



燕山大学
YANSHAN UNIVERSITY

本科生毕业设计（论文）

论文题目 法律学习智能助手的设计与实现

作者姓名 郝雁飞

专 业 计算机科学与技术

指导教师 马勤勇

2024 年 6 月

燕山大学本科生毕业设计（论文）

法律学习智能助手的设计与实现

学 院： 信息科学与工程学院
专 业： 计算机科学与技术
姓 名： 郝雁飞
学 号： 202011040135
指 导 教 师： 马勤勇
答 辩 日 期： 2024 年 6 月

学位论文原创性声明

郑重声明：所呈交的学位论文《法律学习智能助手的设计与实现》，是本人在导师的指导下，独立进行研究取得的成果。除文中已经注明引用的内容外，本论文不包括他人或集体已经发表或撰写过的作品成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果，并承诺因本声明而产生的法律结果由本人承担。

学位论文作者签名：郝雁飞

日期：2024年5月30日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权燕山大学将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保 密 ☐，在__年解密后适用本授权书。

本学位论文属于

不保密 ☒。

(请在以上相应方框内打“√”)

学位论文作者签名：郝雁飞

日期：2024年5月30日

指导教师签名：马勃勇

日期：2024年5月30日

燕山大学毕业设计(论文)任务书

学院：信息科学与工程学院

专业(方向): 计算机科学与技术

学号	202011040135	学生姓名	郝雁飞	专业班级	20 级计算机 5 班
题目	题目名称	法律学习智能助手的设计与实现			
	题目性质	1.理工类：工程设计（ ）；工程技术实验研究型（ ）； 理论研究型（ ）；计算机软件型（√）；综合型（ ）。 2.文管类（ ）；3.外语类（ ）；4.艺术类（ ）。			
	题目类型	1.毕业设计（√） 2.论文（ ）			
	题目来源	科研课题（ ） 生产实际（ ） 自选题目（√）			
主要内容	本法律学习智能助手的使用者主要有管理员，法律学生以及客服。法律学生可以查询自己 想要知道的法律条文，由于法律条文专业性较强，本法律学习智能助手还可以对法律条文进行解 释。法律学生还可以查询法律案件，针对某一个行为查询量刑原因和判罚的依据。法律学生还可 以通过本法律学习智能助手查看相关法律文献。本法律学习智能助手还可以给法律学生出题考 试，并且收集学生的错题以方便学生日后的复习。				
基本要求	1.毕业设计论文符合撰写规范。 2.查阅相关资料，明确设计题目的意义和应用前景及国内外研究现状。 3.对所设计和开发的软件对社会和经济发展意义和作用进行说明、分析和评价，应能满足用户在 系统功能等方面的需求（如系统功能的完善程度，响应速度，稳定性等），能并给出切实可行的 目标达成度的评价指标和实施效果。				
参考资料	[1] 王吉伟.AIAgent 发展简史，从哲学思想启蒙到人工智能实体落地[J]. 大数据时 代, 2023,(12):6-19. [2] 科大讯飞发布星火认知大模型[J]. 智能建筑与智慧城市, 2023,(05):4.				
周次	12 月 12 日—寒假	1—3 周	4—9 周	10—13 周	14—15 周
应完成的内容	1. 完成相关技术的 学习和掌握 2. 对系统关键部分 进行编程验证	1. 需求分析 2. 撰写开题报 告和文献综述	1. 系统总体 设计 2. 系统详细 设计 3. 完成编码	1. 测试完善 2. 撰写毕业论 文	毕业论文答辩
指导教师：马勤勇 职称：讲师 2023 年 12 月 12 日					

摘 要

随着人工智能的发展和自然语言处理等一系列技术的迅速发展，智能助手在生活中的各个方面得到了很大的应用，在法律研究、法律学习等方面也可以用到智能助手。之前法律学习主要依赖于传统的学习方式，包括研究法律文件、研究法律条文和案例等，这些方式对于法律学生来说很有必要，但是在梳理大量信息和概念是可能会需要很多时间，并且法律条文的繁多，信息的复杂也给法律学生带来了挑战。

本法律学习智能助手可以帮助学生更加便捷的学习相关法律知识。法律学习智能助手可以通过机器学习和自然语言处理等技术，帮助学生更好的获取和理解法律知识。法律学习智能助手可以为学生提供案例分析，回答学生的问题，指导学生学习；还可以利用机器学习分析法律案例，帮助法律学生更好的理解法律条文。学生可以使用本系统的论坛进行学术交流，学生可以在论坛发表话题，也可以回复其他学生的话题。本系统主要分为信息管理模块、论坛模块、充值模块、搜索法律条文模块、查看法律文献模块、用户自测模块六个模块，帮助了法律学生更加便捷地学习，方便了法律学生查询所需要的信息。

本系统采用 SQLite 数据库，SQLite 属于一个进程内的库，可以实现自给自足，并且不需要服务器，也不许需要配置。本系统还采用 Qt 框架，Qt 是一个跨平台的 C++ 应用程序开发框架，可以用于开发图形用户界面、嵌入式系统等应用程序，支持 Windows、Linux、Android 和 IOS 等多个操作系统。

关键词：人工智能；法律学习；智能助手；机器学习；自然语言处理；SQLite

Abstract

With the rapid development of artificial intelligence and a series of technologies such as natural language processing, intelligent assistants have been widely applied in various aspects of life, including legal research and legal studies. Previously, legal studies relied mainly on traditional methods, such as studying legal documents, statutes, and cases. While these methods are necessary for law students, they can be time-consuming when dealing with large amounts of information and concepts. The vast number of legal statutes and the complexity of information pose challenges to law students.

This legal study intelligent assistant aims to assist students in learning legal knowledge more conveniently. Leveraging machine learning and natural language processing techniques, the legal study intelligent assistant helps students acquire and comprehend legal knowledge more effectively. It provides case analysis, answers students' questions, and guides their learning. Additionally, it utilizes machine learning to analyze legal cases and help law students better understand legal statutes. Students can use the forum in this system for academic exchanges. They can create topics on the forum and also reply to other students' topics. This system is mainly divided into six modules: Information Management, Forum, Recharge, Legal Articles Search, Legal Literature Viewing, and User Self-assessment. It helps law students to study more conveniently and facilitates their access to the necessary information.

The system adopts the SQLite database, which is an embedded database library that operates within a single process. It is self-contained, requiring no server or complex configuration. The system is built using the Qt framework, a cross-platform C++ application development framework. Qt can be used to develop applications with graphical user interfaces, embedded systems, and more, supporting multiple operating systems such as Windows, Linux, Android, and iOS.

Keywords: Artificial Intelligence, Legal Study, Intelligent Assistant, Machine Learning, Natural Language Processing, SQLite.

目录

摘 要	I
Abstract	II
第 1 章 绪 论	1
1.1 课题的背景	1
1.2 目的和意义	1
1.3 国内外文献综述	2
1.3.1 国内外研究动态	2
1.4 主要研究内容	4
1.4.1 需要解决的问题	5
1.5 论文组织结构	6
1.6 本章小结	6
第 2 章 研究步骤、方法、措施	7
2.1 研究目标	7
2.1.1 确定研究目标	7
2.1.2 数据收集和预处理	7
2.1.3 模型选择和训练	7
2.2 相关技术介绍	9
2.2.1 Qt 框架介绍	9
2.2.2 SQLite 数据库介绍	9
2.2.3 QtCreator 介绍	9
2.2.4 大模型介绍	10
2.3 研究方法和措施	10
2.4 本课题方案设计及对比	12
2.5 本章小结	13
第 3 章 需求分析	15
3.1 可行性分析	15
3.1.1 技术可行性	15
3.1.2 经济可行性	16
3.1.3 社会可行性	16
3.1.4 操作可行性	16
3.2 需求分析	16
3.2.1 功能需求分析	16
3.2.2 非功能需求分析	19

3.3 本章小结	20
第 4 章 总体设计	21
4.1 系统设计原则	21
4.2 总体功能设计	22
4.3 数据库设计	24
4.3.1 数据库实体	24
4.3.2 模块 E-R 图	28
4.3.3 数据库表设计	31
4.4 本章小结	34
第 5 章 详细设计	35
5.1 信息管理模块	35
5.2 论坛模块	38
5.3 充值模块	39
5.4 搜索法律条文模块	41
5.5 查看法律文献模块	42
5.6 用户自测模块	43
5.7 本章小结	44
第 6 章 系统实现	45
6.1 环境配置	45
6.2 关键功能的实现	45
6.2.1 用户登录	45
6.2.2 搜索法律条文	46
6.2.3 咨询法律案件	46
6.2.4 咨询法律问题	47
6.2.5 用户自测	48
6.3 本章小结	48
第 7 章 系统测试	49
7.1 测试步骤	49
7.2 测试方法	50
7.3 功能测试	51
7.4 性能测试	53
7.5 本章小结	54
结 论	55
参考文献	57
致 谢	59

目录

附录 1 开题报告	61
附录 2 文献综述	71
附录 3 中期报告	77
附录 4 外文原文	91
附录 5 外文翻译	105

第1章 绪论

1.1 课题的背景

随着人工智能的发展和自然语言处理等一系列技术的迅速发展，智能助手在生活中的各个方面得到了很大的应用，在法律研究、法律学习等方面也可以用到智能助手。之前法律学习主要依赖于传统的学习方式，包括研究法律文件、研究法律条文和案例等，这些方式对于法律学生来说很有必要，但是在梳理大量信息和概念是可能会需要很多时间，并且法律条文的繁多，信息的复杂也给法律学生带来了挑战。

法律学习智能助手可以通过机器学习和自然语言处理等技术，帮助学生更好的获取和理解法律知识。法律学习智能助手可以为学生提供案例分析，回答学生的问题，指导学生学习；还可以利用机器学习分析法律案例，帮助法律学生更好的理解法律条文。随着智能学习助手的发展，国内外许多机构已经开始探索如何使用这项技术，一些公司也在开发和推出面向法律学习的智能助手产品。

1.2 目的和意义

法律学习智能助手通过使用自然语言处理和机器学习等人工智能技术，可以帮助法律学生迅速找到法律案例，并且可以为用户提供相关的法律条文以及为法律学生提供对条文的解释，可以更好的帮助法律学生理解和应用相关的法律条文。法律学生也可以通过输入关键字搜索相应法律条文，本法律学习智能助手可以迅速检索相关的法律法规条文，并且提供与之相关案例和应用，帮助法律学生迅速找到相应的法律条文。法律学生还可以输入不理解的法律术语，智能助手可以帮助法律学生来解释复杂的法律术语，可以帮助法律学生更好地理解专业性较强的法律法规，帮助学生的学习。本法律学生智能助手还可以帮助法律学生撰写法律文书，通过分析法律学生提供的要求，法律学生智能助手可以自动生成法律文书的模板，帮助法律学生撰写更加专业的文书。智能助手还可以分析使用者的学习需求和兴趣，推荐相关的学习资源，可以帮助用户出题自测，以及找出用户的错题加以整理，为用户学习提供方便，帮助学生日后进行复习。本法律学习智能助手还可以帮助学生之间更加便捷的进行学术交流，为法律学生提供更加便捷的交流平台。

1.3 国内外文献综述

1.3.1 国内外研究动态

近几年，伴随着人工智能技术的发展，国内一些公司也在研究和开发面向法律学习的智能助手产品。

百度法律智能助手是一款基于人工智能技术的法律学习工具，它通过自然语言处理、知识图谱和机器学习等技术，帮助法律学生学习法律知识，为用户进行案例分析和问题解答等，帮助法律学生深入理解法律领域的知识。用户可以向百度法律智能助手提出相关的法律问题，它会根据自然语言处理技术来理解问题，并尝试给出相关的法律知识和解答；用户还可以向百度法律智能助手提供相关案例或者提出问题，百度法律智能助手会分析案例并向用户提供相关的法律解释和类似的案例；百度法律智能助手还可以提供广泛的法律知识，用户可以查询特定的法律知识和相关概念；用户可以用百度法律智能助手搜索相关的法律条文、案例和法律文献，百度法律智能助手会提供相应的检索结果和摘要信息。

现在，人工智能已经第三次开始迅猛地发展。人工智能理论和技术取得了飞速发展，在语音识别、文本识别、视频识别等感知领域取得了突破，已经可以达到甚至超过了人类水准，成为了引领新一轮科技革命和产业变革的战略性技术。人工智能的应用领域也快速向多方向发展，出现在与人们日常生活息息相关的越来越多的场景中^[1]。

大模型是指具有大规模参数的机器学习模型，而大语言模型是指在自然语言处理领域的一种大模型，可以用来进行文本处理。大语言模型的主要特点是在大规模语料库上进行了训练，学习了自然语言的语法，国内外较为常见的有 OpenAI 的 GPT 系列，Google 的 Bard，以及百度的文心一言。

GPT 是一种基于 Transformer 架构的大语言模型。GPT 中文意为生成式预训练变换器。自然语言处理可以解决一些通用的问题，分为两个关键步骤：自然语言理解（NLU）和自然语言生成（NLG）。自然语言理解是指将人类的语言转化成为机器可以理解和处理的形式，它涉及对文本进行语义分析和语法分析，来提取出其中的意义和结构，NLU 的主要目标是使计算机能够更好的理解和解释人类的语言。自然语言生成是指根据计算机内部的信息和逻辑规则，生成自然语言文本的，目标是使计算机能够以自然的方式生成易于人类理解的语言输出^[5]。总之，自然语言理解是

人类语言转化为机器可以理解的形式，而自然语言生成是根据计算机内部的信息和逻辑规则转化为人类可以理解的语言。这两个任务在自然语言处理中相辅相成，共同构建了人机交互、智能对话和信息处理等应用领域的基础。

Bard 是一个由谷歌公司开发的聊天机器人，经过了训练，内容丰富。Bard 接受过了大量文本数据的培训，可以针对各种问题进行交流并且可以生成类似人类写作的文本。最初 Bard 基于 LaMDA 系列的大语言模型，后来基于 PaLM2，现在则是基于 Gemini 系列的大语言模型，是谷歌为了应对 OpenAI 公司的 GPT 系列而研发的。Gemini 是一个多模态大语言模型系列，多模态大语言模型是一个脱胎于大语言模型的模型，在原有的强大泛化能力和推理能力的基础上，进一步引入了多模态的信息处理能力。模态指的是一种信息的来源形式，例如人的听觉，嗅觉等，信息的模态指的是信息的来源媒介，例如文本、视频、音频、图像等，所谓的多模态指的是具有多种信息来源或者多种信息形式。多模态侧重的是不同类型的信息的融合和分析的问题，推动了人工智能更好地了解和认知世界。和以往的多模态方式相比，新兴的多模态大语言模型的模型更大，训练范式更新，具有着传统的多模态模型不具备的能力。

文心大模型是百度旗下的大模型，是一个由百度自主研发的产业级大模型，文心大模型以创新性的知识增强技术为核心，从单模态大模型到跨模态大模型，从通用基础大模型到跨领域大模型持续创新，文心大模型构建了模型层、工具与平台层，加快了人工智能大模型大规模产业化进程，大幅度降低了人工智能开发和应用的门槛。上文提到模态指的是一种信息的来源形式，而单模态顾名思义指的就是仅用一种模态的数据进行学习，而跨模态学习指的是一种在不同模态之间进行的信息传递和理解的学习过程。跨模态是从一个模态提取信息，并且还要用这些信息来理解另一个模态的内容。跨模态学习的核心在于不同模态之间的相关性和互补性。2019 年 3 月，文心大模型初次发布，历经四年，文心大模型在 2023 年 4 月提升至 4.0 版本，经过飞桨框架软硬协同优化训练的文心大模型，在各种能力上都有了全面的提升。飞桨深度学习平台以百度多年的深度学习技术研究和业务应用为基础，是中国首个自主研发的产业级深度学习平台，在开发便捷的深度学习框架、大规模分布式训练、高性能推理引擎等技术上处于国际领先水平，当前飞桨凝聚八百万开发者，基于飞桨创建的八十万个模型，飞桨广泛服务于金融、能源、制造、交通等领域。文心大模型现在也已经广泛地应用在搜索、信息流、智能音箱等互联网产品，并且通过飞

浆深度学习平台赋能工业、能源、教育、金融、通信等各个行业。

1.4 主要研究内容

大语言模型是指具有大规模参数和能力的语言模型，大模型的目标是理解和生成自然语言文本。这些模型使用深度学习等技术，通过学习大量的文本数据来捕捉语言的概率分布。

大语言模型的优势在于它可以处理复杂的语言结构和语义关系。它们可以理解上下文、推理相关信息、生成连贯的文本等。这使得大模型在各种自然语言处理任务中表现出色，比如文本生成、对话系统、摘要生成、机器翻译、问答系统等^[4]。然而，大语言模型也面临一些挑战，其中之一是对计算资源的需求，大语言模型通常需要庞大的计算设备和显著的训练时间。

除了上述这些，模型的输出可能会受到数据偏见、语义歧义和生成不准确性等问题的影响，需要进行后期处理和调整。尽管如此，大语言模型也代表了自然语言处理领域的前沿技术，对推动自然语言的理解和生成的发展具有着重要意义。随着模型规模的扩大和技术的改进，大语言模型将在各个领域发挥越来越重要的作用，为人们提供更智能、自然的语言交互和应用体验。

本法律学习智能助手是一种基于人工智能技术的软件，目的是辅助学生和专业人士学习和理解相关的法律知识。它结合了自然语言处理、机器学习和大语言模型等技术，能够提供准确的法律信息、解答法律问题，并帮助用户进行法律研究。

法律学习智能助手的使用者分为管理员和普通用户和客服，普通用户即为法律学生。管理员拥有独特的管理员账号，管理员登录后拥有账号管理的权限，分为账号封禁和账号解封。客服拥有独特的客服账号，客服登录后可以与管理员和法律学生进行交流，可以充当管理员和法律学生之间的交流媒介，避免管理员和法律学生的直接交流。用户注册登陆后可以查询法律条文，智能助手能够查询和提供相关法律法规的信息，方便用户查找相关法律文本；智能助手可以连接用户到法律论坛和社区，让用户与其他法律学习者和专业人士交流、分享经验和提问问题；法律学习智能助手可以迅速访问各种法律数据库和文献资源，包括法律条文、判例、学术论文等，为用户提供准确和全面的法律知识；用户可以向法律学习智能助手提出具体的法律问题，它会分析问题并给出相应的答案和解释。它可以帮助用户理解法律概念、解释法律原则，并提供相关案例和法律依据。法律学习智能助手可以帮助用户进行

法律研究，提供相关的案例和法律文件，帮助用户找到相关的法律条文和判例，并进行分析和比较，支持用户进行深入的法律研究。法律学习智能助手可以根据用户的学习需求和兴趣，提供个性化的学习推荐。它可以分析用户的学习历史和偏好，并提供相关的学习资源和建议，帮助用户高效地学习和提升法律知识。法律学习智能助手还可以给法律学生出题考试，批阅学生的答卷，监测法律学生对相关法律条文的学习情况和理解情况，为法律学生收集错题集，帮助法律学生日后的复习和自测，法律学习智能助手的功能如图 1-1 所示。

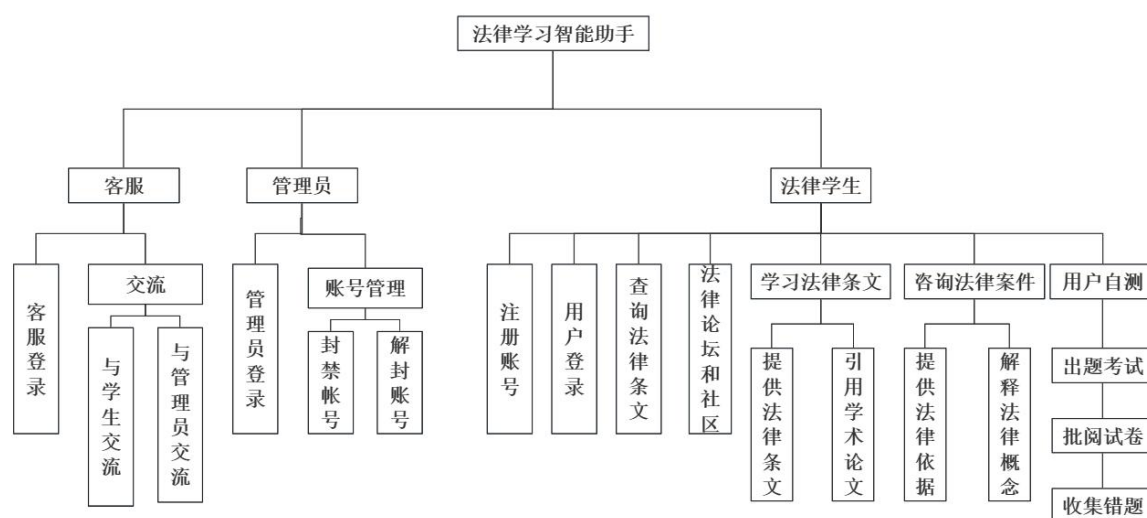


图 1-1 功能结构图

1.4.1 需要解决的问题

法律学习智能助手需要能够获取广泛的法律知识，以便能够高效地检索和应用，这包括收集和整理法律条文、案例、法律解释、法学文献等内容，因此选择使用大语言模型来实现人机交互。

法律法规文本通常具有复杂的语义和句法结构，因此法律学习智能助手需要具备强大的自然语言处理能力，能够理解和解析输入的法律问题，并提取关键信息。这涉及词法分析、句法分析等技术。

法律学习智能助手也需要具备法律问题分析和推理的能力，它需要能够理解用户提出的法律问题，并结合法律知识进行分析和判断。此外，智能助手还需要拥有友好的用户界面和交互方式，使用户能够方便地与其进行沟通和互动，这可以是一个聊天界面、语音交互或基于 Web 的用户界面。

1.5 论文组织结构

论文共分为七章，具体组织结构如下。

第一章为绪论部分，主要描述了本次课题的背景，目的和意义，以及国内外的研究动态，还有本课题的主要研究内容；第二章主要描述了研究步骤、方法和措施，确定了研究目标，介绍了相关技术，探讨了研究方法和措施，对比了其他的方案；第三章主要为需求分析，首先描述了可行性分析，分为技术可行性，经济可行性，社会可行性和操作可行性，然后进行了需求分析，包括功能需求分析和非功能需求分析，并且还配备了用例图对需求分析进行了介绍；第四章主要介绍了总体设计，先介绍了本法律学习智能助手的设计原则，又介绍了总体的功能设计以及对数据库的设计；第五章对本法律学习智能助手进行了详细设计，详细介绍了本法律学习智能助手的功能结构，数据库实体还有数据库表，先介绍了本法律学习智能助手设计遵循的几条原则，然后对法律学习智能助手应该实现的功能进行设计，并且按模块划分了功能，并画出了系统的功能模块图，后面跟着进行了数据库设计，画出了数据库实体属性图和 ER 图并根据这个设计出了数据库表，最后按照模块对主要的各功能以流程图的形式进行详细设计阐述；第六章介绍了系统实现的功能有哪些，先介绍了法律学习智能助手的环境配置，从软件环境方面对本法律学习智能助手展开了叙述，然后分了包括登录注册模块、搜索法律条文模块、查询法律案件模块、咨询法律问题等模块进行了部分功能的界面展示；第七章进行了系统测试，从法律学习智能助手系统测试出发，首先对测试步骤进行逐一简单介绍，紧接着介绍两种常用的测试方法——黑盒测试与白盒测试。随后通过测试用例的执行情况对法律学系智能助手进行了分模块的功能测试。

1.6 本章小结

本章主要介绍了本课题的背景、目的和意义，即本法律学习智能助手可以帮助学生之间更加便捷地进行学术交流，为法律学生提供更加便捷的交流平台。此外还介绍了国内外发展现状，详细介绍了百度的文心一言、谷歌公司的 Bard、OpenAI 公司的 GPT，还有介绍了需要解决的问题，以及本论文的组织结构。

第2章 研究步骤、方法、措施

2.1 研究目标

2.1.1 确定研究目标

设计一个拥有可以回答用户法律问题，帮助法律学生出题考试，批改试卷，收集错题帮助用户复习等功能的学习助手。这个法律学生智能助手允许学生搜索法律条文，查看法律文献，咨询法律案件，法律学生智能助手会提供给学生相应的法律条文和解释，提供法律文献，可以帮学生解答法律问题。法律学习智能助手还可以为学生提供论坛平台来交流。

2.1.2 数据收集和预处理

数据收集：确定需要收集的法律数据的类型和来源，包括法律条文、案例、文献等。根据确定的数据来源，收集相关的法律数据。这可以通过网络爬取、数据下载、数据库查询等方式进行，然后将收集到的数据划分为训练集、验证集和测试集。训练集用于进行模型的训练，验证集用于进行模型的调优和选择超参数，测试集用于评估模型的性能和泛化能力^[7]。

预处理：将收集到的法律文本进行分词，将文本划分为单词或子词的序列，可以使用现有的分词工具或库进行处理。将分词后的文本转化为词向量表示，以便模型能够处理。可以使用预训练的词向量模型或自行训练词向量模型。将文本序列填充或截断为固定的长度，以保证输入数据的一致性。这可以使用填充和截断技术，如将较长的文本截断为固定长度，或通过添加特殊的填充符号将较短的文本序列填充到相同长度。将文本数据进行编码，以适应模型的输入要求^[8]。

2.1.3 模型选择和训练

大语言模型是指具有大规模参数和计算能力的自然语言处理模型，这些模型通常可以通过大量的数据和参数进行训练，来生成人类可以理解的语言。大模型与小模型的主要区别在于模型大小、训练数据量和计算资源不同。大语言模型的表现经常

遵循着扩展法则，但对于“涌现能力”，往往只有模型规模到达一定程度之后才可以体现出来，其中具有着代表性的能力包括上下文学习、指令遵循、逐步推理等。

对于本项目的研究来说，首先需要研究不同的大语言模型，并且要了解它们的特点和适用场景，然后根据研究目标 and 需求，选择适合法律学习智能助手的大语言模型，还要考虑模型的预训练能力、对法律领域的了解程度、语义理解和生成能力等因素。

本项目选择 GPT 模型，GPT 是一种基于 Transformer 架构的大语言模型。GPT 中文意为生成式预训练变换器。自然语言处理可以解决一些通用的问题，分为两个关键步骤：自然语言理解（NLU）和自然语言生成（NLG）。自然语言理解是指将人类的语言转化成为机器可以理解和处理的形式，它涉及对文本进行语义分析和语法分析，来提取出其中的意义和结构，NLU 的主要目标是使计算机能够更好的理解和解释人类的语言。自然语言生成是指根据计算机内部的信息和逻辑规则，生成自然语言文本的，目标是使计算机能够以自然的方式生成易于人类理解的语言输出。

预训练是一个大模型的基础能力，当语言模型的参数量到达千亿级别时，从头预训练一个大模型很会有困难。预训练之后就是适配微调，微调可以进一步增强大模型的能力，并且还可以让大模型更加满足人类的需要。微调主要有两种：指令微调和对齐微调，指令微调通过收集指令的格式来微调大模型，可以增强大模型遵循人类指令的能力，可以让模型更好地范化到未知的任务，对齐微调通过收集人类的反馈数据，利用强化学习进一步微调大模型，可以让大模型更加符合人类的偏好。总而言之，自然语言理解是将人类语言转化为机器可以理解的形式，而自然语言生成是根据计算机内部的信息和逻辑规则转化为人类可以理解的语言。这两个任务在自然语言处理中相辅相成，共同构建了人机交互、智能对话和信息处理等应用领域的基础。

针对某个具体的任务，通常会有多种模型可供用户选择，对同一个模型也可能会有很多组参数，通过分析、评估模型的泛化误差，然后选择泛化误差最小的模型。被测试的数据集全集未知，再使用测试集对其进行泛化测试，这样得到的测试误差即为泛化误差的近似^[9]。

除此之外，用户还可以使用数据收集和预处理步骤中准备的法律数据作为模型的训练数据，然后将数据划分为训练集、验证集和测试集，确保训练数据的充分性和代表性。使用测试集评估训练好的模型的性能和泛化能力，用户可以根据评估结

果选择最佳模型，选择的模型要在验证集和测试集上表现良好，并且还能够准确地回答法律问题。

2.2 相关技术介绍

法律学习智能助手主要采用 QT 框架，通过 SQLite 数据库，通过 QtCreator 设计开发。

2.2.1 Qt 框架介绍

Qt 是一个跨平台的 C++ 应用程序开发框架，可以用于开发图形用户界面、嵌入式系统等应用程序，支持 Windows、Linux、Android 和 IOS 等多个操作系统^[14]。Qt 框架提供了许多功能和工具，其中包括图形用户界面设计、数据库操作、网络编程、文件处理等。同时 Qt 还是一个非常灵活的框架，Qt 可以允许使用者通过 C++ 语言来开发程序，开发者可以根据自己的需求使用。Qt 框架经过与时俱进地优化，可以提供更高性能的应用程序，与此同时 Qt 还提供了大量的内置工具和 API，使 Qt 适用于各种开发。Qt 还包括了一个庞大的类库，用户可以利用这些类使得开发过程更加便捷。Qt 还具有良好的文档和社区支持，因此广泛用于各种类型的应用程序开发，包括桌面应用、移动应用、嵌入式系统等^[16]。

2.2.2 SQLite 数据库介绍

SQLite 属于一个进程内的库，可以实现自给自足，并且不需要服务器，也不需要配置，SQLite 引擎不是独立的，它可以按照应用程序的需求进行静态或者动态链接，SQLite 可以直接访问存储文件^[11]。

SQLite 不需要一个单独的服务器进程或者操作，也不需要安装和管理，一个完整的 SQLite 数据库可以存储在一个跨平台的磁盘文件中，并且 SQLite 量级较轻，完全配置时小于 400KB，还可以自给自足，不需要外部依赖，同时 SQLite 还支持 SQL92 标准的查询语言功能，此外还提供了简单和易于使用的 API，方便用户使用。

