**编 号：**

**审定成绩：**

|  |  |  |  |
| --- | --- | --- | --- |
| **重庆邮电大学**  **毕业设计（论文）** | | | |
|  | | | |
|  | **中文题目** | [**基于AI语音识别的websocket聊天室**](http://172.20.2.53:84/StudentSelectTopic/javascript:void(0)) |  |
|  |  |
| **英文题目** |  |
|  |  |
| **学院名称** | **软件工程学院** |
| **学生姓名** | **陈俊松** |
| **专 业** | **英语+软件** |
| **班 级** | **1301415** |
| **学 号** | **2014213963** |
| **指导教师** | **宋琦** |
| **答 辩 组**  **负 责 人** |  |
|  | |

**年 月**

**重庆邮电大学教务处制**

软件工程学院本科毕业设计(论文)诚信承诺书

本人郑重承诺：

我向学院呈交的论文《 基于AI语音识别的websocket聊天室 》，是本人在指导教师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明并致谢。本人完全意识到本声明的法律结果由本人承担。

年级 2014级

专业 英语+软件

班级 1301415

承诺人签名

年 月 日

学位论文版权使用授权书

本人完全了解重庆邮电大学有权保留、使用学位论文纸质版和电子版的规定，即学校有权向国家有关部门或机构送交论文，允许论文被查阅和借阅等。本人授权重庆邮电大学可以公布本学位论文的全部或部分内容，可编入有关数据库或信息系统进行检索、分析或评价，可以采用影印、缩印、扫描或拷贝等复制手段保存、汇编本学位论文。

（注：保密的学位论文在解密后适用本授权书。）

|  |  |
| --- | --- |
| 学生签名： | 指导老师签名： |
| 日期： 年 月 日 | 日期： 年 月 日 |

摘要

随着时代的飞速发展，互联网技术日新月异，传统的通信技术已经不能满足我们在日常通信的要求，WebSocket 即时通信技术应运而生，而基于语音识别的即时通信技术更是将我们的生活丰富化，便捷化。技术的发展，引导我们不断的进步。WebSocket 协议本质上是一个基于 TCP 的协议。为了建立一个 WebSocket连接，客户端浏览器首先要向服务器发起一个 HTTP 请求，这个请求和通常的 HTTP请求不同，包含了一些附加头信息，其中附加头信息“Upgrade: WebSocket”表明这是一个申请协议升级的 HTTP请求，服务器端解析这些附加的头信息然后产生应答信息返回给客户端，客户端和服务器端的 WebSocket便可建立连接。双方就可以通过这个连接通道自由的传递信息，并且这个连接会持续存在直到客户端或者服务器端的某一方主动的关闭连接。语音识别技术，也被称为自动语音识别Automatic Speech Recognition，(ASR)，就是让机器通过识别和理解过程把语音信号转变为相应的文本或命令的高技术。　主要包括特征提取技术、模式匹配准则及模型训练技术三个方面。其目标是将人类的语音中的词汇内容转换为计算机可读的输入。

基于AI语音识别的Websocket聊天室主要是通过语音识别技术将我们的声音进行语音识别并及时进行数据传输到服务端，服务端通过验证进行语音识别，并返回响应，还可以通过机器人智能回复，智能查询相关信息，适用于多个应用场景，可流畅的在各个系统上运行。

基于AI语音识别的Websocket聊天室主要是基于C/S架构，前端通过react + ant.design 进行MVVM模式构建，后端通过express + websocket进行服务端实现，数据库通过NO-SQL的MongoDB的进行搭建。

**关键词：**Websocket，语音识别，聊天室

**Abstract**

With the rapid development of the times, Internet technology is changing with each passing day, traditional communication technologies can no longer meet the requirements of our daily communications, WebSocket instant messaging technology came into being, and instant messaging technology based on speech recognition enriches our lives. Facilitation. The development of technology leads us to continuous improvement. The WebSocket protocol is essentially a TCP-based protocol. In order to establish a WebSocket connection, the client browser first sends an HTTP request to the server. This request is different from the usual HTTP request and contains some additional header information. The additional header “Upgrade: WebSocket” indicates that this is an application protocol. Upgraded HTTP requests. The server parses these additional headers and generates response information back to the client. WebSockets on the client and server can establish connections. Both parties can freely transmit information through this connection channel, and this connection will continue to exist until either the client or the server side actively closes the connection. Speech recognition technology, also known as Automatic Speech Recognition (ASR), is a high technology that allows machines to convert speech signals into corresponding text or commands by identifying and understanding the process. It mainly includes three aspects: feature extraction technology, pattern matching criteria, and model training technology. The goal is to convert the vocabulary content of human speech into computer-readable input.

The Websocket chat room based on AI voice recognition mainly uses voice recognition technology to carry out voice recognition of our voices and transmit data to the server in time. The server authenticates and recognizes voices and returns responses. It can also reply intelligently through robots. Query related information, suitable for multiple application scenarios, can run smoothly on various systems.

The Websocket chat room based on AI voice recognition is mainly based on the C/S architecture. The front end constructs the MVVM mode through react + ant.design, and the back end implements the server through express + websocket. The database is constructed through NO-SQL MongoDB

**Keywords**: Websocket , voice recognition, chat room

目录

[第1章 引言](#_Toc469491156)

[1.1 研究背景和意义](#_Toc469491157)

[1.2 国内外研究现状](#_Toc469491158)

[1.3 主要内容和工作安排](#_Toc469491161)

[第2章 开发工具及平台兼容性](#_Toc469491162)

[2.1 编码工具和调试工具的选择](#_Toc469491163)

2.1.1 sublime的开发优势

2.1.2基于chrome扩展程序React Developer Tools的调试

[2.2 Node.js和npm介绍](#_Toc469491164)

2.2.1 Node.js的介绍和开发优势

2.2.2 npm包管理工具的使用

[2.3打包工具webpack介绍](#_Toc469491167)

2.4代码规范及ESLint介绍使用

2.5 MongoDB数据库介绍

[第3章 系统分析与设计](#_Toc469491168)

3.1系统需求分析

3.2 系统概要设计

3.2.1 系统结构设计

3.2.2 数据库设计

[第4章 系统的具体实现](#_Toc469491176)

4.1 前端react + ant.design设计

4.1.1 音频转换模块

4.1.2 语音识别模块

4.1.3 语音朗读模块

4.1.4 登录注册模块

4.1.5 聊天室模块

4.2 后端express + websocket设计

4.2.1 websocket设计模块

4.2.2语音识别接口模块

4.2.3机器人回复模块

4.2.4登录注册模块

4.2.5聊天室模块

4.2.6数据库连接模块

4.3 MongoDB数据库设计

4.3.1数据库user集合设计

4.3.2 数据库聊天内容talking集合设计

[第5章 系统测试](#_Toc469491188)

[参考文献](#_Toc469491211)

[致谢](#_Toc469491212)

[附录A](#_Toc469491213)

[附录B](#_Toc469491214)

[一、英文原文](#_Toc469491215)

[二、英文](#_Toc469491216)

[三、源程序](#_Toc469491217)

[四、其他](#_Toc469491218)

# 

# 第1章 引言

## 1.1 研究背景和意义

大概在 1992 年，一家称作 Nombas 的公司开发了一种叫做 C 减减（C-minus-minus，简称 Cmm）的嵌入式脚本语言。Cmm 背后的理念很简单：一个足够强大可以替代宏操作（macro）的脚本语言，同时保持与 C （和 C ++）足够的相似性，以便开发人员能很快学会。这个脚本语言捆绑在一个叫做 CEnvi 的共享软件中，它首次向开发人员展示了这种语言的威力。Nombas 最终把 Cmm 的名字改成了 ScriptEase。当 Netscape Navigator 崭露头角时，Nombas 开发了一个可以嵌入网页中的 CEnvi 的版本。这些早期的试验被称为 Espresso Page（浓咖啡般的页面），它们代表了第一个在万维网上使用的客户端语言。而 Nombas 丝毫没有料到它的理念将会成为万维网的一块重要基石。

当网上冲浪越来越流行时，对于开发客户端脚本的需求也逐渐增大。此时，大部分因特网用户还仅仅通过 28.8 kbit/s 的调制解调器连接到网络，即便这时网页已经不断地变得更大和更复杂。而更加加剧用户痛苦的是，仅仅为了简单的表单有效性验证，就要与服务器进行多次地往返交互。设想一下，用户填完一个表单，点击提交按钮，等待了 30 秒的处理后，看到的却是一条告诉你忘记填写一个必要的字段。那时正处于技术革新最前沿的 Netscape，开始认真考虑开发一种客户端脚本语言来解决简单的处理问题。

就在 Netscape Navigator 2.0 即将正式发布前，Netscape 将其更名为 JavaScript，目的是为了利用 Java 这个因特网时髦词汇。Netscape 的赌注最终得到回报，JavaScript 从此变成了因特网的必备组件。

随着javascript的迅速发展，其在互联网的角色也越来越鲜明，重要性也越来越高，由此在互联网领域的作用日渐突出。根据许多调查显示，JavaScript连续多年在互联网开发中高居榜首。

‘**能用 JavaScript 编写的程序，都终将会以 JavaScript 编写**’，这句话曾经听起来十分高傲，但是随著 Node.js 的发展，我们渐渐地明白到这句话的远见性──JavaScript 真的越来越强大了。

2009年2月，Ryan Dahl在博客上宣布准备基于V8创建一个轻量级的[Web服务器](https://www.baidu.com/s?wd=Web%E6%9C%8D%E5%8A%A1%E5%99%A8&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dBmW6znywWPy7-nhNWPWm30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHDLP1fLPHbdP1mYPW01rHcdrf)并提供一套库。 2009年5月，Ryan Dahl在GitHub上发布了最初版本的部分Node.js包，随后几个月里，有人开始使用[Node.js开发](https://www.baidu.com/s?wd=Node.js%E5%BC%80%E5%8F%91&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dBmW6znywWPy7-nhNWPWm30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHDLP1fLPHbdP1mYPW01rHcdrf)应用。 2009年11月和2010年4月，两届JSConf大会都安排了Node.js的讲座。 2010年年底，Node.js获得[云计算](https://www.baidu.com/s?wd=%E4%BA%91%E8%AE%A1%E7%AE%97&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dBmW6znywWPy7-nhNWPWm30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHDLP1fLPHbdP1mYPW01rHcdrf)服务商Joyent资助，创始人Ryan Dahl加入Joyent全职负责Node.js的发展。 2011年7月，Node.js在微软的支持下发布Windows版本。Node.js 是一个非常新兴的开发工具，年龄远不如 Python、Ruby、PHP 等老大哥，但是它确实有史以来发展最快的开发工具，没有之一。在这短短的几年间，我们看到了 Node.js 从当初的一无所有到如今的飞速发展，这是没有任何其他开发工具能够媲美的。

