



# 南京大學

## 研究生畢業論文

### (申請工程碩士學位)

論文題目 一種基於某最新框架的Hammer模塊砸碎Nut的方法

作者姓名 張三

學科、專業名稱 工程碩士（軟件工程領域）

研究方向 軟件工程

指導教師 李四 職稱

20xx 年 x 月 xx 日

学 号 : MFxxxxxxx  
论文答辩日期 : 20xx 年 x 月 xx 日  
指 导 教 师 : (签字)



# Using a Hammer to Crack a Nut

By

**San Zhang**

Supervised by

Advisor Title Si Li, Advisor Title Wang Wu

A Thesis

Submitted to the Software Institute

and the Graduate School

of Nanjing University

in Partial Fulfillment of the Requirements

for the Degree of

**Master of Engineering**

Software Institute

May 20xx



# 南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目： 一种基于某最新框架的Hammer模块砸碎Nut的方法  
工程硕士（软件工程领域） 专业 20xx 级硕士生姓名： 张三  
指导教师（姓名、职称）： 李四 职称

## 摘 要

这部分是中文摘要。

注意：本模板使用的是PDFLaTeX编译的，这一编译的好处在于速度快，并能直接引用pdf格式的图形。

以下展示列举（无编号）：

1. 贡献1。
2. 贡献2。
3. 贡献3。

### 常见问题与解答

1. 编写论文时建议拷贝一份论文模板的副本，原模板内的文字是模板使用说明，以及常用LaTeX格式的用法（表格、图形、论文引用、公式、算法等），方便各位同学参考。
2. LaTeX模板并不与Word模板完全一致，也无需与Word模板的格式一致，使用本模板即遵照本模板的要求。
3. LaTeX已经非常成熟，通过搜索引擎可以解决绝大部分问题，模板内也提供了丰富的手册供参考。
4. 推荐使用Windows下的CTeX环境。CTeX是一个编辑器（WinEdt）加编译器（MiKTeX）的整合包，安装完成之后请务必升级MiKTeX（WinEdt或开始菜单均有升级入口）。Linux或Mac用户可以选择TeX Live（编译器，包含一个简单的编辑器）加TeXStudio或texmaker（编辑器）。

关键词： 中文，关键，字



## 南京大学研究生毕业论文英文摘要首页用纸

THESIS: Using a Hammer to Crack a Nut

SPECIALIZATION: Software Engineering

POSTGRADUATE: San Zhang

MENTOR: Advisor Title Si Li, Advisor Title Wang Wu

### **Abstract**

This is English abstract.

以下展示的是圆点列举（无编号）做些修改：

- First Contribution.
- Second Contribution.
- Third Contribution.

**Keywords:** English, Keywords





# 目 录

表目录 .....	ix
图目录 .....	xi
<b>第一章 项目背景与意义 .....</b>	<b>1</b>
1.1 项目背景 .....	1
1.2 相关系统研究现状 .....	2
1.3 本文主要的工作 .....	2
1.4 本文组织结构 .....	2
<b>第二章 技术综述 .....</b>	<b>3</b>
2.1 模块化介绍 .....	3
2.2 Spring技术介绍 .....	3
2.2.1 依赖注入和控制反转 .....	3
2.2.2 SpringBoot技术介绍 .....	4
2.2.3 Eureka技术介绍 .....	4
2.3 Oracle技术介绍 .....	4
2.4 Vue框架介绍 .....	4
2.4.1 Element技术介绍 .....	4
2.5 Tesseract技术介绍 .....	4
<b>第三章 系统需求分析与概要设计 .....</b>	<b>5</b>
3.1 项目总体规划 .....	5
3.1.1 系统合理性概述 .....	5
3.1.2 涉众分析 .....	5
3.2 系统功能性需求分析 .....	6
3.2.1 元数据管理模块功能需求 .....	6
3.2.2 发送规则模块功能需求 .....	6

3.2.3	任务调度管理模块功能需求 .....	6
3.2.4	文件处理模块功能需求 .....	6
3.2.5	文件路由模块功能需求 .....	7
3.2.6	文件发送模块功能需求 .....	7
3.3	系统非功能性需求分析 .....	7
3.4	系统外部接口需求分析 .....	7
3.5	系统概要设计 .....	7
3.5.1	系统总体架构设计 .....	7
3.5.2	数据库设计 .....	8
<b>第四章</b>	<b>系统详细设计与实现 .....</b>	<b>13</b>
4.1	元数据管理模块详细设计与实现 .....	13
4.2	发送规则模块详细设计与实现 .....	13
4.3	任务调度管理模块详细设计与实现 .....	13
4.4	文件路由模块详细设计与实现 .....	13
4.5	文件处理模块详细设计与实现 .....	13
4.6	文件发送模块详细设计与实现 .....	13
<b>第五章</b>	<b>文件自动发送平台测试 .....</b>	<b>15</b>
5.1	测试环境 .....	15
5.2	功能测试 .....	15
5.3	对外接口测试 .....	15
5.4	本章小结 .....	15
<b>第六章</b>	<b>总结与展望 .....</b>	<b>17</b>
6.1	总结 .....	17
6.2	进一步工作展望 .....	17
<b>第七章</b>	<b>标题 .....</b>	<b>19</b>
7.1	这是节标题 .....	19
7.1.1	这是小节标题 .....	19

<b>第八章 正文</b> .....	<b>21</b>
8.1 正文书写的小技巧 .....	21
8.2 一些正文中的标记 .....	21
8.3 注意软换行的使用 .....	21
<b>第九章 表格</b> .....	<b>23</b>
9.1 表格与表格引用的基本概念.....	23
9.2 基本表格 .....	23
9.3 表格单元跨列 .....	23
9.4 表格单元跨行 .....	24
9.5 表格与图形位置.....	24
<b>第十章 图形</b> .....	<b>25</b>
10.1 基本图形 .....	25
10.2 引用代码 .....	26
<b>第十一章 公式</b> .....	<b>27</b>
11.1 公式5.1与论证 .....	27
11.2 公式5.2与论证 .....	27
11.3 公式5.3与论证 .....	28
<b>第十二章 算法</b> .....	<b>29</b>
<b>第十三章 论文引用</b> .....	<b>31</b>
13.1 引文相关 .....	31
13.2 页眉测试 .....	32
<b>参考文献</b> .....	<b>33</b>
<b>简历与科研成果</b> .....	<b>35</b>
<b>致谢</b> .....	<b>37</b>



