户的需求恰好与已经定义好的某个需求相互匹配时，计算机就会直接对用户进行相应回

复。静态对话行为常用于某一特定领域或特定任务的对话系统，比如时间、日期查询，机票、酒店的预订等，例如：“我想预订一张下个星期五早上八点云南到北京的飞机票”，计算机会对这句话进行识别然后找到已经定义好的回复，直接对用户的需求做出反应，这就是静态对话行为。与此相对，动态对话行为计算机没有提前进行定义，而是在人机对话过程中对用户的对话行为进行动态识别。这类对话行为常见于开放域对话系统，也就是计算机可以进行比较自由的回复。例如：“今天的天气很好”，计算机对这句话的识别后可以自由的回复，就是动态对话行为。

(2) 对话状态识别：对话状态与对话的即时性有很大的联系，当用户输入一个问题后，机器人需要能很快的进行回复，如果聊天机器人回复很慢，用户今天输入的问题，机器人明天才能做出相应的回答，这样的聊天机器人肯定会被淘汰。因此，聊天机器人对用户的问题应该及时回答。

(3) 对话策略学习：对话策略学习是对话管理中非常重要的模块，为了使人 and 计算机的对话像人与人之间的对话一样自然，需要研究人与人对话过程中，对对话行为，对话状态进行分析，找到指导人机对话的方法。

自然语言生成

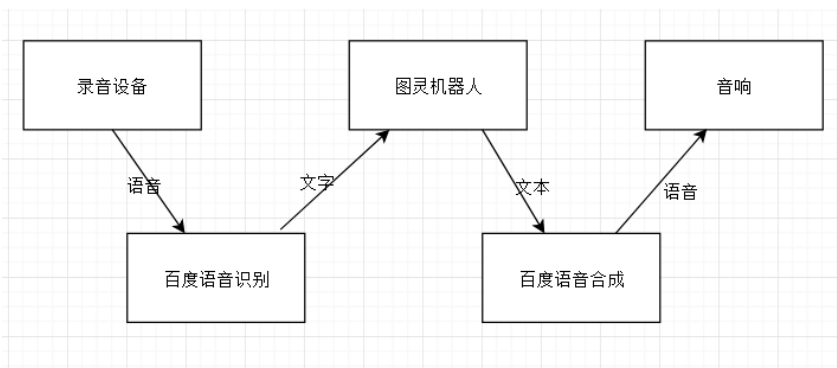
自然语言生成，英文名：Natural Language Generation，简称 NLG，和自然语言处理（NLU）一样是人工智能的重要研究方向，都是人类智慧的结晶。语言不仅具有多样的结构还有许多歧义，所以在人机对话过程中，需要计算机能够正确理解的用户意图，根据对话管理模块产生的非语言信息生成的相应的文本。自然语言生成主要分为检索式对话生成技术和生成式对话生成技术。

(1) 检索式对话生成技术：检索式是计算机提前将语料库定义好，然后根据用户的输入从语料库中找到与之相匹配的回复，采用检索式对话生成技术的回复一般比较固定、单一，采用这种技术的聊天机器人多用于一些比较简单、机械重复的问题，比如预定酒店，查询时间等等。

(2) 生成式对话生成技术：生成式不需要在语料库中进行匹配回答，而是计算机通过对相关的机器学习产生对用户的回复，因为这个回复是计算机自己创造的，因此不一定出现在已有的语料库里，基于生成式对话生成技术的聊天机器人的回复一般比较丰富且不唯一，采用这种技术的聊天机器人多用于回答开发式的问题。

第 1.4 节 聊天机器人实现简介

基于树莓派实现语音聊天功能包含以下的步骤，首先用户通过麦克风进行录音，然后系统把录制成功的语言发送给百度语音识别，它返回文字再发给图灵机器人，图灵机器人比对自己的数据库，再返回回答的文本，把文本经过语音合成后保存到本地，然后把它播放出来，整个流程是线性流程。



图（2）聊天机器人实现流程

物件清单：

- 免驱动麦克风——1
- 树莓派——1
- 音响——1

第 1.5 节 主要研究内容

本毕设的主要研究内容是在一块全新的树莓派上实现简单的语音聊天功能，接下来我将依次介绍树莓派系统的烧录，环境配置，基于树莓派如何实现简单的语音聊天功能，和用户能进行一些比较简单的语音聊天，以及自定义关键字离线唤醒。

第 1.6 节 本章小结

本章首先根据语音聊天机器人的发展现状，分析了对聊天机器人的研究意义。接着阐述了聊天机器人实现的关键技术自然语言处理的发展现状，然后介绍了在树

莓派上实现语音聊天机器人的基本框架，以及聊天机器人系统各个模块的相关功能。

第 2 章 语言聊天功能实现与原理

第 2.1 节 树莓派的基本配置

树莓派，英文名：RaspberryPi，简写 RPi，是世界上目前最流行的微型电脑主板，是开源硬件的领导产品，树莓派设计的初衷是为了普及推广学生计算机编程教育，虽然他只有一张信用卡大小，但它的功能却是十分的强大，开发人员能够在树莓派上开发编程，比如开发游戏。除此之外，树莓派价格低廉，大多数人都能够负担起，支持 linux 操作系统。最重要的是相关的开发资料完善，并且社区活跃。自从树莓派问世以来，就得到了无数计算机爱好者的追捧。