HTML5草案的前身名为 Web Applications 1.0，于2004年被WHATWG提出，于2007年被W3C接纳，并成立了新的 HTML 工作团队。HTML 5 的第一份正式草案已于2008年1月22日公布。HTML5 仍处于完善之中。然而，大部分[现代浏览器](https://baike.baidu.com/item/%E7%8E%B0%E4%BB%A3%E6%B5%8F%E8%A7%88%E5%99%A8)已经还仍不具备 HTML5 支持。

“语音识别”的终极梦想，是真正能够理解人类语言甚至是方言环境的系统。但几十年来，人们并没有一个有效的策略来创建这样一个系统，直到人工智能技术的爆发。在过去几年中，人们在人工智能和深度学习领域的突破，让语音识别的探索跨了一大步。其中的精髓便是对话式AI，对话式AI包含有两个主要类别：人机界面，以及人与人沟通的界面。在人机界面中，人类与机器往往通过语音或文本交互，届时机器会理解人类 (尽管这种理解方式是有限的) 并采取相应的一些措施。

基于HTML5的版本兼容性和JavaScript以及AI语音识别技术的不断演变，为了实现真正的纯JavaScript语言演绎的PC端语音识别系统，由此开发了基于AI语音识别的聊天室。

基于AI语音识别的聊天室摒弃了传统的JSP开发模式，前端页面和后端服务器接口都采用纯JavaScript语言编写，实现技术的一致性，减轻前后端数据连通的复杂性，可以使开发人员清晰的了解数据的流动，实现前后端的无缝式开发。而前端部分通过对HTML5标准进行二次包装，以实现语音格式转换，语音朗读TTS引擎的兼容性，语音识别部分通过封装的AI接口，智能识别并返回机器人响应。完善了HTML5标准的部分接口和AI语音识别的信息响应，实现了纯PC端JavaScript开发的AI语音识别聊天室。

## 1.2 国内外研究现状

### 1.2.1 国外研究现状

语音识别的研究工作可以追溯到20世纪50年代AT&T贝尔实验室的Audry系统，它是第一个可以识别十个英文数字的语音识别系统。但真正取得实质性进展，并将其作为一个重要的课题开展研究则是在60年代末70年代初。实验室语音识别研究的巨大突破产生于20世纪80年代末：人们终于在实验室突破了大词汇量、连续语音和非特定人这三大障碍，第一次把这三个特性都集成在一个系统中，比较典型的是卡耐基梅隆大学（CarnegieMellonUniversity）的Sphinx系统，它是第一个高性能的非特定人、大词汇量连续语音识别系统。

20世纪90年代前期，许多著名的大公司如IBM、苹果、AT&T和NTT都对语音识别系统的实用化研究投以巨资。语音识别技术有一个很好的评估机制，那就是识别的准确率，而这项指标在20世纪90年代中后期实验室研究中得到了不断的提高。比较有代表性的系统有：IBM公司推出的ViaVoice和DragonSystem公司的NaturallySpeaking，Nuance公司的NuanceVoicePlatform语音平台，Microsoft的Whisper，Sun的VoiceTone等。

其中IBM公司于1997年开发出汉语ViaVoice语音识别系统，次年又开发出可以识别上海话、广东话和四川话等地方口音的语音识别系统ViaVoice‘98。它带有一个32，000词的基本词汇表，可以扩展到65，000词，还包括办公常用词条，具有“纠错机制”，其平均识别率可以达到95%。该系统对新闻语音识别具有较高的精度，是目前具有代表性的汉语连续语音识别系统。

2002 年 8 月，W3C 建立了第 4 个 HTML 工作组 （实际上是第 3 个 XHTML 工作组），目标是完成 HTML 到 XHTML 的转变，包括完成以尽可能多的 使用通用的 XML 技术为设计目标的 XHTML 2.0。同 时，HTML 工作组发布了 XHTML 2.0 第一个公开工 作草案，并在声明中指出，尽管 XHTML 2.0 来源于 HTML 4、XHTML 1.0 和 XHTML 1.1，XHTML 2.0 并 不向后兼容早期版本。

HTML 5 在 HTML 4 的基础上新增了较多元素和 属性，同时，也废除了部分元素和属性。与 HTML 4 相比，HTML 5 在文档结构、页面元素、图形支持、 多媒体支持、本地存储、离线应用、多线程和地理位置信息等方面有较大的变化，这些变化极大地提 升了 HTML 5 对 Web 应用的开发支持。

浏览器是 HTML 5 的重要解析器，因此，浏览 器对 HTML 5 的支持直接影响 HTML 5 的应用。由 于主要浏览器厂商直接参与了 HTML 5 的标准制定 工作，因此，多数浏览器厂商发布的新版本浏览器 都加强了对 HTML 5 的支持。

### 1.2.2国内研究现状

我国语音识别研究工作起步于五十年代，但近年来发展很快。研究水平也从实验室逐步走向实用。从1987年开始执行国家863计划后，国家863智能计算机专家组为语音识别技术研究专门立项，每两年滚动一次。我国语音识别技术的研究水平已经基本上与国外同步，在汉语语音识别技术上还有自己的特点与优势，并达到国际先进水平。

中科院自动化所、声学所、清华大学、北京大学、哈尔滨工业大学、上海交通大学、中国科技大学、北京邮电大学、华中科技大学等科研机构都有实验室进行过语音识别方面的研究，其中具有代表性的研究单位为清华大学电子工程系与中科院自动化研究所模式识别国家重点实验室。

HTML 5 标准涉及浏览器开发商、工具开发商及 应用开发商等，技术覆盖面较广，标准制定难度较大。 我国为了推动基于 HTML 5 的互联网和移动互联网 产业发展，我国标准研制主要聚焦于聚焦于浏览器兼 容性规范、浏览器编程接口及标准符合性测试规范 等基础性标准，并结合 HTML 5 的后续发展，逐步 建立起符合我国产业发展需求的标准体系。

目前，HTML 5 标准由 W3C 和 WHATWG 联合 制定。我国的百度、腾讯、UC、华为、盛大、360 等互联网企业已于近年加入 W3C，并派代表参加 HTML 工作组。

## 1.3 主要内容和工作安排

本文主要按以下方式组织：

第一章：主要介绍系统的研究目的和意义，研究背景，研究现状，以及标准实现的不同情况。

第二章：主要介绍系统实现的所需工具优势，调试软件的使用，包管理器的开发和使用，打包工具的使用，以及数据库开发的优势

第三章：主要介绍系统的设计方向，包括系统需求分析设计，系统概要设计，系统结构设计，数据库设计方向。

第四章：主要介绍系统的编码实现，包括HTML5标准的实现以及兼容处理，包括对不支持浏览器的容错处理，语音识别的格式转换问题，前端框架架构处理，后端接口数据处理，以及机器人回复处理。

第五章：主要介绍系统测试结果，包括语音识别结果，服务器压力测试，系统黑盒和白盒测试，数据库相关测试。

# 第2章开发工具及平台兼容性

## 2.1编码工具和调试工具的选择

## 2.2.1 sublime的开发优势

Sublime Text 是一个代码编辑器，也是HTML和散文先进的文本编辑器。Sublime Text是由程序员Jon Skinner于2008年1月份所开发出来，它最初被设计为一个具有丰富扩展功能的[Vim](https://baike.baidu.com/item/Vim)。

Sublime Text具有漂亮的用户界面和强大的功能，例如代码缩略图，[Python](https://baike.baidu.com/item/Python)的插件，代码段等。还可自定义键绑定，菜单和工具栏。Sublime Text 的主要功能包括：拼写检查，书签，完整的 Python API ， Goto 功能，即时项目切换，多选择，多窗口等等。Sublime Text 是一个跨平台的编辑器，同时支持[Windows](https://baike.baidu.com/item/Windows)、[Linux](https://baike.baidu.com/item/Linux)、[Mac OS X](https://baike.baidu.com/item/Mac%20OS%20X)等操作系统。作为主流前端开发编辑器，它不仅体积较小，运行速度快并且文本功能强大，同时支持编译功能且可在控制台看到输出，内嵌python解释器支持插件开发以达到可扩展目的和内置Package Control——可支持的大量插件可通过其进行管理。

## 2.2.1 基于chrome扩展程序React Developer Tools的调试

React Developer Tools是一款由facebook开发的有用的[Chrome 浏览器扩展](http://www.cnplugins.com/)，可以通过 Chrome Web存储获取。使用 Chrome Devtools 进行调试时，可以查看应用程序的 React 组件分层结构，而不是更加神秘的浏览器 DOM 表示。添加react developer tools到chrome，是对chrome开发工具的React调试工具。React的开发工具是开源Chrome DevTools延伸反应的JavaScript库。它允许你检查React在Chrome开发者工具组件的层次结构（原名WebKit Web Inspector）。你会得到新的标签要求在你的Chrome DevTools反应。这表明你的根反应组件在页面渲染，以及他们最终渲染组件。

插件安装成功后，浏览器的右上方出现现图示的log，用户如果想要调试的话就点击log就可以了。通过选择树中的一个组件，你可以在右边的面板中检查和编辑它的当前的道具和状态。在面包屑，你可以检查所选组件，创建它的组件，创建一个组件，等等。如果您使用常规元素选项卡检查页面上的一个反应元素，然后切换到“React”选项卡，该元素将在“React树”中自动选择。