## 表 目 录

3.1	代理商表结构 .....	9
3.2	发送规则表结构 .....	9
3.3	任务运行表结构 .....	9
3.4	文件处理表结构 .....	10
3.5	传真发送表结构 .....	10
3.6	邮件发送表结构 .....	11
9.1	实验系统中函数调用与数据依赖的交集 .....	23
9.2	错误率与函数特征之间的关联 .....	24
9.3	五个实验系统概述 .....	24



## 图 目 录

3.1 模块关系图 .....	8
4.1 代理商POJO操作代码 .....	13
10.1 以含错误的RTM为输入的五系统上三个实验（Call, Data, Call+Data）的错误率（Incorrectness） .....	25
10.2 VoD系统中的代码片段 .....	26
13.1 在dblp上下载Bibtex .....	31
13.2 Bibtex详细信息 .....	32





## 第一章 项目背景与意义

### 1.1 项目背景

随着办公自动化的兴起以及对自然环境的保护需要，人们更多的使用计算机来数字化地创建、收集、存储、处理，并传播完成办公室任务所需的信息。由于信息的数字化，任务相关人员能够更加方便快捷地共享信息，高效地协同工作，极大地优化了以往复杂、低效的办公室工作过程[1]。以微软的office套件为典型代表，一系列的优秀办公软件应运而生。现代企业间存在大量文件传输的需求。通常文件会伴随有下面的一些特点：文件类型相对固定，办公的文件除了行业特有文件类型外，通常以PDF，Excel，Word文件居多；发送方式相对固定，主流的方式一般有传真和邮件发送；发送时间相对固定，多数文件需要每日定时进行发送，可能是交易日或自然日。比如对于中国的基金公司，按照基金的管理规定，公司的产品主要是由银行等金融机构来进行代理销售的，公司需要每天和金融公司确认前一天基金的申购、赎回、转换的金额，这里以PDF和Excel文件的格式居多，邮件或传真的方式发送给金融机构。随着业务的发展需要，基金公司合作的代销商通常都有上百个，意味着公司每天需要发送几百个文件。还有金融公司受国家银监会，证监会监督，每天需要定时上报大量文件。这些文件数量众多，人工发送繁琐，容易出错，公司都会开发一些小工具来达到自动发送的目的。但是公司不同部门对于文件的发送时间，文件类别，发送方式要求多样，各自开发造成了系统的繁杂和臃肿。需要开发一个平台对文件发送进行统一管理。

基于以上背景和实际需求，本文详细的介绍了从公司内已有接口为基础，使用微服务架构，设计实现了一个适合公司统一发送文件的平台。该平台隐藏了个文件格式的细节差异，集成了最常使用的传真和邮件发送方式。提供了文机的定时，生成，上传，解析，加密，确认，发送，监控等功能。方便用户对发送文件的生命周期进行管理。开放了发送的API接口方便开发人员的调用，免去了文件处理的细节，提供了额外的功能封装。通过该平台可以满足大部分公司的文件发送需求，免去了大量重复繁琐的工作，提高了文件发送的准确性和易用性。

文件发送平台对邮件传真接口进行整合，提高了基础接口的易用性，免去了开发人员使用接口前需要对文件的处理。对软件系统来说，兼容了多种文件和发送方式，减少系统的重复建设。对使用人员来说，提供了简单，明确的页

面操作，降低了使用难度，支持通用的发送规则，降低了人力成本。

## 1.2 相关系统研究现状

今年来，随着无纸化办公的逐步推进，以微软的office工具作为对于提高生产力工具的开发也

## 1.3 本文主要的工作

本文主要对兴全基金文件自动发送平台的设计和实现过程进行阐述。首先，介绍该平台的开发背景，然后简单介绍了开发过程中所用到的相关技术，继而进行业务描述，从功能性和非功能性两方面对该平台进行需求分析，规范了系统边界，简述总体设计和主要模块，最后详述了各模块的逻辑并展示该平台的实际运行效果。

## 1.4 本文组织结构

## 第二章 技术综述

本文设计和实现的文件自动发送平台使用SpringBoot为主体框架进行开发，根据功能不同将平台分为多个模块进行开发，每一个模块单独放在一个包中，公共组件类放入公共包中。底层数据库使用金融系统常见的Oracle数据库。采用SpringCloud中的Eureka提供对外接口。前端使用Vue进行开发，UI使用ElementUI框架。采用Tesseract进行OCR识别PDF文字。

### 2.1 模块化介绍

模块化设计的历史非常悠久，覆盖范围广泛。不仅广泛应用于UI设计、软件开发之中，在报纸版面、室内设计、机械制造、土木工程中也无处不在。软件设计中的模块化设计是指将系统整理根据一定的原则划分为多个小块，称之为模块。模块之间通过定义良好的接口进行交互，合理有序的组织整个系统。每个模块独立进行设计开发，可以并行的推进系统的开发进度，更合理的分配人员，提升工作效率。模块可以提高软件的复用性，职责明确，功能单一的模块可以方便的供给开发人员调用，不需要开发人员对相同的功能进行多个设计，保证了设计的一致性，降低了成本，提升效率。依照原则划分的模块通常都符合高内聚，低耦合的原则。模块间相对独立，便于升级和维护，移植性高。

### 2.2 Spring技术介绍

Spring框架是一个基于Java平台，为开发Java应用程序提供全面的基础架构支持。Spring负责处理基础结构，开发人员负责业务逻辑。Spring从POJO来构建应用程序，并将业务非侵入性地应用与POJO。

#### 2.2.1 依赖注入和控制反转

Java广泛应用于互联网应用的开发，这些应用包含了大量的对象，对象之间相互合作，相互依赖。但Java缺乏将松散对象组合为整体的有效方法。可以通过诸如工厂模式、抽象工程模式、建造者模式、组合模式等设计模式来组织应用的类和实例。但是，设计模式只是给出了模式的定义，应用场景，和如何设计，实际开发中仍然需要开发人员自己实现他们。

Spring框架的IOC通过另外一种方法将多个对象组织为一个完整应用。

### **2.2.2 SpringBoot技术介绍**

### **2.2.3 Eureka技术介绍**

## **2.3 Oracle技术介绍**

## **2.4 Vue框架介绍**

### **2.4.1 Element技术介绍**

## **2.5 Tesseract技术介绍**

## 第三章 系统需求分析与概要设计

### 3.1 项目总体规划

#### 3.1.1 系统合理性概述

证券投资基金是证券市场发展的必然产物，在发达国家已有上百年的历史。一些人认为早在1822年荷兰国王威廉姆一世就创立了私人信托投资基金。也有人倾向1868年英国成立的“海外及殖民地政府信托基金”看作最早的基金。中国基金从1992年开始开始发展，基金规范化运作不断提高，法律体系日益完善，基金品种日益丰富，基金规模持续增长。

