

厦門大學

本科毕业设计

基于 RAG 技术的智能标书撰写系统

Intelligent Bid Document Writing System Based on RAG Technology

姓名：朱纪玉

学号：22920212204333

学院：信息学院

专业：网络空间安全

年级：2021 级

校内指导教师：罗晔 副教授

二〇二五年四月三十日

厦门大学本科学位论文诚信承诺书

本人呈交的学位论文是在导师指导下独立完成的研究成果。本人在论文写作中参考其他个人或集体已经发表的研究成果，均在文中以适当方式明确标明，并符合相关法律法规及《厦门大学本科毕业论文（设计）规范》。

该学位论文为（ ）课题（组）的研究成果，获得（ ）课题（组）经费或实验室的资助，在（ ）实验室完成（请在以上括号内填写课题或课题组负责人或实验室名称，未有此项声明内容的，可以不作特别声明）。

本人承诺辅修专业毕业论文（设计）（如有）的内容与主修专业不存在相同与相近情况。

学生声明（签名）：

年 月 日

摘 要

啥也没写

关键词：检索增强生成；大语言模型；标书

Abstract

empty

Key Words: Undergraduate Thesis; LaTeX; Xiamen University

目 录

第一章 绪论	1
1.1 招投标行业数字化发展现状与挑战	1
1.2 RAG 技术	2
1.2.1 RAG 简介	2
1.2.2 RAG 技术在智能标书撰写工具中的应用潜力	2
1.3 本文工作	3
1.4 论文结构	3
第二章 相关工作	4
2.1 招投标领域的研究现状	4
2.1.1 传统投标方法	4
2.1.2 现有智能标书撰写工具	5
2.1.3 RAG 技术研究现状	5
第三章 智能标书撰写系统设计	7
3.1 需求分析	7
3.2 系统架构设计	7
3.3 企业信息管理	7
3.4 方案库（知识图谱）	7
3.5 资历库	9
3.6 标书内容生成	10
3.6.1 识别招标需求	10
3.6.2 生成标书目录	10
3.6.3 生成标书正文	10
3.7 风格迁移	11
3.8 决策参考	11
3.9 后期处理	12
第四章 系统实现与实验	13
4.1 系统开发环境与工具	13
4.1.1 系统环境	13
4.1.2 开发工具	13

4.1.3 系统框架	13
4.2 系统功能实现	13
4.2.1 方案库 (知识图谱)	13
4.2.2 招标需求识别模块	14
4.2.3 标书目录生成模块	14
4.2.4 标书正文生成模块	15
4.2.5 风格迁移模块	15
4.2.6 决策参考模块	15
4.2.7 后期处理模块	15
4.2.8 模板管理模块	15
4.3 系统测试	15
4.3.1 测试方法	15
4.3.2 测试结果	15
第五章 总结与展望	16
参考文献	17
附录	18
致谢	19

Contents

Chapter 1 Introduction	1
1.1	1
1.2	2
1.2.1	2
1.2.2	2
1.3	3
1.4	3
Chapter 2	4
2.1	4
2.1.1	4
2.1.2	5
2.1.3	5
Chapter 3	7
3.1	7
3.2	7
3.3	7
3.4	7
3.5	9
3.6	10
3.6.1	10
3.6.2	10
3.6.3	10
3.7	11
3.8	11
3.9	12
Chapter 4	13
4.1	13
4.1.1	13
4.1.2	13

4.1.3	13
4.2	13
4.2.1	13
4.2.2	14
4.2.3	14
4.2.4	15
4.2.5	15
4.2.6	15
4.2.7	15
4.2.8	15
4.3	15
4.3.1	15
4.3.2	15
Chapter 5 Conclusion and Future Work	16
References	17
Appendix	18
Acknowledgements	19

第一章 绪论

1.1 招标投标行业数字化发展现状与挑战

随着我国经济社会的快速发展，招标投标行业作为市场经济的重要组成部分，其作用日益凸显。招标投标行业的健康发展对于促进经济增长、提高市场竞争力、维护社会公平正义具有重要意义。招标投标行业具有广泛的市场覆盖面，包括基础设施建设、公共服务、工业生产等多个领域。据统计，2024 年，我国招标采购规模约 34.74 万亿元，其中，政府公开采购招标规模 2.49 万亿元，工程招标采购 26.82 万亿元，其他采购招标 5.43 万亿元；^[2] 目前，在国家进一步“放管服”、加速推进“优化营商环境”的大背景下，招标采购加速向信息化、数字化发展。^[1] 一方面，招标采购交易全流程逐渐实现网络信息化、数字化：2024 年全国电子招标投标平台日均处理量突破 100 万单，AI 评标系统在中部六省试点准确率达 92%^[3]。另一方面，招标投标的服务模式从单一、程序性服务向复合型、综合性咨询服务转型。