如图2.1所示显示了安装了 React Developer Tools 的 example3 组件分层结构外观。

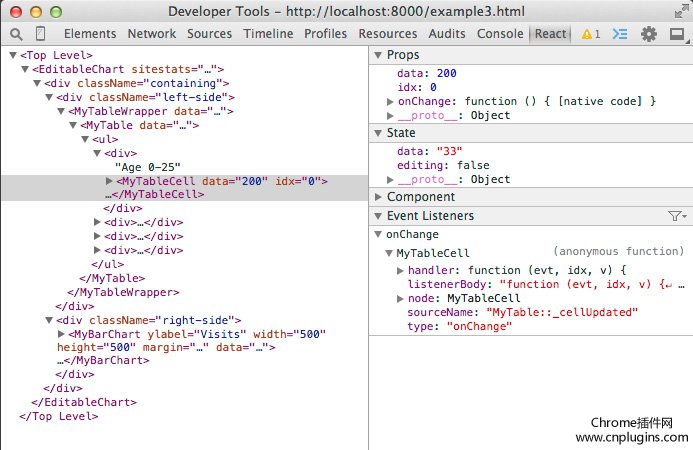


图2.1 React Developer Tools 的 example3 组件分层结构外观图

类似地，如图2.2.2，如果在一个组件的渲染阶段有一个断点，将在“React”选项卡中自动选择。这允许你在渲染树中一步一步看到一个组件如何影响另一个。通过打破错误，你可以很容易地找到哪些组件在渲染过程中抛出一个错误，和什么样的道具导致它

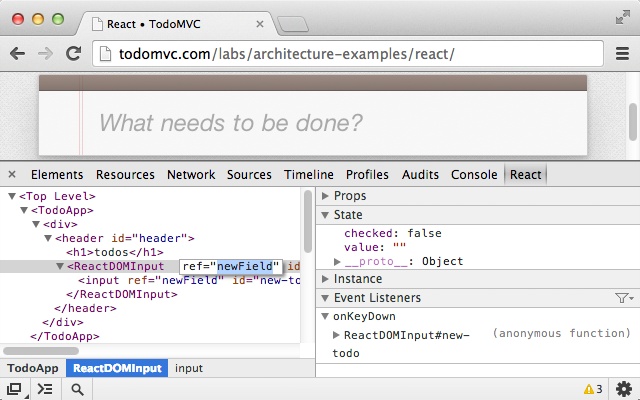


图2.2 错误断点调试

## 2.2 Node.js和npm介绍

## 2.2.1 Node.js的介绍和开发优势

Node.js是一个Javascript运行环境(runtime environment)，发布于2009年5月，由Ryan Dahl开发，实质是对Chrome V8引擎进行了封装。Node.js对一些特殊用例进行优化，提供替代的API，使得V8在非浏览器环境下运行得更好。

V8引擎执行Javascript的速度非常快，性能非常好。[1]  Node.js是一个基于Chrome JavaScript运行时建立的平台， 用于方便地搭建响应速度快、易于扩展的网络应用。Node.js 使用[事件驱动](https://baike.baidu.com/item/%E4%BA%8B%E4%BB%B6%E9%A9%B1%E5%8A%A8)， 非阻塞[I/O](https://baike.baidu.com/item/I%2FO/84718) 模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。

V8引擎本身使用了一些最新的编译技术。这使得用Javascript这类[脚本语言](https://baike.baidu.com/item/%E8%84%9A%E6%9C%AC%E8%AF%AD%E8%A8%80)编写出来的代码运行速度获得了极大提升，又节省了开发成本。对性能的苛求是Node的一个关键因素。 Javascript是一个[事件驱动](https://baike.baidu.com/item/%E4%BA%8B%E4%BB%B6%E9%A9%B1%E5%8A%A8)语言，Node利用了这个优点，编写出可扩展性高的服务器。Node采用了一个称为“事件循环(event loop）”的架构，使得编写可扩展性高的服务器变得既容易又安全。提高服务器性能的技巧有多种多样。Node选择了一种既能提高性能，又能减低开发复杂度的架构。这是一个非常重要的特性。并发编程通常很复杂且布满地雷。Node绕过了这些，但仍提供很好的性能。

Node采用一系列“非阻塞”库来支持事件循环的方式。本质上就是为文件系统、数据库之类的资源提供接口。向文件系统发送一个请求时，无需等待硬盘（[寻址](https://baike.baidu.com/item/%E5%AF%BB%E5%9D%80)并检索文件），硬盘准备好的时候非阻塞接口会通知Node。该模型以可扩展的方式简化了对慢资源的访问， 直观，易懂。尤其是对于熟悉[onmouseover](https://baike.baidu.com/item/onmouseover)、onclick等[DOM](https://baike.baidu.com/item/DOM/50288)事件的用户，更有一种似曾相识的感觉。

虽然让Javascript运行于服务器端不是Node的独特之处，但却是其一强大功能。不得不承认，浏览器环境限制了我们选择编程语言的自由。任何服务器与日益复杂的浏览器客户端应用程序间共享代码的愿望只能通过Javascript来实现。虽然还存在其他一些支持Javascript在服务器端 运行的平台，但因为上述特性，Node发展迅猛，成为事实上的平台。

### 2.2.2 npm包管理工具的使用介绍

NPM的全称是Node Package Manager，是一个NodeJS包管理和分发工具，已经成为了非官方的发布Node模块（包）的标准。

Nodejs自身提供了基本的模块，但是开发实际应用过程中仅仅依靠这些基本模块则还需要较多的工作。幸运的是，Nodejs库和框架为我们提供了帮助，让我们减少工作量。但是成百上千的库或者框架管理起来又很麻烦，有了NPM，可以很快的找到特定服务要使用的包，进行下载、安装以及管理已经安装的包。

它是世界上最大的软件注册表，每星期大约有 30 亿次的下载量，包含超过 600000 个 包（package） （即，代码模块）。来自各大洲的开源软件开发者使用 npm 互相分享和借鉴。包的结构使您能够轻松跟踪依赖项和版本。

npm 由三个独立的部分组成:网站，注册表（registry），命令行工具（CLI）。[网站](https://npmjs.com/) 是开发者查找包（package）、设置参数以及管理 npm 使用体验的主要途径。注册表 是一个巨大的数据库，保存了每个包（package）的信息。[CLI](https://docs.npmjs.com/cli/npm) 通过命令行或终端运行。开发者通过 CLI 与 npm 打交道。

## 2.3打包工具webpack介绍

现今的很多网页其实可以看做是功能丰富的应用，它们拥有着复杂的JavaScript代码和一大堆依赖包。为了简化开发的复杂度，前端社区涌现出了很多好的实践方法

* 模块化，让我们可以把复杂的程序细化为小的文件;
* 类似于TypeScript这种在JavaScript基础上拓展的开发语言：使我们能够实现目前版本的JavaScript不能直接使用的特性，并且之后还能转换为JavaScript文件使浏览器可以识别；
* Scss，less等CSS预处理器

这些改进确实大大的提高了我们的开发效率，但是利用它们开发的文件往往需要进行额外的处理才能让浏览器识别,而手动处理又是非常繁琐的，这就为WebPack类的工具的出现提供了需求。

webpack 是一个现代 JavaScript 应用程序的静态模块打包器(module bundler)。当 webpack 处理应用程序时，它会递归地构建一个依赖关系图(dependency graph)，其中包含应用程序需要的每个模块，分析你的项目结构，找到JavaScript模块以及其它的一些浏览器不能直接运行的拓展语言（Scss，TypeScript等），并将其转换和打包为合适的格式供浏览器使用。

## 2.4代码规范及ESLint介绍使用

ESLint是一个用来识别 ECMAScript 并且按照规则给出报告的代码检测工具，使用它可以避免低级错误和统一代码的风格。ESLint被设计为完全可配置的，主要有两种方式来配置ESLint：

* 在注释中配置：使用JavaScript注释直接把配置嵌入到JS文件中。
* 配置文件：使用下面任一的文件来为全部的目录和它的子目录指定配置信息。
  + javascript：使用.eslintrc.js文件并导出一个包含配置的对象。
  + YAML：.eslintrc.yaml或者.eslintrc.yml
  + JSON：.eslintrc.json，并且此文件允许使用JS形式的注释
  + 废弃的用法：.eslintrc，此文件可以是JSON或者YAML
  + package.json：在package.json文件中创建eslintConfig属性，所有的配置包含在此属性中。

这些文件的优先级则是按照以上出现的顺序（.eslintrc.js > .eslintrc.yaml > .eslintrc.yml > .eslintrc.json > .eslintrc > package.json）。

可以被配置的信息主要分为3类：

* Environments：你的 javascript 脚步将要运行在什么环境（如：nodejs，browser，commonjs等）中。
* Globals：执行代码时脚步需要访问的额外全局变量。
* Rules：开启某些规则，也可以设置规则的等级。

## 2.5MongoDB数据库介绍

MongoDB  是一个基于分布式文件存储的数据库。由[C++](https://baike.baidu.com/item/C%2B%2B)语言编写。旨在为WEB应用提供可扩展的高性能数据存储解决方案。

MongoDB  是一个介于[关系数据库](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E6%95%B0%E6%8D%AE%E5%BA%93)和非关系数据库之间的产品，是非关系数据库当中功能最丰富，最像关系数据库的。他支持的数据结构非常松散，是类似[json](https://baike.baidu.com/item/json)的[bson](https://baike.baidu.com/item/bson)格式，因此可以存储比较复杂的数据类型。Mongo最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立[索引](https://baike.baidu.com/item/%E7%B4%A2%E5%BC%95)。

它的特点是高性能、易部署、易使用，存储数据非常方便

主要功能特性有：

\*面向集合存储，易存储对象类型的数据。

\*模式自由。

\*支持动态[查询](https://baike.baidu.com/item/%E6%9F%A5%E8%AF%A2)。

\*支持完全索引，包含内部对象。

\*支持查询。

\*支持复制和故障恢复。

\*使用高效的二进制数据存储，包括大型对象（如视频等）。

\*自动处理碎片，以支持云计算层次的扩展性。

\*支持[RUBY](https://baike.baidu.com/item/RUBY)，[PYTHON](https://baike.baidu.com/item/PYTHON)，[JAVA](https://baike.baidu.com/item/JAVA)，[C++](https://baike.baidu.com/item/C%2B%2B)，[PHP](https://baike.baidu.com/item/PHP)，[C#](https://baike.baidu.com/item/C%23)等多种语言。

\*文件存储格式为BSON（一种JSON的扩展）。

\*可通过[网络](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C)访问。

# 第3章系统分析与设计

## 3.1 系统需求分析

本系统是基于AI语音识别的websocket聊天室，主要的功能是通过AI语音识别聊天内容并采用websocket传输方式，进行实时数据交流，并使用机器人将语音识别的内容进行智能回复。