因为监管要求，基金公司不直接持有投资人基金，投资人的基金多是商业银行代为保管。另外银行银行，证券公司还负责一部分基金的销售。伴随着业务的高速增长，各基金公司合作的商业伙伴也成倍增长，每日需要将业务情况进行汇总，定时向合作公司发送报表，账单，文件。人工发送文件效率低，浪费时间。文件内容多涉及商业机密，手工发送容易出错。开发的小工具适应范围小，功能各有高低，重复建设严重。需要将公司内部已有服务进行整合，开发一个文件发送平台，兼容当前的多种文件格式和发送需求，统一进行文件发送管理，降低使用和开发成本。

本项目整体前后端分离，后端使用SpringBoot进行搭建，前端使用Vue和ElementUI搭建。选用Java编程语言。传统Web项目开发效率低，代码配置多，构建部署复杂，随着Spring框架的发展，具由开箱即用，约定由于配置的SpringBoot成为了快速开发的主流框架。并且和公司当前的技术体系一致。

Vue是轻量级的框架，支持双向数据绑定，对控件的交互性能好，学习成本低，易于和其他框架结合使用。Java是Web开发主流的语言，虚拟机运行，不依赖特定平台；支持多线程，性能高；语言成熟安全，有大量参考资料。Java的接口与运行绑定给系统提供了一定的扩展性，有利于方便以后业务的新增。

#### 3.1.2 涉众分析

文件自动发送平台的使用人群主要是公司的业务人员和开发人员。业务人员需要将导出的大量报表发送给相应的合作商，开发人员可以调用封装的发送API，免去繁琐的文件处理。

## 3.2 系统功能性需求分析

本文中，文件自动发送平台主体可以分为元数据管理模块，发送规则模块，任务调度管理模块，文件处理模块，文件路由模块和文件发送模块

### 3.2.1 元数据管理模块功能需求

元数据管理模块主要功能为提供联系人的展示和修改，联系人主要是从其他表中由运维人员手动导入，本系统只需提供联系人的修改和展示功能。

### 3.2.2 发送规则模块功能需求

通常的业务逻辑是用户从另外的系统导出报表或者手动生成好报表文件，然后把文件放到机器的共享文件下，由系统约定的时间去读取约定的文件夹下的文件，根据文件名的格式确定收件人进而发送。规则模块就是用户和系统约定文件名的格式，选择文件发送频率、发送方式，文件是否需要加密，是否约定发送时间，发送前是否需要用户点击确认以及生成调度任务等功能。

### 3.2.3 任务调度管理模块功能需求

本模块负责不间断轮询未完成的任务，协调任务在不同模块间推进，管理任务的生命周期。任务调度的任务主要分为三类，一种是由规则定期生成的任务，第二种是用户使用前端页面提交的任务，第三种是开发人员使用API提交的任务。这些任务的初始状态不一，规则生成的任务文件通常还未生成，用户前端页面上传的文件是压缩包，API调用传递的文件是base64字符串。不同文件的放松方式可能有传真、邮件，或者二者皆有。任务调度模块负责解析任务当前状态，合理调用文件相关接口和发送相关接口处理任务，推进任务进行、中断任务或者重置任务的状态。另外还需要提供前端展示任务的数据接口。

### 3.2.4 文件处理模块功能需求

业务的报表文件存在相当一部分是从另外的系统中导出的，用户希望能将这一部分文件生成的功能也集成到平台中。不同类型的报表中的格式通常不一致，为了能够支持多种文件，文件生成模块定义了一组文件相关的接口，主要涉及文件的生成，转化，加密。文件的生成是基础功能，转化是为了适应传真的底层接口参数需要，加密则是使用邮件发送时存在对附件文件加密的需求。开发人员通过实现这组接口来支持不同类型的文件。此外还需要支持文件的下载功能，方便用户下载自动生成的文件，查看文件的正确性后决定是否发送。

### 3.2.5 文件路由模块功能需求

对于未明确指定文件接收人的文件，识别文件内容，分析文件接收人。

### 3.2.6 文件发送模块功能需求

公司已有邮件服务器和传真服务器，但是二者的参数差异大，并且对发送重试的支持不友好。文件发送模块主要提供了传真、邮件、短信服务的封装，支持发送任务的错误重试。

## 3.3 系统非功能性需求分析

易操作性：使用交互性良好的Vue前端框架，尽可能简化操作页面，降低公司业务人员的使用成本。

可扩展性：文件处理特别是其中的文件生成具有特定的业务场景要求，定义一组规范的文件接口，使得系统能具备扩展性，支持导出新的文件内容而不影响现有业务。

## 3.4 系统外部接口需求分析

文件自动发送平台提供对常见文件类型的传真和邮件的支持，开发API接口供开发人员调用。

## 3.5 系统概要设计

### 3.5.1 系统总体架构设计

早期公司已经搭建有邮件服务器，开发人员通过SMTP协议即可发送邮件，通用性较好但每次使用都需要自己构造MimeMessage对象，没有开发专用发送文件的接口，期望平台能够提供直接传入文件的接口。传真服务器的参数要求传入图片文件的base64编码的字串，我们通常使用的PDF和Excel都不能直接使用接口，需要将他们转换为tif格式的图片再使用base64进行编码。此外，传真服务未提供失败重试功能。

针对以上需求，本文所介绍的文件自动发送平台采用模块化开发的方案，将系统分为元数据管理模块，发送规则模块，任务调度管理模块，文件处理模块，文件路由模块和文件发送模块。模块化是一种处理复杂系统分解为更好的可管理模块的方式，对业务进行模块化拆分后，可以使各业务模块间业务逻辑清晰，功能单一，提高开发效率。