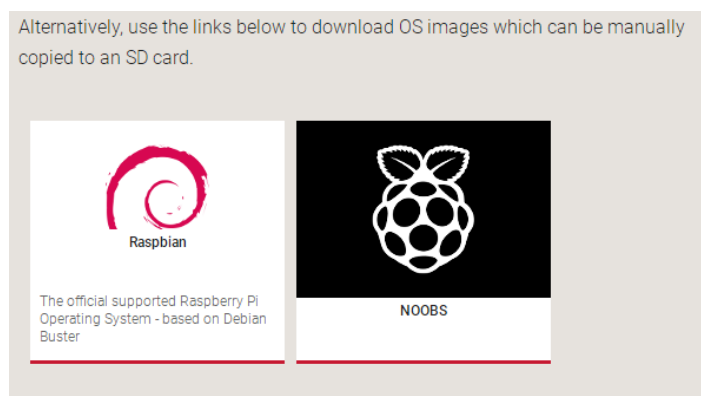


图 (3)：树莓派

2.1.1、烧录系统

树莓派需要安装相应的系统之后才能正常使用，进行编程开发，常用的系统为 Raspbian（一种基于 Debian 的一种单纯的 Arm 版的 Linux 系统）。本次实验中，树莓派安装使用的系统也是 Raspbian。

打开浏览器，输入网址 <https://www.raspberrypi.org/downloads> 或者直接输入关键字树莓派官网，进入树莓派官网后下载需要烧录的 raspbian 系统（选择右边的版本）。下载成功后的文件是压缩包文件，需要对其进行解压。



图（4）：树莓派系统

SD 卡相当于树莓派这个微型电脑的硬盘，将 raspbian 系统烧录到提前准备好的 SD 卡（本次实验使用的内存为 32G）之前，需要对 SD 卡进行一次格式化处理，保证 SD 中没有其他文件或者系统。将 SD 卡插入读卡器在电脑上使用格式化工具进行格式化，然后使用烧录软件 Win32DiskImager 将已经完成解压的系统镜像文件 raspbian 烧录 SD 卡中，烧录完成的时候可能会跳出一个对话框，询问是否再次对 SD 卡进行格式化处理，这里不能再对 SD 格式化，因为会破坏镜像系统，导致树莓派不能正常工作。

2.1.2、WiFi 网络配置

Raspbian 系统烧录成功之后，使用读卡器，在电脑上打开 SD 卡，找到文件中的 boot 分区，也就是树莓派的 /boot 目录，在这个目录下创建一个名叫 Text.txt 的文件，将下面的参考代码写入该文件并进行保存，然后将 Text.txt 文件重命名为 wpa_supplicant.conf 文件。