主要的业务是

1. 调用HTML5标准录音接口对接收到的音频进行接收处理，分离，转换，解码成可以进行语音识别的音频内容。
2. 调用语音识别接口，对转换成功的音频文件进行语音识别。
3. 采用websocket进行实时的数据传输，保证数据传输的实时性。
4. 调用机器人接口对语音识别成功的内容进行智能回复。

系统的功能需求是

1. 登录注册页面的成功实现
2. 录音功能的实现
3. 聊天室功能的实现
4. 系统UI界面友好，交互性好，易于操作

## 3.2系统概要设计

### 3.2.1 系统结构设计

系统结构图，如图3-1所示：

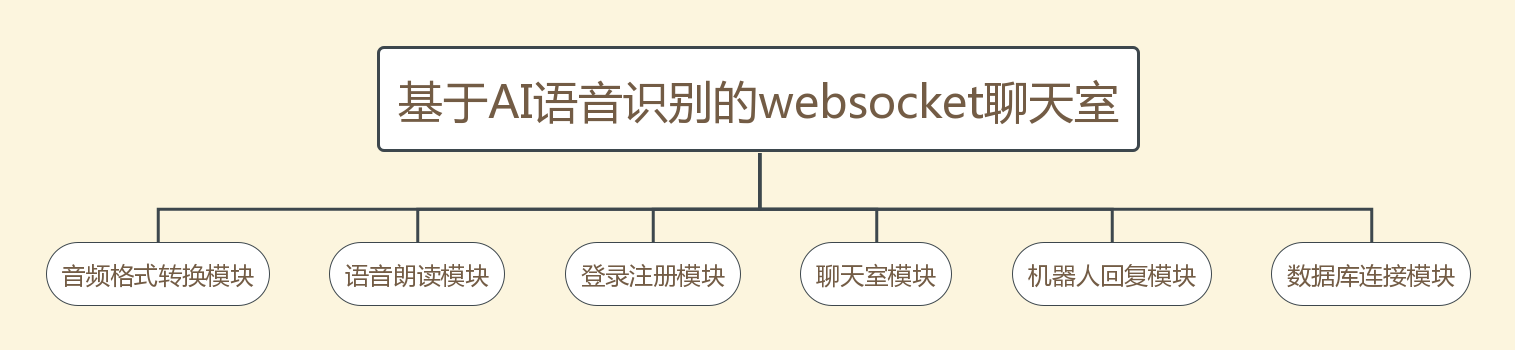


图3-1 系统结构图

系统登陆注册流程图，如图3-2：

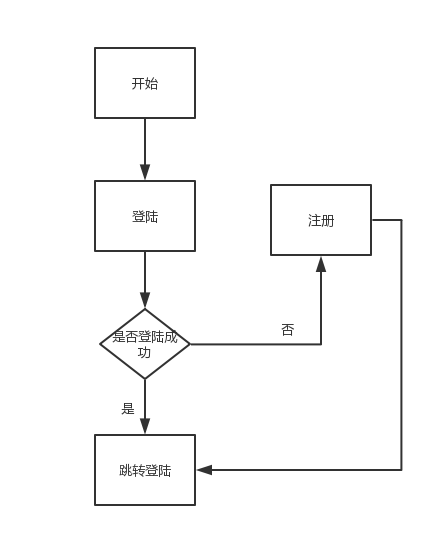


图3.2 登陆注册流程图

# 第4章 系统的具体实现

## 4.1 前端的具体实现

### 4.1.1 音频转换模块

获取HTML5音频对象：

var context = new AudioContext();

创建音频输入对象：

var audioInput = context.createMediaStreamSource(stream);

创建录音对象：

var recorder = context.createScriptProcessor(4096, 1, 1);

初始化音频转换变量：

size: 0, //录音文件长度

buffer: [], //录音缓存

inputSampleRate: context.sampleRate, //输入采样率 默认为 44.1k

inputSampleBits: 16, //输入采样数位 8, 1， 百度语音识别: 原始语音的录音格式目前只支持评测 8k/16k 采样率 16bit 位深的单声道语音

outputSampleRate: 16000, //输出采样率

oututSampleBits: 16, //输出采样数位 8, 16

对接收到的音频进行压缩合并的函数：

compress: function () { //合并压缩

//合并

var data = new Float32Array(this.size);

var offset = 0;

for (var i = 0; i < this.buffer.length; i++) {

data.set(this.buffer[i], offset);

offset += this.buffer[i].length;

}

//压缩

var compression = parseInt(this.inputSampleRate / this.outputSampleRate , 10);

var length = data.length / compression;

var result = new Float32Array(length);

var index = 0, j = 0;

while (index < length) {

result[index] = data[j];

j += compression;

index++;

}

return result;

},

对音频进行编码的函数：

encodeWAV: function () {

var sampleRate = Math.min(this.inputSampleRate, this.outputSampleRate);

var sampleBits = Math.min(this.inputSampleBits, this.oututSampleBits);

var bytes = this.compress();

var dataLength = bytes.length \* (sampleBits / 8);

var buffer = new ArrayBuffer(44 + dataLength);

var data = new DataView(buffer);

var channelCount = 1;//单声道

var offset = 0;

var writeString = function (str) {

for (var i = 0; i < str.length; i++) {

data.setUint8(offset + i, str.charCodeAt(i));

}

}

// 资源交换文件标识符

writeString('RIFF'); offset += 4;

// 下个地址开始到文件尾总字节数,即文件大小-8

data.setUint32(offset, 36 + dataLength, true); offset += 4;

// WAV文件标志

writeString('WAVE'); offset += 4;

// 波形格式标志

writeString('fmt '); offset += 4;

// 过滤字节,一般为 0x10 = 16

data.setUint32(offset, 16, true); offset += 4;

// 格式类别 (PCM形式采样数据)

data.setUint16(offset, 1, true); offset += 2;

// 通道数

data.setUint16(offset, channelCount, true); offset += 2;

// 采样率,每秒样本数,表示每个通道的播放速度

data.setUint32(offset, sampleRate, true); offset += 4;

// 波形数据传输率 (每秒平均字节数) 单声道×每秒数据位数×每样本数据位/8

data.setUint32(offset, channelCount \* sampleRate \* (sampleBits / 8), true); offset += 4;

// 快数据调整数 采样一次占用字节数 单声道×每样本的数据位数/8

data.setUint16(offset, channelCount \* (sampleBits / 8), true); offset += 2;

// 每样本数据位数

data.setUint16(offset, sampleBits, true); offset += 2;

// 数据标识符

writeString('data'); offset += 4;

// 采样数据总数,即数据总大小-44

data.setUint32(offset, dataLength, true); offset += 4;

// 写入采样数据

if (sampleBits === 8) {

for (let i = 0; i < bytes.length; i++ , offset++) {

var s = Math.max(-1, Math.min(1, bytes[i]));

var val = s < 0 ? s \* 0x8000 : s \* 0x7FFF;

val = parseInt(255 / (65535 / (val + 32768)), 10);

data.setInt8(offset, val, true);

}

} else {

for (let i = 0; i < bytes.length; i++ , offset += 2) {

let s = Math.max(-1, Math.min(1, bytes[i]));

data.setInt16(offset, s < 0 ? s \* 0x8000 : s \* 0x7FFF, true);

}

}

return new Blob([data], { type: 'audio/wav' });

}

开始录音，并调用HTML5音频输入函数，并连接recorder，进行音频录入的开始：

this.start = function () {

audioInput.connect(recorder);

recorder.connect(context.destination);

}

停止录音：

this.stop = function () {

recorder.disconnect();

}

获取音频文件：

this.getBlob = function () {

this.stop();

return audioData.encodeWAV();

}

回放：

this.play = function (audio) {

audio.src = window.URL.createObjectURL(this.getBlob());

}

音频采集：

recorder.onaudioprocess = function (e) {

audioData.input(e.inputBuffer.getChannelData(0));

}

进行HTML5音频采集初始化操作：

H5Recorder.init = function (callback) {

navigator.mediaDevices.getUserMedia({ audio: true })

.then(function (stream) {

var rec = new H5Recorder(stream);

callback(rec);

})

.catch(function (err) {

console.error(err)

})

}

绑定到当前window的全局变量，利于react中进行组件初始化调用：

window.H5Recorder = H5Recorder;

### 4.1.2 语音识别模块

在react组件中设置语言识别状态：

setState (status) {

if( status === 'recording') {

this.recording = true

this.recognizing = false

} else if( status === 'recognizing') {

this.recognizing = true

} else {

this.recording = false

this.recognizing = false

}

}

发送语言到后台服务器进行语音识别：

emitMes() {

var \_this = this;

if (\_this.recognizing) {

return

}

if (! \_this.recording) {

window.H5Recorder.init(function (rec) {

\_this.recorder = rec

\_this.recorder.start()

})

\_this.setState('recording')

} else {

let buffer = \_this.recorder.getBlob()

this.socket.emit('chat message', buffer)

\_this.setState('recognizing')

}

### 4.1.3 语音朗读模块

### HTML5语音Web Speech API: HTML5中和Web Speech相关的API实际上有两类，一类是“语音识别(Speech Recognition)”，另外一个就是“语音合成(Speech Synthesis)”，这两个名词听上去很高大上，实际上指的分别是“语音转文字”，和“文字变语音”。

“语音识别”和“语音合成”看上去像是正反两方面，应该带有镜面气质，实际上，至少从兼容性来看，两者并无法直接对等。

“语音识别(Speech Recognition)”目前的就Chrome浏览器和死忠小弟Opera浏览器默认支持，并且需要webkit私有前缀，如图4-1：（本文不调用此Web Speech API，语音识别有另外的API可寻。）