图 1-1: 中国招标投标公共服务平台已经支持 CA 互认和远程异地评标和评标专家库共享

目前，地方政府公共资源交易平台功能日臻完善，各大央企集团大多已建成使用招标采购电子交易平台，行业领域内招标采购业务活动基本实现全流程在线化，目前正在加快向数字化和智能化方向转型，AI 应用基础良好。总体而言，与招标人相比，投标人对人工智能技术应用驱动力不足，在功能开发和应用方面均不如招标人及资源交易中心广泛和深入。投标人也已经逐步意识到大语言模型对招标采购行业的深远影响，目前已经有投标人使用 GPT 编制投标文件部分章节，如施工组织方案。^[4]

从开发者的角度来看，开发为投标人服务的智能工具存在以下难点：1. 招标、投标文件往往包含大量的专业术语，需要遵循特定的逻辑或行业规则，目前的自然语言处理技术理解和生成这些内容时存在一定局限性。2. 在投标过程中，

投标人需要综合市场、企业、客户、竞争对手等多方面信息进行决策，仅仅依靠自然语言处理技术或简单的检索技术难以提供有参考价值的决策建议。3. 招标采购行业涉及大量的商业机密和敏感信息，尤其是对于提供 SAAS（Software as a Service，软件即服务）的服务商而言，在技术上和社会信任上都需要长期努力。

1.2 RAG 技术

1.2.1 RAG 简介

RAG（Retrieval-Augmented Generation，检索增强生成）是一种连接外部数据源以增强大语言模型（LLM）输出质量的技术。这种技术帮助 LLM 访问私有数据或特定领域的数据，并解决幻觉问题。因此，RAG 已被广泛用于许多通用的生成式 AI（GenAI）应用中，如 AI 聊天机器人和推荐系统。RAG 技术一般可以划分为信息检索和生成两个阶段。

在信息检索阶段，系统解析用户输入的文本，从构建的专业知识库中，通过向量检索、图检索等方式筛选相关程度最高的文本片段。在生成阶段，模型将检索到的文档片段作为上下文与用户输入的文本一起输入到 seq2seq 模型中，得到输出。

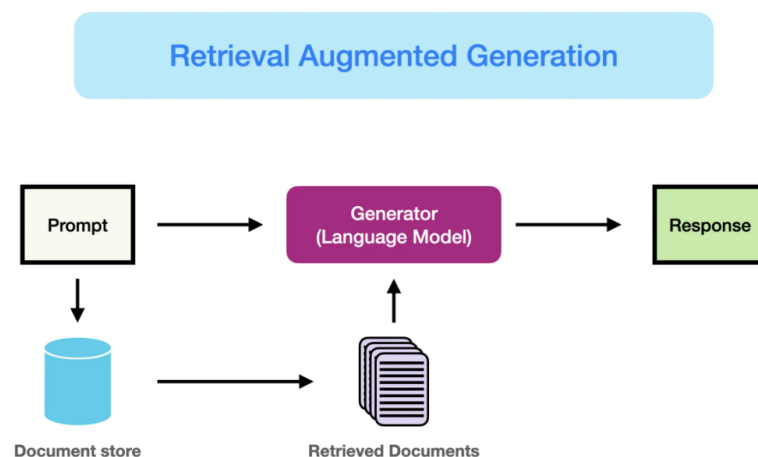


图 1-2: RAG 技术基本原理

1.2.2 RAG 技术在智能标书撰写工具中的应用潜力

标书撰写的过程可以分解为招标要求理解和标书内容生成两个任务，传统的大语言模型能够满足一般的文本理解和内容生成任务，但受限于模型的训练数据，对于标书撰写这样涉及专业知识和企业信息以及多种综合因素的复杂任

务，存在一定局限性。而 RAG 技术通过检索和利用外部知识库中的相关信息，能够为标书撰写提供更丰富的知识支持和更准确的生成内容。

RAG 技术预计可以实现以下功能：1、理解招标要求：通过检索与招标文件相关的行业标准、法规和历史案例，RAG 模型能够更精准地理解招标文件的核心要求，并利用相关知识指导内容生成。2、生成标书内容：结合检索到的外部知识和输入的招标文件要求，RAG 模型可以生成可靠的标书正文内容，并且依靠检索提供的证据链，为人工复核提供便利。3、提供决策参考：RAG 模型可以检索市场数据、竞争对手信息等外部知识，结合预设的竞价、博弈模型，为标书撰写提供有价值的决策参考。