具体代码如下：

```
country=CN//代表是在中国
```

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
```

```
update_config=1
```

```
network={
```

```
ssid="YuYin"//树莓派可以连接的 WiFi 名称为: YuYin
```

```
psk="11111111"//WiFi: YuYin 的密码为: 11111111
```

```
key_mgmt=WPA-PSK
```

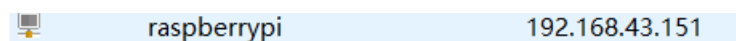
priority=1//树莓派连接 WiFi 的优先级,当有多个 WiFi 时,priority 值越大,表明该 WiFi 接入的优先级越高,但不可以小于零。

2.1.3、开启 SSH 服务:

和上一步操作类似,在 boot 分区下新建一个文件,文件命名为 ssh (注意没有任何后缀名),但这次不需要在该文件中写入任何代码,当树莓派启动后会自动检测到这个 ssh 文件然后启用 ssh 服务。

2.1.4、查看树莓派的 IP 地址:

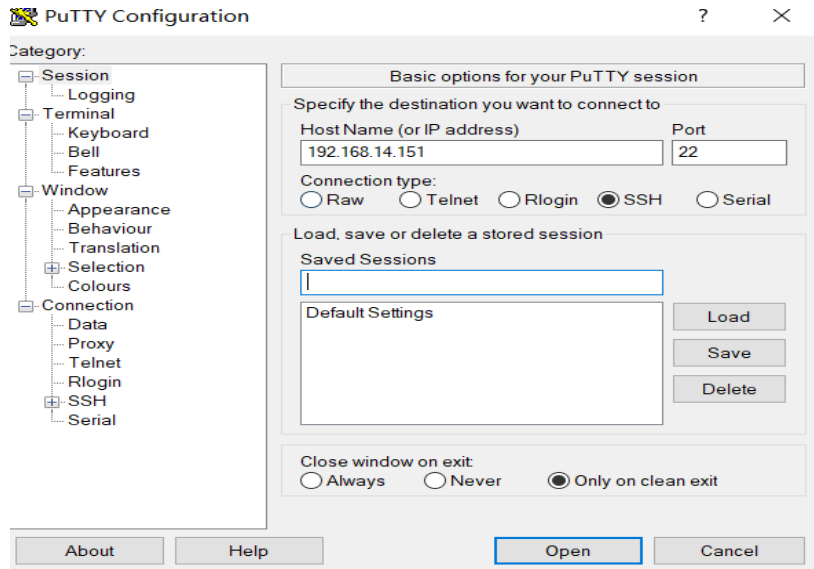
为了操作简单,可以用手机热点创建名为:YuYin,连接密码为:11111111 的 WiFi,使电脑和树莓派都连接上这个 WiFi,然后使用 IP 地址查看软件:Advanced IP Scanner 查看树莓派的 IP 地址为:192.168.43.151。



图(5): 树莓派 IP 地址

2.1.5、使用 putty 登录系统

putty 是 ssh 客户端的一种工具



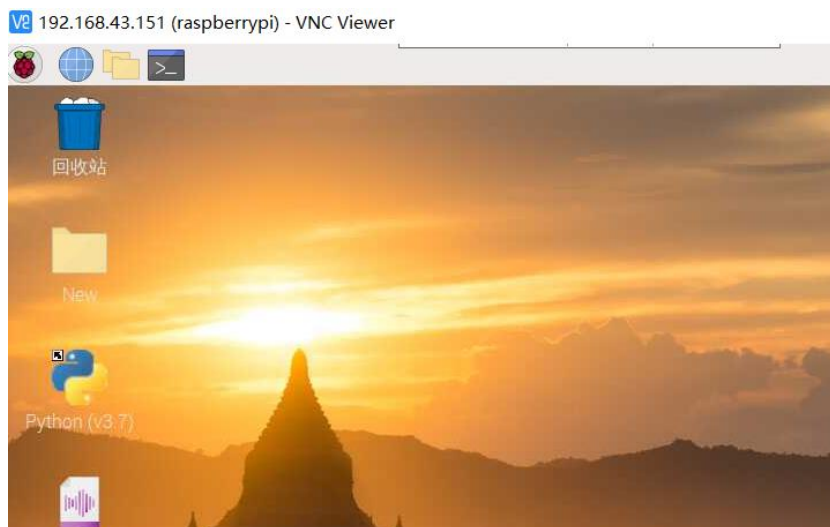
图（6）：putty 登陆系统

2.1.6、开启 vnc 远程

raspbian 已经自帶了 realvnc 远程服务，只需要启用 raspi-config 的 vnc 服务就可以启用 vnc 服务。具体步骤如下：

- (1) : `sudo raspi-config`
- (2) Interfacing Options --> VNC --> Yes
- (3) 下载 VNC 的客户端，打开运行 VNC，输入树莓派的 IP 地址 192.168.43.151

最后登陆成功的界面如下图所示：



图（7）：树莓派界面

第 2.2 节 语音聊天功能的实现

聊天功能的实现可以概括为：通过麦克风录音之后将语音发送给百度语音识别，百度语音识别将音频转化为相应的文本再发给图灵机器人，图灵机器人比对自己的数据库，再返回匹配的回复文本，然后该文本经过语音合成后保存到本地，最后用音响把它播放出来，具体的开发流程如下：

2.2.1、测试录音

可以采用树莓派自带的 `arecord` 实现录音，16000 的采样频率，保存为 `Test.wav`

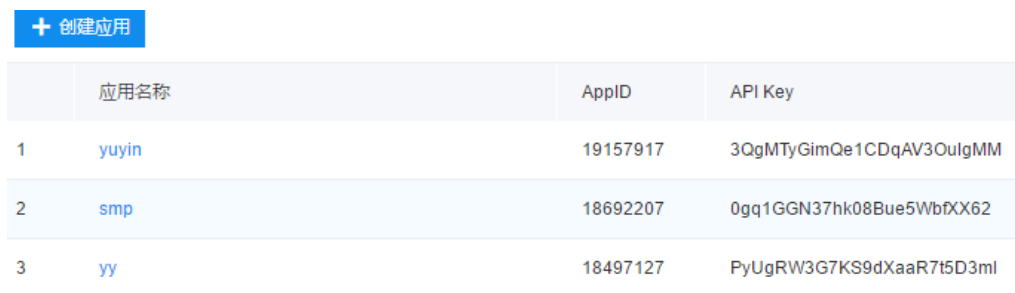
```
arecord -D "plughw:1" -f S16_LE -r 16000 Test.wav
```

其中，`-D` 参数后接麦克风设备，当树莓派连接了麦克风后，树莓派上有 2 个设备：内部设备和外部麦克风设备，`plughw:1` 表示使用外部设备。`-f` 表示录制语音的格式，`-r` 表示语音的采样频率。由于后面提到的百度语音识别对音频文件格式是有一定要求的，所以需要录制成符合要求的格式。另外，因为在这里没有限制录制的时间，所以系统会一直录制下去，需要我们手动按下 `ctrl-c` 结束录音。录制后的音频文件保存为 `Test.wav`，点击打开 `Test.wav`，如果一切正常的话，音响会播放刚刚录制的内容。