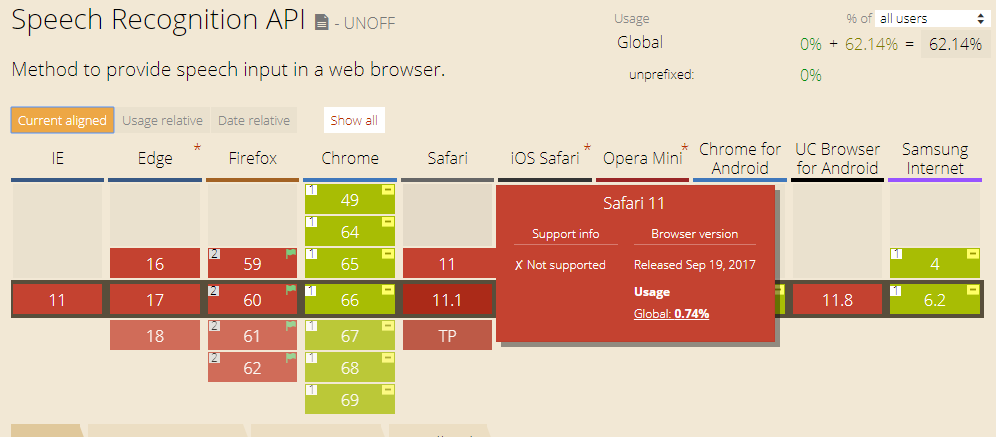


图4-1 Speech Recognition兼容性

“语音合成(Speech Synthesis)”的兼容性要好上太多了，如图4-2：

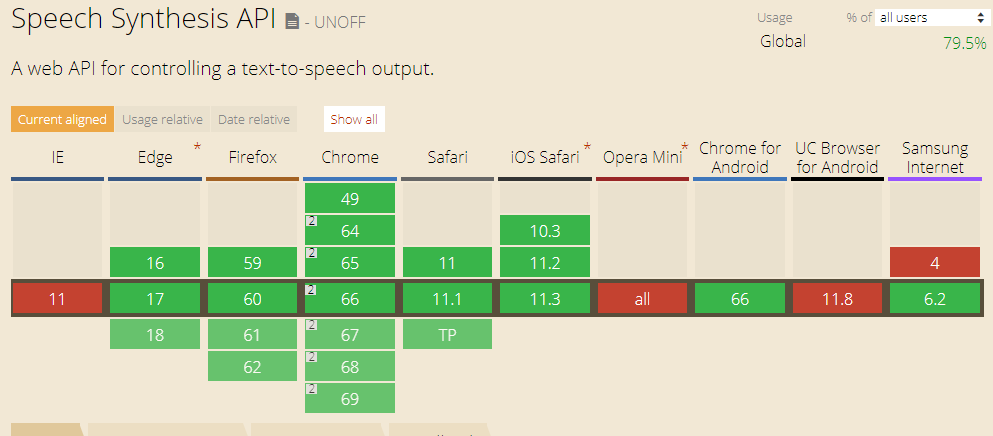


图4-2 Speech Synthesis兼容性

语音朗读模块主要调用Web Speech API的语音合成API，进行语音的朗读：

function synthVoice(text) {

// 声明speechSynthesis对象

const synth = window. speechSynthesis

// 新建语音合成实例

const utterance = new SpeechSynthesisUtterance()

// 设置朗读语音为中文

utterance.lang = 'zh';

utterance.text = text

synth.speak(utterance)

}

并且语音合成需要系统内置TTS模块，如果当前操作系统为精简版操作系统或者系统未安装TTS模块，将导致语音合成失败。

### 4.1.4 登录注册模块

登录注册页面背景采用css3技术绘制动画：

@keyframes change-bac {

50% {

background-image: url(../imgs/login1.jpg) ;

background-size:100% 100%;

background-attachment: fixed;

background-position: center center;

background-repeat: no-repeat;

}

100% {

background-image: url(../imgs/login2.jpg) ;

background-size:100% 100%;

background-attachment: fixed;

background-position: center center;

background-repeat: no-repeat;

}

}

登录页面和注册页面通用一个模块：

class Login extends Component {

render() {

return (

<div className="background">

<div className="login">

<LoginForm history = { this.props.history } />

</div>

</div>

);

}

}

export default Login;

class Login extends Component {

render() {

return (

<div className="background">

<div className="register">

<RegisterForm history = { this.props.history } />

</div>

</div>

);

}

}

export default Login;

登录注册两个页面使用ant.design框架的form表单

采用fetch模块获取数据，并向后台服务器发送数据获取相应：

fetch('http://localhost:8000/login', {

method: "POST",

headers: {

"Content-Type": "application/json"

},

body: JSON.stringify(values)}).then((res) => {

return res.json();

}).then((res) => {

// 返回响应成功 注册成功

if (res.success) {

console.log(res)

openNotification(res.info + '将自动跳转');

setTimeout(() => \_this.props.history.push('/talking'), 2000);

}else {

openNotification(res.info);

setTimeout(() => \_this.props.history.push('/register'), 2000);

}

});

### 4.1.5 聊天室模块

聊天室模块使用react + ant.design设计，前端在模块初始化时向后台获取用户的基本信息，包括聊天室的聊天人数，聊天的历史记录，聊天的搜索内容等等，父组件和子组件之间的通信采用react的props传输，内部状态通过this.state保存：

constructor(props) {

super(props);

this.state = {

talker: ''

}

}

changeTalker(talker) {

console.log(talker);

this.setState({talker: talker})

// this.talker = talker;

}

## 4.2 后端express + websocket设计

### 4.1.1 websocket设计模块

websocket模块采用koa-socket进行开发，它是搭载在koa框架上的一款socket模块，使用koa-socket能实现websocket的全部功能。

开始先在koa上装载app:

io.attach(app)

然后开始监听：

io.on('connection', (socket) => {

console.log('a user connected')

})

再对事件进行监听，以便能及时的收到用户的消息：

io.on('chat message', (socket) => {})

### 4.1.2 语音识别接口模块

语音识别使用baidu-ai的SDK，通过调用baidu-ai的接口，可以很方便的进行语音的智能识别并返回可处理的错误信息。

首先使用之前需要在baidu-ai注册当前的app\_id，通过app\_id可以拿到当前app的api\_key和secret\_key，然后在连接语音识别时注册当前app的客户端：

const client = new AipSpeech(APP\_ID, API\_KEY, SECRET\_KEY)

然后上传到baidu-ai进行语音识别：

client.recognize(voiceBuffer, 'wav', 16000, { lan: 'zh' })

.then(function (res) {

console.log('This is what you say')

console.log(res)

if (res.err\_no === 0) {

let text = res.result[0]

SendChat(text)

.then((res) => {

console.log('This is return message')

console.log(res.text)

console.log(res)

io.broadcast('bot reply', { say: text, replay: res.text })

})

.catch((err) => {

console.error(err)

})

### 4.1.3 机器人回复模块

### 4.1.4 登录注册模块

### 4.1.5 聊天室模块

### 4.1.6 数据库连接模块

## 4.3 MongoDB数据库设计

### 4.3.1 数据库user集合设计

### 4.3.2 数据库聊天内容talking集合设计

# 

# 第5章 科学道德与学风

仅参考此格式进行排版。

学位论文撰写时要遵循科学道德，树立良好学风。本章介绍科学道德与学风中和学位论文撰写密切相关的四个方面，在撰写学位论文时要拒绝科研不端行为，规避科研不当行为，注意科研规范和引文规范。

## 5.1 科学道德与学风问题

科学道德与学风问题是指科技工作者在科研规范、行为准则、治学精神、治学态度、治学风气、治学原则等方面出现的失范现象。反映了现代科研体制在科研活动中存在的问题和漏洞，既有科技工作者精神层面的伦理道德问题，也有行为层面的科研规范问题[3]。

中国科协《科技工作者科学道德规范（试行）》规定的违反科学道德的学术不端行为主要表现为七种类型，具体描述见附录B。

## 5.2 科研不端行为

本节内容均摘自文献[3]，后文未一一引用。

### 5.2.1 科研不端行为的定义

国际科技界将严重违反基本的科学诚信的行为称为科研不端行为(misconduct in science，或称scientific misconduct)，这种行为与科研违规行为、科研越轨行为的内涵十分接近。科研不端行为主要有以下三方面特征：第一，违反科学界通用的道德标准，或严重背离相关研究领域的行为规范；第二，不端行为是蓄意的、明知故犯的或是肆无忌惮的；第三，不端行为不包括诚实的错误或者观点的分歧。

综上所述，科研不端行为的理解如图5.1所示。

2007年，中国科学院发布《中国科学院关于加强科研行为规范建设的意见》，明确将科研不端行为进行定义，并分为以下几类：

1）在研究和学术领域内有意做出虚假的陈述，包括：编造数据；篡改数据；改动原始文字记录和图片；在项目申请、成果申报，以及职位申请中做虚假的陈述。

定义科研不端行为的要素

主观故意

客观危害

行为违反科学界通用道德标准

主要类型：

造假；篡改；剽窃

例外类型：

诚实的错误

其他类型：

科研不当行为

图5.1 科研不端行为的理解

2）损害他人著作权，包括：侵犯他人的署名权，如将做出创造性贡献的人排除在作者名单之外，未经本人同意将其列入作者名单，将不应享有署名权的人列入作者名单，无理要求著者或合著者身份或排名，或未经原作者允许用其他手段取得他人作品的著者或合著者身份。剽窃他人的学术成果，如将他人材料上的文字或概念作为自己的发表，故意省略引用他人成果的事实，使人产生为其新发现、新发明的印象，或引用时故意篡改内容、断章取义。

3）违反职业道德利用他人重要的学术认识、假设、学说或者研究计划，包括：未经许可利用同行评议或其它方式获得的上述信息；未经授权就将上述信息发表或者透露给第三者；窃取他人的研究计划和学术思想据为己有。

4）研究成果发表或出版中的科学不端行为，包括：将同一研究成果提交多个出版机构出版或提交多个出版物发表；将本质上相同的研究成果改头换面发表；将基于同样的数据集或数据子集的研究成果以多篇作品出版或发表，除非各作品间有密切的承继关系。

5）故意干扰或妨碍他人的研究活动，包括故意损坏、强占或扣压他人研究活动中必需的仪器设备、文献资料、数据、软件或其他与科研有关的物品

6）在科研活动过程中违背社会道德，包括骗取经费、装备和其他支持条件等科研资源；滥用科研资源，用科研资源谋取不当利益，严重浪费科研资源；在个人履历表、资助申请表、职位申请表，以及公开声明中故意包含不准确或会引起误解的信息，故意隐瞒重要信息。

### 5.2.2 科研不端行为的表现形式

从表现形式看，有三类科研不端行为，即杜撰、篡改、剽窃(FFP)。

**1）杜撰(fabrication)**

杜撰一般指按照某种科学假说和理论演绎出的期望值伪造虚假的观察与实验结果，从而支持理论的正确性或者确认实验结果的正确性。它表现为对科学和实验结果的不尊重，按照个人主观意愿无中生有，捏造事实。