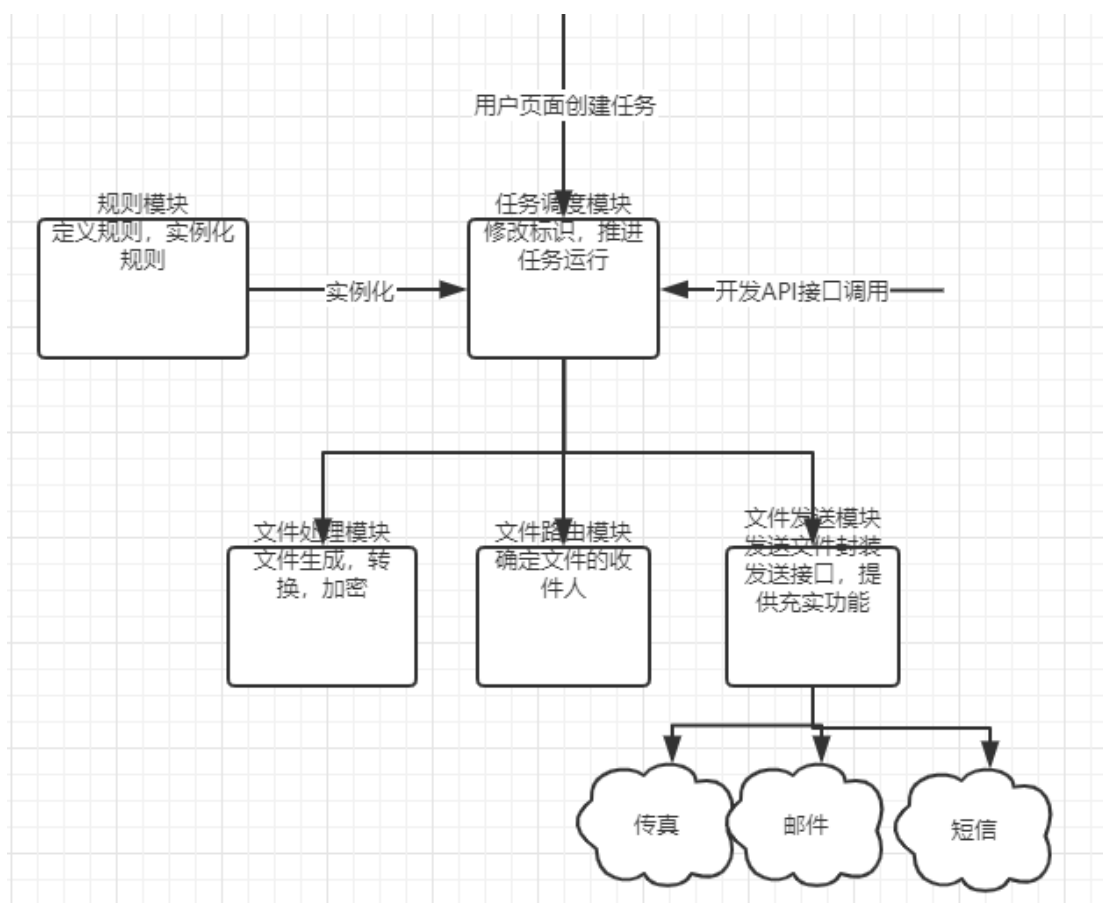


图 3.1: 模块关系图

各个模块之间的关系如图 3.1 所示

### 3.5.2 数据库设计

系统主要涉及的数据表有代理商信息表T\_TA\_AGENCY\_INFO\_RECORD (表 3.1)、发送规则表T\_TA\_RULE\_RECORD (表 3.2)、任务运行表T\_TA\_TASK\_RECORD (表 3.3)、文件处理表T\_TA\_FILE\_RECORD (表 3.4)、传真发送表T\_TA\_SEND\_FAX\_RECORD (表 3.5)、邮件发送表T\_TA\_SEND\_MAIL\_RECORD (表 3.6)、



表 3.1: 代理商表结构

字段名	类型	备注
VC_AGENT_NO	VARCHAR2 (20)	销售商编号
VC_AGENT_NAME	VARCHAR2 (100)	销售商名称
VC_AGENT_TEL	VARCHAR2 (20)	销售商手机号
VC_AGENT_MAIL	VARCHAR2 (60)	销售商邮箱
VC_AGENT_FAX	VARCHAR2 (20)	销售商传真号

表 3.2: 发送规则表结构

字段名	类型	备注
L_RULE_ID	NUMBER(10,0)	规则ID
VC_RULE_TYPE	VARCHAR2 (20)	规则类型
SEND_TYPES	VARCHAR2 (20)	发送类型
C_FILE_ENCRYPTED	Bit (1)	是否文件加密
C_AUTO_SEND	Bit (1)	是否自动发送

表 3.3: 任务运行表结构

字段名	类型	备注
L_TASK_ID	NUMBER(10,0)	任务ID
D_TASK_DATE	VARCHAR2 (20)	任务时间
VC_TASK_NAME	VARCHAR2 (20)	任务名称
L_RULE_ID	NUMBER(10,0)	规则ID
L_MAIL_ID	NUMBER(10,0)	邮件ID
L_FILE_ID	NUMBER(10,0)	文件ID
L_FAX_ID	NUMBER(10,0)	传真ID
TASK_STAGE	NUMBER(10,0)	任务状态
VC_TASK_TYPE	VARCHAR2 (20)	任务类型
VC_RULE_TYPE	VARCHAR2 (20)	规则类型

表 3.4: 文件处理表结构

字段名	类型	备注
L_FILE_ID	NUMBER(10,0)	文件ID
D_SCHEDULED_DATE_TIME	VARCHAR2 (20)	调度最早时间
D_FILE_START_DATE	VARCHAR2 (20)	文件数据开发时间
D_FILE_END_DATE	VARCHAR2 (20)	文件数据结束时间
VC_FILE_SUFFIX	VARCHAR2 (20)	文件后缀名
VC_FILE_REALLY_KEY	VARCHAR2 (20)	文件密钥
VC_FILE_SRC_PATH	VARCHAR2 (100)	原始文件路径
VC_FILE_FINAL_PATH	VARCHAR2 (100)	处理后文件路径
VC_FILE_FAX_PATH	VARCHAR2 (100)	图片文件路径
C_SUCCESS	Bit (1)	任务是否成功
C_SKIP_SRC_GENERATE	Bit (1)	是否跳过原始文件生成
C_LACK_DATA	Bit (1)	是否允许缺失数据
C_OVER	Bit (1)	任务是否结束
C_HORIZONTAL	Bit (1)	是否是A4水平排版
VC_ERROR_DESC	VARCHAR2 (500)	错误详情
D_FILE_CREATE_TIME	VARCHAR2 (20)	文件生成时间
VC_RULE_TYPE	VARCHAR2 (20)	规则类型