2.2.2、下载相应的 API 和软件

下载相应的 API 和软件：

- (1)：导入百度语音 API 包：`pip install baidu_aip`。
- (2)：安装播放器：`sudo apt-get -y install mpg321`。
- (3)：注册百度语音识别账号 (<https://ai.baidu.com/>)，并获取开发所需的 API KEY、APIID。



The screenshot shows a table of applications created on the Baidu AI Cloud platform. At the top left, there is a blue button with a plus sign and the text '创建应用' (Create Application). The table has four columns: an index, application name, AppID, and API Key.

	应用名称	AppID	API Key
1	yuyin	19157917	3QgMTyGimQe1CDqAV3OulgMM
2	smp	18692207	0gq1GGN37hk08Bue5WbTX62
3	yy	18497127	PyUgRW3G7KS9dXaaR7t5D3ml

图 (8)：百度语音

- (4)：注册图灵机器人账号 (<http://www.turingapi.com/>)，并获取开发所需的 API KEY。



The screenshot displays a list of Turing API robot accounts. Each row includes a robot icon, the robot name, the number of chats for the day, and the API key with a copy icon.

	图灵机器人	今日聊天数量: 0	apikey: 2811... 
	图灵机器人	今日聊天数量: 0	apikey: 8da1... 
	ROBOT	今日聊天数量: 0	apikey: e704f... 
	坤坤	今日聊天数量: 0	apikey: 7585... 

图 (9)：图灵机器人

百度语音识别和图灵机器人都需要进行实名认证后才能正常使用，在使用图灵机器人时可以自定义一些问题，创建属于自己的语料库，比如：

问：上半年感觉啥也没做就过去了

答：不要气馁，伟大的事业都是在下半年完成的，比如七一建党，八一建军，十一建

国，而上半年，不是五四青年节，就是六一儿童节，都不太成熟。

问题 (字符小于64)	答案
再歇一个试试	试试就试试
你歇啥	歇你咋地
上半年感觉啥也没做就过去了	不要气馁，伟大的事业都是在下半年完成的，比如七一建党，八一建军，十一建国，而上半...
关机	你个刁民，竟然还想事朕
北京化工大学校训	宏德博学、化育天工
介绍一下北京化工大学?	北京化工大学创办于1958年,原名北京化工学院,是新中国为“培养尖端科学技术所需要的高级...
北京化工大学在哪里?	北京化工大学地址在北京市朝阳区北三环东路15号,该校是中国教育部直属的全国重点大学、...

图（10）：部分自定义问题

2.2.3、测试百度语音识别的 API 是否正常使用

百度语音识别的基本流程就是把需要识别的语音信息、语音数据、token 等发送给百度的语音识别服务器，获取到对应的文本。使用语句 Sudo nano yuyin.py 创建一个名叫 yuyin.py 文件，输入下面代码：