按照科研的内容和程序分类，杜撰主要分两种：第一，科研申请中的杜撰。主要指在项目资金申请、科研成果申报，以及职位申请等其他科研活动中做虚假的陈述，如杜撰学历、杜撰论文或书刊发表记录、提供虚假获奖证书、文献引用证明，等等；第二，科研过程中的杜撰。主要指在科研过程中，未经过试验、调查，仅根据局部科学现象甚至根本没有根据，凭空编造、虚拟出一些试验数据、结果或事实、证据来作为支持自己论点的论据，证明某理论的正确性。而凭空编造出来的数据或实验结果不具有可重复性，与真实的数据互不兼容。

**2）篡改(falsification)**

篡改，主要是指在科研过程中，用作伪的手段按自己的期望随意改动、任意取舍原始数据或试验使得结果符合自己的研究结论、支持自己的论点。

篡改数据违背了科研规范中的一个基本要求，就是要忠实的记录和保存原始数据。用个人主观意愿对科研结果横加干预，其实验结果必然不具有可重复性。

篡改行为的表现形式主要包括两种：第一，篡改数据，主要指以一些实验结果为基础推测实验结果，对另一些与推测其他结果不同的实验结果、实验记录和图片进行修改。第二，拼凑数据，主要指按期望值主观取舍、任意组合实验结果，或者把与期望值不符的实验结果删除，只保留与期望值一致的实验结果。

**3）剽窃（(plagiarism)**

剽窃是指将他人的科研成果或论文全部或部分原样照抄，并以自己名义发表的欺诈行为。它不仅包括对他人作品字句、内容的直接使用，也包括对他人学术论著的思想、观点、结构、体系等元素作为自己论著的基本元素加以使用并发表的行为。通常表现为不尊重他人学术思想、学术观点，不注明学术思想、学术观点的出处来源而随意使用。

在学术研究中，对已有成果的了解是必需的，对已有成果的借鉴也是不可避免的，因此是否适当引用就成为判断剽窃或借鉴的关键。正确的引用包括两个方面的含义：一是凡借鉴就要引用，引用就要对原出处进行明示；二是引用只能反映研究者对本研究领域已有研究成果的了解和借鉴，或反映已有成果与自己研究的关系，不能构成自己研究成果的主体内容。

## 5.3 科研不当行为

本节内容均摘自文献[3]。

### 5.3.1 科研不当行为的定义

科研不当行为(questionable research practice，QRP)是指虽然违反科学的目的、精神和科学研究事业的基本道德原则，但没有直接触犯明确规定的研究活动的道德底线的行为。

科研不当行为的特征主要为：第一，科研不当行为以明确不违反科学共同体规约为前提，更不是一种违法行为。第二，科研不当行为虽然不是科学共同体规约所明确禁止，但它是不合理的，具有不合理、不公正、不合乎科学道德的特征。

### 5.3.2科研不当行为的表现形式

科研不当行为的表现形式很多，一般来说，科研不当行为主要可分为五种类型：

**1）数据的不当使用**

(1) 根据本能感觉，排除本人认为不精确的观测或数据点；或因匆忙完成项目而偷工减料；

(2)未能在合理的期间内获得重要研究数据；

(3)维持不充分的研究记录，特别是那些用于发表或被他人所依赖的结果；

(4)运用不恰当的统计学或其他计量方法提高研究结论的重要性；

(5)在论文中给出理由的情况下将异常值从数据集中剔除；

(6)为了提高研究的重要性，运用不恰当的统计方法；

(7)窃取供应品、书籍或数据；

(8)操纵实验以获得本人想要的结果；

(9)未经许可复制数据、论文或软件程序；

(10)在合理的期间内，未能保持良好的研究记录或研究数据。

**2）违反科学规则**

(1)忽视材料处理政策的细节（如生物安全、放射性材料等）；运用一个项目的资金完成其他项目；

(2)在人体研究实验中没有报告不良事件；

(3)在研究中不珍惜动物资源；

(4)违反本人所在研究机构的生物安全规定而未尽告知义务，将员工和学生暴露于生物风险之中。

**3）不当的同行关系**

(1)通过与论文研究无重要关联的特殊服务获取署名；

(2)在同事没有对论文作出重大贡献的情况下将其列为作者，以作为人情回报；

(3)为了确保本人成为唯一的发明人，未告知合作者本人申请专利的意图；

(4)未经授权运用他人的想法，或对这种使用未给予应有的感谢；

(5)与同事讨论本人所正在承担的期刊论文审稿工作中获得的保密数据；

(6)规避同行审查程序并通过媒体发布会公布本人的研究结果，而未给予同行足够的时间评估本人的工作；

(7)在文献综述中未能表明在该领域的其他人或相关前期工作的贡献；

(8)妨害他人的工作；

(9)评审他人论文时未经认真阅读即拒绝论文的发表；

(10)在评审工作中做出贬损的评论乃至贬损他人人格；

(11)拒绝同行接触作为已发表论文之支撑的独一无二的研究材料或数据。

**4）不当的师生关系**

(1)基于财、物、性等交易行为许诺学生以更好的成绩；

(2)过度使用、忽略或剥削研究生或博士后的劳动；

(3)提供过于正面或过于负面的推荐信。

**5）基于产出压力的不当科研**

(1)在基金项目的申请中夸大事实，以说服评审人此项目将会对该领域做出重大贡献；

(2)为了应对资助方的压力，修改研究的设计、方法或者结果；

(3)在论文或计划中保留方法或结果的细节；

(4)将推测歪曲为事实或者公布初步的研究结果（特别是在大众媒体上），而未能提供充分的数据使得同行可以评判结果的有效性或重复该实验；

(5)在两种或两种以上不同的期刊上发表相同的论文，而未告知编辑，即一稿多投或一稿多发；

(6)在工作申请或建立中夸大事实；

(7)在资助本人研究的公司中拥有实质性数量的股份而未披露此种经济利益；

(8)故意夸大新药的临床效果以获得经济利益。

## 5.4 科研规范

本节内容均摘自文献[3]。

科研规范（或学术规范）主要是指从事科研活动的行为规范，是以科研道德为基础，以科学共同体为主体，对科研及其相关行为作出的规制性安排。当代科技工作者应坚持的科研规范包括：诚实原则、公开原则、公正原则、尊重知识产权、声明与回避原则。

### 5.4.1 研究数据收集、记录和保存中的规范

数据直接影响到研究成果，因此应当从源头上抓好数据的规范行为。

在数据收集过程中，首先应保证获得数据的条件是真实的，而不是虚构的；其次要确保收集和保存实验数据的完整性；第三，不能为某种目的或获取利益对原始数据进行人为加工和篡改；第四，收集特殊数据应当事先获得授权许可。

数据记录应当与数据的获得同步。数据记录必须精准，必须严格按照有关程序和规则记录数据。

在数据保存方面，第一，应以严谨的方式保存数据。如果是书面记录，就要存放在安全的地方；如果是计算机文件，就应备份，并注意将备份的数据保存在安全处，备份数据应与原始数据分开保存，并且定期为所保存的数据重新备份。第二，原始数据应由产生这些数据的研究机构和科研人员共同保存。第三，要慎重保存涉及机密或危险的数据。第四，应做好数据保存相关事项的预先协议。第五，遵守数据保存期限但不应有意隐蔽数据。

### 5.4.2 研究数据使用中的规范

在未通过发表物或公开宣布研究成果而确立优先权之前，科研人员可以独自使用已经得到确证或有效的数据。一旦科研人员将实验结果公开发表，其他人就可以自由地获取实验涉及到的所有数据，包括最终结果，以便于检验和使用。

在数据使用和处理成图像过程中，首先，应当保证原始数据的真实性，并且保证图像是对数据的真实体现；其次，论著中的数据图像必须是原始记录的完整体现；第三，他人制作完成的数据图像应当在论著中予以说明；第四，应当熟知和合理运用现有相关处理数据的计量方法；第五，应当预先了解拟投稿的相关出版社或期刊的数据和图像处理规范或相关指南；第六，应当了解哪些行为是会受到处罚，以及将会受到怎样的处理。

### 5.4.3 引文的规范

引文的规范可参考中国国家标准局2005年发布并实施的《文后参考文献著录规则》。通常已经发表的论著或文章可以不经作者授权就自行引用，对于未正式发表的资料，未经所有者的许可，不应随意引用。

引用时需要避免的七种行为：

**1）著而不引。**一些作者把原作者的研究进行改头换面，再用自己的语言叙述出来，并当作自己的论述而不注明出处。这种行为虽然在表达上可能是作者自己的话，实际上，作者只是挪用了别人的观点、想法或理念，并不是作者自己的研究，所以是一种剽窃行为。

**2）引而不著。**利用引注或者改写/转述引文，并以之构成自己论著作的主要部分或核心内容，即为引而不著。这种对引注的不恰当或过度使用，也是一种剽窃行为。

**3）有意漏引。**在引用文献综述特定领域的研究、或者佐证自己的研究时，应当公正地涵盖已有的研究，如果为了减少工作量而故意不去查阅一部分文献，或者只选择对自己研究有利的研究，或者为了突出自己研究的意义而不提及某些已有研究，等等，均为有意漏引。这是一种欺骗行为。

**4）过度他引。**引文应当是作者在撰写论著时确实参考或引用过的文献，如果为了给人一种阅读了大量文献资料、研究基础扎实的印象，而故意在论著中加入大量实际没有参考或引用过的、或者与本文论题根本不相干的文献，做不相关引用、无效引用，就是过度他引了。这是一种伪注。

**5）不当自引。**作者撰写论著时，出于提高引用率，或扩大影响等目的，不必引而偏引，进行不必要的过度自我引用。这是一种欺骗行为。

**6）相互引用。**引用应当完全出于学术目的，但有一些作者为了提高彼此的引用率，采取“团体作战”的方式，在小团体之间进行，以提高彼此引用率为目的的相互引用。这样做即使提高了引用率，也是圈内相互消化的结果，并不体现真实的引用率和论文质量。这是一种作假和欺骗行为。