2.2.3 QtCreator 介绍

QtCreator 是一种跨平台的轻量级集成开发环境，支持跨平台运行，支持 Linux，Mac 和 Windows。QtCreator 可以让开发人员能够利用 Qt 这个框架更加便捷地完成开

发任务^[15]。

2.2.4 大模型介绍

语言模型是一种自然语言处理的基本概念，是一种用来计算语言概率的数学模型，语言模型先理解语言的结构和规律，并且对文本下一阶段的文本中的词语做出预测，可以让语言模型能够识别语法、语义等语言特征。

随着深度学习的发展，神经网络语言模型逐渐成为顶流，其中包括循环神经网络和长短期记忆网络等模型，这些模型通过学习大量文本数据来建立词语之间的联系，从而实现对文本的预测。在自然语言处理中，语言模型可以用来进行机器翻译、文本生成等任务，在文本生成方面，语言模型可以用来进行自动写作、智能对话等^[6]。

在过去几年中，大语言模型迅速发展，其中 GPT（生成式预训练转换器）和 BERT（双向编码表示器）代表了自然语言处理领域的进步，重新定义了语言理解和生成的范式。以 GPT 和 BERT 为代表的大语言模型利用 Transformer 架构的优势，又通过大规模的预训练来学习语言的表示。其中，GPT 采用了自回归的与训练目标，在生成方面的任务上表现得很出色。

大语言模型的训练是一个复杂的过程。首先需要进行数据集的选择，其中数据集的质量和规模对训练很重要，一般来说需要选择大规模的文本作为训练数据；其次就是预训练和微调的过程，在预训练阶段，模型会在大规模文本上进行无监督学习，在微调阶段，模型会在特定的数据上进行有监督学习；之后进行计算资源与训练成本，一般情况下训练大语言模型需要大量的计算机时间和时间；最后要避免过拟合和模型泛化，比如进行正则化，数据增强等。

2.3 研究方法和措施

（1）方案一

大语言模型是一类基础模型，是经过大量数据训练，使法律学习智能助手能够理解和生成自然语言和其他类型的内容，可以让法律学习智能助手来执行各种任务。大语言模型在将生成式人工智能带到公共利益最前沿方面所发挥的作用让大语言模型被更多的人所认识。随着生成式人工智能的新发展，在企业环境之外，大语言模型似乎是突然出现。然而，包括 IBM 在内的许多公司多年来一直在不同层面实施大语言模型，用来增强自然语言理解和自然语言处理的能力。这与机器学习、机器学

习模型、算法、神经网络和为人工智能系统提供架构的变换器模型的进步同步发生。

大语言模型是一类基础模型，大语言模型经过大量数据训练，用来提供推动多个用例和应用程序以及解决大量任务所需的基础功能。这与为每个用例单独构建和训练域特定模型的想法形成鲜明对比，这种想法在许多标准下令人望而却步，会抑制协同作用，甚至可能导致性能下降。

大语言模型代表自然语言处理和人工智能领域的重大突破，用户可以通过 Open AI 的 Chat GPT-3 和 GPT-4 等界面轻松访问，这些界面已经获得了 Microsoft 的支持。总而言之，大语言模型旨在经过大量数据训练，像人类一样理解和生成文本以及其他形式的内容。这种模型有能力从环境中推断问题的能力，生成连贯并且还与环境相关的响应，翻译成英语以外的语言，总结文本，回答问题，甚至可以协助完成创造性写作或代码生成任务。它们之所以如此神通广大，是因为有数十亿个参数使模型能够捕获语言中的复杂模式并执行各种与语言相关的任务。大语言模型正在彻底改变从聊天机器人和虚拟助理到内容生成、研究协助和语言翻译各个领域的应用程序。随着大语言模型的不断发展和改进，大语言模型将会重塑我们与技术交互和获取信息的方式，使它们成为现代数字环境的关键部分。

(2) 方案二

TF-IDF（词频-逆向文件频率）是一种用于信息检索与文本挖掘的常用加权技术，是一种统计的方法，用来评估一个字词对于一个文件集或者是一个语料库中的其中一份文件的重要程度。显而易见的是，字词的重要性随着它在文件中出现的次数成正比增加，但同时也会随着它在语料库中出现的频率成反比下降^[10]。

TF-IDF 的主要思想是：如果某个单词在一篇文章中出现的频率高，并且在其他文章中很少出现，那么就可以认为这个词或者短语具有很好的类别区分能力，适合用来分类。

TF（词频）指的是在一个文档中某个词语出现的频率。词频可以用简单的计数来表示，即为这个词语在文档中出现过的次数，计算方式是用这个词语在文档中出现的次数除以文档中的总词数。词频反映了词语在文档中是否重要，一个词的词频越高，说明它的重要性越大。IDF（逆文档频率）指的是衡量某个词语在整个语料库中的重要程度。逆文档频率通过计算某个词语在语料库中出现的文档频率的倒数，来衡量这个词语的稀有程度。逆文档频率可以有效地减少一些常见的词语对于整个文本的重要性的影响，从而可以提高对于特定词语的重视程度。TF-IDF 模型将词频

和逆文档频率相乘，可以得到每个词语的 TF-IDF 值^[10]。

TF-IDF 值高的词语在检测的文本中出现的频率更高，但是在整个语料库中却较为稀有，因此就会具有更高的重要性。TF-IDF 模型通过计算每个词语的权重，将文本表示为一个向量，可以作用于文本的相似度计算、信息检索、文本分类等任务。TF-IDF 模型的优点是简单有效，这个模型可以很好地捕捉到词语在文本中的重要程度。然而，这个模型同时也存在一些限制，比如没有办法考虑词语的顺序和语义关系，对于长文本的检测效果可能有限。在实际应用中，可以结合其他提取方法和机器学习算法来进一步提高文本处理的性能。

2.4 本课题方案设计及对比

大语言模型上下文理解能力强，通过预训练，可以学习到庞大的语言知识和语义关系，能够更好地理解和生成文本，大语言模型在预训练阶段使用了大规模的通用语料，大语言模型旨在经过大量数据训练，像人类一样理解和生成文本以及其他形式的内容。这种模型有能力从环境中推断问题的能力，生成连贯并且还与环境相关的响应，翻译成英语以外的语言，总结文本，回答问题，甚至可以协助完成创造性写作或代码生成任务。

大语言模型计算资源需求高通常需要大量的计算资源和显存来进行预训练和推理，对硬件设施的要求较高，预训练需要大规模的文本数据，对于特定领域的任务，可能需要额外的领域数据进行微调，模型通常具有大量的参数和复杂的网络结构，导致模型较大，难以部署和运行在资源受限的环境中。

TF-IDF 模型计算简单，不需要大量的计算资源，词频可以用简单的计数来表示，即为这个词语在文档中出现过的次数，词频反映了词语在文档中是否重要，一个词的词频越高，说明它的重要性越大，逆文档频率通过计算某个词语在语料库中出现的文档频率的倒数，来衡量这个词语的稀有程度。逆文档频率可以有效地减少一些常见的词语对于整个文本的重要性的影响，从而提高对于特定词语的重视程度。可以根据关键词的重要性进行文本排序和匹配，还可以提供关键词的权重信息，帮助理解文本的重要内容和关键信息^[18]。

TF-IDF 模型只考虑词频和文档频率，忽略了词语之间的语义关系，不能很好地理解上下文和语义，对于复杂的自然语言处理任务的性能有限，对于未见过的词汇没有有效的表示，无法处理未知词汇的情况。

2.5 本章小结

Qt 是一个跨平台的 C++ 应用程序开发框架，可以用于开发图形用户界面、嵌入式系统等应用程序，支持 Windows、Linux、Android 和 IOS 等多个操作系统。Qt 框架提供了许多功能和工具，其中包括图形用户界面设计、数据库操作、网络编程、文件处理等。同时 Qt 还是一个非常灵活的框架，Qt 可以允许使用者通过 C++ 语言来开发程序，开发者可以根据自己的需求使用。Qt 框架经过与时俱进地优化，可以提供更高性能的应用程序，与此同时 Qt 还提供了大量的内置工具和 API，使 Qt 适用于各种开发。Qt 还包括了一个庞大的类库，用户可以利用这些类使得开发过程更加便捷。Qt 还具有良好的文档和社区支持，因此广泛用于各种类型的应用程序开发，包括桌面应用、移动应用、嵌入式系统等，因此选用 Qt 框架开发。

大语言模型上下文理解能力强，通过预训练，可以学习到庞大的语言知识和语义关系，能够更好地理解和生成文本，但同时也需要大量的计算资源和数据，TF-IDF 模型简单高效，适用于文本检索任务，但在语义理解和复杂任务上表现有限。而法律学习智能助理需要迅速访问各种法律数据库和文献资源，需要能够理解用户提出的法律问题，并结合法律知识进行推理、分析和判断，因此选择大语言模型来实现法律学习智能助手的人机交互。

第3章 需求分析

3.1 可行性分析

随着人工智能技术的迅速发展，人工智能技术被应用到了越来越多的领域。把人工智能技术应用到法律学习领域将能很好地发挥计算机技术处理效率更高的优势，更好地帮助法律学习者学习法律，是法律学生学习更加便捷。法律学习智能助手是否可行是在正式开发设计项目之前必须考虑的问题。评估一个项目要从很多方面进行，本节主要是从技术方面、经济方面、社会方面和操作方面四个方面对法律学习智能助手进行可行性分析。

3.1.1 技术可行性

Qt 是一个由 Qt Company 开发的跨平台 C++ 图形用户界面应用程序开发框架。它既可以开发 GUI 程序，也可用于开发非 GUI 程序，比如控制台工具和服务器。Qt 是一个面向对象的框架，使用特殊的代码生成扩展，Qt 很容易扩展，并且允许真正的组件编程。Qt 的良好封装机制使 Qt 的模块化程度非常高，可重用性较好，对于用户开发来说是非常方便的。Qt 提供了一种称为 signals/slots 的安全类型来替代 callback，这使得各个元件之间的协同工作变得十分简单^[14]。

大语言模型上下文理解能力强，通过预训练，可以学习到庞大的语言知识和语义关系，能够更好地理解和生成文本，大语言模型在预训练阶段使用了大规模的通用语料，大语言模型旨在经过大量数据训练，像人类一样理解和生成文本以及其他形式的内容。这种模型有能力从环境中推断问题的能力，生成连贯并且还与环境相关的响应，翻译成英语以外的语言，总结文本，回答问题，甚至可以协助完成创造性写作或代码生成任务，还可以处理复杂的自然语言问题。

GPT 是一种基于 Transformer 架构的大语言模型。GPT 中文意为生成式预训练变换器。自然语言处理可以解决一些通用的问题，分为两个关键步骤：自然语言理解（NLU）和自然语言生成（NLG）。自然语言理解是指将人类的语言转化成为机器可以理解和处理的形式，它涉及对文本进行语义分析和语法分析，来提取出其中的意义和结构，NLU 的主要目标是使计算机能够更好的理解和解释人类的语言。自然

语言生成是指根据计算机内部的信息和逻辑规则，生成自然语言文本的，目标是使计算机能够以自然的方式生成易于人类理解的语言输出。

3.1.2 经济可行性

Qt 框架经过与时俱进的发展已经很成熟，可以直接利用，开发成本低。此外，对大语言模型的利用使得法律学习更加便捷，减少了时间成本与经济成本，在经济方面是可行的。

3.1.3 社会可行性

在人工智能蓬勃发展的今天，生活中处处离不开人工智能的影子，人工智能已经被广泛应用，同时由于大语言模型的强大的数据处理能力使得学习更加简单，因此在社会方面是可行的。

3.1.4 操作可行性

Qt 框架经过发展可以使用户更加方便地使用通过 QtCreator 开发的程序，可以时用户的操作更加简便，不存在操作方面的困难。

3.2 需求分析

需求分析是进行系统开发设计的基础，需求分析的调研将会直接影响到后期系统的设计。本章首先对法律学系智能助手进行了可行性分析，从经济方面、社会方面、技术方面和操作方面四个方面分析了法律学习智能助手的实现是否可行，然后对系统需要达到的基本目标进行了分析，最后又进行了需求分析。

3.2.1 功能需求分析

（1）用户信息管理。使用本法律学习智能助手需要先进行注册登录，本法律学习智能助手不支持游客登录，防止非法用户以及非法访问对数据库的破坏。使用本法律学习智能助手首先需要注册用户名以及密码，注册合法的用户名和密码后会被写入数据库，登录时需要输入和数据库中对应的用户名和密码即可使用本法律学习智能助手。用户登录后可以看到自己的信息，此外用户还可以修改自己的密码。法律学习智能助手用户信息管理功能用例图如图 3-1 所示。

(2) 用户权限管理。本法律学习智能助手的使用者包括管理员，法律学生和客服，在登录时需要选择身份。管理员可以对法律学生的账号进行封禁和解封，被封禁的法律学生不能在论坛中发言，若法律学生对封禁行为有异议，可以向客服了解情况，客服再去联系管理员。客服没有给法律学生解封账号的权限。法律学习智能助手用户权限管理功能用例图如图 3-2 所示。

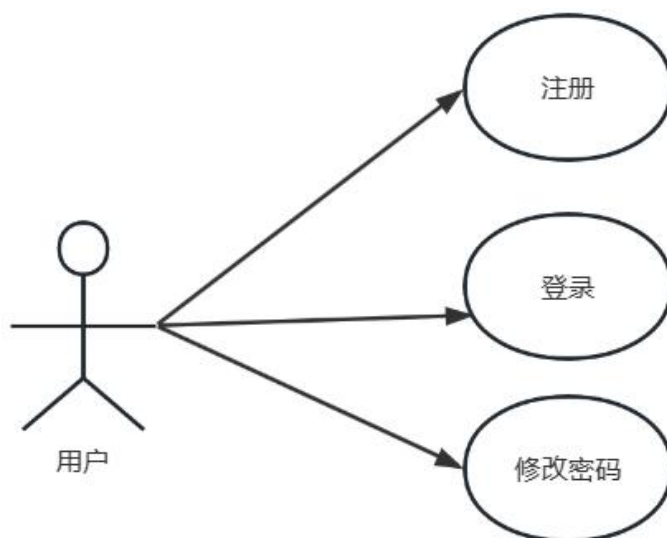


图 3-1 法律学习智能助手用户信息管理用例图

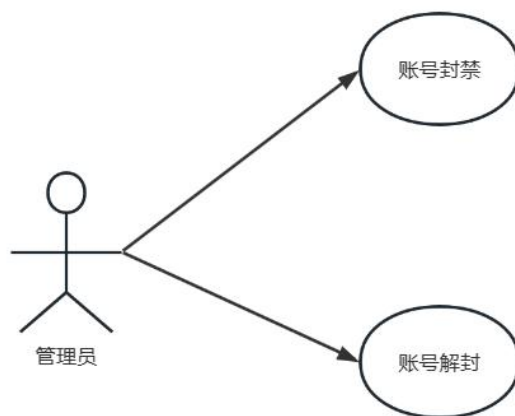


图 3-2 法律学习智能助手用户权限管理用例图

(3) 论坛。法律学生可以在论坛上发表话题，其他学生可以进行回复，发表话题的法律学生可以删除回复，所有学生都可以举报话题和回复，举报信息会发送给管理员，管理员经过核实后可以对被举报账号进行封禁。法律学习智能助手论坛功能用例图如图 3-3 所示。

(4) 查询法律条文。法律学生可以向法律学习智能助手提问某部法律的具体某

条法律的内容，由于法律条文具有专业性，学生可能不理解法律条文含义，可以向法律学习智能助手继续提问法律条文的解释，当法律学生向系统继续提问法律条文的详细解释之后，本法律学习智能助手可以给出具体解释。法律学习智能助手查询法律条文功能用例图如图 3-4 所示。

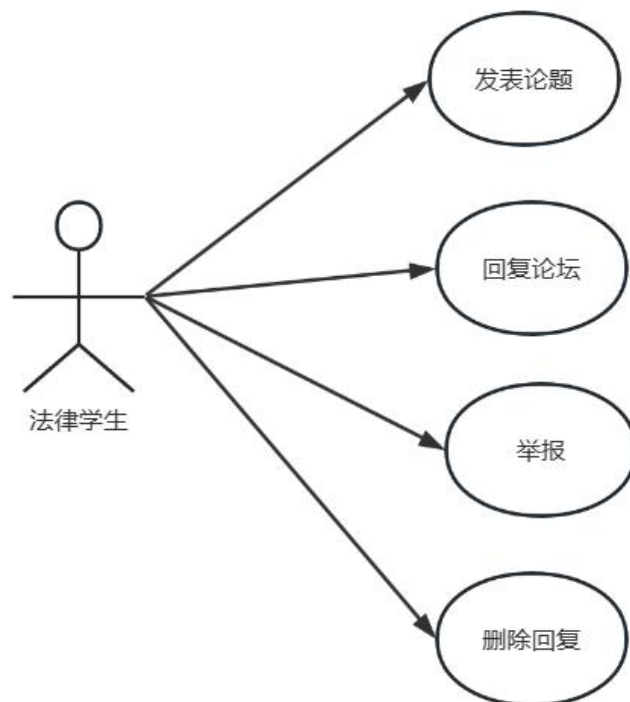


图 3-3 法律学习智能助手论坛功能用例图

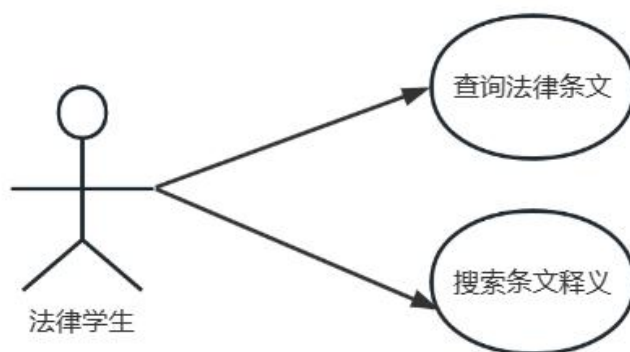


图 3-4 法律学习智能助手查询法律条文功能用例图

(5) 咨询法律案件。法律学生可以向法律学习智能助手询问法律案件的处理依据，法律学习智能助手可以给法律学生解释案件的宣判原因。法律学习智能助手咨询法律案件功能用例图如图 3-5 所示。

(6) 用户自测。法律学习智能助手可以给法律学生出题，学生做题之后法律学

习智能助手可以帮学生批改错题，还可以收集学生的错题帮助学生日后的复习。法律学习智能助手用户自测功能用例图如图 3-6 所示。

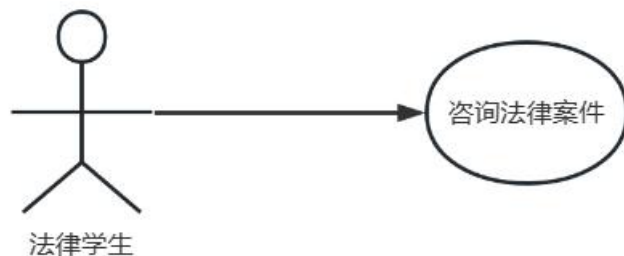


图 3-5 法律学习智能助手咨询法律案件功能用例图

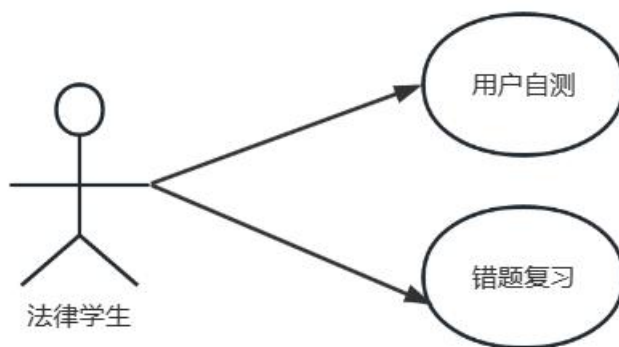


图 3-6 法律学习智能助手用户自测功能用例图

3.2.2 非功能需求分析

非功能需求和功能需求是从不同的角度出发进行考虑的。对于功能需求来说，系统的可用性是功能性需求考虑的主要对象。而非功能需求考虑的是系统的易用性，下面将从性能需求、技术环境需求以及安全需求等三方面对法律学习智能助手的非功能需求进行分析。

(1) 性能需求。首先要求本法律学习智能助手对用户操作的响应速度要快，需要选用适当的大模型，需要具有较高的性能，其次法律条文具有专业性，需要将机器语言转化为专业的法律名词，最后在开发本法律学习智能助手时应该尽量降低法律学习智能助手的操作复杂度和难度，并且能够保障本法律学习智能助手可以长期稳定运行，在遇到故障时本法律学习智能助手可以快速响应故障，保障本法律学习智能助手的可用性和易用性。

(2) 软硬件环境需求。QtCreator 是一种跨平台的轻量级集成开发环境，支持跨

平台运行，支持 Linux，Mac 和 Windows。QtCreator 可以让使用它的人能够利用 Qt 这个框架更加便捷地完成开发任务。Qt 框架提供了许多功能和工具，其中包括图形用户界面设计、数据库操作、网络编程、文件处理等。Qt 还具有良好的文档和社区支持，因此广泛用于各种类型的应用程序开发，包括桌面应用、移动应用、嵌入式系统等^[15]。SQLite 不需要一个单独的服务器进程或者操作，也不需要安装和管理，一个完整的 SQLite 数据库可以存储在一个跨平台的磁盘文件中，并且 SQLite 量级较轻，还可以自给自足，不需要外部依赖，同时 SQLite 还支持 SQL92 标准的查询语言功能，此外还提供了简单和易于使用的 API，方便用户使用。

（3）安全需求。法律学习智能助手需要具备较高的安全性，防止使用本法律学习智能助手的法律学生的信息被私自篡改。首先在用户登录时需要进行用户的权限的划分，防止非法用户进入，同时对可以使用本法律学习智能助手的人员也有着泾渭分明的权限。

3.3 本章小结

本章从经济可行性、技术可行性、社会可行性等多个方面对法律学习智能助手的可行性进行了分析，分析了法律学习智能助手应该具备的功能，之后进行需求分析，并且还配备了用例图对本法律学习智能助手的功能性需求分析进行了详细的介绍，之后也从性能、安全等方面对本法律学习智能助手进行了非功能性的需求分析。

第4章 总体设计

本章主要描述了本法律学习智能助手的设计流程，在设计流程的基础上提出了在法律学习智能助手中应该遵循的基本原则，之后用流程图对各个快进行了具体的设计，最后还给出了数据库设计，其中数据库设计包括数据库实体设计，模块 ER 图设计，以及数据库表设计，最后是本章小结对本章的总结。

4.1 系统设计原则

本法律学习智能助手的设计时为了帮助法律学生更加方便快捷地学习专业法律知识，所以系统设计时需要考虑用户也就是法律学生的实际需求，以大语言模型为核心，以方便快捷为目的来进行，与此同时，系统的管理与更新、扩展和维护也需要考虑，因此要考虑本法律学习智能助手的易用性、实用性等，以下内容时法律学习智能助手进行设计的几个基本原则。

（1）易用性。法律学习智能助手的目的是让用户学习法律更加方便快捷，因此本法律学习智能助手需要用更简单的操作来完成更复杂的任务，要让使用者可以更加快速地学会如何使用本法律学习智能助手，在设计界面时主要功能要更加突出，方便用户可以更好地使用。

（2）实用性。设计法律学习智能助手需要从用户的实际需求出发，需要把需求分析作为前提基础并且还要结合具体生活需要来设计本法律学习智能助手要实现的功能，还要确保设计的功能可以很好地满足用户的需求。

（3）稳定性。设计的法律学习智能助手需要在使用过程中可以稳定地运行，因此在系统功能设计任务完成之后，还需要经过预训练和微调来保证设计的功能可以按照设计预期来实现。

（4）技术先进。法律学习智能助手的实现必须采用先进的技术，本法律学习智能助手采用 Qt 框架，使用 QtCreator 进行编程，并结合 SQLite 数据库，都是当前在

（5）安全性。在使用法律学习智能助手时需要保证个人信息等数据的安全可靠，管理员与普通用户即法律学生设置不同的权限，不同角色用户身份要拥有不同的权限，防止出现权限交叉。

（6）可扩展性。随着后期对法律学习智能助手的使用，还需要增加更多功能使

法律学生学习法律更加方便快捷。

4.2 总体功能设计

本法律学习智能助手的使用者主要有管理员，法律学生以及客服。

客服主要用来接收来自法律学生的申诉，可以和法律学生和管理员进行交流，避免管理员和法律学生直接的接触，管理员和客服的账号由系统分配，不能注册管理员和客服的账号，注册时只能注册学生的账号。客服的功能结构图如图 4-1 所示。

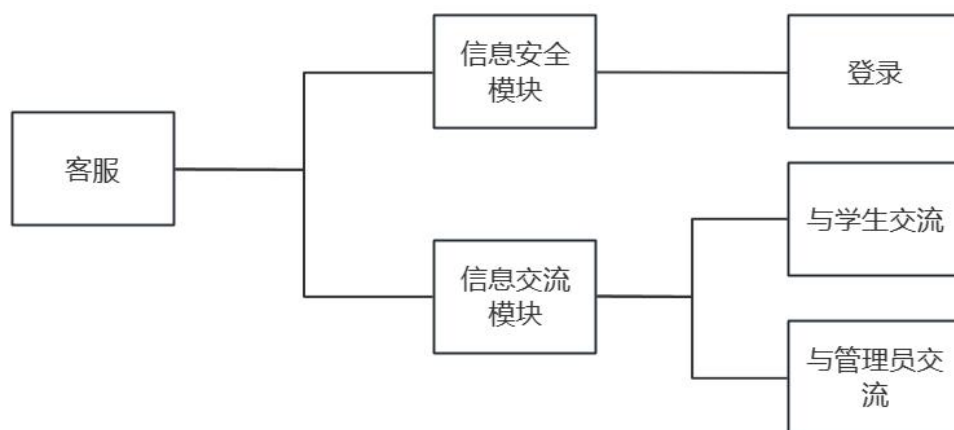


图 4-1 功能结构图（客服）

法律学生在注册登录后进入个人信息页面，可以完善或修改个人信息，也可以修改密码，但账号经过注册后不可修改。法律学生还可以在论坛发表论题，也可以回复别人的论题，发表论题的学生有删除回复的权限，所有学生都有举报发言的权限，举报信息会直接发送给管理员，管理员审核后可以对被举报账号进行封禁，被封禁的账号不能再在论坛发言。法律学生还可以查询自己想知道的法律条文，比如某一部法律第几条的内容是什么，由于法律条文专业性较强，本法律学习智能助手还可以对法律条文进行解释。法律学生还可以查询法律案件，针对某一个行为查询量刑原因和判罚的依据。法律学生还可以通过本法律学习智能助手查看相关法律文献。本法律学习智能助手还可以给法律学生出题考试，并且收集学生的错题以方便学生日后的复习。法律学生的功能结构图如图 4-2 所示。

管理员可以进行对用户账号的封禁和解封，管理员还可以进行公告的发布，比如封禁账号时需要发布公告说明封禁的账号以及封禁原因。管理员的功能结构图如

图 4-3 所示。



图 4-2 功能结构图（法律学生）

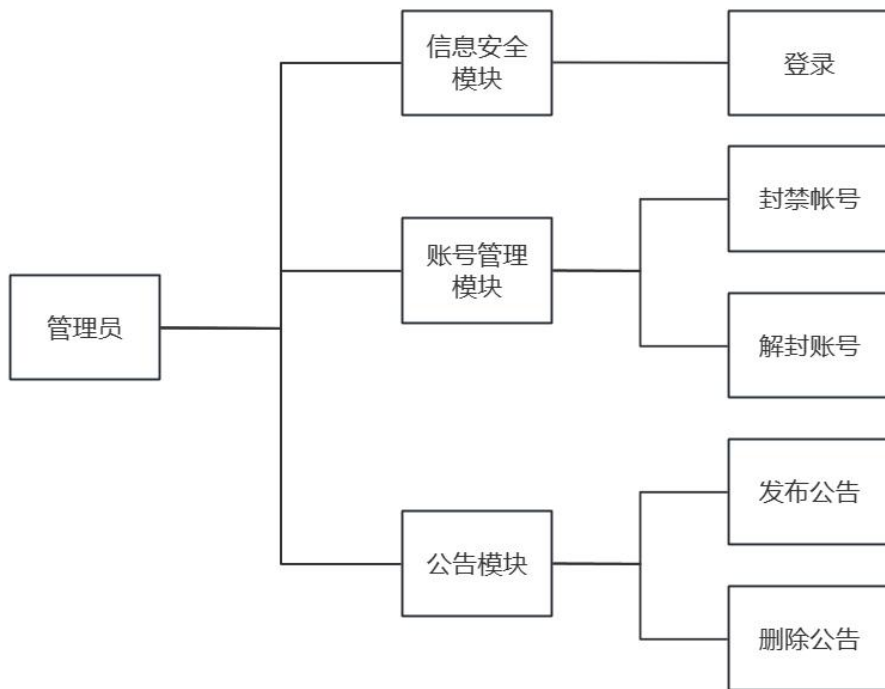


图 4-3 功能结构图（管理员）

4.3 数据库设计

4.3.1 数据库实体

在概念模型设计中，E-R 模式是最常用的一种，本小节主要介绍了 12 个实体属性，具体实体属性如下。

（1）管理员的实体属性包括账号，密码，用户名，电话和邮箱，其中账号、密码和用户名不可为空，电话和邮箱可以为空，管理员实体属性如图 4-4 所示。

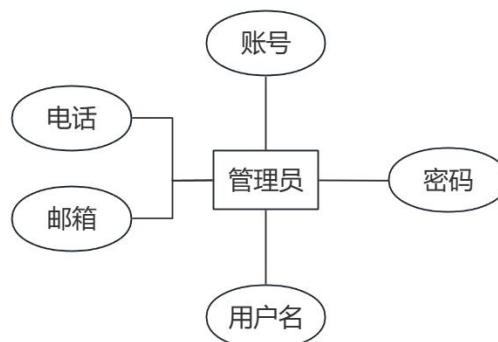


图 4-4 管理员实体属性图

(2) 客服的实体属性包括账号, 密码, 用户名, 电话和邮箱, 客服的账号, 密码和用户名不可为空, 电话和邮箱可以为空, 客服实体属性如图 4-5 所示。

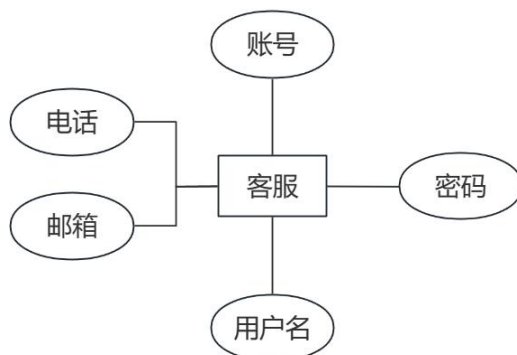


图 4-5 客服实体属性图

(3) 法律学生的实体属性包括账号, 密码, 用户名, 帐号状态, 电话和邮箱, 其中学生的账号, 密码和用户名不可为空, 电话和邮箱可以为空, 法律学生实体属性如图 4-6 所示。