```
# encoding: utf-8

from aip import AipSpeech

APP_ID="19157917" //百度语音识别 API ID

API_KEY="3QgMTyGimQe1CDqAV3OulgMM" //百度语音识别 API KEY

SECRET_KEY="7N8xMXfaHvVA11Krz65uq9Tqp7pTKq6V"

#初始化语音识别客户端

client=AipSpeech(APP_ID,API_KEY,SECRET_KEY)

//客户端合成文本生成结果，vol-指定语速

result=client.synthesis(text='曾经有一段真挚的爱情在我的面前，我没有珍惜，直到失去后，才后悔莫及!',options={'vol':5})
```

```
//生成为语音格式文件

if not isinstance(result,dict):

with open('1.mp3','wb') as f:

f.write(result)

else:

print(result)
```

在树莓派上使用语句 `python yuyin.py` 编译运行后，在树莓派本地会生成一个叫做 `1.mp3` 的文件，点击打开这个 `1.mp3` 文件，如果音响成功播放：曾经有一段真挚的爱情在我的面前，我没有珍惜，直到失去后，才后悔莫及。说明百度语音识别能正常的使用。

2. 2. 4、测试图灵机器人是否能正常使用

使用图灵机器人对我们的问题进行相应的回复，从而实现人机交流能顺畅进行。在树莓派上使用语句 `sudo nano turling.py` 创建一个 `turling.py` 文件，在 `turling.py` 文件中输入下面的代码,并保存。

```
# -*- coding: utf-8 -*-

import urllib//导入需要的模板

import json//传送图灵回复的信息

def getTurling(url):

    words = urllib.urlopen(url)

    turling = words.read()

    return turling

if __name__ == '__main__':

    key = '75858c17fe044c7fb390298e4453e881'//图灵机器人的 API KEY
```

```
api = 'http://www.tuling123.com/openapi/api?key=' + key + '&info='
```

```
while True:
```

```
    info = raw_input('我: ')//进行问题输入
```

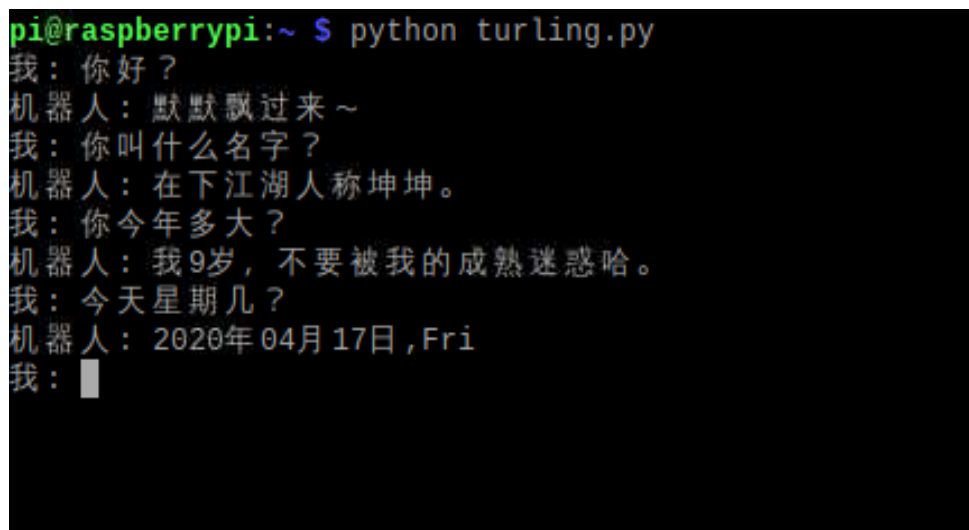
```
    request = api + info
```

```
    response = getTurling(request)
```

```
    dic_json = json.loads(response)
```

```
    print '机器人:'.decode('utf-8') + dic_json['text']//输出图灵回复
```

使用语句: `python turling.py` 进行 `turling.py` 编译运行, 如果能实现一些简单文字聊天功能, 说明图灵机器人能正常调用。