**7）模糊引注。**为逃避被指责为抄袭的可能，一些论著在直接引用了他人的相关文献后，并不标出具体的引文出处，如分册数、页码等，而将它们笼统地放在文后参考文献，从而给人从总体上只是参考了某一文献的印象。

## 5.5 本章小结

介绍了科研不端行为和科研不当行为以及科研规范。

# 第6章 参考文献的标注和要求

仅参考此格式进行排版。参考文献具体格式及排版见后面“参考文献”部分。

学位论文必须要列参考文献，以说明著述内容的科学根据和出处，进而方便读者扩展性阅读的查找。若引用他人成果，应列出引文出处，以尊重他人的科学研究成果[2,15]。

## 6.1 参考文献的重要性

参考文献反映论文作者的科学态度和论文具有真实、广泛的科学依据，也反映论文的起点和深度。方便论文作者与前人的成果区别开来，是对他人劳动成果的尊重。方便读者检索和查找有关资料。有利于节省论文篇幅，有助于科技情报人员进行情报研究和计量学研究。

## 6.2 顺序编码体系

参考文献标著录有如下4种体系：著者－出版年体系（Harvard体系），顺序编码体系，数字字母混合体系和出版年顺序体系[4]。我校本科学位论文的参考文献标注采用顺序编码体系。即参考文献以文献在整个学位论文中出现的次序用[1]、[2]、[3]…形式统一排序、依次列出，置于文中提及的文字末尾的右上角，视引文表标注情况，置于标点符号前或后[4]。

### 6.2.1正文中引用文献的标注方法

正文中引用文献的标示应置于所引内容最后一个字的右上角，所引文献编号用阿拉伯数字置于方括号“[ ]”中，用小4号字体的上角标，引用单篇文献时如例1所示；引用两篇文献时，各篇文献序号置于同一个方括号内，其间用逗号（不是顿号）分开，如例2所示，如果连续序号多于两个以上时，可用范围号“~”（中文还可用全身“—”，外文用en-dash“-”）连接起讫序号，如例3所示；如果文献序号作为叙述文字的一部分，则文献号与正文平排，并且每条文献都要加方括号，如例4和例5所示；如果同一文献在文中不同处被重复引用，全文只在其第一次出现时标应标的序号，以后各处均标这同一序号；若必须标出引文页码，可把页码标在方括号外，如例6所示，也可用其他明确的方式标出。

例1：……，表明已低到2500 m的高度[2]，……。

例2：原位生成的TiB主要有针状或晶须状[21,22]

例3：复杂网络是当今学术界的研究热点[3-6]，早期的研究结果[2,4,6-9] 表明，……。

例4：文献[2]指出，此高度已低到2500 m。

例5：紫色土壤主要分布在我国西南地区（参见文献[11]、[12]、[32]）。

例6：……，表明已低到2500 m的高度[2,35]，……。

不得将引用文献标示置于各级标题处。

### 6.2.2 文后参考文献表的著录方法

按论文中引用的顺序号排列参考文献，不按著者，不分语种。多著者时，著者间用西文逗号隔开，只列前3人，后加“等(et al．)”。在句中，凡是西文符号后应空半格。

参考文献的著录格式严格按照以下形式书写（含标点符号）：

**1）专著**：作者.书名.版本(第1版不著录)[M].出版地:出版者,出版年:引用起止页码.

**2）译著**：作者.书名［M］.译者,译.出版地:出版者,出版年: 引用起止页码.

**3）期刊**：作者.题名［J］.刊名,出版年份,卷(期):起止页码.

**4）会议论文集**：作者.题名［C］// 编者.论文集名.出版地:出版者,出版年:起止页码.

**5）学位论文**：作者.题名:［D］.保存地:保存者,年份.

**6）专利文献**：专利申请者.题名.专利国别,专利号［P］.公告日期或公开日期.

**7）标准**：责任者.标准代号标准名称［S］.出版地:出版者,出版年.

**8）电子文献标注格式**：主要责任者.题名: 其它题名信息[文献类型标志/文献载体标志].出版地: 出版者, 出版年(更新或修改日期)（引用日期）.获取和访问路径.

参考文献著录时还应注意：参考文献中的外文作者名、外文刊名的缩写一律不用缩写点。外文著者一律用姓在名前，采用首字母缩写（中国人用全名不缩写）。姓和名之间不加逗号，名2个以上大写首字母，2名间空一格。文献作者3名以内全部列出，4名以上则列前3人，后加“et al”。各著者间不加“and”、“和”等，应用逗号分开。外文题名第一个单词首字母大写，其余单词（专有名词除外）均不大写。外文刊名应按国际标准规定缩写，不加缩写点。

特别提醒，对西文作者，正文中引用时应遵循西文规范，即名在前姓在后，缩写部分要加缩写点。以参考文献[10]为例，其参考文献条目为：

[10] Atzori L, Iera A, Morabito G. The internet of things: a survey[J]. Computer Networks, 2010, 54(15): 2787 -2805.

正文中引用时的文字为：“L. Atzori等人对物联网的进展进行了综述，指出……”。西文人名乱用、不当使用是常见的写作问题。

通过查非发现，大部分论文参考文献格式都存在各种问题，应该严格规范执行。更多样例见本模板参考文献部分。如有未尽之处，可参看发表的重邮学报论文参考文献标注样例。需要特别说明的是，由于不是所有已发表论文或高年级撰写的学位论文都严格执行了同一规范，请勿将不规范的格式当做模仿对象，以免给自己带来后续问题。

## 6.3 参考文献要求

1）所有被引用文献均要列入参考文献中，必须按顺序标注，但同一篇文章只用一个序号。

2）参考文献数量不得少于15篇，各二级学院应按不同专业提出外文参考文献具体要求，教科书和硕士学位论文不得多于5本,其中外文文献一般不少于2条。参考文献中近三年的文献数一般应不少于总数的1/3，并应有近两年的参考文献。(网上参考文献和各类标准不包含在上述规定的文献数量之内)。

3）教材、产品说明书、未公开发表的研究报告（著名的内部报告如PB、AD报告及著名大公司的企业技术报告等除外）等通常不宜作为参考文献引用。一些未公开发表的内容引用时，可以采用注释如脚注的方式。

4）引用网上参考文献时，应注明该文献的准确网页地址。因为网上文献大都不规范，除特殊情况，原则上不引用，尽量用脚注方式给出。如测试数据来源网址，在正文首次出现时均可直接脚注或正文中给出出处。

5）序号应按文献在论文中的被引用顺序编排。换行时与作者名第一个字对齐。若同一文献中有多处被引用，则要写出相应引用页码，各起止页码间空一格，排列按引用顺序，不按页码顺序。

参考文献内容中文宋体，英文Times New Roman，小4号宋体，1.5倍行距。

Word也提供了参考文献引用的基本功能，但维护一致性比较困难。目前有较多的参考文献管理软件或工具，各有利弊，可根据情况使用，相互交流使用经验，更好地做好文献管理。

## 6.4本章小结

介绍了参考文献的标注方法、著录方法和相关要求。

# 第7章 总结与展望

仅参考此格式进行排版。

学位论文应有结论，可以从论文的主要工作、创新点和后续的研究工作等方面进行总结。

## 7.1 主要工作与创新点

学位论文的结论是最终的、总体的结论，不是正文中各段的小结的简单重复。结论应该观点明确、严谨、完整、准确、精炼。文字必须简明扼要。可以从论文的主要工作、创新点和后续的研究工作等方面总结。

如果不可能导出应有的结论，也可以没有结论而进行必要的讨论。

可以在结论或讨论中提出建议、研究设想、仪器设备改进意见、尚待解决的问题等。不要简单重复罗列实验结果，要认真阐明本人在科研工作中创造性的成果和新见解，在本领域中的地位和作用，新见解的意义。对存在的问题和不足应作出客观的叙述。应严格区分自己的成果与他人（特别是导师的）科研成果的界限。

一般应按四级标题的方式给出，根据需要设置数量。如本文主要工作和创新点如下：

1）阐述第一个创新工作。不要把阅读文献当成创新工作。

2）阐述第二个创新工作。

3）阐述第三个创新工作。

4）阐述第四个创新工作。

特别提醒，不应简单和中文摘要内容相互拷贝。同一段文字或句子在本文中原则上只出现一次。

## 7.2 后续研究工作展望

针对工作不足或问题，说明更下一步深入的研究。如内容较多，也应用四级标题方式列出。

# 参考文献

参考文献是毕业设计(论文)不可缺少的组成部分，它反映论文作者的科学态度和毕业设计(论文)的取材来源、广博程度和可靠程度，同时能方便地把作者的研究成果与他人的成果区别开来。一份完整的参考文献也是向读者提供的一份有价值的信息资料。参考文献应列入主要的中外文献，数量不少于15篇，教科书**和硕士论文**不多于5本。按论文中参考文献出现的次序，用中括号以数字连续编号，格式参考毕业设计（论文）模版。

1. 中国标准研究中心等. GB/T7713.1-2006.学位论文编写规则[S]. 北京:中国标准出版社, 2006.
2. 汪继祥. 作者编辑手册[M]. 北京:科学出版社, 2004: 109, 118, 146-148.
3. 全国科学道德和学风建设宣传教育领导小组. 科学道德与学风建设宣传参考大纲(试用本). 北京: 2011.
4. 中国标准研究中心等. GB/T7714-2005. 文后参考文献著录规则[S]. 北京: 中国标准出版社, 2005.
5. 李娜芬.障碍环境中Swarm突现计算模型研究及行为控制:[D]. 重庆: 重庆邮电大学, 2013.
6. 谢希仁. 计算机网络(第五版）[M]. 北京: 电子工业出版社, 2008: 349-352.
7. Thomas L Floyd. Digital Fundamentals (Seventh Edition) [M]. 北京: 科学出版社, 2002: 349-352.
8. Robert D. Thompson. 数字电路简明教程[M]. 马爱文, 赵霞, 李德良等. 北京: 电子工业出版社, 2003: 449-452.
9. 吕学勤, 陈树果, 林静. 求解0/1背包问题的自适应遗传退火算法[J]. 重庆邮电大学学报(自然科学版), 2013, 25(1): 138-142.
10. Atzori L, Iera A, Morabito G. The internet of things: a survey[J]. Computer Networks, 2010, 54(15): 2787 -2805.
11. Mohammed M，Kasim A. Multi-Criteria Vertical Handover by TOPSIS and Fuzzy Logic[C]//2011 International Conference on Communications and Information Technology (ICCIT). [s. l.]：IEEE Press, 2010: 97-102.
12. AgrawalL R, Gehrke J, Gunopulos D, et al. Automatic subspace clustering of high dimensional data for data mining applications[C]//In Proceedings of the 1998 ACM SIGMOD. Seattle: ACM Press, 1998:94-105.
13. 伏梦盈. 基于博弈论的协作通信中继节点选择:[D]. 长沙: 湖南大学, 2012.
14. 工业与信息化部电信研究院. 中国物联网白皮书[R]. 201:1 -3
15. 中国标准研究中心等. GB/T7713.1-2006. 学位论文编写规则[S]. 北京: 中国标准出版社, 2006.
16. 陈国平, 张百珂, 马耀辉. 一种基于蓝牙技术的手机防盗防遗失报警方法. 中国, 201110369930.0[P]. 2014.
17. MagicBoy, 李鹏. GPS定位基本原理浅析: Mobile On Lion镜像站点[OL], 2010-12-09 (2015-01-27). http://www.cnblogs.com/magicboy110/archive/2010/12/09/1901669.html.
18. 工业和信息化部关于电信服务质量的通告(2014年第1号）[OL].(2014-03-03). http://www.miit.gov.cn/n11293472/n11293832/n11293907/n11368223/15864477.html.