表 3.5: 传真发送表结构

字段名	类型	备注
L_FAX_ID	NUMBER(10,0)	传真ID
VC_FAX_NUMBER	VARCHAR2 (20)	收件人传真号
VC_SUBJECT	VARCHAR2 (20)	传真主题
VC_RECEIVER	VARCHAR2 (20)	收件人名称
VC_MESSAGE	VARCHAR2 (20)	传真信息
D_SCHEDULED_DATETIME	VARCHAR2 (20)	调度最早时间
VC_ACCESS_NUMBER	VARCHAR2 (20)	传真任务响应编号
VC_RESULT	VARCHAR2 (20)	传真任务响应码
L_SEND_TIMES	NUMBER(10,0)	发送次数
L_SEND_PAGES	NUMBER(10,0)	发送页数
D_SEND_DATETIME	VARCHAR2 (20)	发送时间
L_FILE_ID	NUMBER(10,0)	发送的文件ID
C_OVER	Bit (1)	任务结束标识
C_SUCCESS	Bit (1)	任务成功标识
VC_ERROR_DESC	VARCHAR2 (500)	错误详情
C_SEND_FLAG	Bit (1)	允许发送标识

表 3.6: 邮件发送表结构

字段名	类型	备注
L_MAIL_ID	NUMBER(10,0)	邮件ID
VC_MAIL_ADDRESS	VARCHAR2 (20)	收件人邮件地址
VC_TEL	VARCHAR2 (20)	收件人手机号
VC_SUBJECT	VARCHAR2 (20)	收件人主题
VC_BODY	VARCHAR2 (20)	邮件信息
D_SCHEDULED_DATETIME	VARCHAR2 (20)	调度最早时间
L_FILE_ID	NUMBER(10,0)	发送的文件ID
C_OVER	Bit (1)	任务结束标识
C_SUCCESS	Bit (1)	任务成功标识
VC_ERROR_DESC	VARCHAR2 (500)	错误详情
C_SEND_FLAG	Bit (1)	允许发送标识



## 第四章 系统详细设计与实现

### 4.1 元数据管理模块详细设计与实现

元数据管理模块主要负责代理商信息和系统基本配置信息的管理。

SpringBoot框架的JPA提供了对POJO的基本操作封装,简化了对数据库的操作。这里使用JpaRepository来实现对POJO的访问,如图 4.1 所示

---

```
package com.xqfunds.xqfaxtask.repository;

import com.xqfunds.xqfaxtask.model.AgencyInfoRecord;
import org.springframework.data.jpa.repository.JpaRepository;

public interface AgencyInfoRepository
    extends JpaRepository<AgencyInfoRecord, String> {
}
```

图 4.1: 代理商POJO操作代码

### 4.2 发送规则模块详细设计与实现

### 4.3 任务调度管理模块详细设计与实现

### 4.4 文件路由模块详细设计与实现

### 4.5 文件处理模块详细设计与实现

### 4.6 文件发送模块详细设计与实现



## 第五章 文件自动发送平台测试

### 5.1 测试环境

### 5.2 功能测试

### 5.3 对外接口测试

### 5.4 本章小结





## 第六章 总结与展望

### 6.1 总结

### 6.2 进一步工作展望



## 第七章 标题

这是章节标题。注：一般而言，标题不要比小节标题更小，即不要出现1.2.3.4这种标题（本模板支持此类标题，即Subsubsection）。

### 7.1 这是节标题

#### 7.1.1 这是小节标题



## 第八章 正文

### 8.1 正文书写的小技巧

CTeX自带的pdf浏览器，双击每段文字之后会自动回到WinEdt的编辑位置。

只有间隔一个明显的换行才会自然段分段。

因此，建议把一个自然段中的每句话都单独作为一行。这样的好处是，每次双击一句话，都可以回到WinEdt编辑器具体的一行。如果编辑时也按照自然段组织，则双击时会返回到一大段，不能定位到具体位置。不便于快速定位到出现问题的地方。

### 8.2 一些正文中的标记

斜体

加粗

代码元素格式

居中，左右对齐同理。

这里展示脚注。<sup>1</sup>

一个小建议，中文后直接跟上述格式标记（包含各种引用）可能会出现一些问题。因此，在中文字和格式标记的斜杠之间加入一个波浪号是一个常用的习惯。双波浪线等价于一个强制空格，有时比键盘输入的空格要好用。

### 8.3 注意软换行的使用

论文一般会引用代码，本模板建议将代码声明为 `class.this()` 格式。在引用代码时，较长的函数名会导致函数名超出文本边界的情况，因此可以考虑手动进行软换行，请参考以下例子。

“图XX 展示了从AquaLush 系统中抽取的函数调用依赖示例，其中 `UIController.buildLogScrn()` 是为了实现新功能 “the control panel shows log message” 而在新版本中添加的函数。”

---

<sup>1</sup>数字列举和圆点列举见摘要部分



## 第九章 表格

表格是LaTeX中少数没有Word好用的功能。但word的表格依然存在行间距的问题，而LaTeX也有简洁美观，相对易用（相对）的三线表。

### 9.1 表格与表格引用的基本概念

表格的编号和表目录都是自动生成并持续编号的，无需人工修改。只要对表格有标注（label），则在正文中引用该表的label，就可以随时保持最新编号。

注意：如果一个新表格加入，并被引用，编辑器将需要连续编译两次到三次，才能完成全部标题、引用和目录的更新。可以理解为第一次编译引入新表格，此时还不知表格引用位置的具体编号，需要留待第二次编译完成。而有可能第三次编译才将表格信息写入开头的表目录。类似的情况也会出现在图形和论文引用这两部分，其中尤以论文引用部分最为奇特，详见相关章节。

### 9.2 基本表格

表 9.1（这里是一个表引用！）是一个简单的三线表，双击表格可以在编辑界面内见到具体设置。

具体解释一下表格的设置：第一个table体内首先先声明标记位置以及字体大小；随后声明表格对齐方式；其次描述表标题；之后进入具体的表内容（tabular，此时还要声明表格单元中的内容如何对齐）；依次画出三线并填充内容；如果表格内容较多，可以相应的加入横线来划分（hline）；之后退出tabular；最后给表起名以实现全局引用，并退出表格。

表 9.1: 实验系统中函数调用与数据依赖的交集

	Call	Data	Overlap
<b>VoD</b>	222	899	66
<b>GanttProject</b>	5560	24243	1042
<b>jHotDraw</b>	3943	14555	893