```
pi@raspberrypi:~ $ python turling.py
我: 你好?
机器人: 默默飘过来~
我: 你叫什么名字?
机器人: 在下江湖人称坤坤。
我: 你今年多大?
机器人: 我9岁, 不要被我的成熟迷惑哈。
我: 今天星期几?
机器人: 2020年04月17日, Fri
我: 
```

图 (11): 图灵机器人回复

将上面这些都测试完, 如果麦克风能正常的录音、语音识别能正常调用、图灵机器人能正常回复, 说明树莓派开发环境没有问题, 下面我将介绍如何在树莓派上实现具体的语音聊天功能。

第 2.3 节 语音聊天功能的实现

语音聊天功能的实现可以概括为: 语音识别把人发出的声音转换为文字, 经过自然语音处理得到意图, 再输入数据库获取回答的文字, 经过语音合成就能实现语音对

话，而后续这些功能可通过调用 API 来实现。

API (Application Programming Interface) 又被称作应用程序接口，是一些人们已经预先定义好的函数，可以用于实现不同的功能。在实际开发的过程中，开发人员不需要访问函数内部的代码，更不用去理解函数内部是如何实现工作机制的细节，只需要进行调用即可，而图灵机器人和百度人工智能平台为我们的开发提供了许多的免费接口。

硬件：

1. 音响 (播放声音)
2. 无驱动麦克风 (录制声音)
3. 树莓派

软件：

1. python2.7 (模块: pyaudio wave requests urllib urllib2 sys)

总共 5 步：

第一步: 录音

第二步: 语音识别

第三步: 图灵回复

第四步: 语音合成

第五步: 播放

2.3.1、录音

树莓派使用无驱动麦克风进行录音，录音使用的是 linux 的 arecord。

```
def recordVoice(self):
```

```
    print “开始录音...”
```

```
    os.system(‘arecord -D “plughw:1” -f S16_LE -r 16000 ’%self.RECORD_PATH)
```

```
print “录音结束...”
```

2.3.2、语音识别

调用流程:

创建账号及应用: 浏览器输入 `ai.baidu.com` 打开百度语音识别控制台, 创建应用, 勾选开通“语音技术”短语音识别、短语音识别极速版能力。获取开发所需的百度语音识别的 AppID、API Key、Secret Key, 并通过请求鉴权接口换取 token 。

创建识别请求: 采用 POST 方式, 音频可通过 JSON 和 RAW 两种方式提交。JSON 方式音频数据由于 base64 编码, 数据会增大 1/3。其他填写具体请求参数 。

返回识别结果: 百度语音识别结果需要采用 JSON 格式进行封装并且马上返回, 如果百度语音识别成功, 语音聊天系统会将识别的结果放在 JSON 的“result”字段中, 并且统一采用 utf-8 方式进行编码。

2.3.3、图灵回复

因为要实现聊天功能, 所以需要机器人能对用户的提问的问题进行相应的回复, 所以使用图灵机器人进行回复。图灵机器人根据百度语音识别转化的文本, 对用户的提问做出相匹配的回复。

```
def Reply(self,words): //图灵获取回复

    turling = {"key": self.Tuling_API_KEY,

              "info": words.encode("utf-8")}

    reply = requests.post(self.URL, data=turling, verify=True)

    if res:

        date = json.loads(reply.text)

        print date["text"]

        return date["text"]

    else:
```

```
print "对不起,未获取到回复信息"
```

2.3.4、语音合成

调用图灵机器人进行回复后，使用百度语音将图灵机器人回复的文本转化为相应的音频文件，合成的文件格式一般为 mp3，pcm（8k 及 16k），wav（16k）。

```
def voiceSynthesis(self,word): //语音合成
```

```
    token = self.access_token
```

```
    cuid = self.CUID
```

```
    URL="http://tsn.baidu.com/text2audio?tex="+word+"&lan=zh&cuid="+cuid+"&ctp=1&tok="+token+"&per=4"
```

```
    return URL
```

```
class TurLingAPI:
```

```
    def __init__(self):
```

```
        self.Tuling_API_KEY = "75858c17fe044c7fb390298e4453e881"//图灵机器人  
APIKEY
```

```
        self.URL = "http://www.tuling123.com/openapi/api"
```

2.3.5、播放

将百度语音合成的音频使用音响进行播放输出。

```
def playVoice(self,url): //播放声音
```

```
    print url
```

```
    os.system('mpg123 "%s"%url)
```

第 2.4 节 本章小结

本章详细介绍了树莓派开发的基本配置，如何开启 VNC 远程，以及实现语音聊

天的流程，包括测试录音，导入软件，测试百度识别的 API 是否能正常使用，测试机器人是否能正常调用。具体的语音聊天功能的实现，大体可以概括为麦克风进行录音，百度语音识别，图灵机器人进行回复，语音语音合成，播放回复音频五个步骤。

第 3 章 离线语音唤醒实现

在现实生活中，我们使用苹果手机的语音功能前都会对着手机说几声：“Siri, Siri”，这时苹果手机的 Siri 会被唤醒，我们可以和它进行聊天了，但当我们说其他的唤醒词时，Siri 并不会被我们唤醒。语音唤醒的使用，不仅方便使用语音聊天功能，也提供了一定的安全性。

第 3.1 节 离线语音唤醒

可以使用语音唤醒引擎 snowboy 实现语音唤醒功能，下面将介绍如何使用 snowboy 实现语音唤醒，并且如何修改唤醒词。

3.1.1、snowboy 介绍、安装与测试

Snowboy 是一款语音唤醒引擎，可以实现树莓派的离线语音唤醒功能，即树莓派在不需联网的情况下，依然可以实现语音唤醒。唤醒词可以用于发起语音聊天功能，除此之外，唤醒词也具有一定的安全性，使用了唤醒词后，语音聊天机器人就可以避免意外启动。

Snowboy 的特点：

1. 高度可定制。允许自行修改和定义唤醒词。
2. 可以一直监听，但是保护个人的隐私。Snowboy 不需要连接到网络，可以实现离线唤醒。

3. 轻巧的、可嵌入，可以让您在 Raspberry Pi 上运行。轻巧的、可嵌入，可以让您在 Raspberry Pi 上运行。

3.1.2、snowboy 安装

1、首先需要下载 snowboy 的安装包，连接如下：

<https://s3-us-west-2.amazonaws.com/snowboy/snowboy-releases/rpi-arm-raspbian-8.0-1.2.0.tar.bz2>

Snowboy 的压缩包下载完成后对其进行解压。

2、首先卸载树莓本身的 pyaudio 软件，然后重新进行安装。