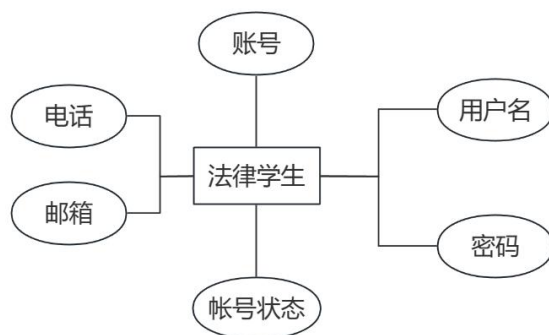


图 4-6 法律学生实体图

(4) 论坛的实体属性包括话题, 话题的时间以及论坛的回复, 论坛实体属性如图 4-7 所示。

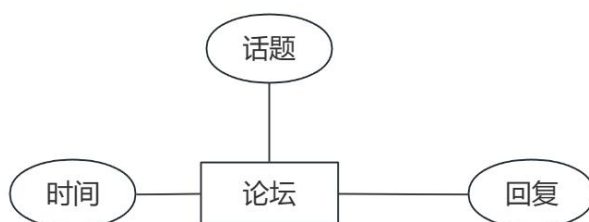


图 4-7 论坛实体图

（5）题库的实体属性包括题库的序号，内容，状态和类型，题库实体属性如图 4-8 所示。

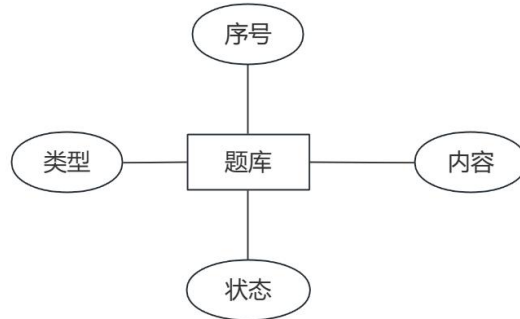


图 4-8 题库实体图

（6）错题集的实体属性包括错题集的序号，内容，状态，类型和时间，错题集实体属性如图 4-9 所示。

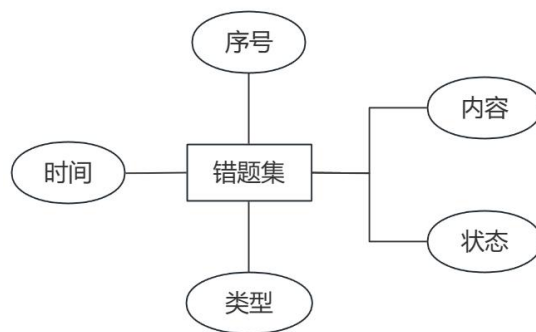


图 4-9 错题集实体图

（7）法律条文的实体属性包括序号，内容，状态，类型和时间，法律条文实体属性如图 4-10 所示。

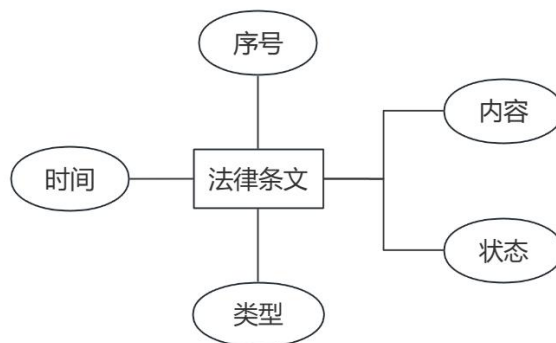


图 4-10 法律条文实体图

(8) 公告的实体属性包括公告的序号，内容以及时间，公告实体属性如图 4-11 所示。

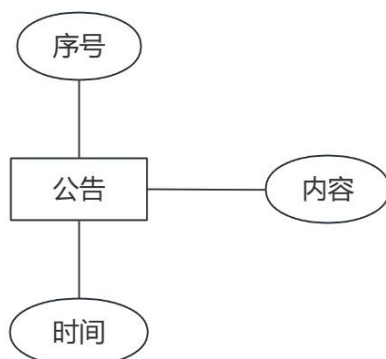


图 4-11 公告实体图

(9) 法律文献的实体属性包括法律文献的序号、内容和类型，法律文献实体属性如图 4-11 所示。

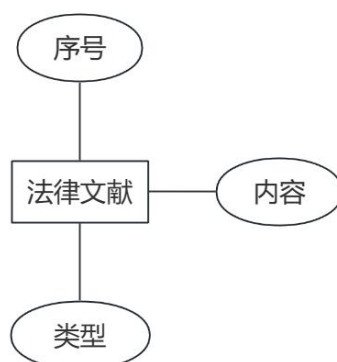


图 4-11 法律文献实体图

(10) 充值的实体属性包括时间、余额和记录，充值实体属性如图 4-12 所示。

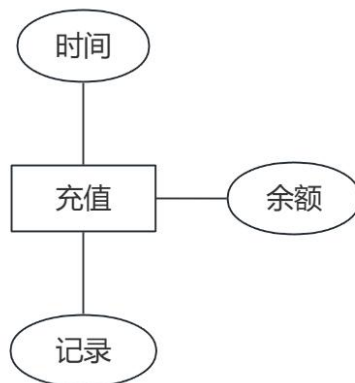


图 4-12 充值实体图

（11）会员的实体属性包括账号、密码、余额和用户名，会员实体属性如图 4-13 所示。

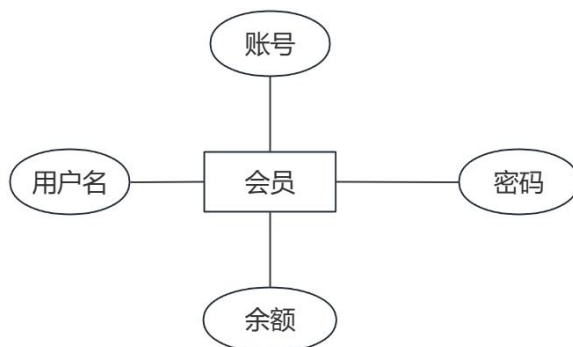


图 4-13 会员实体图

（12）收藏夹的实体属性包括法律条文，法律文献，题目和错题，收藏夹实体属性如图 4-14 所示。

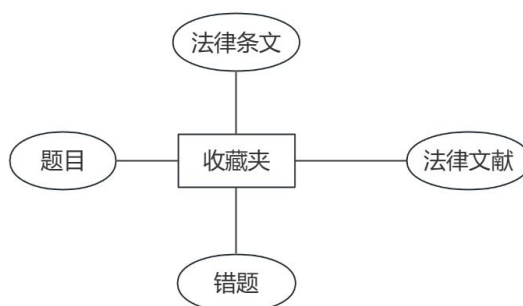


图 4-14 收藏夹实体图

4.3.2 模块 E-R 图

（1）管理员管理法律学生账户主要涉及到四个实体，管理员可以封禁和解封法律学生的账号，学生如果对封禁有异议的话可以向客服申诉，客服可以和管理员沟通，客服的存在可以避免管理员和学生之间的直接沟通/经过管理员再次审核后可以考虑解封法律学生的账号，如果再次审核后还决定封禁则需要发布公告，说明封禁原因。账号管理模块 E-R 图如图 4-15 所示。

（2）搜索法律条文模块主要涉及到两个实体，法律学生可以在法律学习智能助手搜索法律条文，还可以查看法律条文的解释，搜索法律条文模块 E-R 图如图 4-16 所示。

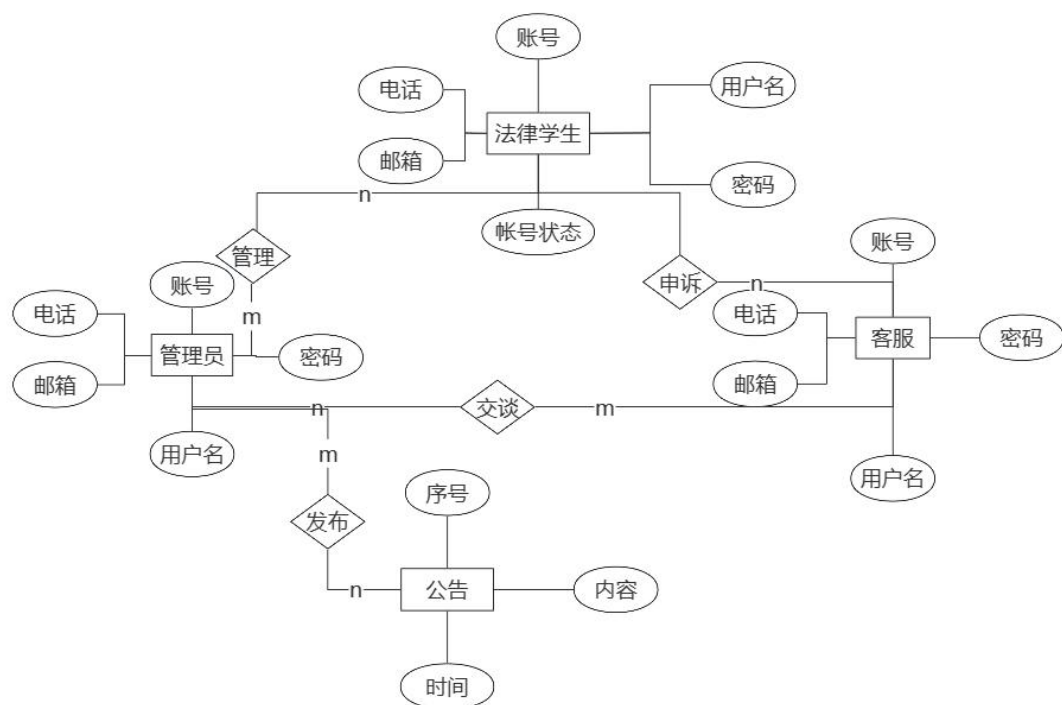


图 4-15 账号管理模块 E-R 图

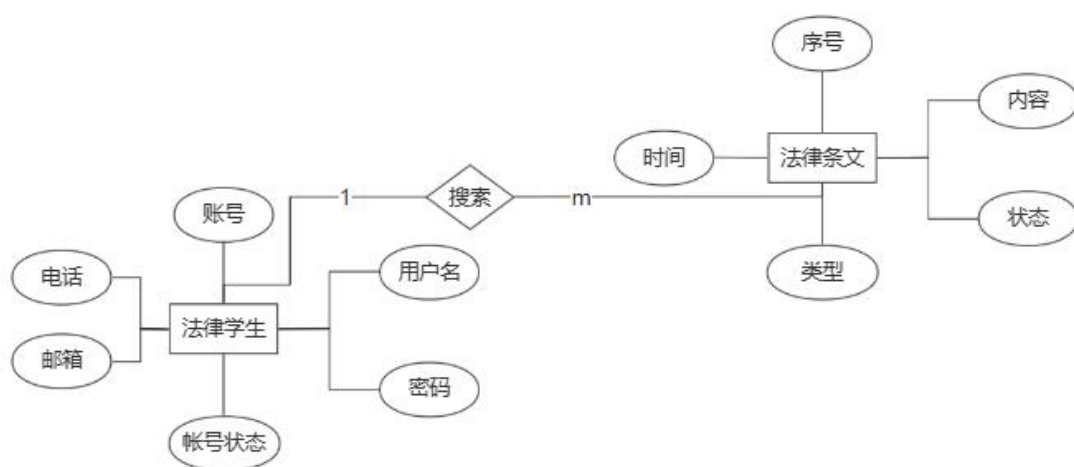


图 4-16 搜索法律条文模块 E-R 图

(3) 浏览法律文献主要涉及到两个实体，法律学生可以通过法律学习智能助手浏览法律文献，本法律学习智能助手会根据学生的需求推荐法律文献。浏览法律文献模块 E-R 图如图 4-17 所示。

(4) 用户自测模块主要涉及到三个实体，法律学生可以通过法律学习智能助手从题库中按照类型做题，法律学习智能助手可以帮法律学生批阅学生做的题，错题

会被法律学习智能助手记录，方便用户日后的复习。用户自测模块 E-R 图如图 4-18 所示。



图 4-17 浏览法律文献模块 E-R 图

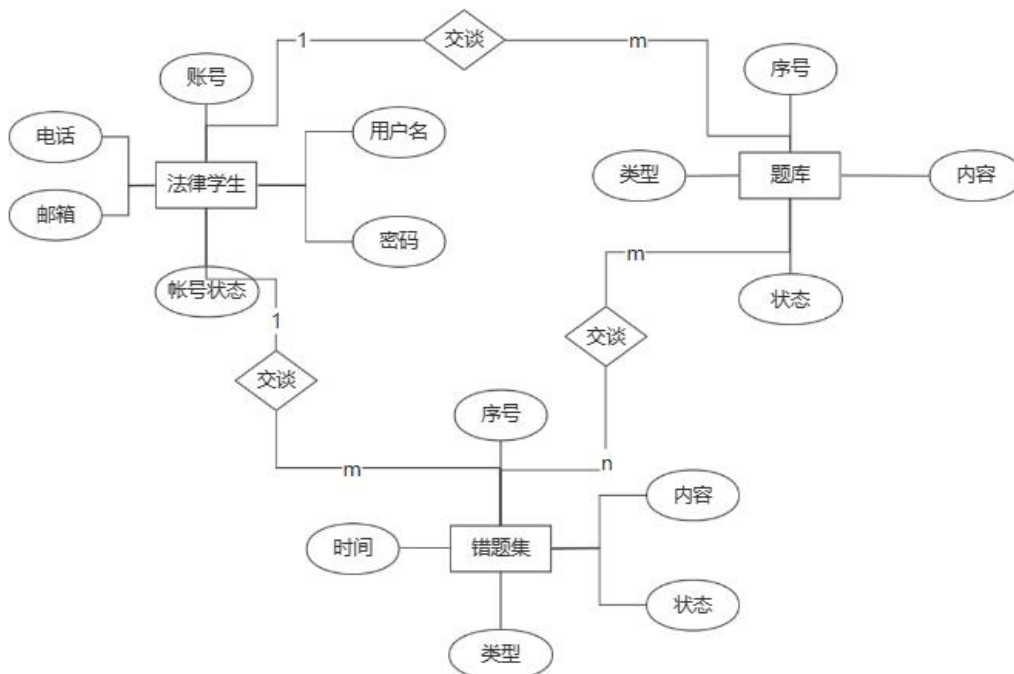


图 4-18 用户自测模块 E-R 图

4.3.3 数据库表设计

在法律学习智能助手的设计与实现中，数据库的设计是非常关键的，数据库模型影响着整个系统。本法律学习智能助手使用 SQLite 数据库进行数据的存储，来记录信息，共设计了 12 个数据库表，各个表的设计结果如下。

(1) 管理员信息表包含管理员账号，管理员密码，管理员昵称，管理员的电话号码，管理员的邮箱，其中管理员账号为主键，管理员的账号，密码和昵称都不能以为空，管理员信息表如表 4-1 所示。

表 4-1 管理员信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
adminid	varchar	15	是	否	管理员账号
adminpwd	varchar	15	否	否	管理员密码
adminname	varchar	20	否	否	管理员昵称
adminphone	int	11	否	是	管理员电话
adminmail	varchar	20	否	是	管理员邮箱

(2) 用户信息表包含用户账号，用户密码，用户昵称，用户的电话号码，用户的邮箱和用户的帐号状态，其中用户的账号为主键，用户的账号，密码和昵称都不能以为空，并且用户的电话号码必须为 11 位数字，电子邮箱的格式必须正确用户信息表如表 4-2 所示。

表 4-2 用户信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
userid	varchar	15	是	否	用户账号
userpwd	varchar	15	否	否	用户密码
username	varchar	20	否	否	用户昵称
userphone	int	11	否	是	用户电话
usermail	varchar	20	否	是	用户邮箱
userstate	bool	--	否	否	账号状态

(3) 题库信息表包含题目序号，题目内容，题目的状态，即题目是否作答过和题目类型，其中题目序号为主键，题目的序号，内容以及题目的状态都不能以为空，题库信息表如表 4-3 所示。

表 4-3 题库信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
questionid	int	5	是	否	题目序号
questiontopic	varchar	255	否	否	题目内容
questionstate	bool	--	否	否	题目是否作答过
questiontype	varchar	10	否	是	题目类型

（4）错题集信息表包含题目序号，错题题目，错题是否作答过，错题时间和题目类型，其中错题序号为主键，错题的题目需要，错题的题目还有错题的状态都不可以为空，错题集信息表如表 4-4 所示。

表 4-4 错题集信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
mistakeid	int	5	是	否	错题序号
mistaketopic	varchar	255	否	否	错题题目
mistakestate	bool	--	否	否	错题是否答对过
mistaketime	datetime	--	否	否	错题时间
questationtype	varchar	10	否	是	题目类型

（5）法律条文信息表包含法律条文序号，法律条文类型，法律条文内容，法律条文是否浏览过，以及浏览时间，其中法律条文的条纹序号为主键，法律条文信息表如表 4-5 所示。

表 4-5 法律条文信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
lawid	int	5	是	否	法律条文序号
lawtype	varchar	10	否	否	法律条文类型
lawtopic	varchar	255	否	否	法律条文内容
lawstate	bool	--	否	否	是否浏览过
lawtime	datetime	--	否	是	浏览时间

（6）论坛信息表包括论坛的话题，论坛的回复和话题的发布时间以及回复的时间，其中论坛的话题，论坛的回复以及话题的发布时间都不可以为空，论坛信息表如表 4-6 所示。

表 4-6 论坛信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
forumtopic	varchar	255	否	否	论坛话题
forumreply	varchar	255	否	否	论坛回复
forumtime	datetime	--	否	否	发布时间

(7) 客服信息表包含客服的账号, 客服的密码, 客服的昵称, 客服的电话和邮箱, 客服信息表如表 4-7 所示。

表 4-7 客服信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
customerid	varchar	15	是	否	客服账号
customerwd	varchar	15	否	否	客服密码
customername	varchar	20	否	否	客服昵称
customerphone	int	11	否	是	客服电话
customermail	varchar	20	否	是	客服邮箱

(8) 公告信息表包含公告序号, 公告内容和公告的发布时间, 公告信息表如表 4-8 所示。

表 4-8 公告信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
noticeid	varchar	15	是	否	公告序号
notice	varchar	15	否	否	公告内容
noticetime	datetime	--	否	否	发布时间

(9) 法律文献信息表包含文献序号, 文献内容和文献类型, 法律文献信息表如表 4-9 所示。

表 4-9 法律文献信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
informationid	varchar	15	是	否	文献序号
information	varchar	65535	否	否	文献内容
information type	varchar	20	否	否	文献类型

(10) 会员信息表包含会员账号, 会员密码, 会员昵称, 会员的余额, 会员信

息表如表 4-10 所示。

表 4-10 会员信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
memberid	varchar	15	是	否	会员账号
memberpwd	varchar	15	否	否	会员密码
membername	varchar	20	否	否	会员昵称
membervalue	int	4	否	否	会员余额

（11）收藏夹信息表包含收藏的法律条文，收藏的法律文献，收藏的题目和后舱的错题，收藏夹信息表如表 4-11 所示。

表 4-11 收藏夹信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
collaw	varchar	65535	否	否	收藏条文
colinfor	varchar	65535	否	否	收藏文献
colquestion	varchar	65535	否	否	收藏题目
colmis	varchar	65535	否	否	收藏错题

（12）充值信息表包含充值时间，用户的余额没充值记录，充值信息表如表 4-12 所示。

表 4-12 充值信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
chargetime	datetime	--	否	否	充值时间
value	varchar	4	否	否	余额
chargerecord	varchar	20	否	否	充值记录

4.4 本章小结

本章首先介绍了本法律学习智能助手的设计流程，在介绍完设计流程后又提出了在本法律学习智能助手的设计中应该遵循的原则，之后对法律学习智能助手进行了整体的体系结构和功能设计，之后用流程图对本法律学习智能助手的各个模块进行了具体的设计，最后还给出了本法律学习智能助手的数据库设计。

第 5 章 详细设计

该章从信息管理模块、论坛模块、充值模块、搜索法律条文模块、查看法律文献模块、用户自测模块六个模块分别对法律学习智能助手的设计与实现过程展开了详细叙述，并且还附带了各个模块的流程图对各个模块的功能实现进行了详细的介绍说明。

5.1 信息管理模块

(1) 用户登录

用户登录功能流程图如图 5-1 所示。

主要参与者：法律学生、管理员、客服。

功能相关条件：用户登录：用户分为管理员、客服和普通法律学生三种，管理员和客服拥有系统分配的账号和密码，登录前不需要注册，可以直接登录管理员和客服的账号，并且客服和管理员的账号不可以修改，但是普通用户即法律学生在登录之前需要先注册账号，注册时需要填写用户账号和密码，登录时需要输入对应的账号和密码，账号和密码输入正确后进入本法律学习智能助手，登录之后法律学生还可以完善自己的信息，也可以进行信息的修改，系统会根据不同的用户身份分配不同用户操作权限。

前置条件：法律学生已经注册完成账号。

系统的保证：避免非法用户进入法律学习智能助手，注册过后即可成为合法用户，就可以成功使用本法律学习智能助手。

使用过程：管理员、客服和普通法律学生登录窗口相同，但需要选择用户身份，选择了账号对应的身份即可以登录，管理员、客服和普通法律学生用户登录后跳转的页面不相同；

用户输入账号和密码，选择自己的身份，点击登录按钮；

法律学习智能助手验证输入的账号和密码是否正确，用户的账号是否和用户选择的身份对应，用户是否为合法用户；

若信息输入正确则可以进入本法律学习智能助手，本法律学习智能助手会根据用户的身份分配不同的权限，如果信息错误则会显示登录失败。

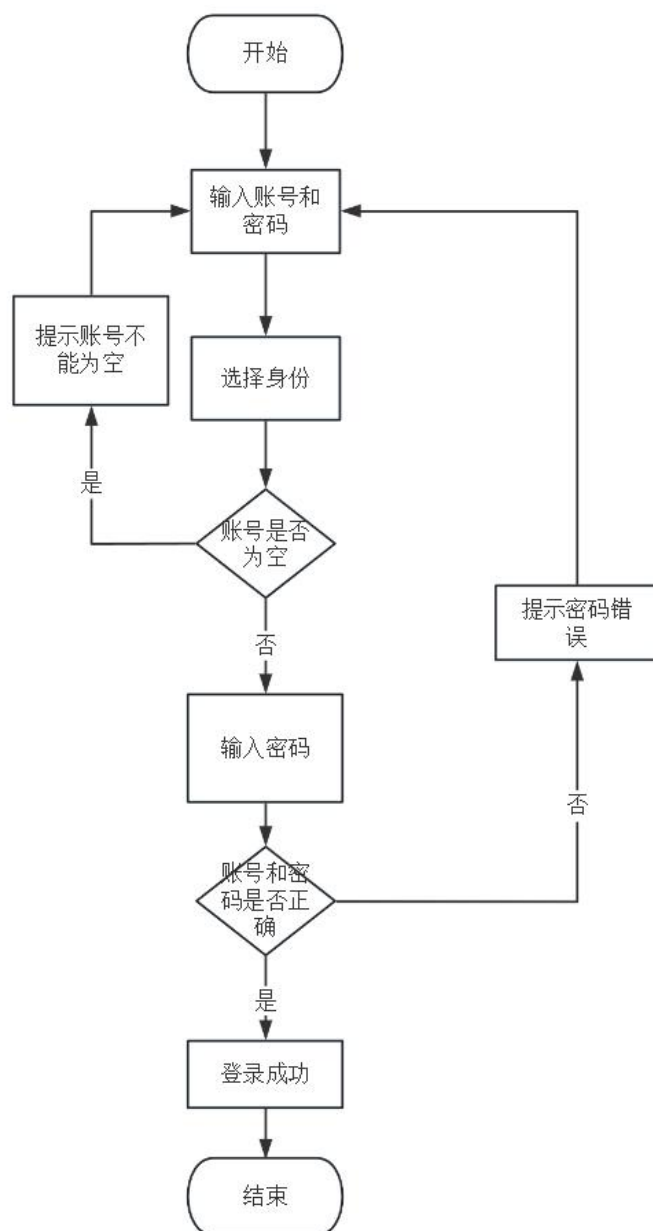


图 5-1 用户登录流程图

（2）密码修改

用户密码修改功能流程图如图 5-2 所示。

主要参与者：普通法律学生。

功能相关条件：用户登录：用户分为管理员、客服和普通法律学生三种，管理员和客服拥有系统分配的账号和密码，登录前不需要注册，可以直接登录管理员和客服的账号，普通用户即法律学生在登录之前需要先注册账号，注册时需要填写用户账号和密码，登录时需要输入对应的账号和密码，账号和密码输入正确后进入系

统，登录之后还可完善自己的信息，系统会根据不同的用户身份分配不同用户操作权限。

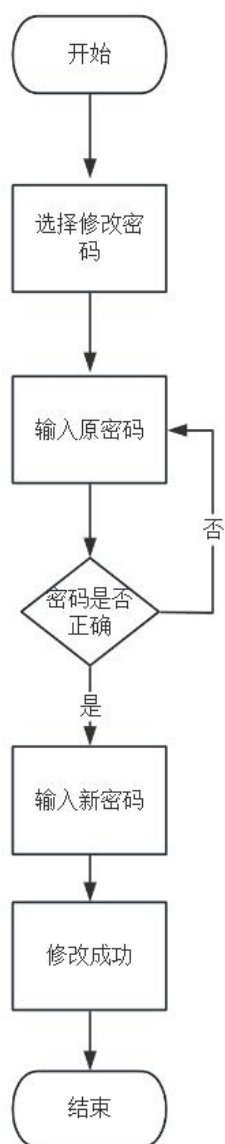


图 5-2 密码修改流程图

前置条件：法律学生已经完成登录。

系统的保证：修改密码后，旧密码无法登录本法律学习智能助手但新密码可以登录本法律学习智能助手。

使用过程：法律学生和管理员还有客服进入系统登录页面，管理员、律师和普通法律学生分别有对应的登录页面；

法律学生在成功登录后可以点击密码修改按钮来修改密码；

输入原本的旧密码和要使用的新密码；

若输入的旧密码正确，就可以进行密码修改，若输入的密码错误则提示用户密码输入错误，密码修改成功后下次登录新密码生效。

5.2 论坛模块

论坛模块流程图如图 5-3 所示。

论坛模块主要用于法律学生进行学术交流，在法律学生注册登录后可以使用论坛模块，法律学生可以在论坛模块发表话题，也可以在其他法律学生发表的话题下进行回复。若发表话题的学生发现不文明回复可以删除回复，发表话题的学生还可以删除自己发表的话题，但是其他法律学生不可以进行删除操作，其他同学也可以向管理员进行举报，管理员收到举报后可以向客服询问情况，客服可以登录系统并查看论坛，经查证后可以反馈给管理员，管理员可以对被举报者进行封禁，被封禁的用户不能在论坛发言。

本小节对论坛模块进行相关介绍。

主要参与者：管理员、客服和法律学生。

功能相关条件：用户登录：用户分为管理员、客服和普通法律学生三种，管理员和客服拥有系统分配的账号和密码，登录前不需要注册，可以直接登录管理员和客服的账号，普通用户即法律学生在登录之前需要先注册账号，注册时需要填写用户账号和密码，登录时需要输入对应的账号和密码，账号和密码输入正确后进入系统，登录之后还可完善自己的信息，系统会根据不同的用户身份分配不同用户操作权限。

前置条件：管理员、客服和法律学生登录本法律学习智能助手

系统的保证：被封禁帐号的用户不能在论坛发言

使用过程：法律学生需要首先进行注册登录；

法律学生可以浏览论坛，在论坛中发表话题以及对感兴趣的话题进行回复；

如果遇到不文明的话题或者回复，法律学生可以向管理员进行举报；

管理员可以向客服核实举报信息，然后对被举报账号进行封禁，被举报的用户有异议可以向客服申诉；

客服会根据申诉信息联系管理员，经过核实后可以管理员需要进行账号的解封，解封后的用户可以进行浏览论坛以及可以在论坛发言，管理员还可以进行公告的发布来说明封禁和解封的原因。