### 9.3 表格单元跨列

表 9.2展示如何实现表格单元跨列。

表 9.2: 错误率与函数特征之间的关联

	Parameters		Return Value		Is Constructor	
	with	without	with	without	with	without
<b>VoD</b>	8.99%	9.20%	6.10%	9.51%	9.43%	8.46%
<b>GanttProject</b>	9.53%	6.05%	8.43%	6.71%	5.14%	8.09%
<b>jHotDraw</b>	4.40%	3.89%	4.36%	3.88%	2.91%	4.39%

## 9.4 表格单元跨行

表 9.3展示如何实现表格单元跨行（Average Number那一行）。此外，本表格的字体尺寸为scriptsize，比上一个表格的footnotesize要更小。

表 9.3: 五个实验系统概述

	VoD	Chess	GanttProject	jHotDraw	iTrust
<b>Version</b>	-	0.1.0	2.0.9	7.2	13.0
<b>Programming Language</b>	Java	Java	Java	Java	Java
<b>KLOC</b>	3.6	7.2	45	72	43
<b>Executed methods</b>	165	316	2741	1755	250
<b>Evaluated requirements</b>	12	7	17	21	34
<b>Average Number of Methods</b>	45	173	387	121	12
<b>Implementing a Requirement</b>	(9-148)	(23-288)	(78-815)	(1-555)	(1-33)
<b>Size of the golden RTM</b>	1980	2212	46597	36855	8500
<b>Requirement traces</b>	534	1210	6584	2547	353
<b>Random Chance of guessing</b>	0.5-7.5%	1-13%	0.2-1.7%	0.003-1.5%	0.01-0.4%
<b>Method Call Dependencies</b>	210	439	4830	3848	319
<b>Method Data Dependencies</b>	905	976	30452	17316	5329

## 9.5 表格与图形位置

常用选项[htbp]是浮动格式：

『h』当前位置。将图形放置在正文文本中给出该图形环境的地方。如果本页所剩的页面不够，这一参数将不起作用。

『t』顶部。将图形放置在页面的顶部。

『b』底部。将图形放置在页面的底部。

『p』浮动页。将图形放置在一只允许有浮动对象的页面上。

一般使用[htb]这样的组合，只用[h]是没有用的。这样组合的意思就是LaTeX会尽量满足排在前面的浮动格式，就是h-t-b这个顺序，让排版的效果尽量好。图形章节会有更多位置符号的例子。



## 第十章 图形

### 10.1 基本图形

相对于表格而言，LaTeX中的图形就简单多了，需要注意的是本模板推荐将所有图形都转化为pdf，具体内容参见图 10.1。该图形放在本模板的本地文件夹FIGs中。图 10.1是将Excel的五个子图形排布在一个ppt页面上，之后保存为pdf文件，最终得到的图形可以保证是矢量图。

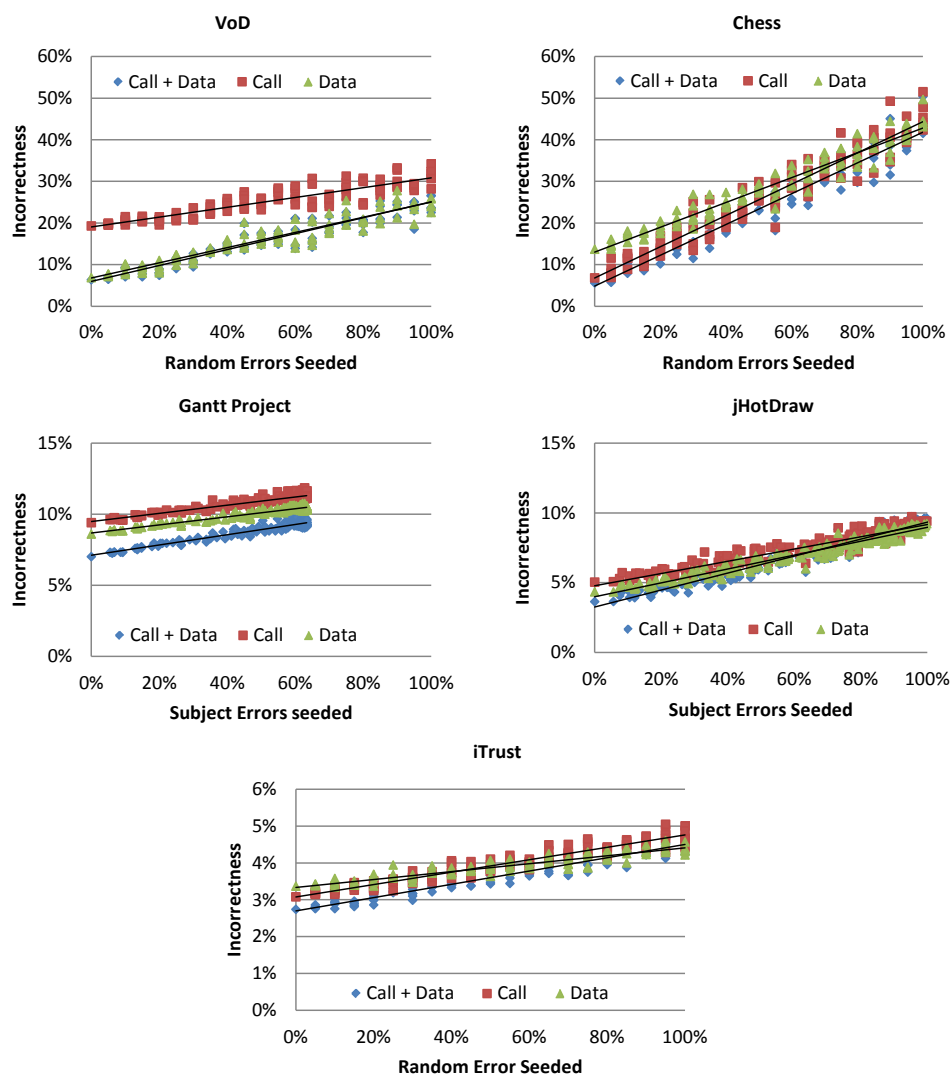


图 10.1: 以含错误的RTM为输入的五五个系统上三个实验（Call，Data，Call+Data）的错误率（Incorrectness）

注意：不要删除FIGs下面的njulogo和njuname这两个文件，这是论文封面的校徽和手写体南大校名。

## 10.2 引用代码

```
class VODClient{
    public final void init(){
        ...
        server = new ServerReq( "127.0.0.1", s);
        server.connect();
        listframe = new ListFrame(server, this);
    }
}

class ListFrame{
    public ListFrame(ServerReq serverReq,
        VODClient vODClient){
        ...
        ser = serverReq;
        parent = vODClient;
        ...}
    void buttonControl3_actionPerformed(...) {
        ...
        String s = listControll1.getSelectedItem();
        if (s != null){
            Movie movie = ser.getmovie(s);
        }
        ...}
}
```