# 致谢

致谢二字一级标题：黑体3号字居中，段前17磅，段后16.5磅，1.5倍行距，致谢二字与致谢内容之间不空行。致谢内容正文样式：宋体小四号，1.5倍行距。

可以从下列方面致谢：协助完成研究工作和提供便利条件的组织或个人；在研究工作中提出建议和提供帮助的人；给予转载和引用权的资料、图片、文献、研究思想和设想的所有者；其他应感谢的组织或个人。

主要感谢导师和对论文工作有直接贡献及帮助的人士和单位。学位申请人的家属及亲朋好友等与论文无直接关系的人员，一般不列入致谢的范围。

致谢辞应谦虚诚恳，实事求是，切忌浮夸与庸俗之词。

# 附录A 科技写作中非学术性低级错误的主要表现

对于一些不宜放入正文中，但作为毕业设计（论文）又不可缺的组成部分或具有重要参考价值的内容，可编入毕业设计(论文)的附录中，例如，公式的推演、源程序代码、附图等内容。附录的内容为备选项目，作者可根据内容的需要决定附录的项目数，用附录A、附录B方式编号。附录的篇幅不宜太多。附录与主体部分一起编制页码。若附录部分有手工制作或复印件，手工制作或复印件部分要装订在内但可以不计页码。附录的文字按照正文格式进行排版。

本附录主要针对学位论文写作或中文科技论文写作，供重庆邮电大学学位论文查非工作参考。未尽事宜，可参考重庆邮电大学论文写作要求、重庆邮电大学学报编辑部等国内期刊社、出版社的通用出版规定。

推荐阅读《科学出版社作者编辑手册》、《科学道德与学风建设宣传参考大纲（试用本）》等写作指导性书籍或资料，可了解更多、更详尽的通用写作出版规范。

| 类别 | 细目 | 主要表现 | 常见问题描述 |
| --- | --- | --- | --- |
| 基本语言规范 | 字词 | * 错字 * 别字 * 漏字 * 赘字 | * “的”、“地”、“得”不分。如将“为了更好地验证本算法的效”误为“为了更好的验证本算法的效果”。 * 拼音等输入法录入错误同音或同形词。 * 将硕士学位论文翻译为paper（应为thesis）。 * 常识错误。比如把地区称为国家。 * 数字中的小数点错误。 |
| 句法 | * 病句 * 错句 * 中文式英文 | * 句子缺乏主语。比如用短语当句子。 * 一句话内前后主语不一致，逻辑混乱，可读性差。 * 按中文习惯翻译英文摘要。比如直接在英文摘要中用paper等当主语。 * 用机器软件翻译中文摘要导致的各种字、词、句低级错误。 |
| 标点  符号 | * 该用不用 * 乱用 * 误用 | * 不断句。比如一长段话仅一个句号。 * 引用他人文字没有使用引号（将导致学术不当行为）。 * 句后缺乏句号。 * 括号、引号等符号没有成对使用，正确配套。 * 用“如下”等引出下文时缺乏冒号。 * 乱用符号。比如在句字中随意打逗号分割断句，或该用句号的用逗号。 * 破折号“——”与连接号“－”混用。 |
| 语种 | * 中西文语言错误混用 * 中西文符号错误混用 | * 中文论文中提供的图、表、公式中错误出现英文文字。 * 中英文标点符号错误混用。如英文摘要中出现中文逗号；公式编号等处的阿拉伯数字编号用中文括号配套。 |
| 通用内容规范 | 一致性 | * 前后内容不一致 | * 多处出现的对象、变量、名称等前后不统一（包括同一参数在图、表中与在正文中的不一致，同一参数大小写的不统一）。 * 中西文摘要的内容、关键词等不一致。 |
| 完备性 | * 文档结构缺项 * 内容缺乏 | * 中文摘要后缺乏关键词等文档要素。 * 首次出现变量、缩写、专业名词等未说明含义。 * 摘要、正文的内容分别不能自解释。比如摘要中含引用论文编号或变量名、缩写等不明信息。 * 变量、数字出现时缺乏名称、量纲说明。 * 出现图、表时缺乏相应说明。 |
| 相关性 | * 出现无关内容 * 内容和形式不相关 | * 有隐藏的无关信息。 * 文本内容被故意图形化。 * 图、表混用。应用表说明的误用图，或表目与表身分界不清晰。 |
| 非必要冗余 | * 内容多处重复 | * 一段文字或句子在摘要、正文等处出现不止一次。 |
| 专业写作规范 | 文字 | * 口语化 * 不遵循行业惯用语要求 * 违反保密等规定 | * 写作中使用口语。 * 硕士学位论文摘要、正文等地用“我们”或“we”当主语。 * 随意更改行业惯用语。如用小写英文字母表示行业约定的缩写。 * 涉及隐私、保密等专业要求时未按规定处理。如保密项目名称直接明文出现。 * 数量转行。如同一数量，一部分在上一行，部分在下一行。 * 公式转行不规范。公式通常在加、减、乘运算符号处转行，如果是乘运算应在上一行末加点乘号。如果是矢量运算应在合适的地方转行或附加说明。 |
| 算法、程序 | * 不完备 * 不规范 | * 使用伪码描述算法时不规范。 * 算法、程序中出现未解释或注释的内容。 |
| 图表、公式 | * 类型混乱 * 内容缺乏 * 质量差 * 编号错误 * 放置位置错误 | * 未按行业规范的图形类型制图。如功能模块图混合系统流程图，动态、静态图不分；程序流程图出口众多。 * 随意编造图形元素，错误制图。如箭头流转混乱；图上的线条无法区分，随意交叉。 * 图元素缺乏。如判断框分支标识不全。 * 未按要求插入矢量图，截图质量差，不清晰。 * 有错误或多余信息。如从word中截取图表，图上还有回车符号。 * 无坐标单位，无图例。 * 图、表格式、风格不一致，比如字体、字号、线条粗细等。 * 图中字太小，难以辨认。 * 图名在图之前，或图出现在正文说明之前，甚至缺少相应图、表说明。 * 表名在表之后。 * 公式编号混乱，或编号中出现“公式”字样。 |
| 摘要和结论 | * 不规范 * 总结归纳不够 | * 摘要写作不规范，不能对论文进行高度概括。 * 没有对取得的成果进行提炼，仅仅是重复描述所做的工作，没有与绪论相呼应。 |
| 参考  文献 | * 格式错误 * 要素缺乏 | * 未按相关的规范整理格式。 * 作者姓和名顺序混乱（应姓在前，名在后），未按要求缩写其名、缩写名未大写、或乱加缩写点（应不加）。 * 缺乏卷期、出版地、页码、文献类型等要素。 * 拼音、西文中的大小写、标点符号未按要求统一。 * 多处格式细节不统一（并非对标准理解不同导致）。 * 文后参考文献未按照在文中引用的先后顺序排列。 |
| 引用 | * 引用错误 * 引用不当 * 该引不引 | * 章节标题上加引用符号。 * 使用他人的公式、图、表未明确给出引用出处（将导致学术不当行为）。 * 使用相对引用时出错。如“见下图”时，文字下并没有图，已经被排版到别的地方。 * 正文中参考文献、公式、图表等的引用关系混乱。如错位引用。 * 正文中提及参考文献作者时，未按原语种惯例排列其姓和名的顺序（中文应姓前名后，英文反之），西文缩写名的字母未大写或其后未加缩写点。 |
| 编辑出版规范 | 模版 | * 不符合标准格式 * 格式不统一 * 格式错误 | * 随意更改全文字体、字号、行距、对齐方式等。 * 全文各处格式不一致。如一个图、表内的字号、字体不统一，忽大忽小；如一个变量在正文里以公示出现时随意缩放大小，明确区别于正文文字，或影响正常行距。 * 变量等格式错误。如正体、斜体、粗体等未正确使用（如向量、矩阵未用粗斜体）；变量下标未编辑为下标；变量无量纲、无单位或单位错误。 * 章节、图表、公式、参考文献等未按要求编号。如学位论文中公式、图表等要求按章独立编号。 * 公式、图表的编号不连续。 * 格式细节错误。如公式编号未右对齐，同时公式本身居中；要求居中的标题等内容而未居中；在设置变量斜体、粗体时把前后括号、标点符号一律包含在内。 * 参考文献格式不按模板要求，信息不全。 |
| 版面 | * 版面混乱 * 行距不一 | * 文中随意“开天窗”，出现不合理的空白行（如3-5行以上）并严重影响版面完整性。 * 正文和引用的图表距离太远。 * 图、表的标题和其内容分离在不同页面，被粗暴截断。 * 标题出现在页面的最后一行。 |
| 其他要求 | 特别  规定 | * 未执行 * 执行不严格、完整 | * 盲审论文未按要求隐去相关信息。如正文、致谢、成果列表等中故意留有个人信息。 * 保密论文提供方式违规。 |

# 附录B 英文翻译

指导教师制定与专业相关的外文文献内容，由学生独立翻译成中文，其外文文献内容不得少于3000字符。译文和原文附于附录部分，按照正文格式进行排版。若原文没有电子版只是原版的复印件，复印件装订在内可不计页码。