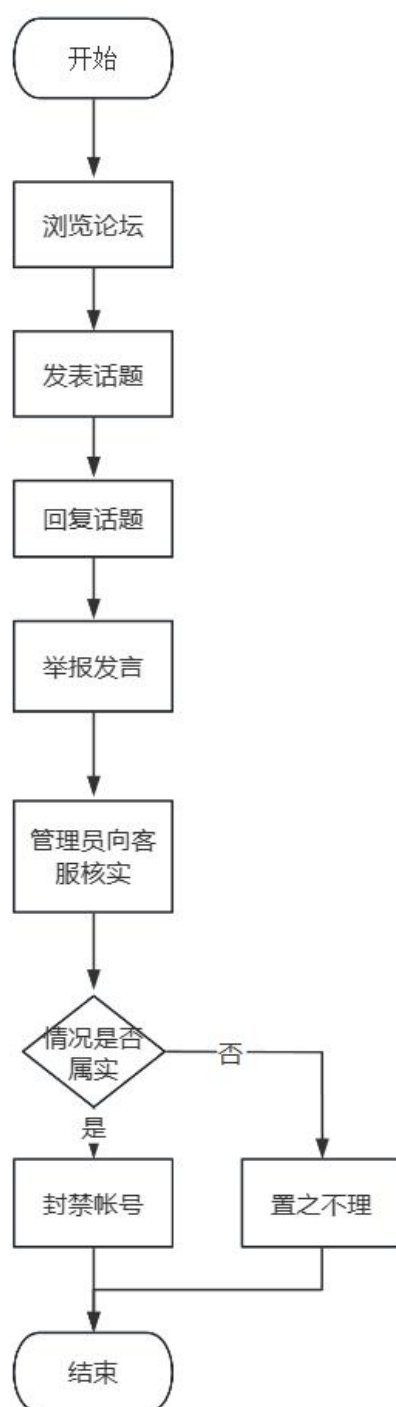


图 5-3 论坛模块流程图

5.3 充值模块

充值功能流程图如图 5-4 所示。

主要参与者：法律学生。

功能相关条件：用户登录：用户分为管理员、客服和普通法律学生三种，法律学生登录后可以使用充值功能。

前置条件：法律学生完成用户登录。

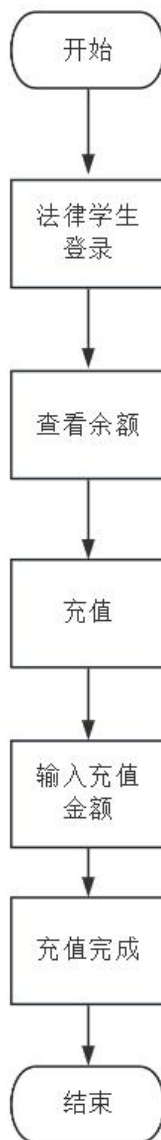


图 5-4 充值模块流程图

使用过程：法律学生登录后可以查看自己的余额，可以自愿进行充值来使用更多功能。

充值模块主要用于进行账号的充值，法律学生使用本法律学习智能助手时需要按月缴费，缴费之后才可以使用搜索法律条文，查看法律文献，用户自测等功能，如果不充值则只能使用论坛等功能。

5.4 搜索法律条文模块

搜索法律条文模块的流程图如图 5-5 所示。

主要参与者：法律学生。

功能相关条件：用户登录：用户分为管理员、客服和普通法律学生三种，法律学生登录后可以使用搜索法律条文功能。

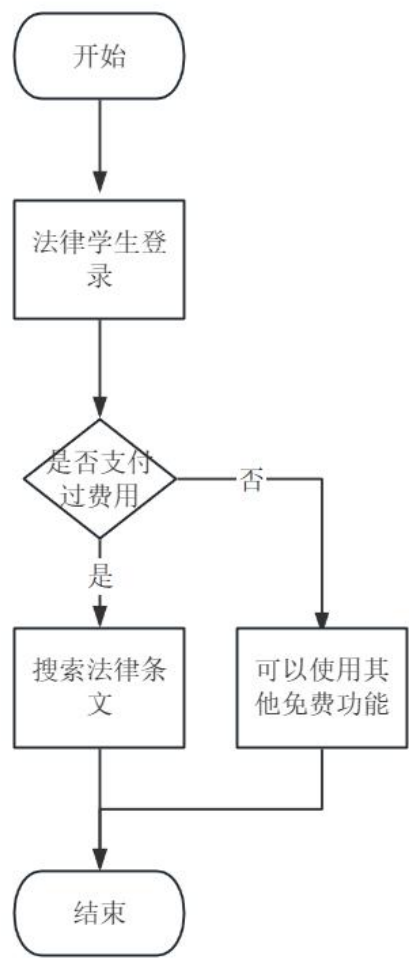


图 5-5 搜索法律条文模块的流程图

前置条件：用户登录。

系统的保证：若用户支付过本月使用本法律学习智能助手费用即可以使用搜索法律条文的功能。

使用过程：法律学生需要先进行登录本系统；

登录后若法律学生支付过使用本系统的费用，就可以使用搜索法律条文的功能，若没有支付过费用则需要支付才可以使使用本功能或者使用其他免费功能。

搜索法律条文模块可以让法律学生使用本法律学习智能助手搜索具体的法律条文，因为法律条文具有较强的专业性，因此法律学生也可以查看本法律学习智能助手对法律条文进行的解释。

5.5 查看法律文献模块

查看法律文献模块的流程图如图 5-6 所示。

主要参与者：法律学生。

功能相关条件：用户登录：用户分为管理员、客服和普通法律学生三种，法律学生登录后可以使用搜索法律条文功能。

前置条件：用户登录。

系统的保证：若用户支付过本月使用本法律学习智能助手费用即可以使用查看法律文献的功能。

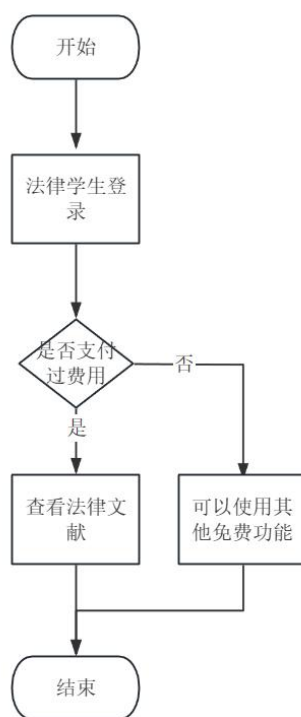


图 5-6 查看法律文献模块的流程图

使用过程：法律学生需要先进行登录本系统；

登录后若法律学生支付过使用本系统的费用，就可以使用查询法律文献的功能，若没有支付过费用则需要支付才可以使用本功能或者使用其他免费功能。

查看法律文献模块可以让法律学生使用本法律学习智能助手查看具体的法律文献内容，法律学生可以通过本法律学习智能助手直接搜索法律文献，可以让学生更加全面地理解法律条文的含义以及应用场景，可以参考其他人对法律条文的理解，方便了法律学生进行法律学习。

5.6 用户自测模块

本法律学习智能助手可以帮助法律学生进行用户自测。首先法律学生选择用户自测功能后本法律学习智能助手可以给用户出题，法律学生可以自由选择题目类型以及题目的范围，在法律学生输入想要回答的题目类型和题目的范围后，本系统可以根据法律学生的需求为法律学生生成相应的题目，法律学生答完题之后本系统可以给法律学生进行批阅，本法律学习智能助手会生成标准的答案提供给法律学生进行参考，法律学生可以选择收藏回答错误的题目或者有价值的题目，本法律学习智能助手可以收集法律学生回答错的题目，可以方便法律学生进行日后的复习。

用户自测流程图如图 5-7 所示。

本小节对用户自测模块的流程进行相关介绍。

主要参与者：法律学生。

功能相关条件：户登录：用户分为管理员、客服和普通法律学生三种，法律学生登录后可以自由选择使用搜索法律条文功能，咨询法律案件功能，查询法律文献功能以及用户自测功能。

前置条件：法律学生进行用户登录，在法律学生登录本系统后选择进行用户自测功能。

系统的保证：本法律学习智能助手可以根据法律学生的需求为法律学生生成相应的题目，法律学生答完题之后本系统可以给法律学生进行批阅，本法律学习智能助手会生成标准的答案提供给法律学生进行参考，法律学生可以选择收藏回答错误的题目或者有价值的题目，方便学生日后的复习。

使用过程：法律学生登录系统后即可进行用户自测；

法律学生可以根据自己的需求生成对应的题目类型；

法律学生可以选择自己想要作答的题目范围；

法律学生作答完题目后本法律学习智能助手会为用户进行批阅试题；

法律学生可以对试题进行收藏方便日后进行复习。

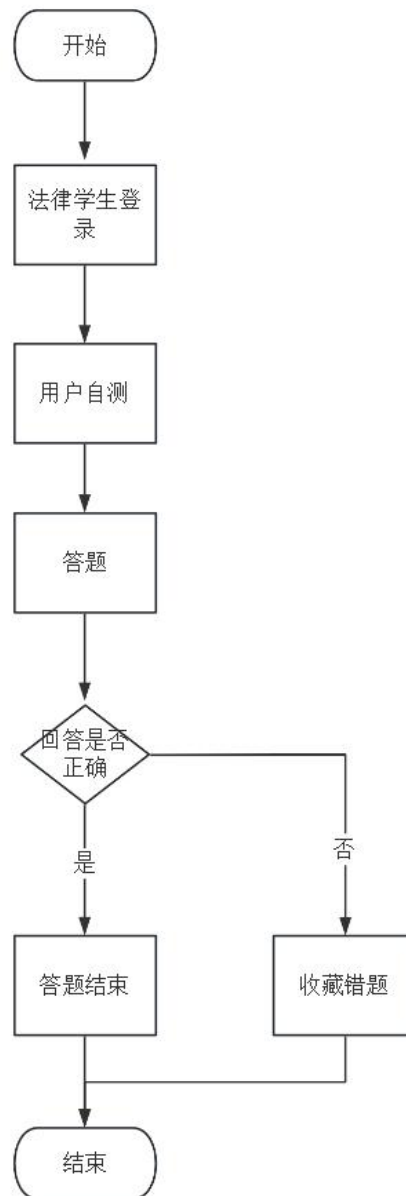


图 5-7 用户自测流程图

5.7 本章小结

本章详细介绍了系统的功能结构，数据库实体和数据库表。本章首先介绍了本系统设计应遵循的几条原则，然后以用户对法律学习智能助手的需求为基础，对系统应实现的功能进行设计，按模块划分功能，并据此画出系统的功能模块图，紧接着进行数据库设计，画出数据库实体属性图和 ER 图并据此设计数据库表，最后按照模块对主要的各功能以流程图的形式进行详细设计阐述。

第 6 章 系统实现

本章主要是在系统设计的基础上来完成系统实现，并对系统实现的主要功能进行测试，以便进一步丰富和完善系统。首先对系统的软件开发与运行环境进行了介绍，然后对系统中的重点功能实现进行了说明并展现系统实现效果图。

6.1 环境配置

法律学习智能助手采用 Qt 框架,使用 QtCreator 开发环境和 SQLite 数据库设计。QtCreator 是一种跨平台的轻量级集成开发环境,支持跨平台运行,支持 Linux, Mac 和 Windows。QtCreator 可以让开发人员能够利用 Qt 这个框架更加便捷地完成开发任务。Qt 框架提供了许多功能和工具,其中包括图形用户界面设计、数据库操作、网络编程、文件处理等。Qt 还具有良好的文档和社区支持,因此广泛用于各种类型的应用程序开发,包括桌面应用、移动应用、嵌入式系统等。SQLite 不需要一个单独的服务器进程或者操作,也不需要安装和管理,一个完整的 SQLite 数据库可以存储在一个跨平台的磁盘文件中,并且 SQLite 量级较轻,还可以自给自足,不需要外部依赖,同时 SQLite 还支持 SQL92 标准的查询语言功能,此外还提供了简单和易于使用的 API,方便用户使用。法律学习智能助手的软件环境如下。

操作系统: Windows11

数据库: SQLite

服务器: Tomact

软件开发工具: QtCreator

开发语言: C++

6.2 关键功能的实现

6.2.1 用户登录

用户登录是保障法律学习智能助手中的数据安全的屏障,输入用户名和密码并选择用户身份,在数据库中存在且相匹配即可成功登录,如果用户第一次使用本法律学习智能助手还没有账号,可以先选择注册,注册后可以拥有本法律学习智能助

手的账号和密码，之后在进行登录，用户登录界面图如图 6-1 所示。



图 6-1 用户登录界面图

6.2.2 搜索法律条文

搜索法律条文是本法律学习智能助手最主要的功能之一，用户登录并达成使用条件后可以使用。用户在登录后可以点击搜索法律条文按钮，在下方输入框输入想要搜索的法律条文，点击确定后即可输出本法律学习智能助手提供的内容。搜索法律条文如图 6-2 所示。

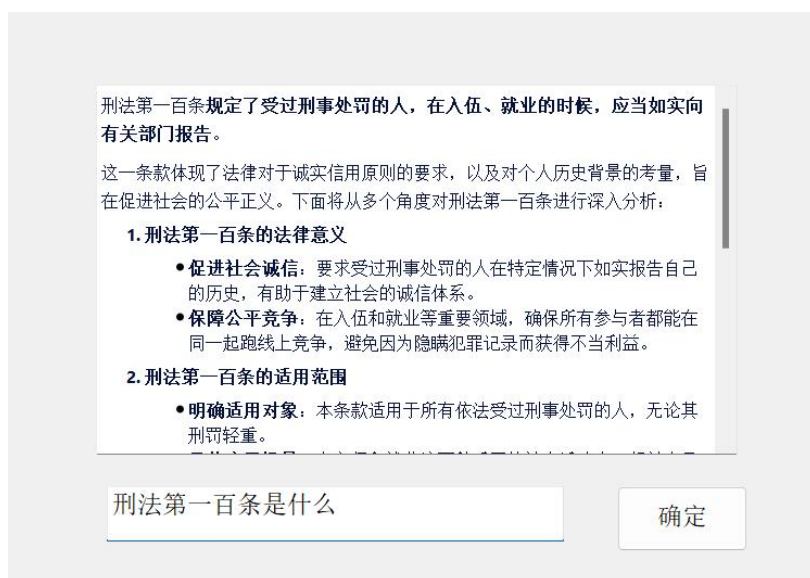


图 6-2 搜索法律条文界面图

6.2.3 咨询法律案件

咨询法律案件是本法律学习智能助手最主要的功能之一，用户登录并达成使用

条件后可以使用。用户在登录后可以点击咨询法律案件按钮，在下方输入框输入想要咨询的法律案件，点击确定后即可输出本法律学习智能助手提供的内容。咨询法律案件如图 6-3 所示。

酒驾，即饮酒后驾驶机动车的行为，在我国法律中被严格禁止并对此类行为给予严厉的刑事处罚。根据《刑法修正案（八）》的规定，醉酒驾驶机动车的行为可被处以拘役并处罚金，这是对公共安全的保护以及对交通法规的严格执行。酒驾被判三年通常指的是因醉酒驾驶造成严重后果或多次酒驾累犯的情况，下面将从多个角度进行深入分析：

1. 法律规定

- **法律条文：**根据我国法律规定，醉酒驾驶属于危险驾驶罪，可处以拘役并处罚金。
- **刑罚依据：**酒驾判刑的具体年限依据驾驶行为的危险性、造成的后果以及是否存在累犯等情况而定。

2. 司法解释

- **量刑标准：**最高人民法院等部门会对酒驾案件的量刑标准给出具体

为什么酒驾判了三年

确定

图 6-3 咨询法律案件界面图

6.2.4 咨询法律问题

咨询法律问题是本法律学习智能助手最主要的功能之一，用户登录并达成使用条件后可以使用。用户在登录后可以点击咨询法律问题按钮，在下方输入框输入想要咨询的法律案件，点击确定后即可输出本法律学习智能助手提供的内容。咨询法律问题如图 6-4 所示。

肇事逃逸是违法的，并且是一项严重的交通违法行为。

肇事逃逸通常指在发生交通事故后，当事人为逃避法律责任而逃离事故现场的行为。这种行为不仅违反了《道路交通安全法》，还可能触犯《刑法》中的交通肇事罪，根据案件的具体情节，可能会受到刑事处罚。下面将通过多个维度来分析肇事逃逸的违法性：

1. 法律规定：根据《最高人民法院关于审理交通肇事刑事案件具体应用法律若干问题的解释》第三条的规定，肇事后逃逸是指在发生交通事故后，为逃避法律追究而逃跑的行为。同时，《中华人民共和国刑法》第一百三十三条规定，交通肇事后逃逸，情节严重的，可处三年以上七以下有期徒刑。

2. 司法解释：最高人民法院发布的相关案例和指导意见明确了肇事逃逸的认定标准，即不仅限于逃离事故现场，还包括在有责任保护现场、抢救伤者、迅速报案等情况下故意逃避这些义务的行为。

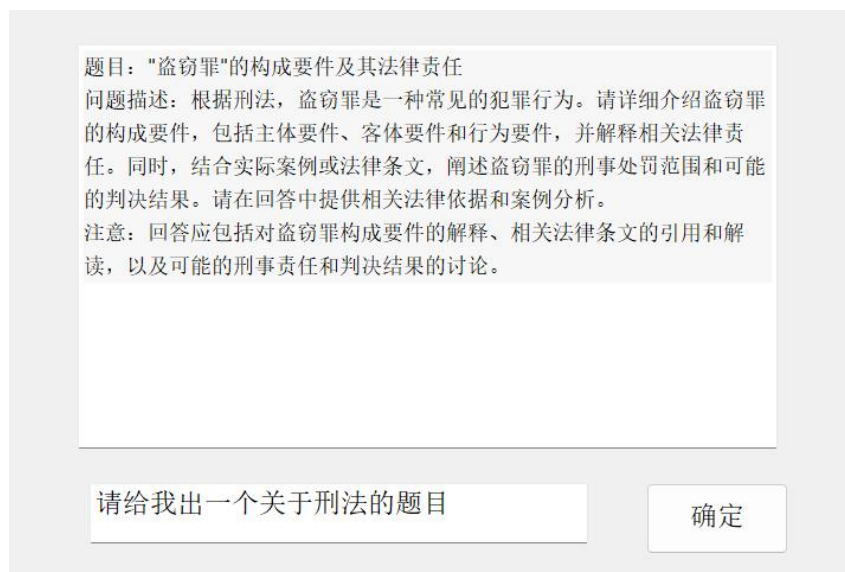
肇事逃逸违法吗

确定

图 6-4 咨询法律问题界面图

6.2.5 用户自测

用户自测是本法律学习智能助手最主要的功能之一，用户登录并达成使用条件后可以使用。用户在登录后可以点击用户自测按钮，在下方输入框输入想要作答的问题，点击确定后即可输出本法律学习智能助手提供的内容。用户自测如图 6-4 所示。



题目："盗窃罪"的构成要件及其法律责任

问题描述：根据刑法，盗窃罪是一种常见的犯罪行为。请详细介绍盗窃罪的构成要件，包括主体要件、客体要件和行为要件，并解释相关法律责任。同时，结合实际案例或法律条文，阐述盗窃罪的刑事处罚范围和可能的判决结果。请在回答中提供相关法律依据和案例分析。

注意：回答应包括对盗窃罪构成要件的解释、相关法律条文的引用和解读，以及可能的刑事责任和判决结果的讨论。

请给我出一个关于刑法的题目

确定

图 6-4 用户自测界面图

6.3 本章小结

本章首先介绍了法律学习智能助手环境配置，从软件环境展开了叙述，然后分模块包括登录注册模块、搜索法律条文模块、查询法律案件模块、咨询法律问题模块进行了部分功能的界面展示。

第 7 章 系统测试

系统测试是软件开发过程中的一个关键阶段，本章将对本法律学习智能助手进行系统测试，通过测试用例验证本法律学习智能助手的各模块功能是否能正常使用，系统性能与响应时间是否合格，是否符合期望。

7.1 测试步骤

通常来讲，系统的测试主要可以分为四个步骤：单元测试、集成测试、系统测试、验收测试。

(1) 单元测试。单元测试是指对软件中的最小可测试单元进行测试，单元一般指的是人为规定的最小的被测试的功能模块，比如 C 语言的一个函数。单元测试是在软件开发的过程中要进行的最低级别的测试活动，在传统的结构化编程语言中，比如对于 C 语言来说，要进行测试的单元一般是函数或者是子过程，而对于面向对象的语言，比如 C++ 语言来说，进行单元测试的基本对象一般是类。经常与单元测试联系起来的另一些开发活动包括代码走读，静态分析和动态分析。静态分析指的是对软件的源代码进行研读，查找一些错误，并不需要对代码进行编译和执行；动态分析是通过观察软件运行时的动作进行跟踪，测试覆盖度方面的信息。

(2) 集成测试。集成测试也叫组装测试，是单元测试的逻辑扩展。集成测试的过程是把两个已经测试过的单元组合成一个组件，测试接口。组件是指多个单元的集成聚合。在测试过程中，许多单元组合成组件，这些组件又聚合成为程序的更大部分，然后进行对测试片段的组合，并扩展成进程，再将模块与其他组的模块一起测试，最后再将构成进程的所有模块一起测试。集成测试是在单元测试的基础上，测试在将所有的软件单元按照概要设计的要求组装成模块、子系统或者系统的过程中各部分的工作是否达到预期结果。集成测试是单元测试的逻辑扩展，集成指的是多个单元的聚合。集成测试采用的方法是测试软件单元的组合能否正常工作，以及与其他组的模块集成之后能否正常工作。所有的软件项目的开发都不能摆脱集成这个阶段，软件单元只有通过集成才能形成一个有机整体。

(3) 确认测试。在集成测试之后组件以及被完全组装起来，接口的错误也已经被排除，那么就可以进行确认测试。确认测试测试的是软件能否按照设想正常工作。

实现软件的确认测试是要进行一系列的黑盒测试。确认测试之前需要进行测试计划的制定，测试计划应该考虑到测试的种类和测试的进度。在测试的过程中应该用一些特殊的测试用例，确保软件与需求一致。应该着重考虑软件是否满足需求的所有功能和需求，确定人机交互界面令用户满意。目前广泛使用的两种测试方法为 α 测试和 β 测试。 α 测试指的是软件开发公司内部组织人员对软件进行测试，在开发现场进行； β 测试指的是软件开发公司组织各个方面的典型用户在日常生活中实际使用，并对软件提出意见，属于一种现场测试。

（4）系统测试。系统测试是对整个系统的测试，把软件、硬件和操作人员看作一个整体。这种测试方法可以发现系统分析和设计中的错误。系统测试是将经过集成测试的软件作为计算机系统的一个部分，在实际运行环境下对计算机系统进行的一系列严格有效的测试，发现软件潜在的问题，确保软件可以正常运行。主要测试软件功能是否正常，也测试软件系统在异常的情况下能否正常运行。

（5）验收测试。验收测试是部署软件之前的最后一个测试阶段，是一个在软件完成了单元测试、集成测试和系统测试之后进行的测试活动，也成为交付测试，确保软件准备就绪，可以让最终用户使用软件设计的功能。经过集成测试之后，以及把模块组装成一个完整的系统，进一步测试软件的有效性。验收测试是系统开发生命周期的一个阶段，相关用户和测试人员根据测试计划和结果对系统进行测试。

7.2 测试方法

软件测试的方法多种多样，目前较为常用的软件测试方法主要分为黑盒测试和白盒测试。

（1）黑盒测试。黑盒测试通过测试来检测每个功能能否正常使用。在测试的过程中，把程序看成一个不能打开的黑盒子，在不考虑内部结构的情况下，在程序接口进行测试，只检测程序能否正常使用。黑盒测试不看重内部逻辑结构，而是侧重于外部结构，主要针对软件界面和功能进行测试。黑盒测试又叫作功能测试，注重于测试软件的功能性需求，在软件测试的各个阶段，黑盒测试都发挥着重要作用，特别是在系统测试过程中，黑盒测试是其他测试方法无法取代的。黑盒测试主要是检测是否有功能错误，是否有遗漏的功能，还检测能否进行正常的输入输出，以及是否有程序初始化和终止方面的错误。

（2）白盒测试。白盒测试是一种基于代码的测试，也被称为结构测试。白盒测

试的盒子指的是被测试的软件，白指的是盒子可视，也就是说可以看清楚盒子里面的东西，以及里面的东西是怎么运行的。白盒测试的测试方法有代码检查法、静态结构分析法、基本路径测试法等。白盒测试法的覆盖标准有逻辑覆盖、循环覆盖和基本路径测试。逻辑覆盖指的是语句覆盖、判定覆盖、条件覆盖、判定条件覆盖、条件组合覆盖和修改条件判断覆盖。

7.3 功能测试

法律学习智能助手的模块比较多，进行功能测试的时候需要以模块为单位进行测试，首先需还要制订试验计划，将期望的结果记录下来，查看实际情况与所期望的是否相符。

(1) 在测试系统登录时，先输入正确的用户名和密码，然后进行登陆操作，预期结果时可以登录到管理页面，再输入错误的用户名和密码，然后进行登录操作，系统提示登录失败，用户名和密码不正确，再输入不存在的用户名和密码，然后进行登录操作，系统提示登录失败，用户名或密码不正确，最后不输入用户名和密码，然后进行登录操作，系统提示用户名和密码不能为空。登录系统测试用例如表 7-1 所示。

表7-1 登录系统测试用例

测试方案	测试数据	预期结果
输入正确的用户名和密码，然后进行登录操作	用户名：admin，密码：123456	能够登录到管理界面
输入错误的用户名或密码，然后进行登录操作	用户名：cuowu，密码：123456	系统提示“登录失败，用户名或密码不正确！”
输入不存在的用户名，然后进行登录操作	用户名：abc，密码：123	系统提示“登录失败，用户名或密码不正确！”
不输入用户名和密码，直接进行登陆操作	不输入用户名和密码	系统提示“用户名或密码不能为空”

(2) 在进行搜索法律条文测试时，先输入想要搜索的法律条文，然后点击确定按钮，与其结构时法律学习智能助手会提供相关的法律条文和解释，如果不输入任何信息点击确定按钮，那么本法律学习智能助手将不会有任何反应，如果输入不存

在的法律条文之后点击确定按钮，那么法律学习智能助手会输出错误提示。搜索法律条文模块测试用例如表 7-2 所示。

表7-2 搜索法律条文测试用例

步骤	测试方案	预期结果
1	输入想要搜索的法律条文， 点击确定按钮	法律学习智能助手提供 相关的法律条文和解释
2	不输入任何信息点击确定按钮	法律学习智能助手没有反应
3	输入没有的法律条文后点击 确定按钮	法律学习智能助手输出错误 提示
4	输入的信息不准确	法律学习智能助手提示请输 入详细信息

（3）在进行咨询法律条文测试时，先输入想要知道的法律案件后点击确定按钮，本法律学习智能助手会提供给学生这样处罚的依据，包括法律条文等，如果不输入任何信息直接点击确定按钮，那么法律学习智能助手不做出任何反应。咨询法律案件模块测试用例如表 7-3 所示。

表7-3 咨询法律案件测试用例

步骤	测试方案	预期结果
1	输入想要知道的法律案件后 点击确定按钮	法律学习智能助手提供给学生 这样处罚的依据，包括法律 条文等
2	不输入任何信息，直接点击 确定按钮	法律学习智能助手不做任何 反应

（4）在进行咨询法律问题的测试时，用户先输入想要知道的法律问题后点击确定按钮，那么法律学习智能助手会提供给学生解释，包括法律法规条文和司法解释，

如果不输入任何信息直接点击确定按钮，那么法律学习智能助手不会做出任何反应。咨询法律问题模块测试用例如表 7-4 所示。

表7-4 咨询法律问题模块测试用例

步骤	测试方案	预期结果
1	输入想要知道的法律问题后 点击确定按钮	法律学习智能助手提供给学 生解释，包括法律法规条文 和司法解释等
2	不输入任何信息直接点击确 定按钮	法律学习智能助手没有任何 反应

（5）在进行用户自测功能模块测试时，如果用户先输入想要作答的题目类型后点击确定按钮，那么法律学习智能助手会为用户提供问题，如果不输入任何信息直接点击确定按钮，那么本法律学习智能助手不做任何反应。用户自测模块测试测试用例如表 7-5 所示。

表 7-5 用户自测模块测试用例

步骤	测试方案	预期结果
1	输入想要作答的题目类型后 点击确定按钮	法律学习智能助手为用户出 题
2	不输入任何信息直接点击确 定按钮	法律学习智能助手不做任何 回应

7.4 性能测试

性能测试在软件的质量保证中起着重要作用，性能测试的内容丰富多样。性能测试是通过自动化的测试工具模拟各种条件来对软件的各种性能指标进行测试。最常用的有负载测试和压力测试，通过负载测试可以检测各项工作在负载下的系统的性能，可以测试当负载逐渐增加时系统各项性能指标的变化。压力测试时通过确定一个系统不能接受的点，来确定系统能提供的最大服务。

7.5 本章小结

本章从法律学习智能助手系统测试出发，首先对测试步骤进行逐一简单介绍，紧接着介绍两种常用的测试方法——黑盒测试与白盒测试。随后通过测试用例的执行情况对社区疫情防控管理信息系统进行了分模块的功能测试，最后基于测试系统响应时间和服务器内存等性能进行压力测试和时间测试。

结 论

随着人工智能的发展和自然语言处理等一系列技术的迅速发展，智能助手在生活中的各个方面得到了很大的应用，在法律研究、法律学习等方面也可以用到智能助手。之前法律学习主要依赖于传统的学习方式，包括研究法律文件、研究法律条文和案例等，这些方式对于法律学生来说很有必要，但是在梳理大量信息和概念是可能会需要很多时间，并且法律条文的繁多，信息的复杂也给法律学生带来了挑战。