图 10.2: VoD系统中的代码片段

这里给出一个代码引用的推荐实践。引用代码时先将代码放入word的文本框中，调整结束后，将该文本框页面另存为pdf文件，之后再作为图形来引用，如图 10.2所示。

## 第十一章 公式

这里直接给出几个较为复杂的公式的例子，可一一进行参照。若有未包含的数学符号或公式格式，请参阅本模板所包含的手册（本地manual文件夹）或百度必应谷歌。

### 11.1 公式5.1与论证

“从直接代码依赖的角度出发，从一个初始域外的类 $C_{out}$ 出发我们尝试找到一个通往初始域内的类 $C_{in}$ 的路径。一条合法的路径需要满足以下两点要求：（1）这一路径是单向的，即 $C_{out}$ 传递性地到达 $C_{in}$ 或 $C_{in}$ 传递性地到达 $C_{out}$ ；（2）路径中只能包含一个 $C_{in}$ （为了避免重复路径的出现）。为了恰当的估计一条合法路径所代表的交互程度，我们计算路径上所有直接代码依赖的紧密度值的几何平均。我们用如下公式来重新计算给定 $C_{out}$ 的IR值（ $IR_{DC}$ ）：”

$$IR_{DC} = IR_{origin} + (IR_{top} - IR_{origin})^{|PATH|} \sqrt{\prod_{x \in PATH} Closeness_{DC}(x)} \quad (11.1)$$

“其中 $IR_{origin}$ 代表 $C_{out}$ 的初始IR值， $IR_{top}$ 代表 $C_{in}$ 被提升过的IR值， $PATH$ 代表 $C_{out}$ 与 $C_{in}$ 之间的路径内所有的直接代码依赖，而 $Closeness_{DC}(x)$ 则代表每一条直接代码依赖关系的紧密度值。在同一对 $C_{out}$ 和 $C_{in}$ 之间可能存在多条合法路径，我们只保留其中能使 $IR_{DC}$ 值最大的那条路径。”

### 11.2 公式5.2与论证

“由于IR方法返回的是一个按照IR值大小倒序排列的候选线索列表，因此一种常用的比较IR方法的方式是在不同的查全率水平上比较不同方法之间的精确度，通常用Precision-Recall曲线表示。为了进一步衡量IR方法返回结果的整体质量，我们选用了另外两个常用的实验度量：AP（Average Precision）与MAP（Mean Average Precision）。其中，AP用于度量全部查询（需求）所检索的相关文档的排序质量，计算方式如下：”

$$AP = \frac{\sum_{r=1}^N (Precision(r) \times isRelevant(r))}{|RelevantDocuments|} \quad (11.2)$$

“其中， $r$  表示被查询对象（类）在列表中的排序， $Precision(r)$  表示前 $r$  个类的准确率。 $isRelevant()$  为一个二值函数，如果文档是相关的，则返回1，若无关，则返回0。”

### 11.3 公式5.3与论证

“由此，我们为类数据依赖定义紧密度 $Closeness_{CD}$  如下：”

$$Closeness_{CD} = \frac{\sum_{x \in \{DT_i \cap DT_j\}} idtf(x)}{\sum_{y \in \{DT_i \cup DT_j\}} idtf(y)} \quad (11.3)$$

“其中 $idtf(x)$  代表共享数据类型的idf值， $DT_i$  与 $DT_j$  的交集代表该数据依赖上的共享数据类型，而 $DT_i$  与 $DT_j$  的并集则代表 $C_i$  和 $C_j$  在全部代码上所访问的数据类型。 $Closeness_{CD}$  的取值范围是0到1之间。”

## 第十二章 算法

同样是定义+引用的方式，参见算法 1。本算法已包含大量常用格式，如有未包含的数学符号或格式，请参阅本模板所包含的手册或询问百度必应谷歌。如论文中无需算法则不用强加。

---

**Algorithm 1:** 初始需求域外追踪线索的重排

---

```

1 topIRValue  $\leftarrow$  initialRegion.topIRValue;
2 foreach link in candidateList do
3   if !initialRegion.contains(link.class) then
4     foreach c in initialRegion do
5       pathList  $\leftarrow$  findValidPaths(link.class, c);
6       gMean  $\leftarrow$  0;
7       foreach path in pathList do
8          $gMean \leftarrow \max(\text{GeometricMean}(\text{Closeness}_{DC}(\text{path})), gMean)$ ;
9       link.IRValue  $\leftarrow$  link.IRValue + gMean(topIRValue - link.IRValue);
10      if hasDataDependencies(c, link.class) then
11        link.IRValue  $\leftarrow$  link.IRValue + ClosenessCD(c,
12          link.class)(topIRValue - link.IRValue);
13      if link.IRValue > topIRValue then
14        link.IRValue  $\leftarrow$  topIRValue;
14 candidateList.reorderByIRValue();

```

---



## 第十三章 论文引用

### 13.1 引文相关

此处的论文引用采用的是类似于IEEE的按出现位置的数字编号格式。建议将被引用的论文全名放入dblp网站（必应谷歌搜索dblp）搜索，之后进入该论文详细信息，如图 13.1 所示。

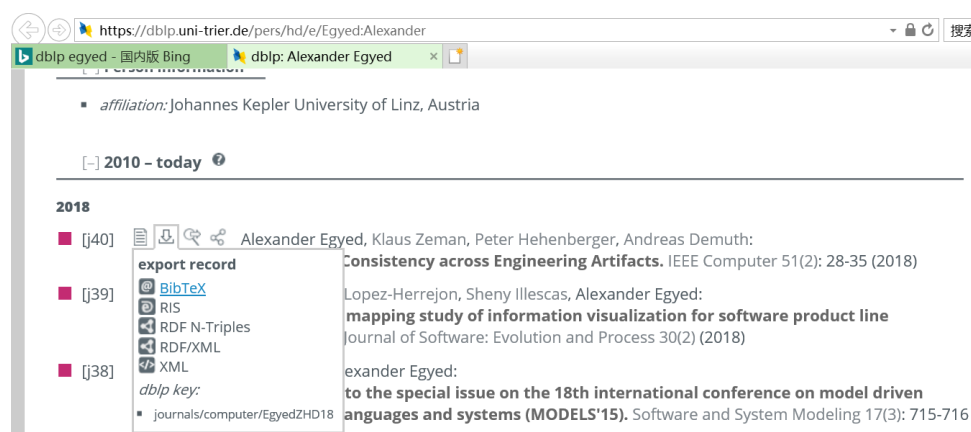


图 13.1: 在dblp上下载Bibtex

点击该链接之后将得到Bibtex信息，如图 13.2所示。打开本地文件夹下的reference.bib文件，完整添加该信息。并在需要引用的位置添加这一引用 [2]。格式为bibtex信息中的开头，例如图中的“*DBLP:journals/computer/EgyedZHD18*”。（此处是一个典型的因为长字符串导致的*bad box*，请参考上述章节的内容手动完整软换行）。