```
sudo apt remove python-pyaudio
```

```
sudo apt install libasound-dev
```

```
sudo apt-get install libatlas-base-dev sox swig
```

接下来是安装 pyaudio，这里需要先进行下载，浏览器打开网址 <http://www.portaudio.com/download.html> 下载 pa_stable_v190600_20161030.tgz。



图（12）：pyaudio 安装包

对 pa_stabel_v190600_20161030.tgz 进行解压，通过 tar-xzf pa_stable_v190600_20161030.tgz 进行解压，然后进行安装。进到解压之后的文件目录下面，进行如下操作：

```
sudo ./configure
```

```
sudo make
```

```
sudo make install
```

```
sudo apt-get install python-pyaudio python3-pyaudio
```

Pyaudio 这样就配置好了。

3、如果树莓派新增了麦克风的输入设备，需要修改配置文件让 usb 声卡作为默认的音频输入。

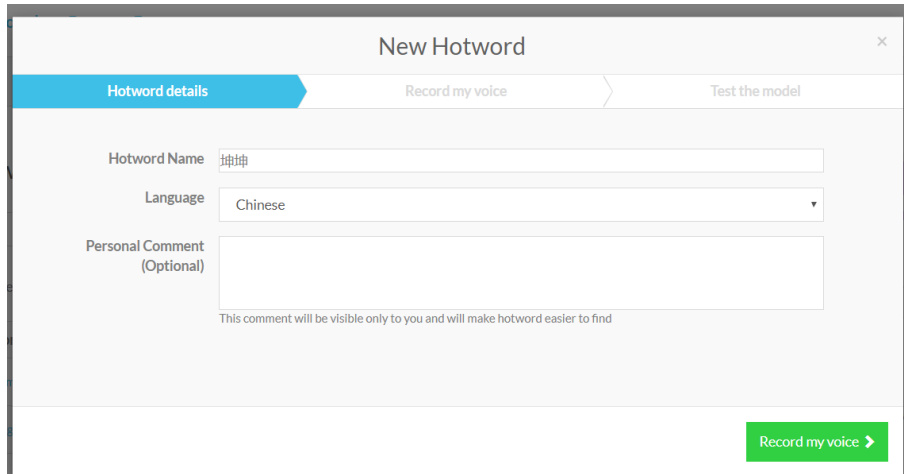
3.1.3、自定义唤醒词

浏览器进到 snowboy 的网址 (<https://snowboy.kitt.ai/hotword/>) 需要使用谷歌浏览器，其他的浏览器效果不好。这里需要注册一个 github 账号，右上角登录，然后会跳转到一个界面，点击创建唤醒词。



图 (13) : snowboy 创建唤醒词

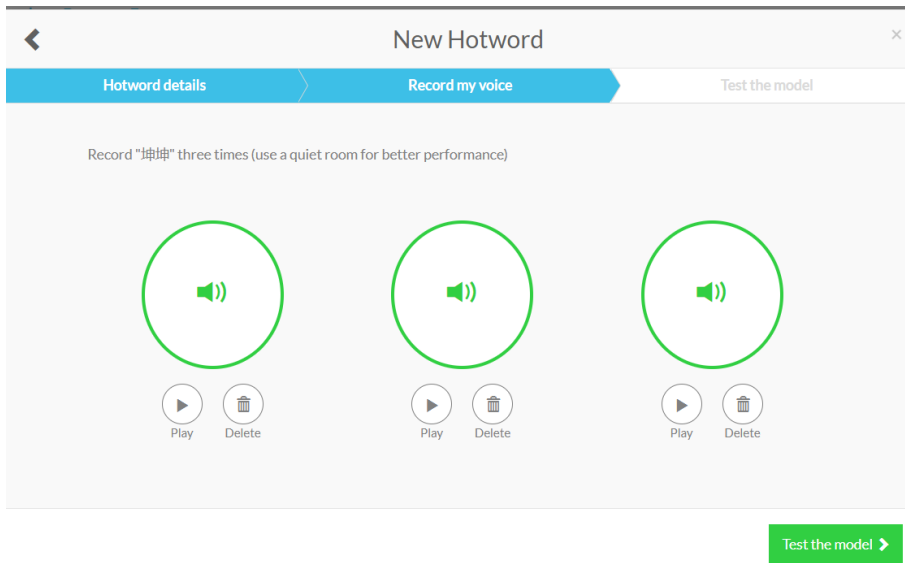
这里接下来开始输入一些唤醒词名字呀，然后语言选择中文，点击右下角创建，开始录音，建议在树莓派上面录好之后上传到页面，创建录音文件命令：`arecord -D "plughw:1,0" -d 3 -r 16000 -t wav -f S16_LE 文件名.wav`，分别创建 Test1.wav, Test2.wav, Test3.wav。



图（14）：snowboy 创建关键字

这样做的好处在于，用树莓派的录音功能录得音，然后识别出来的模型就更加真实贴近你用树莓派唤醒的方式。

上传完之后如下所示：



图（15）：snowboy 进行训练

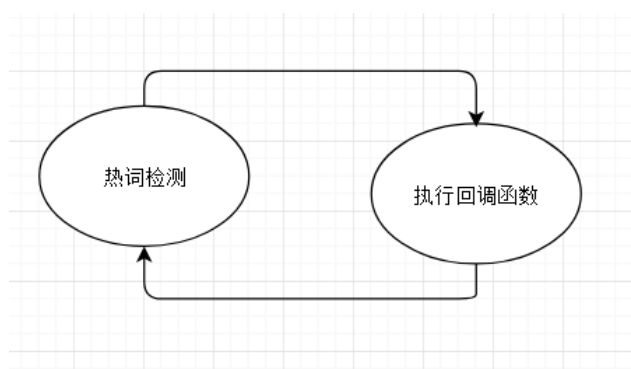
点击右下角开始测试模型，国内网速不是很好，这里需要耐心等待一下，当然也可以连接外网，速度比较快。然后说一句设置好的唤醒词，如果唤醒成功了，就可以点击右下角的 Save and download 了，下载成功后会出现一个坤坤.pmdl，然后需要重命名为 snowboy.pmdl。

最后一步，将 snowboy.pmdl 放到前面的 rpi-arm-raspbian-8.0-1.2.0.tar.bz2 解压好的文件夹中，运行如下指令，python demo.py snowboy.pmdl，说一句设置的唤醒词，假如这时会听见叮叮的声音，表明已经实现语音唤醒了。

第 3.2 节 语音唤醒和语音识别结合

3.2.1 语音唤醒和语音识别结合的原理

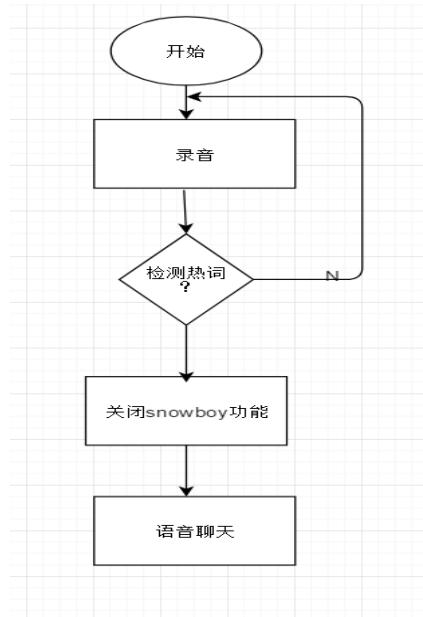
语音唤醒的流程如下图：



图（16）：语音唤醒流程图

执行回调函数主要实现语音聊天系统的语音聊天功能。使用关键字唤醒系统

后，系统执行回调函数。语音唤醒和语音聊天都需要使用麦克风设备，如果同时使用会造成设备冲突，语音聊天无法正常进行。因此语音唤醒成功后应当关闭 snowboy，释放麦克风设备，这样语音聊天才能正常启动。具体实现程序流程图如下：



图（17）：唤醒流程图

3.2.2 语音唤醒和语音识别结合的实现