本法律学习智能助手通过机器学习和自然语言处理等技术，帮助学生更好的获取和理解法律知识。法律学习智能助手为学生提供案例分析，回答学生的问题，指导学生学习；还利用机器学习分析法律案例，帮助法律学生更好的理解法律条文。随着智能学习助手的发展，国内外许多机构已经开始探索如何使用这项技术，同时还有一些公司也在开发和推出面向法律学习的智能助手产品。

本法律学习智能助手的使用者主要有管理员，法律学生以及客服。管理员可以进行对用户账号的封禁和解封，管理员还可以进行公告的发布，比如封禁账号时需要发布公告说明封禁的账号以及封禁原因，客服主要用来接收来自法律学生的申诉，可以和法律学生和管理人员进行交流，避免管理员和法律学生直接的接触，管理员和客服的账号由系统分配，不能注册管理员和客服的账号，注册时只能注册学生的账号。法律学生在注册登录后进入个人信息页面，可以完善或修改个人信息，也可以修改密码，但账号经过注册后不可修改。法律学生还可以在论坛发表论题，也可以回复别人的论题，发表论题的学生有删除回复的权限，所有学生都有举报发言的权限，举报信息会直接发送给管理员，管理员审核后可以对被举报账号进行封禁，被封禁的账号不能再在论坛发言。法律学生还可以查询自己想知道的法律条文，比如某一部法律第几条的内容是什么，由于法律条文专业性较强，本法律学习智能助手还可以对法律条文进行解释。法律学生还可以查询法律案件，针对某一个行为查询量刑原因和判罚的依据。法律学生还可以通过本法律学习智能助手查看相关法律文献。本法律学习智能助手还可以给法律学生出题考试，并且收集学生的错题以方便学生日后的复习。

本项目致力于帮助学生进行更加便捷地学习，可以为法律学生答疑解惑，此外还可以帮助学生更加便捷地获取需要的资源，为学生学习提供便利。

参考文献

- [1] 黄彩梅. 基于费尔克拉夫三维框架的美国人工智能决策联盟涉华话语分析[J/OL]. 情报杂志, 1-9[2024-03-27].
- [2] 潘庆. 赋能、挑战与反制: 思想政治教育微缩课堂技术化的实践展开——以 ChatGPT 等新一代生成式人工智能的应用为例 [J/OL]. 信阳师范学院学报(哲学社会科学版), 1-7[2024-03-27].
- [3] 陆玉武, 曾碧卿. 面向不同层次学生的“人工智能”课程教学探索[J]. 工业和信息化教育, 2024,(03):24-29.
- [4] 王帆. 全球大模型应用百花齐放腾讯发布长文模型: 单次可处理超 38 万字符[N]. 每日经济新闻, 2024-05-20(008).
- [5] 金光, 包阳. 基于大模型的智能系统试验数据分析技术初探[J/OL]. 系统工程与电子技术, 1-13[2024-03-27]
- [6] 刘海涛, 亓达. 大语言模型的语用能力探索——从整体评估到反语分析[J/OL]. 现代外语, 1-13[2024-04-27].
- [7] 赵平广, 赵员康. 大语言模型赋能文化生产的可能与反思[J/OL]. 青年记者, 1-5[2024-05-27].
- [8] 张溢, 潘旭东, 杨珉. JADE-DB: 基于靶向变异的大语言模型安全通用基准测试集[J]. 计算机研究与发展, 2024,61(03):1113-1127.
- [9] 林萌, 戴程威, 郭涛. 基于多模态大语言模型的攻击性模因解释生成方法[J]. 计算机研究与发展, 2024,61(03):1206-1217.
- [10] 张爱华, 孙嘉鸿. 基于 TF-IDF 算法的运营商客户投诉原因研究[J]. 北京邮电大学学报(社会科学版), 2024,26(02)
- [11] 李旭东, 冯宇康, 陈俊升. 面向 SQLite 的多密钥页级别加密系统[J/OL]. 计算机应用, 1-14[2024-04-27].
- [12] 梁杰, 吴志镛, 符景洲, 等. 数据库管理系统模糊测试技术研究综述[J/OL]. 软件学报, 1-25[2024-03-27].
- [13] 李瑞山, 包芳, 闫文敬, 等. 基于 Sqlite 数据库的网络远程操作安全性研究与应用[J]. 电工技术, 2023,(21):233-236.
- [14] 常兴治, 李俊豪, 虞菊花. 基于 Qt 框架的课堂实训任务智能辅助评分系统研究[J]. 常州信息职

- 业技术学院学报, 2024,23(02):21-25.
- [15] 周志增, 宋林涛, 顾荣军, 等. 基于 Qt 的雷达数据处理软件开发与应用[J]. 火控雷达技术, 2024,53(01):87-91.DOI:10.19472/j.cnki.1008-8652.2024.01.015.
- [16] 朱明哲, 蒋培培, 王随. 基于 Qt 的 BP 神经网络演示软件设计[J]. 无线互联科技, 2024,21(06):71-74.
- [17] 刘诚浩, 葛明涛. 基于 Qt 的风光发电数据采集软件设计[J]. 河南科技, 2024,51(06):6-10.DOI:10.19968/j.cnki.hnkj.1003-5168.2024.06.001.
- [18] 关硕, 赵雪, 刘毅. 长三角一体化发展特征与动力探究——基于 TF-IDF 算法与格兰杰检验[J]. 科技和产业, 2024,24(05):40-47.
- [19] 王耀祖, 李擎, 戴张杰, 等. 大语言模型研究现状与趋势[J/OL]. 工程科学学报, 1-14[2024-05-27].
- [20] 谷业凯. 国产大模型加速赋能产业发展[N]. 人民日报, 2024-05-13(017).
- [21] 符晓波. “AI+医疗”落地看病体验升级[N]. 科技日报, 2024-05-13(006).
- [22] Mehrzad T. Prompt Engineering Using ChatGPT: Crafting Effective Interactions and Building GPT Apps[M]. Mercury Learning and Information: 2024-05-24.
- [23] Chung W S, Choi Y, Roy P, et al. Train small, model big: Scalable physics simulators via reduced order modeling and domain decomposition[J]. Computer Methods in Applied Mechanics and Engineering, 2024, 427: 117041-.
- [24] Qu Y, Wei C, Du P, et al. Integration of cognitive tasks into artificial general intelligence test for large models[J]. iScience, 2024, 27(4): 109550-109550.

致 谢

时光荏苒，岁月如梭。转眼间我的大学生活即将结束，六月总是充满了别离，我也将要与我的大学生活告别。

在论文完稿之际，我也要对帮助我很大的老师和同学表示感谢。感谢马勤勇老师在我做毕业设计时对我的指导，我的水平有限，能力一般，还常常犯了懒惰的毛病，但老师总是认真负责地帮助我。感谢马勤勇老师在开始做毕业设计时就帮助我选题，时光总是在不经意间溜走，我也没有准备好我的毕业设计到底要做什么，是马勤勇老师帮助我选定了我的毕业设计的题目，还给出了适合我的进度安排。由于我的粗心大意，我的报告格式常常出错，从开题报告开始马勤勇老师就在帮我修改我的报告，经常能找出我看不出来的错误，帮助我一次又一次地完善我的开题报告。老师不仅仅为我的毕业设计方向做出了指导，还帮助我寻找完成毕业设计的资料。由于我的能力有限，独立查询文献资料的方法不恰当，仅凭我自己而没有老师的指导断然完不成这次毕业设计。在撰写中期报告时马勤勇老师也给了我很大的帮助，没有因为我一次又一次的低级错误而不耐烦，反而要求我把一次又一次改过的报告发给老师审核，马勤勇老师一次又一次地帮助我审核我的报告。在完成最终的论文时，论文的要求更加严格，字数也更加多，但马勤勇老师却没有丝毫的厌烦，一次又一次地指出我的每个错误。

同时也要感谢我的答辩组老师，由于一开始我对我的选题还不是很了解，在开题的时候有很多不理解的问题，答辩组老师也在耐心地指出我的问题，虽然由于我的紧张，一开始并没有听进去，但老师们还是耐心的指出我理解上的错误，在时候重新看我的答辩过程时发现老师们给出的建议对我的毕业设计有很大帮助。

最后再次由衷感谢指导老师和答辩组的老师对我的帮助。

附录 1 开题报告

一、综述本课题国内外研究动态，说明选题的依据和意义

法律学习智能助理是指利用人工智能技术为法律学习者提供辅助和支持的工具或平台。它可以通过自然语言处理、机器学习和知识图谱等技术，帮助用户获取法律领域的知识、解答问题、提供案例分析和法律研究等功能。

1.1 国内外研究动态

(1) 国内发展现状

百度法律智能助理是一款基于人工智能技术的法律学习工具，它通过自然语言处理、知识图谱和机器学习等技术，为用户提供法律知识、案例分析和问题解答等功能，帮助用户深入理解法律领域的知识，并提供法律咨询和研究的支持。用户可以向百度法律智能助理提出相关的法律问题，它会根据自然语言处理技术来理解问题，并尝试给出相关的法律知识和解答；用户还可以向百度法律智能助理提供相关案例信息或问题描述，百度法律智能助理会分析案例并向用户提供相关的法律解释和类似案例的参考；百度法律智能助理还可以提供广泛的法律常识，用户可以查询特定的法律领域的知识和相关概念；用户可以用百度法律智能助理搜索相关的法律法规、判例和法律文献，百度法律智能助理会提供相应的检索结果和摘要信息。

中国法律智能化研究中心是中国法学会设立的研究机构，专注于推动法律智能化的研究和应用。通过研究和开发、学术交流与合作以及人才培养与推广等工作，该研究中心致力于促进法律与人工智能技术的融合，推动法律智能化的发展。其应用成果包括智能法律助手、智能合同平台和法律数据分析工具等，为法律实践和决策提供智能化支持。中国法律智能化研究中心开展法律智能化相关的研究和开发工作，包括法律知识图谱构建、智能问答系统开发、智能合同和智能裁判的研究等。该研究中心组织学术研讨会、论坛和交流活动，促进学界、业界和政府之间的合作，推动法律智能化领域的学术研究和实践探索。中国法律智能化研究中心培养相关领域的专业人才，包括举办培训班、开展人才培养项目等，推广法律智能化的应用和发展。

(2) 国外发展现状

Ross Intelligence（简称 Ross）是一家总部位于加拿大多伦多的人工智能技术公

司，成立于 2014 年。Ross 是一个法律智能助手，利用自然语言处理和机器学习等技术，为律师和法律团队提供智能化的法律研究和分析支持。Ross 为律师和法律团队提供了一个名为 ROSS 的智能助手平台。该平台通过自然语言处理和机器学习技术，使律师能够利用人工智能技术来进行法律研究、查找相关案例和法规、分析法律文件等。ROSS 平台具有强大的搜索和分析功能，可以从大量的法律数据库和文献中提取和整理相关信息，帮助律师更高效地进行法律研究和案件准备工作，还包括了一个智能问答系统，律师可以通过与该系统交互来提出法律问题，系统会根据问题提供智能化的答案和建议。

Judicata 是一家位于美国加利福尼亚州的法律技术公司，成立于 2012 年。Judicata 致力于利用人工智能和自然语言处理技术改进法律研究和分析的效率和准确性。Judicata 的核心功能之一是智能法律搜索。他们开发了一种高度精确和全面的法律搜索引擎，可以从各种法律数据库和文献中提取有关案例、法规、法律条款等相关信息。Judicata 的搜索引擎采用了先进的自然语言处理和机器学习技术，能够理解用户的查询意图，并返回与查询最相关的结果。这使得律师和法律研究人员能够更快速地找到所需的法律信息。Judicata 还提供了文书分析和评估的功能。他们的系统可以自动分析和解析法律文件，如法院判决、合同和法律文件等，提取其中的关键信息和条款。通过该功能，律师可以更准确地理解和评估法律文件的含义和影响，快速找到关键条款和案例，为法律研究和案件准备提供支持。Judicata 构建了一个法律知识图谱，将各种法律概念、案例、法规等相关信息进行组织和链接。这样的知识图谱可以提供全面和结构化的法律信息，通过法律知识图谱，律师可以更好地理解和浏览法律领域的知识体系，也可以快速找到相关案例和法规，并进行深入的法律研究和分析。

1.2 选题依据和意义

法律学习智能助手可以提供对法律案例的分析和解释。通过自然语言处理和机器学习技术，智能助手可以帮助用户快速找到相关案例，并提供关键信息、法律案例和相关判例的摘要，加快用户对案例的理解和应用；可以提供便捷的法律规章查询功能。用户可以通过输入关键词或问题，智能助手可以快速检索相关的法规和条款，并提供解释和相关案例的引用。这样，用户可以快速找到适用的法律规定，为法律问题提供准确的依据；可以帮助用户解释和理解复杂的法律术语。通过建立法律术语的知识图谱和语义关系，智能助手可以提供术语的定义、用法、相关法规和

案例等信息，帮助用户更好地理解和运用法律术语；可以辅助用户进行法律文书的撰写。通过分析用户提供的信息和要求，智能助理可以生成法律文书的模板、建议和结构，提供相关法规和案例的引用，并帮助用户提高文书的准确性和专业性；可以根据用户的学习需求和兴趣，推荐相关的学习资源，如法学教材、学术论文、法学期刊和在线课程等。智能助理可以通过分析用户的学习历史和偏好，提供个性化的推荐，帮助用户更高效地获取学习材料。

二、研究的基本内容，拟解决的主要问题

2.1 研究的基本内容

大语言模型是指具有大规模参数和能力的语言模型，其目标是理解和生成自然语言文本。这些模型使用深度学习技术，通过学习大量的文本数据来捕捉语言的概率分布和语义表示。

大语言模型的优势在于其能够处理复杂的语言结构和语义关系。它们可以理解上下文、推理相关信息、生成连贯的文本等。这使得它们在各种自然语言处理任务中表现出色，如文本生成、对话系统、摘要生成、机器翻译、问答系统等。然而，大语言模型也面临一些挑战和限制。其中之一是计算资源的需求，大语言模型通常需要庞大的计算设备和显著的训练时间。

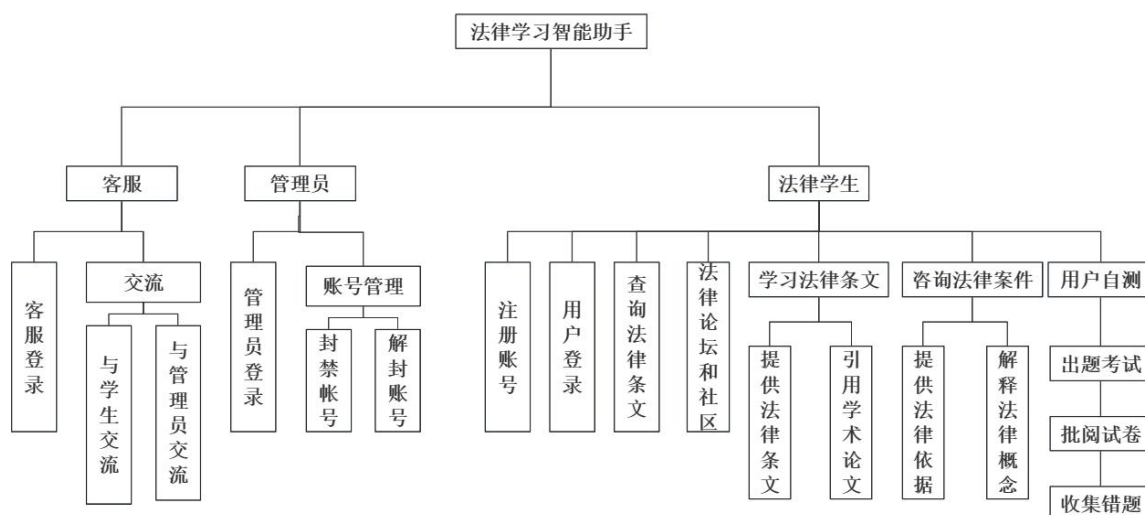
此外，模型的输出可能受到数据偏见、语义歧义和生成不准确性等问题的影响，需要进行后处理和调整。尽管如此，大语言模型代表了自然语言处理领域的前沿技术，对于推动自然语言理解和生成的发展具有重要意义。随着模型规模的扩大和技术的改进，大语言模型将在各个领域发挥越来越重要的作用，为人们提供更智能、自然的语言交互和应用体验。

GPT 是一种基于 Transformer 架构的大语言模型。GPT 中文意为生成式预训练变换器。自然语言处理可以解决用户更为通用的问题，分为两个关键步骤：自然语言理解（NLU）和自然语言生成（NLG）。自然语言理解是指将人类语言转化为机器可以理解和处理的形式过程，它涉及对文本进行语义和语法分析，以提取出其中的意义和结构，NLU 的主要目标是使计算机能够理解和解释人类的语言输入。自然语言生成是指根据计算机内部的信息和逻辑规则，生成自然语言文本的过程，目标是使计算机能够以自然的方式生成易于理解和流畅的语言输出。自然语言理解是将人类语言转化为机器可以理解的形式，而自然语言生成是根据计算机内部的信息和逻辑规则生成自然语言文本。这两个任务在自然语言处理中相辅相成，共同构建了

人机交互、智能对话和信息处理等应用领域的基础。

法律学习智能助手是一种基于人工智能技术的软件工具，旨在辅助学生和专业人士学习和理解法律知识。它结合了自然语言处理、机器学习和大语言模型等技术，能够提供准确的法律信息、解答法律问题，并帮助用户进行法律研究。

法律学习智能助手可以迅速访问各种法律数据库和文献资源，包括法律条文、判例、学术论文等，为用户提供准确和全面的法律知识。用户可以向法律学习智能助手提出具体的法律问题，它会分析问题并给出相应的答案和解释。它可以帮助用户理解法律概念、解释法律原则，并提供相关案例和法律依据。法律学习智能助手可以帮助用户进行法律研究，提供相关的案例和法律文件，帮助用户找到相关的法律条文和判例，并进行分析和比较，支持用户进行深入的法律研究。法律学习智能助手可以根据用户的学习需求和兴趣，提供个性化的学习推荐。它可以分析用户的学习历史和偏好，并提供相关的学习资源和建议，帮助用户高效地学习和提升法律知识。法律学习智能助手还可以给法律学生出题考试，批阅学生的答卷，监测法律学生对相关法律条文的学习情况和理解情况，为法律学生收集错题集，帮助法律学生日后的复习和自测。法律学习智能助手的功能如图 1 所示。



2.2 需要解决的问题

法律学习智能助手需要能够获取广泛的法律知识，并将其组织成结构化的形式，以便能够高效地检索和应用，这包括收集和整理法律条文、案例法、法律解释、法学文献等内容，因此选择使用大语言模型来实现人机交互。

法律文本通常具有复杂的语义和句法结构。智能助理需要具备强大的自然语言处理能力，能够理解和解析输入的法律问题或查询，并提取关键信息。这涉及词法分析、句法分析、命名实体识别、语义角色标注等技术。

智能助理需要具备法律问题分析和推理的能力。它需要能够理解用户提出的法律问题，并结合法律知识进行推理、分析和判断。这可能涉及到逻辑推理、法律条文解释、案例类比等技术。

智能助理需要能够根据用户的具体需求，提供准确、全面的法律咨询和建议。它应该能够解答用户的法律问题、解释相关法律条文、提供预测性的法律意见等。

智能助理需要拥有友好的用户界面和交互方式，使用户能够方便地与其进行沟通和互动。这可以是一个聊天界面、语音交互或基于 Web 的用户界面。

智能助理需要选择部分法律条文给法律学生出题，收集学生的错题，帮助法律学生日后的复习。

三、研究步骤、方法及措施

3.1 研究步骤

3.1.1 确定研究目标

设计一个拥有可以回答用户法律问题，帮助法律学生出题考试，批改试卷，收集错题帮助用户复习等功能的学习助理。

3.1.2 数据收集和预处理

数据收集：确定需要收集的法律数据的类型和来源。这可以包括法律条文、案例法、法学文献、法庭文件等。根据确定的数据来源，收集相关的法律数据。这可以通过网络爬取、数据下载、数据库查询等方式进行。确保收集到的数据覆盖广泛的法律领域和具有代表性。对收集到的数据进行清洗，去除不需要的信息和噪声。将收集到的数据划分为训练集、验证集和测试集。训练集用于模型的训练，验证集用于模型的调优和选择超参数，测试集用于评估模型的性能和泛化能力。

预处理：将收集到的法律文本进行分词，将文本划分为单词或子词的序列。这可以使用现有的分词工具或库进行处理。将分词后的文本转化为词向量表示，以便模型能够处理。可以使用预训练的词向量模型（如 Word2Vec、GloVe、FastText）或自行训练词向量模型。将文本序列填充或截断为固定的长度，以保证输入数据的一致性。这可以使用填充和截断技术，如将较长的文本截断为固定长度，或通过添加特殊的填充符号将较短的文本序列填充到相同长度。将文本数据进行编码，以适应

模型的输入要求。

3.1.3 模型选择和训练

研究不同的大语言模型，并了解它们的特点、能力和适用场景。根据研究目标 and 需求，选择适合法律学习智能助理的大语言模型。考虑模型的预训练能力、对法律领域的了解程度、语义理解和生成能力等因素。

使用数据收集和预处理步骤中准备的法律数据作为模型的训练数据。将数据划分为训练集、验证集和测试集，确保训练数据的充分性和代表性。

使用训练数据对模型进行训练，并在每个训练周期（epoch）结束后使用验证集评估模型的性能。监控训练过程中的损失函数和验证指标，以判断模型的收敛情况和性能表现。

使用测试集评估训练好的模型的性能和泛化能力。根据评估结果选择最佳模型，该模型在验证集和测试集上表现良好，并能够准确地回答法律问题。

3.2 研究方法和措施

（1）方案一

大语言模型是指基于深度学习和自然语言处理技术构建的规模庞大的语言生成模型。它能够理解和生成自然语言文本，具备广泛的语言理解和生成能力。大语言模型的核心是深度神经网络，通常使用递归神经网络（RNN）或者变种，如长短期记忆网络（LSTM）或门控循环单元（GRU）。这些神经网络模型能够处理和学习复杂的语言结构和语义关系。

大语言模型的训练是基于大规模的文本语料库进行的。语料库可以包括互联网上的网页、新闻文章、书籍、论文等各种文本资源。通过大规模的数据训练，模型可以捕捉到语言的统计规律和语义表示，从而具备生成和理解文本的能力。大语言模型可以生成各种类型的文本，如文章、故事、对话等。它可以用于自动写作、内容生成、机器翻译等任务。大语言模型可以通过理解和分析问题，生成准确的回答。它可以用于构建智能问答系统，提供快速、准确的答案和解决方案。

大语言模型可以自动提取文本摘要，概括文章的主要内容。它也可以对文本进行分类，识别文本的情感、主题等。大语言模型可以实现与人类用户的自然对话交互。它可以理解用户的问题和指令，并回答问题、提供建议等。大语言模型可以根据用户的需求，检索和推荐相关的文本信息。它可以帮助用户找到所需的资料、文章或其他相关资源。

（2）方案二

TF-IDF (Term Frequency-Inverse Document Frequency) 模型是一种常用的文本特征表示方法，用于衡量词语在文本中的重要程度。它结合了词频 (Term Frequency) 和逆文档频率 (Inverse Document Frequency)，通过对文本语料库进行统计分析，得到每个词语的权重值。

TF (词频) 指的是在一个文档中某个词语出现的频率。TF 可以用简单的计数表示，即某个词语在文档中出现的次数，或者使用相对词频，即某个词语在文档中出现的次数除以文档中的总词数。TF 反映了词语在文档中的重要性，频率越高，重要性越大。IDF (逆文档频率) 指的是衡量词语在整个语料库中的重要程度。IDF 通过计算某个词语在语料库中出现的文档频率的倒数，来衡量词语的稀有程度。常用的计算公式是使用总文档数目除以包含该词语的文档数目的对数，然后取对数的倒数。IDF 能够减少一些常见的词语对整个文本的重要性影响，提高对于特定词语的重视程度。TF-IDF 模型将词频和逆文档频率相乘，得到每个词语的 TF-IDF 值。

TF-IDF 值高的词语在文本中出现频繁且在整个语料库中较为稀有，因此具有更高的重要性。TF-IDF 模型通过计算每个词语的权重，将文本表示为一个向量，可以用于文本相似度计算、信息检索、文本分类等任务。TF-IDF 模型的优点是简单有效，能够捕捉到词语在文本中的重要程度。然而，它也存在一些限制，比如未考虑词语的顺序和语义关系，对于长文本的表示效果可能有限。在实际应用中，可以结合其他特征提取方法和机器学习算法来进一步提高文本处理的性能。

（3）对比分析

大语言模型上下文理解能力强，通过预训练，可以学习到庞大的语言知识和语义关系，能够更好地理解和生成文本；大语言模型在预训练阶段使用了大规模的通用语料，可以适应各种不同领域和任务，具有较强的泛化能力；大语言模型可以生成具有语法正确性和上下文连贯性的文本，可以用于自动摘要、机器翻译等任务；大语言模型可以处理复杂的自然语言问题，如问答、文本分类、命名实体识别等。

大语言模型计算资源需求高通常需要大量的计算资源和显存来进行预训练和推理，对硬件设施的要求较高；大语言模型的预训练需要大规模的文本数据，对于特定领域的任务，可能需要额外的领域数据进行微调；大语言模型通常具有大量的参数和复杂的网络结构，导致模型较大，难以部署和运行在资源受限的环境中。

TF-IDF 模型计算简单，不需要大量的计算资源，可以快速构建和应用；TF-IDF

模型在信息检索任务中表现良好，可以根据关键词的重要性进行文本排序和匹配；TF-IDF 模型可以提供关键词的权重信息，帮助理解文本的重要内容和关键信息。

TF-IDF 模型只考虑词频和文档频率，忽略了词语之间的语义关系，不能很好地理解上下文和语义；TF-IDF 模型对于复杂的自然语言处理任务，如问答、文本生成等，性能有限；TF-IDF 模型对于未见过的词汇（Out-of-Vocabulary）没有有效的表示，无法处理未知词汇的情况。

综上所述，大语言模型在语言理解和生成方面具有优势，适用于复杂的自然语言处理任务，但需要大量的计算资源和数据；而 TF-IDF 模型简单高效，适用于文本检索任务，但在语义理解和复杂任务上表现有限。法律学习智能助理需要迅速访问各种法律数据库和文献资源，需要能够理解用户提出的法律问题，并结合法律知识进行推理、分析和判断，因此选择大语言模型来实现法律学习智能助手的人机交互。

四、研究工作进度

1—3 周：完成前期的资料搜集及整理，对需要用到的技术进行学习，了解大语言模型；完成开题报告撰写。

4—9 周：完成系统总体设计，详细设计和编码，完成中期答辩。

10—13 周：测试完善，撰写毕业论文。

14—15 周：准备最终答辩所需材料，全面检查、完善毕业设计论文与研究成果，完成毕业设计答辩。

五、主要参考文献

- [1] 张奇, 桂韬, 郑锐等. 大规模语言模型：从理论到实践[M]. 北京：电子工业出版社，2024
- [2] 王子威. 人工智能的“理解”——论语言的工具性与存在性[M]. 上海：东岳论丛，2024-02-29
- [3] 刘晓明, 李丞正旭, 吴少聪等. 文本分类算法及其应用场景研究综述. 西安交通大学电子与信息学部，2024-02-29
- [4] 陈慧敏, 刘知远, 孙茂松. 大语言模型时代的社会机遇与挑战. 清华大学计算机科学与技术系，2024-02-20
- [5] 苏婧琼, 苏艳琼. 基于 LDA 和 TF-IDF 的关键词提取算法研究. 晋中信息学院
- [6] 张凌嘉. 智能个人助理自主性和交互性对用户态度的影响研究，2022-06-01
- [7] 赵栓峰, 李瑶, 李明月. 基于智能教学助理的个性化课堂教学模式研究[M]. 工程

科技 II 辑;社会科学 II 辑, 2021-12-01

[8] 何雨航. 从原理到应用, 人人都懂的 ChatGPT 指南. 京东科技, 2023-05-11.