**注意：**在修改并保存reference.bib文件后，先点击PDFLaTeX旁边的B按钮编译bib文件，之后需要连续使用PDFLaTeX编译两次，直到最后控制台输出的Warnings不再增加，此时才完成一次论文引用的更新。

在bib文件中出现，但并未在论文中被引用的论文不会出现在最后的参考文献中。如果dblp中并未包含你需要的论文，则可以尝试谷歌或百度学术的搜索结果，一般也包含bibtex信息，但可能不完整或不规范。

引用网站链接可以考虑这一格式 [3]（不推荐，网站链接使用脚注更规范些）。

引用书籍可以考虑这一格式 [4]。

中文文献请参考这一格式 [5]（引用标记请避免中文，否则容易出现编译错误）。

以下英文引用用来测试引文排序是否按照插入顺序，以及多引文是否合并 [2, 6]

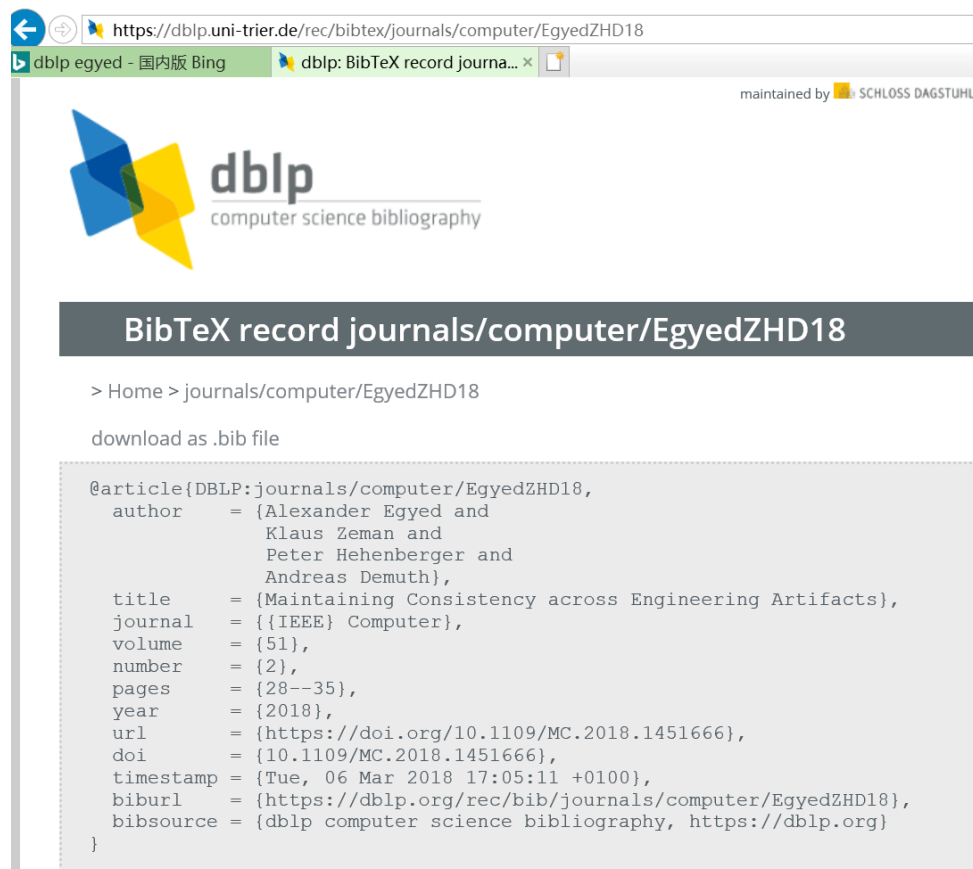


图 13.2: Bibtex详细信息

## 13.2 页眉测试

以下内容测试页眉的相关问题、以下内容测试页眉的相关问题、以下内容测试页眉的相关问题、以下内容测试页眉的相关问题、以下内容测试页眉的相关问题、以下内容测试页眉的相关问题、



## 参考文献

- [1] wikipedia, 办公自动化, <https://zh.wikipedia.org/wiki/%E5%8A%9E%E5%85%AC%E8%87%AA%E5%8A%A8%E5%8C%96>.
- [2] A. Egyed, K. Zeman, P. Hehenberger, A. Demuth, Maintaining consistency across engineering artifacts, IEEE Computer 51 (2) (2018) 28–35.  
URL <https://doi.org/10.1109/MC.2018.1451666>
- [3] GanttProject开发组, Ganttproject主页, <http://www.ganttproject.biz>.
- [4] K. Pohl, Requirements Engineering - Fundamentals, Principles, and Techniques, Springer, 2010.  
URL <http://www.springer.com/computer/swe/book/978-3-642-12577-5?changeHeader>
- [5] 程远国, 尹迪, 基于分布式中间件的传感器网络软件体系结构研究, 计算机工程与科学 (s2) (2006) 4–5.
- [6] K. M. Ting, Y. Zhu, M. J. Carman, Y. Zhu, T. Washio, Z. Zhou, Lowest probability mass neighbour algorithms: relaxing the metric constraint in distance-based neighbourhood algorithms, Machine Learning 108 (2) (2019) 331–376.  
URL <https://doi.org/10.1007/s10994-018-5737-x>



## 简历与科研成果

**基本情况** 张三，男，汉族，1993 年 1 月出生，江苏省南京市人。

### 教育背景

**2015.9～2018.6**    南京大学软件学院    硕士

**2011.9～2015.7**    南京大学软件学院    本科

这里是读研期间的成果（实例为受理的专利）

1. 李四，张三，“一种使用Hammer砸碎Nut的方法”，申请号：20xx1018xywz.a，已受理。



## 致 谢

这里是致谢。一般的感谢顺序：导师，指导老师，师兄姐妹、同学，父母和伴侣。