```
#*_coding:UTF-8_*//将离线语音唤醒和语音聊天功能进行结合
```

```
//导入包
```

```
import snowboydecoder
```

```
import sys
```

```
import signal//导入需要的模板
```

```
import robot //导入实现语音聊天功能的文件
```

```
interrupt= False
```

```
def signal_handler(signal, frame):
```

```
global interrupt

interrupt = True

def interrupt_callback():

    global interrupt

    return interrupt

//执行回调函数，启动语音聊天功能

def callbacks():

    global detector

    // 语音唤醒成功后，音响会提示 ding ding 两声

    snowboydecoder.play_audio_file()

    snowboydecoder.play_audio_file()

    // 关闭语音唤醒功能，将麦克风设备进行释放

    detector.terminate()

    //启动语音聊天功能

    robot.luy()

    //打开 snowboy 功能

    wake_up()

    //关键字唤醒

def wake_up():

    global detector

    model = 'snowboy.pmdl' //唤醒词为：坤坤

    signal.signal(signal.SIGINT, signal_handler)
```

```
detector = snowboydecoder.HotwordDetector(model, sensitivity=0.45)

detector.start(detected_callback=callbacks,
              interrupt_check=interrupt_callback,
              sleep_time=0.05)

//释放麦克风设备，防止使用语音聊天功能时设备冲突

detector.terminate()

if __name__ == '__main__':
    wake_up()
```

第 3.3 节 本章小结

本章首先详细介绍了 snowboy 的功能特点以及使用 snowboy 如何实现语音唤醒，并且自定义关键字唤醒，接着对语音唤醒和语音聊天如何才能结合进行了详细的分析。最后在前章实现的语音聊天功能的基础上，通过对 snowboy 的使用，在树莓派上最终实现了语音唤醒和语音聊天功能的结合。

结 论

本课题研究了如何在树莓派上实现语音聊天功能，通过查阅文献，通过实际实验得到以下结论：

- 1、录音可以采用树莓派自带的 arecord 和 pyaudio 通过麦克风实现录音功能。
- 2、可以使用百度语音识别来识别我们说的话。
- 3、可以使用图灵机器人对我们的聊天内容进行回答，并且可以自定义一些问题。
- 4、可以使用语音唤醒引擎 snowboy 实现离线语音唤醒并且实现自定义关键字唤醒。

参考文献

- [1] 王树良, 李大鹏, 赵柏翔, 耿晶, 张伟, 王海雷. 聊天机器人技术浅析[J/OL]. 武汉大学学报(信息科学版):1-8[2019-12-26]. <https://doi.org/10.13203/j.whugis20190177>.
- [2]程洁. 智能聊天机器人:对话未来[J]. 科技经济导刊, 2018(02):14-15.
- [3]冯升. 聊天机器人问答系统现状与发展[J]. 机器人技术与应用, 2016(04):34-36.
- [4]张润午, 赵海舟, 曾凡瑞, 汤留阳. 基于树莓派的智能语音提醒系统[J]. 物联网技术, 2019, 9(10):79-81+83.
- [5]王辰越. 聊天机器人, 移动互联网下一个金矿[J]. 中国经济周刊, 2013(11):82-83.
- [6]曹祎遐, 何文清. 聊天机器人 换个方式聊天[J]. 上海信息化, 2016(10):16-19.
- [7]孙立茹, 余华云. 基于深度学习的生成式聊天机器人算法综述[J]. 电脑知识与技术, 2018, 14(23):227-228.
- [8]刘超, 马东宇. 智能问答的聊天机器人系统的设计与实现[J]. 信息技术, 2017(05):176-177+180.
- [9]吴灵慧. 问答系统研究综述[J]. 科技传播, 2019, 11(05):147-148.
- [10]李方方, 马昊宇. 基于聊天机器人的智能导购系统[J]. 福建电脑, 2018, 34(02):27-28.
- [11] 人机对话系统与自然语言处理_远洋号 -CSDN 博客
<https://blog.csdn.net/s3FRH3JyN6yymHmT11/article/details/100512834>
- [12] 树莓派简易聊天机器人_yonglisikao 的博客 -CSDN 博客
<https://blog.csdn.net/yonglisikao/article/details/82314512>
- [13]王浩畅, 李斌. 聊天机器人系统研究进展[J]. 计算机应用与软件, 35(12):7-12+95.
- [14]黄家兴. 基于安卓系统的汉语智能聊天机器人[D]. 厦门大学, 2013.

- [15] 易顺明, 胡振宇. 中文聊天机器人原型系统的设计[J]. 沙洲职业工学院学报, 2007(02):5-9.
- [16] Heung-yeung SHUM, Xiao-dong HE, Di LI. From Eliza to XiaoIce: challenges and opportunities with social chatbots[J]. Frontiers of Information Technology & Electronic Engineering, 2018, 19(01):10-26.
- [17] 技术宅. 知心朋友? 聊天机器人是怎么聊天的[J]. 电脑爱好者, 2016(17):62-63.
- [18] Aleksandra Przegalinska, Leon Ciechanowski, Anna Stroz, Peter Gloor, Grzegorz Mazurek. In bot we trust: A new methodology of chatbot performance measures[J]. Business Horizons, 2019.
- [19] Alhagbani E S , Khan M B . Challenges facing the development of the Arabic chatbot[C]// First International Workshop on Pattern Recognition. International Society for Optics and Photonics, 2016.
- [20] Rozga S . Practical Bot Development (Designing and Building Bots with Node.js and Microsoft Bot Framework) || Making the Chat Bot Smarter[M]. 2018.
- [21] Daniel F , Matera M , Zaccaria V , et al. Toward truly personal chatbots: on the development of custom conversational assistants[C]// the 1st International Workshop. IEEE Computer Society, 2018.
- [22] 树莓派打造对话机器人 Python_Python_u013860985 的博客-CSDN 博客 <https://blog.csdn.net/u013860985/article/details/69802766>
- [23] 树莓派安装 raspbian 并配置开发环境_网络_weixin_30715523 的博客-CSDN 博客 https://blog.csdn.net/weixin_30715523/article/details/98497201
- [24] 语音技术 <https://ai.baidu.com/ai-doc/SPEECH/Gk38y8lzk>
- [25] 离线语音唤醒 SnowBoy +树莓派语音交互实现语音机器人_人工智能_木木的博客-CSDN 博客 https://blog.csdn.net/sinat_35162460/article/details/86547013

致 谢

本文的撰写经历了五个月的查找资料和动手在树莓派上一步步实现功能，在这些过程中，收获了许多有关语音聊天，树莓派相关的知识，也再次让我明白了办法总是会比困难多的。

首先我要感谢我的导师胡标老师，感谢他给我这次窥探语音聊天世界的机会，是他为我指明了道路，不断鼓励着我前进，胡标老师的指导让我少走了许多的弯路。

同时我也要感谢我在这四年遇见的小伙伴，尤其是我的室友们，他们非常积极乐观有趣，感谢他们给我的阳光开朗的生活态度和舒适的生活环境，非常幸运遇见这样一群有趣的好伙伴。

最后我要感谢我的父母和我在学习生涯的所有老师，如果没有他们悉心的栽培，我也不会达到今天的高度，我定心怀感恩，继续往前。

当我落下毕业论文的最后一字时，也意味着我大学生活的结束，这四年时光，谈不上完美无瑕，但也不留遗憾，不但收获了知识，更学会了如何去面对生活中的困难——纵使困顿难行，也要砥砺前行。

天下没有不散的宴席，这四年的会是我一生的美好回忆。