附录 2 文献综述

一、课题国内外现状

近几年，伴随着人工智能技术的发展，国内一些公司也在研究和开发面向法律学习的智能助手产品。

百度法律智能助手是一款基于人工智能技术的法律学习工具，它通过自然语言处理、知识图谱和机器学习等技术，帮助法律学生学习法律知识，为用户进行案例分析和问题解答等，帮助法律学生深入理解法律领域的知识。用户可以向百度法律智能助手提出相关的法律问题，它会根据自然语言处理技术来理解问题，并尝试给出相关的法律知识和解答；用户还可以向百度法律智能助手提供相关案例或者提出问题，百度法律智能助手会分析案例并向用户提供相关的法律解释和类似的案例；百度法律智能助手还可以提供广泛的法律知识，用户可以查询特定的法律知识和相关概念；用户可以用百度法律智能助手搜索相关的法律条文、案例和法律文献，百度法律智能助手会提供相应的检索结果和摘要信息。

现在，人工智能已经第三次开始迅猛地发展。人工智能理论和技术取得了飞速发展，在语音识别、文本识别、视频识别等感知领域取得了突破，已经可以达到甚至超过了人类水准，成为了引领新一轮科技革命和产业变革的战略性技术。人工智能的应用领域也快速向多方向发展，出现在与人们日常生活息息相关的越来越多的场景中^[1]。

大模型是指具有大规模参数的机器学习模型，而大语言模型是指在自然语言处理领域的一种大模型，可以用来进行文本处理。大语言模型的主要特点是在大规模语料库上进行了训练，学习了自然语言的语法，国内外较为常见的有 OpenAI 的 GPT 系列，Google 的 Bard，以及百度的文心一言。

GPT 是一种基于 Transformer 架构的大语言模型。GPT 中文意为生成式预训练变换器。自然语言处理可以解决一些通用的问题，分为两个关键步骤：自然语言理解（NLU）和自然语言生成（NLG）。自然语言理解是指将人类的语言转化成为机器可以理解和处理的形式，它涉及对文本进行语义分析和语法分析，来提取出其中的意义和结构，NLU 的主要目标是使计算机能够更好的理解和解释人类的语言。自然语言生成是指根据计算机内部的信息和逻辑规则，生成自然语言文本的，目标是使

计算机能够以自然的方式生成易于人类理解的语言输出^[5]。总之，自然语言理解是将人类语言转化为机器可以理解的形式，而自然语言生成是根据计算机内部的信息和逻辑规则转化为人类可以理解的语言。这两个任务在自然语言处理中相辅相成，共同构建了人机交互、智能对话和信息处理等应用领域的基础。

Bard 是一个由谷歌公司开发的聊天机器人，经过了训练，内容丰富。Bard 接受过了大量文本数据的培训，可以针对各种问题进行交流并且可以生成类似人类写作的文本。最初 Bard 基于 LaMDA 系列的大语言模型，后来基于 PaLM2，现在则是基于 Gemini 系列的大语言模型，是谷歌为了应对 OpenAI 公司的 GPT 系列而研发的。Gemini 是一个多模态大语言模型系列，多模态大语言模型是一个脱胎于大语言模型的模型，在原有的强大泛化能力和推理能力的基础上，进一步引入了多模态的信息处理能力。模态指的是一种信息的来源形式，例如人的听觉，嗅觉等，信息的模态指的是信息的来源媒介，例如文本、视频、音频、图像等，所谓的多模态指的是具有多种信息来源或者多种信息形式。多模态侧重的是不同类型的信息的融合和分析的问题，推动了人工智能更好地了解和认知世界。和以往的多模态方式相比，新兴的多模态大语言模型的模型更大，训练范式更新，具有着传统的多模态模型不具备的能力。

文心大模型是百度旗下的大模型，是一个由百度自主研发的产业级大模型，文心大模型以创新性的知识增强技术为核心，从单模态大模型到跨模态大模型，从通用基础大模型到跨领域大模型持续创新，文心大模型构建了模型层、工具与平台层，加快了人工智能大模型大规模产业化进程，大幅度降低了人工智能开发和应用的门槛。上文提到模态指的是一种信息的来源形式，而但模态顾名思义指的就是仅用一种模态的数据进行学习，而跨模态学习指的是一种在不同模态之间进行的信息传递和理解的学习过程。跨模态是从一个模态提取信息，并且还要用这些信息来理解另一个模态的内容。跨模态学习的核心在于不同模态之间的相关性和互补性。2019 年 3 月，文心大模型初次发布，历经四年，文心大模型在 2023 年 4 月提升至 4.0 版本，经过飞桨框架软硬协同优化训练的文心大模型，在各种能力上都有了全面的提升。飞桨深度学习平台以百度多年的深度学习技术研究和业务应用为基础，是中国首个自主研发的产业级深度学习平台，在开发便捷的深度学习框架、大规模分布式训练、高性能推理引擎等技术上处于国际领先水平，当前飞桨凝聚八百万开发者，基于飞桨创建的八十万个模型，飞桨广泛服务于金融、能源、制造、交通等领域。文心大

模型现在也已经广泛地应用在搜索、信息流、智能音箱等互联网产品，并且通过飞桨深度学习平台赋能工业、能源、教育、金融、通信等各个行业。

二、研究主要成果

语言模型是一种自然语言处理的基本概念，是一种用来计算语言概率的数学模型，语言模型先理解语言的结构和规律，并且对文本下一阶段的文本中的词语做出预测，可以让语言模型能够识别语法、语义等语言特征。

随着深度学习的发展，神经网络语言模型逐渐成为顶流，其中包括循环神经网络和长短期记忆网络等模型，这些模型通过学习大量文本数据来建立词语之间的联系，从而实现对文本的预测。在自然语言处理中，语言模型可以用来进行机器翻译、文本生成等任务，在文本生成方面，语言模型可以用来进行自动写作、智能对话等^[6]。

在过去几年中，大语言模型迅速发展，其中 GPT(生成式预训练转换器)和 BERT(双向编码表示器)代表了自然语言处理领域的进步，重新定义了语言理解和生成的范式。以 GPT 和 BERT 为代表的大语言模型利用 Transformer 架构的优势，又通过大规模的预训练来学习语言的表示。其中，GPT 采用了自回归的与训练目标，在生成方面的任务上表现得很出色。

大语言模型的训练是一个复杂的过程。首先需要进行数据集的选择，其中数据集的质量和规模对训练很重要，一般来说需要选择大规模的文本作为训练数据；其次就是预训练和微调的过程，在预训练阶段，模型会在大规模文本上进行无监督学习，在微调阶段，模型会在特定的数据上进行有监督学习；之后进行计算资源与训练成本，一般情况下训练大语言模型需要大量的计算机时间和时间；最后要避免过拟合和模型泛化，比如进行正则化，数据增强等。

语言模型是一种自然语言处理的基本概念，是一种用来计算语言概率的数学模型，语言模型先理解语言的结构和规律，并且对文本下一阶段的文本中的词语做出预测，可以让语言模型能够识别语法、语义等语言特征。

随着深度学习的发展，神经网络语言模型逐渐成为顶流，其中包括循环神经网络和长短期记忆网络等模型，这些模型通过学习大量文本数据来建立词语之间的联系，从而实现对文本的预测。在自然语言处理中，语言模型可以用来进行机器翻译、文本生成等任务，在文本生成方面，语言模型可以用来进行自动写作、智能对话等^[6]。

在过去几年中，大语言模型迅速发展，其中 GPT(生成式预训练转换器)和 BERT(双向编码表示器)代表了自然语言处理领域的进步，重新定义了语言理解和生成

的范式。以 GPT 和 BERT 为代表的大语言模型利用 Transformer 架构的优势，又通过大规模的预训练来学习语言的表示。其中，GPT 采用了自回归的与训练目标，在生成方面的任务上表现得很出色。

大语言模型的训练是一个复杂的过程。首先需要进行数据集的选择，其中数据集的质量和规模对训练很重要，一般来说需要选择大规模的文本作为训练数据；其次就是预训练和微调的过程，在预训练阶段，模型会在大规模文本上进行无监督学习，在微调阶段，模型会在特定的数据上进行有监督学习；之后进行计算资源与训练成本，一般情况下训练大语言模型需要大量的计算机时间和时间；最后要避免过拟合和模型泛化，比如进行正则化，数据增强等。

三、发展趋势

大语言模型是一类基础模型，大语言模型经过大量数据训练，用来提供推动多个用例和应用程序以及解决大量任务所需的基础功能。这与为每个用例单独构建和训练特定模型的想法形成鲜明对比，这种想法在许多标准下令人望而却步，会抑制协同作用，甚至可能导致性能下降。

大语言模型代表自然语言处理和人工智能领域的重大突破，用户可以通过 Open AI 的 Chat GPT-3 和 GPT-4 等界面轻松访问，这些界面已经获得了 Microsoft 的支持。总而言之，大语言模型旨在经过大量数据训练，像人类一样理解和生成文本以及其他形式的内容。这种模型有能力从环境中推断问题的能力，生成连贯并且还与环境相关的响应，翻译成英语以外的语言，总结文本，回答问题，甚至可以协助完成创造性写作或代码生成任务。它们之所以如此神通广大，是因为有数十亿个参数使模型能够捕获语言中的复杂模式并执行各种与语言相关的任务。大语言模型正在彻底改变从聊天机器人和虚拟助理到内容生成、研究协助和语言翻译各个领域的应用程序。随着大语言模型的不断发展和改进，大语言模型将会重塑我们与技术交互和获取信息的方式，使它们成为现代数字环境的关键部分。

四、主要参考文献

- [1] 陆玉武, 曾碧卿. 面向不同层次学生的“人工智能”课程教学探索[J]. 工业和信息化教育, 2024, (03): 24-29.
- [2] 王帆. 全球大模型应用百花齐放腾讯发布长文模型: 单次可处理超 38 万字符[N]. 每日经济新闻, 2024-05-20 (008).
- [3] 金光, 包阳. 基于大模型的智能系统试验数据分析技术初探[J/OL]. 系统工程与

电子技术, 1-13[2024-03-27]

- [4] 刘海涛, 亓达. 大语言模型的语用能力探索——从整体评估到反语分析[J/OL]. 现代外语, 1-13[2024-04-27].
- [5] 赵平广, 赵员康. 大语言模型赋能文化生产的可能与反思[J/OL]. 青年记者, 1-5[2024-05-27].
- [6] 张谧, 潘旭东, 杨珉. JADE-DB: 基于靶向变异的大语言模型安全通用基准测试集[J]. 计算机研究与发展, 2024, 61(03): 1113-1127.

附录 3 中期报告

一、任务书中本阶段工作目标与任务要求

1.1 课题主要任务

本课题旨在设计与实现法律学习智能助手，法律学习智能助手是一种基于人工智能技术的软件工具，旨在辅助学生和专业人士学习和理解法律知识。它结合了自然语言处理、机器学习和大语言模型等技术，能够提供准确的法律信息、解答法律问题，并帮助用户进行法律研究。

法律学习智能助手的使用者分为管理员和用户。管理员拥有独特的管理员账号，管理员登录后拥有账号管理的权限，分为账号封禁和账号解封。用户注册登陆后可以查询法律条文，智能助手能够查询和提供相关法律法规的信息，方便用户查找相关法律文本；智能助手可以连接用户到法律论坛和社区，让用户与其他法律学习者和专业人士交流、分享经验和提问问题；法律学习智能助手可以迅速访问各种法律数据库和文献资源，包括法律条文、判例、学术论文等，为用户提供准确和全面的法律知识；用户可以向法律学习智能助手提出具体的法律问题，它会分析问题并给出相应的答案和解释。它可以帮助用户理解法律概念、解释法律原则，并提供相关案例和法律依据。法律学习智能助手可以帮助用户进行法律研究，提供相关的案例和法律文件，帮助用户找到相关的法律条文和判例，并进行分析和比较，支持用户进行深入的法律研究。法律学习智能助手可以根据用户的学习需求和兴趣，提供个性化的学习推荐。它可以分析用户的学习历史和偏好，并提供相关的学习资源和建议，帮助用户高效地学习和提升法律知识。法律学习智能助手还可以给法律学生出题考试，批阅学生的答卷，监测法律学生对相关法律法规的学习情况和理解情况，为法律学生收集错题集，帮助法律学生日后的复习和自测。法律学习智能助手的功能结构图如图 1 所示。

1.2 课题主要目标

管理员操作。

管理员可以注册管理员特有的账号

管理员可以进行账号封禁

管理员可以进行账号解封

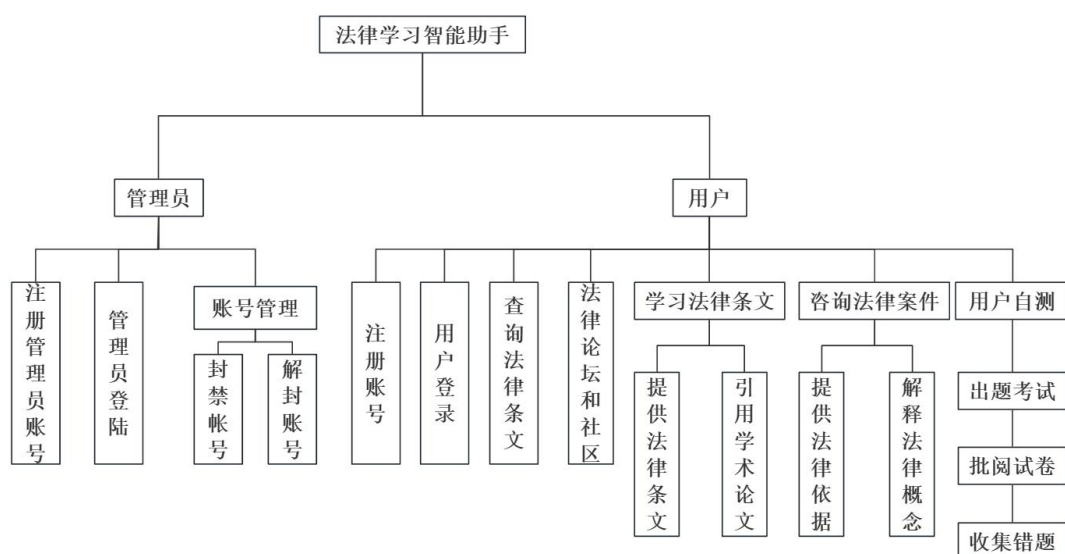


图 1 功能结构图

用户操作。

用户可以注册用户的账号

用户可以搜索查询法律条文

用户可以连接到法律社区和论坛

用户可以学习法律学习智能助手提供的法律条文

用户可以浏览法律学习智能助手提供的法律文献

用户可以学习法律学习智能助手对法律条文的解释

用户可以查询法律学习智能助手对法律案件提供的法律依据

用户可以使用法律学习智能助手提供的题目进行自测

为实现这些功能，查阅了大量资料，我认为技术可行性是最难解决的关键问题。法律学习智能助手需要具备强大的自然语言处理技术，包括文本分析、语义理解等。现代的自然语言处理技术，如预训练语言模型和深度学习算法，可以为智能助手提供强大的语义理解和回答问题的能力。智能助手需要具备友好的用户界面和良好的用户交互体验，使用户能够轻松使用和与助手进行交互。这包括设计清晰的界面布局、合理的操作流程和交互方式，以及响应迅速的交互体验。

二、目前已完成任务情况

2.1 总体设计

需求分析：用户注册登陆后可以查询法律条文，智能助手能够查询和提供相关法律法规的信息，方便用户查找相关法律文本；智能助手可以连接用户到法律论坛

和社区，让用户与其他法律学习者和专业人士交流、分享经验和提问问题；法律学习智能助收可以迅速访问各种法律数据库和文献资源，包括法律条文、判例、学术论文等，为用户提供准确和全面的法律知识；用户可以向法律学习智能助理提出具体的法律问题，它会分析问题并给出相应的答案和解释。它可以帮助用户理解法律概念、解释法律原则，并提供相关案例和法律依据。法律学习智能助理可以帮助用户进行法律研究，提供相关的案例和法律文件，帮助用户找到相关的法律条文和判例，并进行分析和比较，支持用户进行深入的法律研究。法律学习智能助理可以根据用户的学习需求和兴趣，提供个性化的学习推荐。它可以分析用户的学习历史和偏好，并提供相关的学习资源和建议，帮助用户高效地学习和提升法律知识。法律学习智能助手还可以给法律学生出题考试，批阅学生的答卷，监测法律学生对相关法律条文的学习情况和理解情况，为法律学生收集错题集，帮助法律学生日后的复习和自测。

以下是模块设计。

管理员模块：实现管理员账户的注册登录功能，以及对用户账户的封禁和解封功能。用户模块：实现用户账户注册登录功能。

论坛和社区模块：用户连接到论坛，可以在论坛进行沟通交流，分享经验，提问问题等；

咨询法律案件模块：法律学习智能助理可以帮助用户进行法律研究，提供相关的案例和法律文件，帮助用户找到相关的法律条文和判例，并进行分析和比较，支持用户进行深入的法律研究；

学习法律条文模块：法律学习智能助收可以迅速访问各种法律数据库和文献资源，包括法律条文、判例、学术论文等，为用户提供准确和全面的法律知识；

用户自测模块：法律学习智能助手还可以给法律学生出题考试，批阅学生的答卷，监测法律学生对相关法律条文的学习情况和理解情况，为法律学生收集错题集，帮助法律学生日后的复习和自测。

2.2 详细设计

用户在使用本法律学习智能助手时，需要先选择用户身份，若为普通用户，则选择是否有本法律学习智能助手的账号，没有账号则需要先注册，再登录，有账号则可直接登录，登录后用户可以选择操作。用户可选择的操作包括学习法律条文，浏览法律案件，浏览社区和论坛，用户自测等，用户注册登陆后可以查询法律条文，

智能助手能够查询和提供相关法律法规的信息，方便用户查找相关法律文本；智能助手可以连接用户到法律论坛和社区，让用户与其他法律学习者和专业人士交流、分享经验和提问问题；法律学习智能助手可以迅速访问各种法律数据库和文献资源，包括法律条文、判例、学术论文等，为用户提供准确和全面的法律知识；用户可以向法律学习智能助手提出具体的法律问题，它会分析问题并给出相应的答案和解释。它可以帮助用户理解法律概念、解释法律原则，并提供相关案例和法律依据。法律学习智能助手可以帮助用户进行法律研究，提供相关的案例和法律文件，帮助用户找到相关的法律条文和判例，并进行分析和比较，支持用户进行深入的法律研究。法律学习智能助手可以根据用户的学习需求和兴趣，提供个性化的学习推荐。它可以分析用户的学习历史和偏好，并提供相关的学习资源和建议，帮助用户高效地学习和提升法律知识。法律学习智能助手还可以给法律学生出题考试，批阅学生的答卷，监测法律学生对相关法律法规的学习情况和理解情况，为法律学生收集错题集，帮助法律学生日后的复习和自测。若用户身份为管理员，管理员拥有独特的管理员账号，管理员登录后拥有账号管理的权限。本法律学习智能助手的程序流程图如图 2 所示。

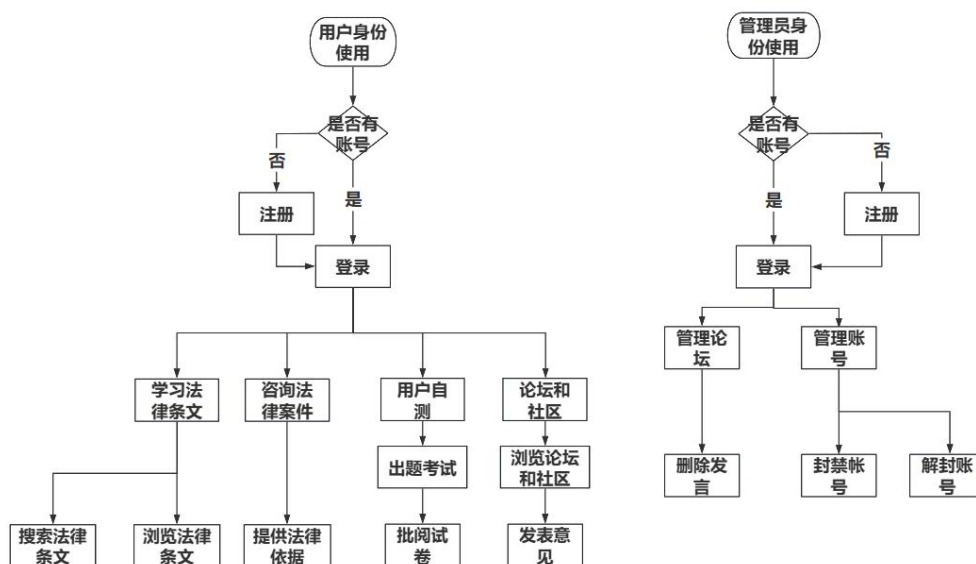


图 2 程序流程图

2.3 语言模型的选择

法律学习智能助手的部分功能选择利用大语言模型。大语言模型是指基于深度学习和自然语言处理技术构建的规模庞大的语言生成模型。它能够理解和生成自然

语言文本，具备广泛的语言理解和生成能力。大语言模型的核心是深度神经网络，通常使用递归神经网络（RNN）或者变种，如长短期记忆网络（LSTM）或门控循环单元（GRU）。这些神经网络模型能够处理和学习复杂的语言结构和语义关系。

大语言模型的训练是基于大规模的文本语料库进行的。语料库可以包括互联网上的网页、新闻文章、书籍、论文等各种文本资源。通过大规模的数据训练，模型可以捕捉到语言的统计规律和语义表示，从而具备生成和理解文本的能力。大语言模型可以生成各种类型的文本，如文章、故事、对话等。它可以用于自动写作、内容生成、机器翻译等任务。大语言模型可以通过理解和分析问题，生成准确的回答。它可以用于构建智能问答系统，提供快速、准确的答案和解决方案。

大语言模型可以自动提取文本摘要，概括文章的主要内容。它也可以对文本进行分类，识别文本的情感、主题等。大语言模型可以实现与人类用户的自然对话交互。它可以理解用户的问题和指令，并回答问题、提供建议等。大语言模型可以根据用户的需求，检索和推荐相关的文本信息。

大模型是指具有大规模参数的机器学习模型，而大语言模型是指在自然语言处理领域的一种大模型，可以用来进行文本处理。大语言模型的主要特点是在大规模语料库上进行了训练，学习了自然语言的语法。

2.4 主要功能

搜索法律条文是本法律学习智能助手最主要的功能之一，用户登录并达成使用条件后可以使用。用户在登录后可以点击搜索法律条文按钮，在下方输入框输入想要搜索的法律条文，点击确定后即可输出本法律学习智能助手提供的内容。搜索法律条文如图 3 所示。

肇事逃逸被判三年是因为什么

由星火AI生成

肇事逃逸被判三年是因为构成了交通肇事罪，并且在事故后有逃逸行为。

根据《中华人民共和国刑法》第一百三十三条的规定，如果违反交通运输管理法规，发生重大交通事故，导致人员重伤、死亡或者造成公私财产重大损失的，将受到刑事处罚。具体来说，如果事故发生后行为人逃逸，将面临三年以上七年以下的有期徒刑；如果因逃逸致人死亡，则将面临七年以上的有期徒刑。

此外，在司法实践中，法院会根据具体的案件情节、性质、手段等多个方面进行综合判断，以确定最终的刑期。例如，如果肇事逃逸的行为导致了严重的后果，如人员死亡或重大财产损失，那么可能会被判处更重的刑罚。相反，如果后果相对较轻，可能会被判处较轻的刑罚。

总之，肇事逃逸被判三年是因为它不仅违反了交通法规，还因为行为人在事故后试图逃避法律责任，这种行为通常会受到法律的严厉制裁。

图 3 搜索法律条文

咨询法律问题是本法律学习智能助手最主要的功能之一，用户登录并达成使用条件后可以使用。用户在登录后可以点击咨询法律案件按钮，在下方输入框输入想要咨询的法律案件，点击确定后即可输出本法律学习智能助手提供的内容。咨询法律问题界面如图 4 所示。



图 4 咨询法律问题

搜索法律案件是本法律学习智能助手最主要的功能之一，用户登录并达成使用条件后可以使用。用户在登录后可以点击搜索法律案件按钮，在下方输入框输入想要搜索的法律案件，点击确定后即可输出本法律学习智能助手提供的内容。搜索法律条文如图 5 所示。



图 5 咨询法律案件

2.6 E-R 图

(1) 管理员管理法律学生账户主要涉及到四个实体，管理员可以封禁和解封法律学生的账号，学生如果对封禁有异议的话可以向客服申诉，客服可以和管理员沟通。账号管理模块 E-R 图如图 6 所示。

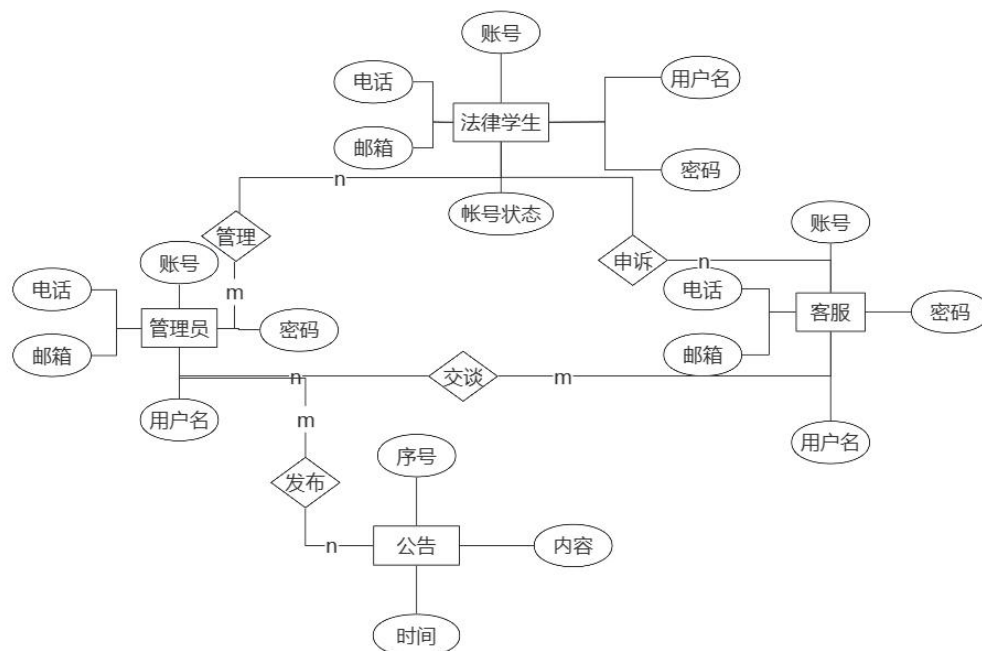


图 6 账号管理模块 E-R 图

(2) 搜索法律条文模块主要涉及到两个实体，法律学生可以在法律学习智能助手搜索法律条文，还可以查看法律条文的解释，搜索法律条文模块 E-R 图如图 7 所示。

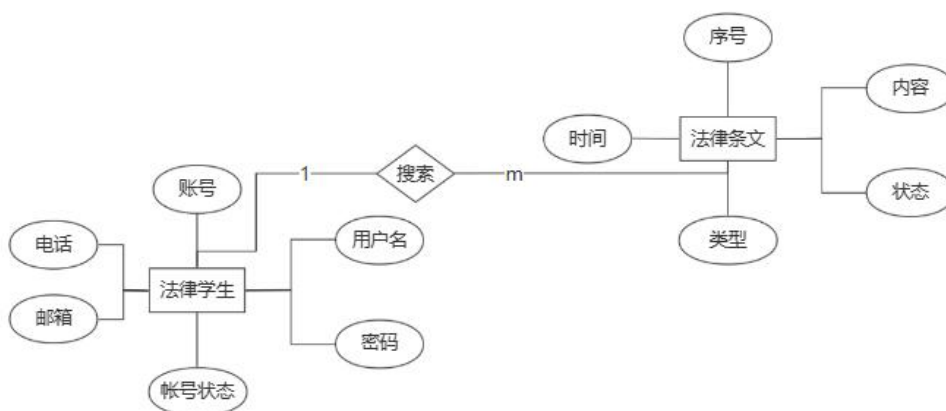


图 7 搜索法律条文模块 E-R 图

（3）浏览法律文献主要涉及到两个实体，法律学生可以通过法律学习智能助手浏览法律文献。浏览法律文献模块 E-R 图如图 8 所示。



图 8 浏览法律文献模块 E-R 图

（4）用户自测模块主要涉及到三个实体，法律学生可以通过法律学习智能助手从题库中按照类型做题，法律学习智能助手可以帮法律学生批阅学生做的题。用户自测模块 E-R 图如图 9 所示。

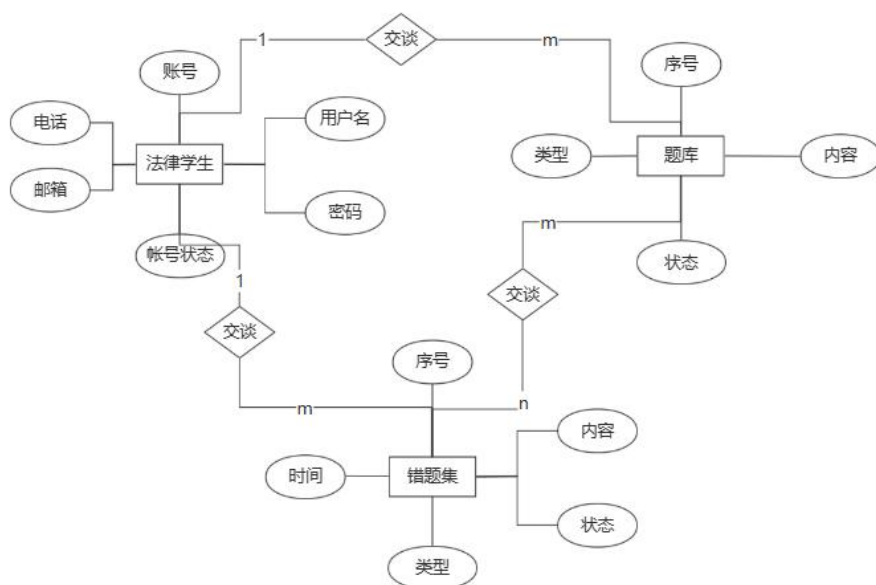


图 9 用户自测模块 E-R 图

2.7 数据库表

在法律学习智能助手的设计与实现中，数据库的设计是非常关键的，数据库模型影响着整个系统。本法律学习智能助手使用 SQLite 数据库进行数据的存储，来记录信息，共设计了 12 个数据库表，各个表的设计结果如下：

(1) 管理员信息表包含管理员账号，管理员密码，管理员昵称，管理员的电话号码，管理员的邮箱，其中管理员账号为主键，管理员的账号，密码和昵称都不可以为空，管理员信息表如表 1 所示。

表 1 管理员信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
adminid	varchar	15	是	否	管理员账号
adminpwd	varchar	15	否	否	管理员密码
adminname	varchar	20	否	否	管理员昵称
adminphone	int	11	否	是	管理员电话
adminmail	varchar	20	否	是	管理员邮箱

(2) 用户信息表包含用户账号，用户密码，用户昵称，用户的电话号码，用户的邮箱和用户的帐号状态，其中用户的账号为主键，用户的账号，密码和昵称都不可以为空用户信息表如表 2 所示。

表 2 用户信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
userid	varchar	15	是	否	用户账号
userpwd	varchar	15	否	否	用户密码
username	varchar	20	否	否	用户昵称
userphone	int	11	否	是	用户电话
usermail	varchar	20	否	是	用户邮箱
userstate	bool	--	否	否	账号状态

(3) 题库信息表包含题目序号，题目内容，题目的状态，即题目是否作答过和题目类型，其中题目序号为主键，题目的序号，内容以及题目的状态都不可以为空，题库信息表如表 3 所示。

表 3 题库信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
questionid	int	5	是	否	题目序号
questiontopic	varchar	255	否	否	题目内容
questionstate	bool	--	否	否	题目是否作答过
questiontype	varchar	10	否	是	题目类型

（4）错题集信息表包含题目序号，错题题目，错题是否作答过，错题时间和题目类型，其中错题序号为主键，错题的题目需要，错题的题目还有错题的状态都不可以为空，错题集信息表如表 4 所示。

表 4 错题集信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
mistakeid	int	5	是	否	错题序号
mistaketopic	varchar	255	否	否	错题题目
mistakestate	bool	--	否	否	错题是否答对过
mistaketime	datetime	--	否	否	错题时间
questationtype	varchar	10	否	是	题目类型

（5）法律条文信息表包含法律条文序号，法律条文类型，法律条文内容，法律条文是否浏览过，以及浏览时间，其中法律条文的条纹序号为主键，法律条文序号，法律条文类型和法律条文的内容都不可为空，法律条文信息表如表 5 所示。

表 5 法律条文信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
lawid	int	5	是	否	法律条文序号
lawtype	varchar	10	否	否	法律条文类型
lawtopic	varchar	255	否	否	法律条文内容
lawstate	bool	--	否	否	是否浏览过
lawtime	datetime	--	否	是	浏览时间

（6）论坛信息表包括论坛的话题，论坛的回复和话题的发布时间以及回复的时间，其中论坛的话题，论坛的回复和话题的发布时间都不是主键并且不可为空，论坛信息表如表 6 所示。

附录 3 中期报告

表 6 论坛信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
forumtopic	varchar	255	否	否	论坛话题
forumreply	varchar	255	否	否	论坛回复
forumtime	datetime	--	否	否	发布时间

(7) 客服信息表包含客服的账号, 客服的密码, 客服的昵称, 客服的电话和邮箱, 客服信息表如表 7 所示。

表 7 客服信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
customerid	varchar	15	是	否	客服账号
customerwd	varchar	15	否	否	客服密码
customername	varchar	20	否	否	客服昵称
customerphone	int	11	否	是	客服电话
customermail	varchar	20	否	是	客服邮箱

(8) 公告信息表包含公告序号, 公告内容和公告的发布时间, 公告信息表如表 8 所示。

表 8 公告信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
noticeid	varchar	15	是	否	公告序号
notice	varchar	15	否	否	公告内容
noticetime	datetime	--	否	否	发布时间

(9) 法律文献信息表包含文献序号, 文献内容和文献类型, 法律文献信息表如表 9 所示。

表 9 法律文献信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
informationid	varchar	15	是	否	文献序号
information	varchar	65535	否	否	文献内容
information type	varchar	20	否	否	文献类型

(10) 会员信息表包含会员账号, 会员密码, 会员昵称, 会员的余额, 会员信

息表如表 10 所示。

表 10 会员信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
memberid	varchar	15	是	否	会员账号
memberpwd	varchar	15	否	否	会员密码
membername	varchar	20	否	否	会员昵称
membervalue	int	4	否	否	会员余额

(11) 收藏夹信息表包含收藏的法律条文, 收藏的法律文献, 收藏的题目和后舱的错题, 收藏夹信息表如表 11 所示。

表 11 收藏夹信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
collaw	varchar	65535	否	否	收藏条文
colinfor	varchar	65535	否	否	收藏文献
colquestion	varchar	65535	否	否	收藏题目
colmis	varchar	65535	否	否	收藏错题

(12) 充值信息表包含充值时间, 用户的余额没充值记录, 充值信息表如表 12 所示。

表 12 充值信息表

字段名	数据类型	长度	是否为主键	是否可为空	备注
chargetime	datetime	--	否	否	充值时间
value	varchar	4	否	否	余额
chargerecord	varchar	20	否	否	充值记录

三、存在的问题和拟解决方法

3.1 存在的问题和解决方法:

(1) 存在的问题。

法律领域的信息庞杂且不断变化, 因此确保数据的可靠性和准确性是一个挑战。

拟解决方法。

确保收集到的法律数据具有高质量和可靠性, 从官方法律数据库、权威出版物和学术研究中获取数据, 采用数据验证和筛选的方法来确保数据的准确性和可靠性。

（2）存在的问题。

法律文件和案例可能存在解释上的歧义，不同的法学派别可能对同一法律问题有不同的见解。

拟解决方法。

提供多个可能的解释，解释其背后的法律理论和争议，并向用户明确说明解释的不确定性，提供相关的法学观点和学术研究，以帮助用户理解不同解释之间的差异。

（3）存在的问题。

法律领域的数据量庞大且多样化，包括法律文件、案例、法规等。

拟解决方法。

可以通过广泛收集、整理和标注大量的法律数据，以建立强大的训练数据集。同时，采用数据增强技术、迁移学习和预训练模型等方法可以提高智能助手对多样数据的处理能力。

（4）存在的问题。

法律学习智能助手需要具备解释法律条款、推理法律逻辑的能力。

拟解决方法。

使用知识图谱和推理引擎，构建法律知识图谱以捕捉法律概念之间的关系，并开发推理算法以进行法律推理和推断。此外，利用机器学习和深度学习技术，进行法律案例的类比和相似性分析，也可以增强智能助手的推理能力。

附录 4 外文原文

Improving Language Understanding by Generative Pre-Training

Abstract

Natural language understanding comprises a wide range of diverse tasks such as textual entailment, question answering, semantic similarity assessment, and document classification. Although large unlabeled text corpora are abundant, labeled data for learning these specific tasks is scarce, making it challenging for discriminatively trained models to perform adequately. We demonstrate that large gains on these tasks can be realized by generative pre-training of a language model on a diverse corpus of unlabeled text, followed by discriminative fine-tuning on each specific task. In contrast to previous approaches, we make use of task-aware input transformations during fine-tuning to achieve effective transfer while requiring minimal changes to the model architecture. We demonstrate the effectiveness of our approach on a wide range of benchmarks for natural language understanding. Our general task-agnostic model outperforms discriminatively trained models that use architectures specifically crafted for each task, significantly improving upon the state of the art in 9 out of the 12 tasks studied. For instance, we achieve absolute improvements of 8.9% on commonsense reasoning (Stories Cloze Test), 5.7% on question answering (RACE), and 1.5% on textual entailment (MultiNLI).

1 Introduction

The ability to learn effectively from raw text is crucial to alleviating the dependence on supervised learning in natural language processing (NLP). Most deep learning methods require substantial amounts of manually labeled data, which restricts their applicability in many domains that suffer from a dearth of annotated resources [61]. In these situations, models that can leverage linguistic information from unlabeled data provide a valuable alternative to gathering more annotation, which can be time-consuming and expensive. Further, even in cases where considerable supervision is available, learning good representations in an unsupervised fashion can provide a significant performance boost. The most compelling evidence for this so far has been the extensive use of pretrained

word embeddings [10, 39, 42] to improve performance on a range of NLP tasks .

Leveraging more than word-level information from unlabeled text, however, is challenging for two main reasons. First, it is unclear what type of optimization objectives are most effective at learning text representations that are useful for transfer. Recent research has looked at various objectives such as language modeling [44], machine translation [38], and discourse coherence [22], with each method outperforming the others on different tasks.¹ Second, there is no consensus on the most effective way to transfer these learned representations to the target task. Existing techniques involve a combination of making task-specific changes to the model architecture [43, 44], using intricate learning schemes [21] and adding auxiliary learning objectives [50]. These uncertainties have made it difficult to develop effective semi-supervised learning approaches for language processing.

In this paper, we explore a semi-supervised approach for language understanding tasks using a combination of unsupervised pre-training and supervised fine-tuning. Our goal is to learn a universal representation that transfers with little adaptation to a wide range of tasks. We assume access to a large corpus of unlabeled text and several datasets with manually annotated training examples (target tasks). Our setup does not require these target tasks to be in the same domain as the unlabeled corpus. We employ a two-stage training procedure. First, we use a language modeling objective on the unlabeled data to learn the initial parameters of a neural network model. Subsequently, we adapt these parameters to a target task using the corresponding supervised objective.

For our model architecture, we use the Transformer [62], which has been shown to perform strongly on various tasks such as machine translation [62], document generation [34], and syntactic parsing [29]. This model choice provides us with a more structured memory for handling long-term dependencies in text, compared to alternatives like recurrent networks, resulting in robust transfer performance across diverse tasks. During transfer, we utilize task-specific input adaptations derived from traversal-style approaches [52], which process structured text input as a single contiguous sequence of tokens. As we demonstrate in our experiments, these adaptations enable us to fine-tune effectively with minimal changes to the architecture of the pre-trained model.

We evaluate our approach on four types of language understanding tasks – natural language inference, question answering, semantic similarity, and text classification. Our general task-agnostic model outperforms discriminatively trained models that employ architectures specifically crafted for each task, significantly improving upon the state of the art in 9 out of the 12 tasks studied. For instance, we achieve absolute improvements of 8.9% on commonsense reasoning (Stories Cloze Test) [40], 5.7% on question answering (RACE) [30], 1.5% on textual entailment (MultiNLI) [66] and 5.5% on the recently introduced GLUE multi-task benchmark [64]. We also analyzed zero-shot behaviors of the pre-trained model on four different settings and demonstrate that it acquires useful linguistic knowledge for downstream tasks.

2 Related Work

Semi-supervised learning for NLP Our work broadly falls under the category of semi-supervised learning for natural language. This paradigm has attracted significant interest, with applications to tasks like sequence labeling [24, 33, 57] or text classification [41, 70]. The earliest approaches used unlabeled data to compute word-level or phrase-level statistics, which were then used as features in a supervised model [33]. Over the last few years, researchers have demonstrated the benefits of using word embeddings [11, 39, 42], which are trained on unlabeled corpora, to improve performance on a variety of tasks [8, 11, 26, 45]. These approaches, however, mainly transfer word-level information, whereas we aim to capture higher-level semantics.

Recent approaches have investigated learning and utilizing more than word-level semantics from unlabeled data. Phrase-level or sentence-level embeddings, which can be trained using an unlabeled corpus, have been used to encode text into suitable vector representations for various target tasks [28, 32, 1, 36, 22, 12, 56, 31].

Unsupervised pre-training Unsupervised pre-training is a special case of semi-supervised learning where the goal is to find a good initialization point instead of modifying the supervised learning objective. Early works explored the use of the technique in image classification [20, 49, 63] and regression tasks [3]. Subsequent research [15] demonstrated that pre-training acts as a regularization scheme, enabling better generalization in deep neural networks. In recent work, the method has been used to

help train deep neural networks on various tasks like image classification [69], speech recognition [68], entity disambiguation [17] and machine translation [48].

The closest line of work to ours involves pre-training a neural network using a language modeling objective and then fine-tuning it on a target task with supervision. Dai et al. [13] and Howard and Ruder [21] follow this method to improve text classification. However, although the pre-training phase helps capture some linguistic information, their usage of LSTM models restricts their prediction ability to a short range. In contrast, our choice of transformer networks allows us to capture longerrange linguistic structure, as demonstrated in our experiments. Further, we also demonstrate the effectiveness of our model on a wider range of tasks including natural language inference, paraphrase detection and story completion. Other approaches [43, 44, 38] use hidden representations from a 2 pre-trained language or machine translation model as auxiliary features while training a supervised model on the target task. This involves a substantial amount of new parameters for each separate target task, whereas we require minimal changes to our model architecture during transfer.

Auxiliary training objectives Adding auxiliary unsupervised training objectives is an alternative form of semi-supervised learning. Early work by Collobert and Weston [10] used a wide variety of auxiliary NLP tasks such as POS tagging, chunking, named entity recognition, and language modeling to improve semantic role labeling. More recently, Rei [50] added an auxiliary language modeling objective to their target task objective and demonstrated performance gains on sequence labeling tasks. Our experiments also use an auxiliary objective, but as we show, unsupervised pre-training already learns several linguistic aspects relevant to target tasks.

3 Framework

Our training procedure consists of two stages. The first stage is learning a high-capacity language model on a large corpus of text. This is followed by a fine-tuning stage, where we adapt the model to a discriminative task with labeled data.

1.1 Unsupervised pre-training Given an unsupervised corpus of tokens $U = \{u_1, \dots, u_n\}$, we use a standard language modeling objective to maximize the following likelihood:

$$L_1(U) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \Theta) \quad (1)$$

where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters Θ . These parameters are trained using stochastic gradient descent [51].

In our experiments, we use a multi-layer Transformer decoder [34] for the language model, which is a variant of the transformer [62]. This model applies a multi-headed self-attention operation over the input context tokens followed by position-wise feedforward layers to produce an output distribution over target tokens:

$$\begin{aligned} h_0 &= UWe + Wp \\ h_l &= \text{transformer_block}(h_{l-1}) \forall i \in [1, n] \\ P(u) &= \text{softmax}(h_n W T e) \end{aligned} \quad (2)$$

where $U = (u_{-k}, \dots, u_{-1})$ is the context vector of tokens, n is the number of layers, We is the token embedding matrix, and Wp is the position embedding matrix.

Supervised fine-tuning

After training the model with the objective in Eq. 1, we adapt the parameters to the supervised target task. We assume a labeled dataset C , where each instance consists of a sequence of input tokens, x_1, \dots, x_m , along with a label y . The inputs are passed through our pre-trained model to obtain the final transformer block's activation h_{m+1} , which is then fed into an added linear output layer with parameters Wy to predict y :

$$P(y|x_1, \dots, x_m) = \text{softmax}(h_{m+1} Wy). \quad (3)$$

This gives us the following objective to maximize:

$$L_2(C) = \sum_{(x,y)} \log P(y|x_1, \dots, x_m). \quad (4)$$

We additionally found that including language modeling as an auxiliary objective to the fine-tuning helped learning by (a) improving generalization of the supervised model, and (b) accelerating convergence. This is in line with prior work [50, 43], who also observed improved performance with such an auxiliary objective. Specifically, we optimize the following objective (with weight λ):

$$L_3(C) = L_2(C) + \lambda * L_1(C) \quad (5)$$

Overall, the only extra parameters we require during fine-tuning are Wy , and embeddings for delimiter tokens (described below in Section 3.3).

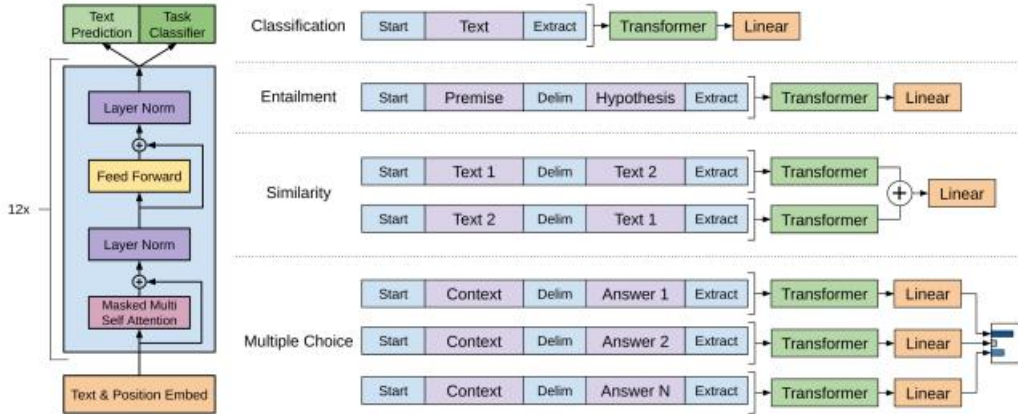


Figure 1: (left) Transformer architecture and training objectives used in this work. (right) Input transformations for fine-tuning on different tasks.

We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

Task-specific input transformations

For some tasks, like text classification, we can directly fine-tune our model as described above. Certain other tasks, like question answering or textual entailment, have structured inputs such as ordered sentence pairs, or triplets of document, question, and answers. Since our pre-trained model was trained on contiguous sequences of text, we require some modifications to apply it to these tasks. Previous work proposed learning task specific architectures on top of transferred representations [44]. Such an approach re-introduces a significant amount of task-specific customization and does not use transfer learning for these additional architectural components. Instead, we use a traversal-style approach [52], where we convert structured inputs into an ordered sequence that our pre-trained model can process. These input transformations allow us to avoid making extensive changes to the architecture across tasks. We provide a brief description of these input transformations below and Figure 1 provides a visual illustration. All transformations include adding randomly initialized start and end tokens (hsi, hei).

Textual entailment For entailment tasks, we concatenate the premise p and hypothesis h token sequences, with a delimiter token (\$) in between.

Similarity For similarity tasks, there is no inherent ordering of the two sentences

being compared. To reflect this, we modify the input sequence to contain both possible sentence orderings (with a delimiter in between) and process each independently to produce two sequence representations h and l which are added element-wise before being fed into the linear output layer.

Question Answering and Commonsense Reasoning For these tasks, we are given a context document z , a question q , and a set of possible answers $\{a_k\}$. We concatenate the document context and question with each possible answer, adding a delimiter token in between to get $[z; q; \$; a_k]$. Each of these sequences are processed independently with our model and then normalized via a softmax layer to produce an output distribution over possible answers.

4 Experiments

4.1 Setup Unsupervised pre-training

We use the BooksCorpus dataset [71] for training the language model. It contains over 7,000 unique unpublished books from a variety of genres including Adventure, Fantasy, and Romance. Crucially, it contains long stretches of contiguous text, which allows the generative model to learn to condition on long-range information. An alternative dataset, the 1B Word Benchmark, which is used by a similar approach, ELMo [44], is approximately the same size but is shuffled at a sentence level - destroying long-range structure. Our language model achieves a very low token level perplexity of 18.4 on this corpus.

Table 1: A list of the different tasks and datasets used in our experiments.

Task	Datasets
Natural language inference	SNLI [5], MultiNLI [66], Question NLI [64], RTE [4], SciTail [25]
Question Answering	RACE [30], Story Cloze [40]
Sentence similarity	MSR Paraphrase Corpus [14], Quora Question Pairs [9], STS Benchmark [6]
Classification	Stanford Sentiment Treebank-2 [54], CoLA [65]

Model specifications Our model largely follows the original transformer work [62]. We trained a 12-layer decoder-only transformer with masked self-attention heads (768 dimensional states and 12 attention heads). For the position-wise feed-forward networks, we used 3072 dimensional inner states. We used the Adam optimization scheme [27] with a max learning rate of $2.5e-4$. The learning rate was increased linearly from zero over the

first 2000 updates and annealed to 0 using a cosine schedule. We train for 100 epochs on minibatches of 64 randomly sampled, contiguous sequences of 512 tokens. Since layernorm [2] is used extensively throughout the model, a simple weight initialization of $N(0, 0.02)$ was sufficient. We used a bytepair encoding (BPE) vocabulary with 40,000 merges [53] and residual, embedding, and attention dropouts with a rate of 0.1 for regularization. We also employed a modified version of L2 regularization proposed in [37], with $w = 0.01$ on all non bias or gain weights. For the activation function, we used the Gaussian Error Linear Unit (GELU) [18]. We used learned position embeddings instead of the sinusoidal version proposed in the original work. We use the `ftfy` library² to clean the raw text in BooksCorpus, standardize some punctuation and whitespace, and use the `spaCy` tokenizer.

Fine-tuning details Unless specified, we reuse the hyperparameter settings from unsupervised pre-training. We add dropout to the classifier with a rate of 0.1. For most tasks, we use a learning rate of $6.25e-5$ and a batchsize of 32. Our model finetunes quickly and 3 epochs of training was sufficient for most cases. We use a linear learning rate decay schedule with warmup over 0.2% of training. λ was set to 0.5.

4.2 Supervised fine-tuning

We perform experiments on a variety of supervised tasks including natural language inference, question answering, semantic similarity, and text classification. Some of these tasks are available as part of the recently released GLUE multi-task benchmark [64], which we make use of. Figure 1 provides an overview of all the tasks and datasets.

Natural Language Inference The task of natural language inference (NLI), also known as recognizing textual entailment, involves reading a pair of sentences and judging the relationship between them from one of entailment, contradiction or neutral. Although there has been a lot of recent interest [58, 35, 44], the task remains challenging due to the presence of a wide variety of phenomena like lexical entailment, coreference, and lexical and syntactic ambiguity. We evaluate on five datasets with diverse sources, including image captions (SNLI), transcribed speech, popular fiction, and government reports (MNLI), Wikipedia articles (QNLI), science exams (SciTail) or news articles (RTE).

Table 2 details various results on the different NLI tasks for our model and previous

state-of-the-art approaches. Our method significantly outperforms the baselines on four of the five datasets, achieving absolute improvements of upto 1.5% on MNLI, 5% on SciTail, 5.8% on QNLI and 0.6% on SNLI over the previous best results. This demonstrates our model’s ability to better reason over multiple sentences, and handle aspects of linguistic ambiguity. On RTE, one of the smaller datasets we evaluate on (2490 examples), we achieve an accuracy of 56%, which is below the 61.7% reported by a multi-task biLSTM model. Given the strong performance of our approach on larger NLI datasets, it is likely our model will benefit from multi-task training as well but we have not explored this currently.

Table 2: Experimental results on natural language inference tasks, comparing our model with current state-of-the-art methods. 5x indicates an ensemble of 5 models. All datasets use accuracy as the evaluation metric.

Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	<u>89.3</u>	-	-	-
CAFE [58] (5x)	80.2	79.0	<u>89.3</u>	-	-	-
Stochastic Answer Network [35] (3x)	<u>80.6</u>	<u>80.1</u>	-	-	-	-
CAFE [58]	78.7	77.9	88.5	<u>83.3</u>		
GenSen [64]	71.4	71.3	-	-	<u>82.3</u>	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	61.7
Finetuned Transformer LM (ours)	82.1	81.4	89.9	88.3	88.1	56.0

Table 3: Results on question answering and commonsense reasoning, comparing our model with current state-of-the-art methods.. 9x means an ensemble of 9 models

Method	Story Cloze	RACE-m	RACE-h	RACE
val-LS-skip [55]	76.5	-	-	-
Hidden Coherence Model [7]	<u>77.6</u>	-	-	-
Dynamic Fusion Net [67] (9x)	-	55.6	49.4	51.2
BiAttention MRU [59] (9x)	-	<u>60.2</u>	<u>50.3</u>	<u>53.3</u>
Finetuned Transformer LM (ours)	86.5	62.9	57.4	59.0

Question answering and commonsense reasoning Another task that requires aspects of single and multi-sentence reasoning is question answering. We use the recently released RACE dataset [30], consisting of English passages with associated questions from middle and high school exams. This corpus has been shown to contain more reasoning type

questions that other datasets like CNN [19] or SQuAD [47], providing the perfect evaluation for our model which is trained to handle long-range contexts. In addition, we evaluate on the Story Cloze Test [40], which involves selecting the correct ending to multi-sentence stories from two options. On these tasks, our model again outperforms the previous best results by significant margins - up to 8.9% on Story Cloze, and 5.7% overall on RACE. This demonstrates the ability of our model to handle long-range contexts effectively.

Semantic Similarity Semantic similarity (or paraphrase detection) tasks involve predicting whether two sentences are semantically equivalent or not. The challenges lie in recognizing rephrasing of concepts, understanding negation, and handling syntactic ambiguity. We use three datasets for this task – the Microsoft Paraphrase corpus (MRPC) [14] (collected from news sources), the Quora Question Pairs (QQP) dataset [9], and the Semantic Textual Similarity benchmark (STS-B) [6]. We obtain state-of-the-art results on two of the three semantic similarity tasks (Table 4) with a 1 point absolute gain on STS-B. The performance delta on QQP is significant, with a 4.2% absolute improvement over Single-task BiLSTM + ELMo + Attn.

Classification Finally, we also evaluate on two different text classification tasks. The Corpus of Linguistic Acceptability (CoLA) [65] contains expert judgements on whether a sentence is grammatical or not, and tests the innate linguistic bias of trained models. The Stanford Sentiment Treebank (SST-2) [54], on the other hand, is a standard binary classification task. Our model obtains an score of 45.4 on CoLA, which is an especially big jump over the previous best result of 35.0, showcasing the innate linguistic bias learned by our model. The model also achieves 91.3% accuracy on SST-2, which is competitive with the state-of-the-art results. We also achieve an overall score of 72.8 on the GLUE benchmark, which is significantly better than the previous best of 68.9.

Table 4: Semantic similarity and classification results, comparing our model with current state-of-the-art methods. All task evaluations in this table were done using the GLUE benchmark. (mc= Mathews correlation, acc=Accuracy, pc=Pearson correlation)

Method	Classification		Semantic Similarity			GLUE
	CoLA (mc)	SST2 (acc)	MRPC (F1)	STSB (pc)	QQP (F1)	
Sparse byte mLSTM [16]	-	93.2	-	-	-	-
TF-KLD [23]	-	-	86.0	-	-	-
ECNU (mixed ensemble) [60]	-	-	-	<u>81.0</u>	-	-
Single-task BiLSTM + ELMo + Attn [64]	<u>35.0</u>	90.2	80.2	55.5	<u>66.1</u>	64.8
Multi-task BiLSTM + ELMo + Attn [64]	18.9	91.6	83.5	72.8	<u>63.3</u>	<u>68.9</u>
Finetuned Transformer LM (ours)	45.4	91.3	82.3	82.0	70.3	72.8

Overall, our approach achieves new state-of-the-art results in 9 out of the 12 datasets we evaluate on, outperforming ensembles in many cases. Our results also indicate that our approach works well across datasets of different sizes, from smaller datasets such as STS-B ($\approx 5.7k$ training examples) – to the largest one – SNLI ($\approx 550k$ training examples).

5 Analysis

Impact of number of layers transferred We observed the impact of transferring a variable number of layers from unsupervised pre-training to the supervised target task. Figure 2(left) illustrates the performance of our approach on MultiNLI and RACE as a function of the number of layers transferred. This indicates that each layer in the pre-trained model contains useful functionality for solving target tasks.

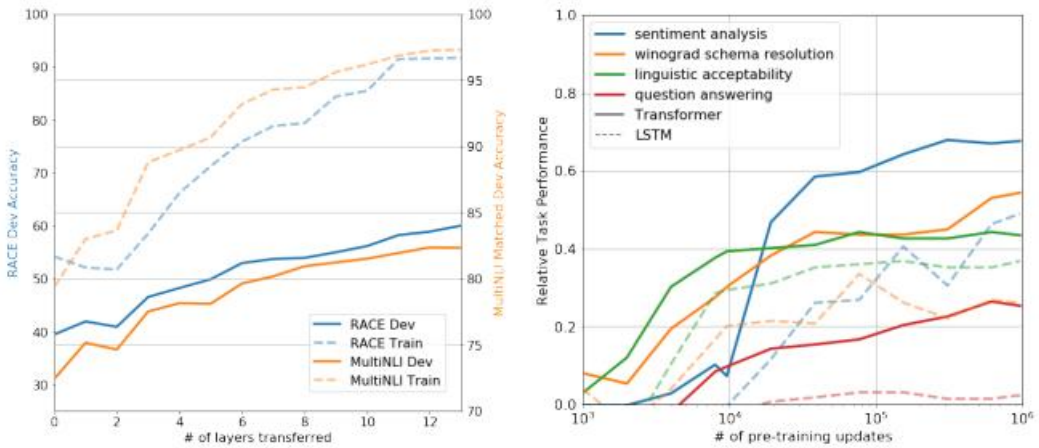


Figure 2: (left) Effect of transferring increasing number of layers from the pre-trained language model on RACE and MultiNLI. (right) Plot showing the evolution of zero-shot performance on different tasks as a function of LM pre-training updates.

Performance per task is normalized between a random guess baseline and the current state-of-the-art with a single model.

Zero-shot Behaviors We’d like to better understand why language model pre-training of transformers is effective. A hypothesis is that the underlying generative model learns to perform many of the tasks we evaluate on in order to improve its language modeling capability and that the more structured

Table 5: Analysis of various model ablations on different tasks. Avg. score is a unweighted average of all the results. (mc= Mathews correlation, acc=Accuracy, pc=Pearson correlation)

Method	Avg. Score	CoLA (mc)	SST2 (acc)	MRPC (F1)	STS _B (pc)	QQP (F1)	MNLI (acc)	QNLI (acc)	RTE (acc)
Transformer w/ aux LM (full)	74.7	45.4	91.3	82.3	82.0	70.3	81.8	88.1	56.0
Transformer w/o pre-training	59.9	18.9	84.0	79.4	30.9	65.5	75.7	71.2	53.8
Transformer w/o aux LM	75.0	47.9	92.0	84.9	83.2	69.8	81.1	86.9	54.4
LSTM w/ aux LM	69.1	30.3	90.5	83.2	71.8	68.1	73.7	81.1	54.6

attentional memory of the transformer assists in transfer compared to LSTMs. We designed a series of heuristic solutions that use the underlying generative model to perform tasks without supervised finetuning. We visualize the effectiveness of these heuristic solutions over the course of generative pre-training in Fig 2(right). We observe the performance of these heuristics is stable and steadily increases over training suggesting that generative pretraining supports the learning of a wide variety of task relevant functionality. We also observe the LSTM exhibits higher variance in its zero-shot performance suggesting that the inductive bias of the Transformer architecture assists in transfer.

For CoLA (linguistic acceptability), examples are scored as the average token log-probability the generative model assigns and predictions are made by thresholding. For SST-2 (sentiment analysis), we append the token very to each example and restrict the language model’s output distribution to only the words positive and negative and guess the token it assigns higher probability to as the prediction. For RACE (question answering), we pick the answer the generative model assigns the highest average token log-probability when conditioned on the document and question. For DPRD [46] (winograd schemas), we replace the definite pronoun with the two possible referents and predict the resolution that the generative model assigns higher average token log-probability to the rest of the

sequence after the substitution.

Ablation studies We perform three different ablation studies (Table 5). First, we examine the performance of our method without the auxiliary LM objective during fine-tuning. We observe that the auxiliary objective helps on the NLI tasks and QQP. Overall, the trend suggests that larger datasets benefit from the auxiliary objective but smaller datasets do not. Second, we analyze the effect of the Transformer by comparing it with a single layer 2048 unit LSTM using the same framework. We observe a 5.6 average score drop when using the LSTM instead of the Transformer. The LSTM only outperforms the Transformer on one dataset – MRPC. Finally, we also compare with our transformer architecture directly trained on supervised target tasks, without pre-training. We observe that the lack of pre-training hurts performance across all the tasks, resulting in a 14.8% decrease compared to our full model.

6 Conclusion

We introduced a framework for achieving strong natural language understanding with a single task-agnostic model through generative pre-training and discriminative fine-tuning. By pre-training on a diverse corpus with long stretches of contiguous text our model acquires significant world knowledge and ability to process long-range dependencies which are then successfully transferred to solving discriminative tasks such as question answering, semantic similarity assessment, entailment determination, and text classification, improving the state of the art on 9 of the 12 datasets we study. Using unsupervised (pre-)training to boost performance on discriminative tasks has long been an important goal of Machine Learning research. Our work suggests that achieving significant performance gains is indeed possible, and offers hints as to what models (Transformers) and data sets (text with long range dependencies) work best with this approach. We hope that this will help enable new research into unsupervised learning, for both natural language understanding and other domains, further improving our understanding of how and when unsupervised learning works.

附录 5 外文翻译

通过生成性预训练提高语言理解能力

摘要

自然语言理解包括文字蕴涵、问答、语义相似性评估和文档分类等多种任务。虽然大量的未标记的文本语料库是丰富的，标记的数据学习这些特定的任务是稀缺的，使它具有挑战性的区别训练模型，以执行充分。我们证明，在这些任务上的巨大收益可以通过在不同的未标记文本语料库上对语言模型进行生成性预训练，然后对每个特定任务进行区分性微调来实现。与以前的方法不同，我们在微调期间利用任务感知的输入转换来实现有效的转换，同时对模型架构要求最小的更改。我们展示了我们的方法在广泛的自然语言理解基准上的有效性。我们的通用任务不可知模型优于使用专门为每个任务设计的架构的区分性训练模型，在所研究的 12 个任务中，有 9 个任务的最新技术得到了显著改进。例如，我们在常识推理 (Stories Cloze Test)、问答 (RACE) 和文字蕴涵 (multili) 方面的绝对改进分别达到了 8.9%、5.7% 和 1.5%。

1 引言

有效地从原始文本中学习的能力对于减轻自然语言处理 (NLP) 中对监督式学习的依赖至关重要。大多数深度学习方法需要大量手工标记的数据，这限制了它们在许多缺乏注释资源的领域的适用性[61]。在这些情况下，可以利用来自未标记数据的语言信息的模型为收集更多注释提供了一种有价值的替代方法，而这种方法可能既耗时又昂贵。此外，即使在可以进行大量监督的情况下，在无监督的情况下学习良好的表示也可以显著提高性能。迄今为止，最令人信服的证据是广泛使用预先训练的词语嵌入技术[10, 39, 42]来提高一系列 NLP 任务的性能[8, 11, 26, 45]。

然而，从未标记的文本中利用超过词级的信息是具有挑战性的，主要原因有两个。首先，目前还不清楚哪种类型的优化目标在学习对于转换有用的文本表示方面最有效。最近的研究着眼于不同的目标，如语言建模[44]、机器翻译[38]和语篇连贯[22]，每种方法在不同的任务上都优于其他方法。现有的技术包括对模型体系结构进行特定于任务的更改，使用复杂的学习方案[21]和添加辅助学习目标[50]。这些不确定性使得难以开发有效的语言处理半监督学习方法。

本文采用无监督预训练和监督微调相结合的方法，探索了一种语言理解任务的

半监督方法。我们的目标是学习一种通用的表示法，这种表示法几乎不适应大范围的任务。我们假设可以访问大量未标记的文本和几个带有手动注释的训练示例(目标任务)的数据集。我们的设置不要求这些目标任务与未标记的语料库在同一域中。我们采用两阶段的培训程序。首先，我们在未标记的数据上使用语言建模目标来学习神经网络模型的初始参数。随后，我们使用相应的监督目标将这些参数适应于目标任务。

对于我们的模型体系结构，我们使用 Transformer [62]，已经证明它在各种任务(如机器翻译[62]、文档生成[34]和语法解析[29])上表现强劲。与循环网络等替代方案相比，这种模型选择为我们提供了更结构化的内存，用于处理文本中的长期依赖关系，从而在不同任务之间产生强大的传输性能。在传输过程中，我们利用从遍历风格方法[52]派生出来的特定于任务的输入适应性，该方法将结构化文本输入作为一个连续的标记序列处理。正如我们在实验中所证明的那样，这些适应性使我们能够有效地调整预先训练的模型的架构，并对其进行最小程度的改变。

我们评估了我们的方法在四种类型的语言理解任务-自然语言推理，问题回答，语义相似性，和文本分类。我们的通用任务不可知模型优于使用专门为每个任务设计的架构的区分性训练模型，在所研究的 12 个任务中，有 9 个任务的最新技术得到了显著改进。例如，我们在常识推理 (Stories Cloze Test) [40]、回答问题 (RACE) [30]、文字蕴涵 (multili) [66] 和最近引入的 GLUE 多任务基准[64] 方面的绝对改进分别达到了 8.9%、5.7%、1.5% 和 5.5%。我们还分析了预先训练模型在四种不同设置下的零拍行为，并证明它为下游任务获得了有用的语言知识。

2 相关工作

NLP 的半监督学习我们的工作大体上属于自然语言的半监督学习范畴。这种范式已经引起了人们的极大兴趣，其应用于序列标记[24, 33, 57]或文本分类[41, 70]等任务。最早的方法使用未标记的数据来计算单词水平或短语水平的统计数据，然后用作监督模型的特征[33]。在过去的几年里，研究人员已经证明了使用文字嵌入的好处[11, 39, 42]，这是训练在未标记语料库，以提高在各种任务的表现[8, 11, 26, 45]。然而，这些方法主要转移词级信息，而我们的目标是捕获更高级别的语义。

最近的方法已经研究了从未标记的数据中学习和利用超过词级语义的方法。短语水平或句子水平的嵌入，可以训练使用一个未标记的语料库，已被用来编码文本

到适当的向量表示各种目标任务[28, 32, 1, 36, 22, 12, 56, 31]。

无监督式学习是半监督学习的一种特殊情况，其目标是找到一个良好的初始点，而不是修改监督式学习的目标。早期的工作探索了将该技术用于图像分类[20, 49, 63]和回归任务[3]。随后的研究[15]表明，预训练作为一种正则化方案，能够在深度神经网络中更好地泛化。在最近的工作中，该方法已被用于帮助训练深层神经网络的各种任务，如图像分类[69]，语音识别[68]，实体消歧[17]和机器翻译[48]。

最接近我们的工作包括使用语言建模目标对神经网络进行预先训练，然后在监督下对目标任务进行微调。Dai 等[13]和 Howard 和 Ruder [21]遵循这种方法来改善文本分类。然而，尽管预训练阶段有助于捕获一些语言信息，但他们对 LSTM 模型的使用将其预测能力限制在短期内。相比之下，我们对变压器网络的选择允许我们捕捉更长范围的语言结构，正如我们的实验所证明的那样。此外，我们还证明了我们的模型在更广泛的任务范围内的有效性，包括自然语言推理，释义检测和故事完成。其他方法[43, 44, 38]使用来自 2 个预先训练的语言或机器翻译模型的隐藏表示作为辅助特征，同时训练目标任务的监督模型。这涉及到每个单独目标任务的大量新参数，而我们在转移过程中需要对我们的模型架构进行最小程度的改变。

辅助训练目标添加辅助无监督训练目标是半监督学习的一种替代形式。Collobert 和 Weston [10]的早期工作使用了各种各样的辅助 NLP 任务，如词性标注、组块、命名实体识别和语言建模，以改进语义角色标注。最近，Rei [50]为他们的目标任务目标添加了一个辅助语言建模目标，并展示了序列标记任务的性能增益。我们的实验也使用辅助目标，但正如我们所显示的，无监督的预训练已经学习了与目标任务相关的几个语言方面。

3 框架

我们的培训程序包括两个阶段。第一阶段是在大量的文本语料库中学习大容量的语言模型。接下来是一个微调阶段，我们使模型适应带有标签数据的区分性任务。

无监督的预训练给定一个无监督的令牌语料库 $u = \{ u_1, a. . . u_n \}$ ，我们使用一个标准的语言建模目标来最大化以下可能性：

$L_1(u) = \sum_i \log p(u_i | u_{i-k}, . . . u_{i-1}; \theta)$ (1)，其中 k 是上下文窗口的大小，条件概率 p 采用参数 θ 的神经网络建模。这些参数使用随机梯度下降进行训练[51]。

在我们的实验中，我们使用多层 Transformer 解码器[34]作为语言模型，这是

变压器[62]的一个变体。该模型对输入上下文令牌应用一个多头自注意操作，然后按位置前馈层生成目标令牌的输出分布：

$$H_0 = UWe + Wp$$

$$H_i = \text{Transformer_Block}(H_{i-1}) \text{ something } i \in [1, n]$$

$$P(u) = \text{softmax}(h_n W T e) \quad (2)$$

其中 $u = (u_k, \dots, u_1)$ 是令牌的上下文向量， n 是层数， We 是令牌嵌入矩阵， Wp 是位置嵌入矩阵。

监督微调

在对模型进行情商目标训练之后。1，我们调整参数来监督目标任务。我们假设一个带标签的数据集 c ，其中每个实例由一个输入标记序列 x_1, \dots, x_m 以及一个标签 y 组成。输入通过我们的预先训练的模型，以获得最终的变压器块的激活 h_{m+1} ，然后馈入一个附加的线性输出层，参数为 y ，以预测 $y: P(y | x_1, \dots, x_m) = \text{softmax}(h_{m+1} Wy)$ 。 (3)

这给了我们以下最大化的目标: $L_2(c) = -\sum (x, y) \log p(y | x_1, \dots, x_m)$ 。 (4)

我们还发现，将语言建模作为微调的辅助目标有助于学习：(a) 提高监督模型的通用性，(b) 加速收敛。这与之前的工作一致[50, 43]，他们也观察到这样一个辅助目标改善了性能。具体而言，我们优化了以下目标(重量 λ) : $L_3(c) = L_2(c) + \lambda * L_1(c)$ (5)

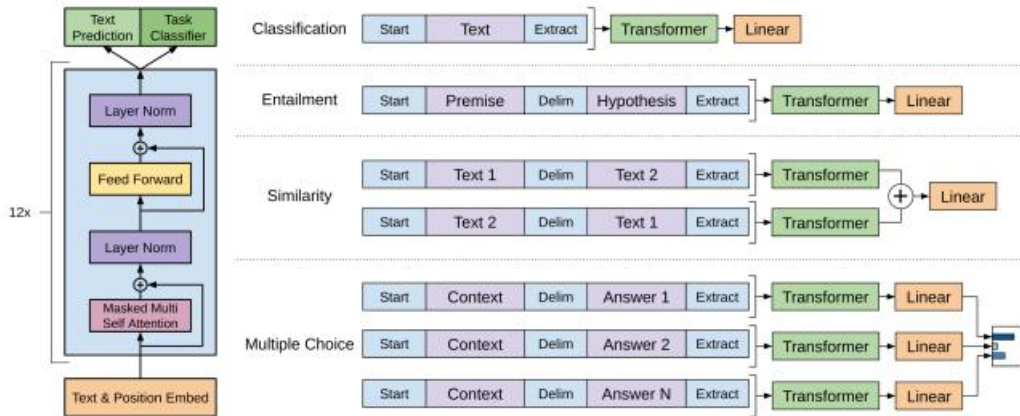


图 1: (左)本工作中使用的 Transformer 体系结构和培训目标。(right) Input transformation for fine-tuning on different tasks (右图)输入转换，用于对不同任务进行微调。

我们将所有结构化输入转换为令牌序列，由我们的预先训练的模型进行处理，

然后是线性 + softmax 层。特定于任务的输入转换对于某些任务，比如文本分类，我们可以像上面描述的那样直接微调我们的模型。某些其他任务，如问题回答或文字蕴涵，有结构化的输入，如有序的句子对，或文档、问题和答案的三元组。由于我们预先训练好的模型是基于连续的文本序列来训练的，我们需要一些修改来将它应用到这些任务中。以前的工作提出了在转移表示之上的学习任务特定架构[44]。这种方法重新引入了大量的特定于任务的定制，并且没有对这些额外的架构组件使用转移学习。相反，我们使用遍历风格的方法[52]，我们将结构化输入转换成一个有序的序列，我们的预先训练的模型可以处理。这些输入转换允许我们避免跨任务对架构进行广泛的更改。我们在下面提供了这些输入转换的简要描述，图 1 提供了一个可视化的说明。所有的转换包括添加随机初始化的开始和结束标记(hsi, hei)。

文字蕴涵对于蕴涵任务，我们将前提 p 和假设 h 标记序列连接起来，中间有一个分隔符标记(\$)。

对于相似性任务，两个被比较的句子没有固有的顺序。为了反映这一点，我们修改输入序列以包含可能的句子顺序(中间有一个分隔符)，并分别处理每个句子以产生两个序列表示 h m l，这两个序列表示在输入线性输出层之前按元素顺序添加。

问题回答和常识推理对于这些任务，我们给出了一个上下文文档 z，一个问题 q 和一组可能的答案{ ak }。我们将文档上下文和问题与每个可能的答案连接起来，在其中添加一个分隔符，以得到[z; q; \$; ak]。这些序列中的每一个都用我们的模型独立处理，然后通过 softmax 层进行标准化，以在可能的答案上产生输出分布。

4.1 设置无监督的预先训练

我们使用 BooksCorpus 数据集[71]来训练语言模型。它包含超过 7000 本独特的未出版书籍，来自各种类型，包括冒险，幻想和浪漫。至关重要的是，它包含了很长一段连续的文本，这使得生成模型能够学会对远程信息进行调整。一个替代的数据集，1B Word Benchmark，被类似的方法 ELMo [44]使用，大小大致相同

表 1: 我们实验中使用的不同任务和数据集的列表。

Task	Datasets
Natural language inference	SNLI [5], MultiNLI [66], Question NLI [64], RTE [4], SciTail [25]
Question Answering	RACE [30], Story Cloze [40]
Sentence similarity	MSR Paraphrase Corpus [14], Quora Question Pairs [9], STS Benchmark [6]
Classification	Stanford Sentiment Treebank-2 [54], CoLA [65]

但是在句子层面上进行了混乱——破坏了长程结构。我们的语言模型在这个语

料库中达到了一个非常低的令牌级困惑度 18.4。

我们的模型在很大程度上遵循了最初的变压器工作[62]。我们训练了一个只有 12 层解码器的变压器，具有掩蔽的自我注意头(768 维状态和 12 个注意头)。对于位置型前馈网络，我们使用了 3072 维内部状态。我们使用 Adam 优化方案[27]，最大学习速率为 2.5×10^{-4} 。在最初的 2000 次更新中，学习率从零线性增加，并使用余弦时间表退火为 0。我们在 64 个随机抽样的小批量 512 个令牌的连续序列上训练 100 个纪元。由于层次范数[2]在整个模型中被广泛使用，因此 $n(0, 0.02)$ 的简单权重初始化是足够的。我们使用了一个字节对编码(BPE)词汇表，其中包含 40,000 次合并[53]和剩余、嵌入和注意力丢失，其正则化率为 0.1。我们还采用了[37]中提出的 12 正则化的修改版本，所有非偏倚或增益权重的 $w = 0.01$ 。对于激活函数，我们使用高斯误差线性单位(GELU)[18]。我们使用了学习位置嵌入，而不是原始工作中提出的正弦版本。我们使用 `ftfy library 2` 清理 BooksCorpus 中的原始文本，标准化一些标点符号和空格，并使用 `spaCy` 标记器。

微调细节除非特别指定，否则我们重复使用无监督的预先培训中的超参数设置。我们以 0.1 的比率将辍学率添加到分类器中。对于大多数任务，我们使用 6.25×10^{-5} 的学习率和 32 的批量大小。我们的模型很快调整，3 个时代的培训足以满足大多数情况。我们使用线性学习速率衰减时间表，热身超过 0.2% 的培训。 Λ 设置为 0.5。

4.2 监督微调

我们在各种监督任务上进行实验，包括自然语言推理、问答、语义相似性和文本分类。其中一些任务是最近发布的 GLUE 多任务基准[64]的一部分，我们使用了这些任务。图 1 提供了所有任务和数据集的概述。

自然语言推理自然语言推理的任务，也称为文字蕴涵的识别，包括阅读一对句子，并从蕴涵、矛盾或中性的一个句子判断它们之间的关系。虽然最近有很多的兴趣[58, 35, 44]，任务仍然具有挑战性，因为存在着各种各样的现象，如词汇含义，共指，词汇和句法歧义。我们使用五个不同来源的数据集进行评估，包括图像标题(SNLI)、语音转录、通俗小说和政府报告(MNLI)、维基百科文章(QNLI)、科学考试(SciTail)或新闻文章(RTE)。

表 2 详细说明了我们的模型和以前最先进的方法的不同 NLI 任务的各种结果。我们的方法明显优于 5 个数据集中的 4 个，在 MNLI、SciTail、QNLI 和 SNLI 方面的绝对改进分别达到 1.5%、5%、5.8% 和 0.6%。这表明我们的模型能够更好地

推理多个句子，并处理语言模糊性的各个方面。在 RTE 上，我们评估的一个较小的数据集(2490 个例子)，我们达到了 56% 的准确率，低于多任务 biLSTM 模型报告的 61.7%。鉴于我们的方法在更大的 NLI 数据集上的强大表现，我们的模型很可能也将受益于多任务训练，但我们目前还没有探索这一点。

表 2: 自然语言推理任务的实验结果，将我们的模型与当前最先进的方法进行比较。5x 表示 5 个模型的集合。所有数据集都使用准确性作为评估指标。

表 3: 回答问题和常识推理的结果，将我们的模型与当前最先进的方法进行比较。.9x 意味着 9 个模型的集合。问题回答和常识推理另一个需要单句和多句推理的任务是问题回答。我们使用最近发布的 RACE 数据集[30]，由中学和高中考试相关问题的英文段落组成。这个语料库已经被证明包含了比其他数据集如 CNN [19] 或 SQuAD [47]更多的推理类型的问题，为我们的模型提供了完美的评估，这个模型被训练用于处理远程上下文。此外，我们评估 Story 完形填空测试[40]，其中涉及从两个选项中选择多句故事的正确结尾。在这些任务上，我们的模型再次以显著的优势超越了之前的最佳结果——在 Story Cloze 上达到了 8.9%，在 RACE 上达到了 5.7%。这证明了我们的模型有效处理远程上下文的能力。

语义相似性语义相似性(或释义检测)任务涉及预测两个句子在语义上是否等价。挑战在于识别概念的重新表述，理解否定，以及处理句法歧义。我们使用三个数据集来完成这项任务——Microsoft Paraphrase corpus (MRPC) [14] (收集自新闻来源)、Quora Question Pairs (QQP)数据集[9]和 Semantic text Similarity benchmark (STS-B) [6]。我们获得了三个语义相似性任务中的两个(表 4)的最新结果，在 STS-B 上获得了 1 点绝对增益。QQP 的性能增量是显著的，比单任务 BiLSTM + ELMo + Attn 绝对提高了 4.2%。

分类最后，我们还评估了两个不同的文本分类任务。语言可接受性语料库 (CoLA) [65]包含了一个句子是否符合语法的专家判断，并测试了训练有素的模型固有的语言偏见。另一方面，斯坦福情感树库 (SST-2) [54]是一个标准的二进制分类任务。我们的模型在可口可乐上得到了 45.4 分，这是一个特别大的飞跃，超过了之前最好的结果 35.0，显示了我们的模型学习的固有语言偏见。该模型在 sst-2 上也达到了 91.3% 的准确率，这与最先进的结果具有竞争力。我们还在 GLUE 基准测试中取得了 72.8 的总分，这比之前最好的 68.9 分要好得多。

表 4: 语义相似性和分类结果，将我们的模型与当前的状态方法进行比较。本表

中的所有任务评估都是使用 GLUE 基准进行的。(mc = Mathews 相关性, acc = 精度, pc = Pearson 相关性)

总的来说,我们的方法在我们评估的 12 个数据集中的 9 个数据集中实现了新的最先进的结果,在许多情况下表现优于集合。我们的结果还表明,我们的方法适用于不同规模的数据集,从较小的数据集,如 STS-B (约 5.7 k 训练示例)-最大的一个-SNLI (约 550k 训练示例)。

5 分析

转移层数的影响我们观察了从无监督的预训练到有监督的目标任务转移不同层数的影响。图 2(左)说明了我们在 multili 和 RACE 上的方法的性能作为转移层数的函数。我们观察到传输嵌入提高性能的标准结果,并且每个变压器层为 multili 上的完全传输提供了高达 9% 的进一步好处。这表明预先训练好的模型中的每一层都包含了解决目标任务的有用功能。

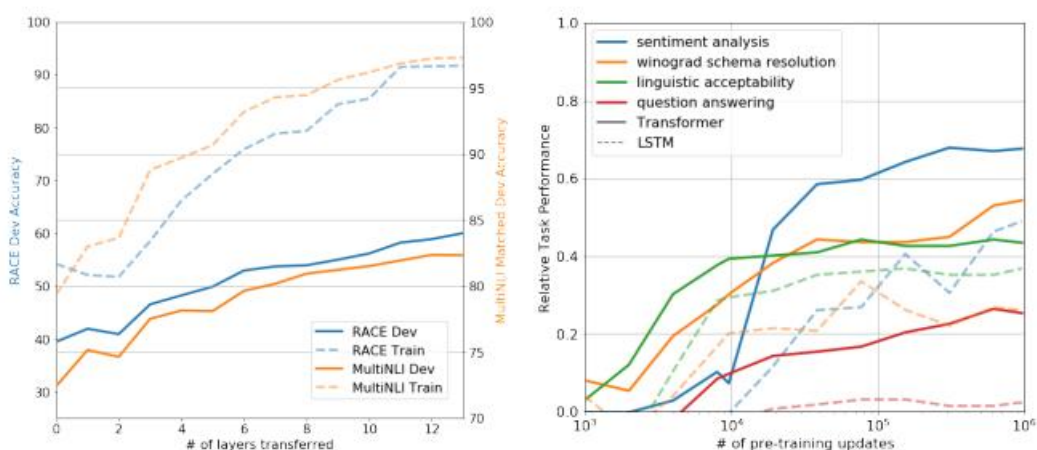


图 2: (左)从 RACE 和 multili 上的预训练语言模型转移越来越多层的效果。(右)作为 LM 预训练更新的函数,显示不同任务零拍性能的演变的 Plot。

每个任务的表现以随机猜测基线和当前最先进的单一模型之间进行标准化。零镜头行为我们想更好地理解为什么语言模型预先训练变压器是有效的。一个假设是,潜在的生成模型学习执行我们评估的许多任务,以提高其语言建模能力,并且结构化程度越高

对于可乐(语言可接受性),例子被评分为平均令牌日志概率的生成模型分配和预测是由阈值。对于 SST-2(sentiment analysis),我们在每个例子中都附加一个标记,并将语言模型的输出分布限制为正负两个词,然后猜测它赋予较高概率的标记

作为预测。对于 RACE（问题回答），我们选择答案生成模型分配最高的平均令牌日志概率条件下的文件和问题。对于 DPRD [46] (winograd 模式)，我们用两个可能的指称代词替换定义代词，并预测了生成模型在替换后赋予序列其余部分较高的平均令牌对数概率的分辨率。

消融研究我们进行三种不同的消融研究(表 5)。首先，我们在微调期间检查我们的方法在没有辅助 LM 目标的情况下的性能。我们观察到辅助目标有助于 NLI 任务和 QQP。总体而言，趋势表明，较大的数据集受益于辅助目标，但较小的数据集不受益。其次，我们通过比较使用相同框架的单层 2048 单元 LSTM 来分析 Transformer 的影响。我们观察到当使用 LSTM 而不是 Transformer 时，平均分下降了 5.6。LSTM 只在一个数据集 MRPC 上优于 Transformer。最后，我们还比较了我们的变压器架构直接训练监督目标任务，没有预先训练。我们观察到，缺乏预先训练会影响所有任务的表现，与我们的完整模型相比，降低了 14.8%。

6 结论

我们引入了一个框架，通过生成性预训练和区分性微调，使用单一的任务不可知模型实现强的自然语言理解。通过在一个多样的语料库上进行预先训练，我们的模型获得了重要的世界知识和处理长距离依赖关系的能力，这些知识和能力随后被成功地转移到解决歧视性任务上，例如问题回答、语义相似性评估、蕴涵确定和文本分类，从而改进了我们研究的 12 个数据集中的 9 个数据集的技术水平。长期以来，使用无监督(预)训练来提高辨别性任务的表现一直是机器学习研究的一个重要目标。我们的工作表明，实现显著的性能增益确实是可能的，并提供了关于什么模型(变形金刚)和数据集(具有长程依赖性的文本)最适合这种方法的提示。我们希望这将有助于对自然语言理解和其他领域的非监督式学习进行新的研究，进一步提高我们对非监督式学习如何以及何时起作用的理解。

燕山大学毕业设计（论文）指导教师意见书

学院：信息科学与工程学院

专业(方向)：计算机科学与技术

题目名称		法律学习智能助手的设计与实现					
学号		202011040135	姓名	郝雁飞	班级	5 班	
序号	评审项目	指标				满分	评分
1	项目过程的管理意识 5 分	(2 分) 1.1 能独立查阅文献和调研；能较好地写出开题报告和中期报告；项目实施过程中各要素组织基本齐全、合理； (3 分) 1.2 有综合、收集和正确利用各种信息及获取新知识的能力，能考虑非技术指标因素。				5	3
2	自我学习意识和工作态度 5 分	(2 分) 2.1 参加了毕业设计（论文）的各个训练环节，按期独立完成规定的工作任务，可以很好的综合运用所学专业知解决遇到的问题；能够独立查阅相关资料内容，可以很好的解决实际问题。 (3 分) 2.2 遵守纪律，学习认真；尊敬指导教师，能够听取指导教师的意见和指导；不存在学术不端行为。				5	3
是否同意参加答辩： <input checked="" type="checkbox"/> 同意 <input type="checkbox"/> 不同意						总分	6
评语： <div style="text-align: center; font-size: 2em; margin-top: 20px;">同意答辩</div> <div style="text-align: right; margin-top: 20px;"> 指导教师签名：马勤勇 日 期：2024.5.30 </div>							

燕山大学毕业设计（论文）答辩评审表

学院： 信息科学与工程学院 专业（方向）： 计算机科学与技术

题目名称		法律学习智能助手的设计与实现				
学号	202011040135	姓名	郝雁飞	班 级	5 班	
指导教师姓名		马勤勇	职称	讲师		
序号	评审项目	指标			满分	评分
1	开发工具运用能力 10 分	(5 分)1.1 设计能力。可以很好的应用开发工具设计符合国家或行业标准的毕业设计(论文)，体现一定分析解决本专业实际问题的综合设计能力，基本达到选题的目标要求。 (5 分) 1.2 系统能力。体现出较为扎实的专业基本知识，基本能够综合运用开发工具解决毕业设计(论文)相关问题。软件系统能够正常运行，数据正确，代码命名规范、缩进风格统一，能够准确的无误的讲解代码的运行逻辑。			10	7
2	工作总结和成果展示 10 分	(5 分) 2.1 工作总结。答辩表述清楚，语言基本流畅，整体内容能够体现毕业设计(论文)主题；体现出较为扎实的专业基本知识，核心概念基本明确。 (5 分) 2.2 成果展示。态度认真，对完成的毕业设计(论文)内容完全掌握；回答问题有理有据，基本概念清楚，主要问题回答准确，有一定深度。			10	6
					总分	13
评语： <div>答辩委员会小组成员： 姓名职称（签名）： 姓名职称（签名）： 姓名职称（签名）： 答辩委员会（小组）负责人签名：</div> <div>姓名职称（签名）： 姓名职称（签名）： 姓名职称（签名）：</div> <div>2024年6月03日</div>						

燕山大学毕业设计（论文）评分表

开题考核：

开题考核（满分 20 分）			
当前社会发展现状的认识 （5 分）	专业领域技术发展现状的调研 （5 分）	问题分析和解决方案 （10 分）	开题成绩

中期考核：

中期考核（满分 30 分）			
设计方案的可行性分析 （8 分）	问题研究能力的评价 （10 分）	系统设计能力的考核 （12 分）	中期成绩

燕山大学毕业设计（论文）评分表

指导教师评分：

导师评分（满分 10 分）		
项目过程的管理意识 （5 分）	自我学习意识和工作态度 （5 分）	导师评分

指导教师签字：_____ 年 月 日

毕业论文答辩：（满分 40 分）

报告和论文的撰写质量 （10 分）	外文资料阅读与翻译 （5 分）	创新和发展意识 （5 分）

开发工具运用能力 （10 分）	工作总结和成果展示 （10 分）	答辩成绩

答辩组组长签字：_____ 年 月 日

综合其开题成绩、中期成绩、导师评分、答辩成绩，

该本科生毕业设计(论文)的总成绩为：_____。

(☐A+ ☐A ☐B+ ☐B ☐C+ ☐C ☐D+ ☐D ☐F)