1.3 本文工作

本文工作主要包括以下几个方面：分析现有的智能标书撰写工具，设计基于 RAG 技术的智能标书撰写系统；实现“招标需求理解-标书框架生成-标书内容生成” workflow；实现基于知识图谱的方案库，将用户上传的文件，解析为可用于 RAG 的知识图谱；实现基于 Prompt 的风格迁移功能，为用户提供预设或自定义的风格选项；实现“数据检索-工具调用-输出决策”的决策参考 workflow；

1.4 论文结构

第一章：本章介绍招投标行业的数字化发展现状，阐述为投标人服务的智能工具的开发难点。随后，简要介绍 RAG 技术并分析标书撰写中的应用潜力。接着，概述本文的主要工作内容，和论文的整体结构。

第二章：本章介绍传统的招投标方法和现有的智能标书撰写工具，分析其优缺点。详细回顾 RAG 技术的相关研究进展，其中重点介绍基于知识图谱的 RAG 技术。

第三章：本章将对智能标书撰写系统进行需求分析。随后阐述系统架构设计。

第四章：本章将描述系统的开发环境和工具，详细介绍系统功能的具体实现。

第五章：本章将总结本文的主要研究成果，并讨论系统的优势和可能的改进方向。

第二章 相关工作

2.1 招投标领域的研究现状

2.1.1 传统投标方法

一般标书的结构分为商务部分、技术部分和报价部分（但招标文件特殊要求格式除外）。

商务部分：一般包括投标人说明、厂家介绍、业绩、合同、产品授权书、法人授权书、三证、资格证书、交货期、付款方式、售后服务、承诺书、商务偏离表、商务应答、备品备件专用工具清单等，要严格按照标书内容要求及顺序编写。

技术部分：一般包括投标设备技术说明、图纸设计、技术参数、产品配置、技术规格偏离表，技术力量简介、安装施工方案、产品质量、产品简介、产品彩页等等，要严格按照标书内容要求及顺序编写。其中技术偏离表需要标注正偏离、负偏离、无偏离；如投标产品的技术指标优于招标要求即为正偏离，反之为负偏离，符合招标要求即为无偏离；

报价部分，一般包括报价一览表、分项报价表、投标函、投标承诺书、投标单位名称、投标代表签字、法人代表签字等，要严格按照招标文件要求及顺序编写。

标书编写可以分为初步目录编排和后期目录编排两种方式。初次目录编排：根据招标文件的要求，初步编写投标文件目录；为了方便收集投标书的资料。对评分点、控标点、优势应在初步目录中标注，其目的是为了标书制作者重视该部分文档。后期目录编排：按事先拟定好的投标文件目录，对正文内容的标题设置为标题 1、标题 2、标题 3、标题 4 等，然后自动生成投标文件的目录。

在标书电子版制作完成后，参与人员需要进行交叉检查，对错误的地方进行指正修改；需要重点审查的地方有：正本制作正副本内容一致；字体、格式是否统一；审查报价产品明细是否符合招标产品需求明细（包括产品型号和数量），分项（分包）报价是否符合和正确；审查报价表中的分报价和总报价的计算、大小写是否正确，报价表、投标一览表、投标函中的报价大小写是否一致；审查开标文件书写格式是否与招标方的要求一致；

在标书打印完成后，需签字处完成签字盖章，最终审查无误后，进行密封。

传统的标书编写方法依赖人工操作，为确保标书的质量和合规性，投标团队

需要投入大量的时间和精力进行信息收集、内容撰写和格式调整，存在较多重复性、繁琐性的任务。开发智能化的标书撰写工具，将投标工作中的重复性和繁琐性任务自动化，有望提高标书编写的效率和准确性。

2.1.2 现有智能标书撰写工具

目前，市面上已经出现了一批智能标书撰写工具，如链企标书、星火投标等。这些工具的开发背景各不相同，部分由传统招标采购行业的专业公司开发，部分由互联网科技巨头或新兴的互联网创业公司开发。在服务形式方面，大多数工具提供 SaaS（Software as a Service，软件即服务）模式，用户通过网络浏览器登录平台使用标书撰写工具，少数工具提供本地部署服务。

这些智能标书撰写工具的核心功能基本相同，仅在功能完成度和用户体验上存在差异。总体而言，这些工具具备以下几项核心功能：招标文件解析：解析用户上传的投标文件，提取标书撰写的参考信息。投标目录生成：根据自动提取或用户输入的参考信息，生成标书目录。投标正文生成：根据标书目录，生成标书内容。除此以外，部分工具有在线编辑、风格迁移、文件云端存储引用、封面模板等附加功能。



图 2-1: 现有 AI 工具的核心功能，图中以链企 AI 为例

目前市面上的智能标书撰写工具在实际应用中仍然存在如下问题。

1、生成的内容不够专业。标书撰写涉及复杂的细分领域，不同行业的招标要求和标书撰写规范存在较大差异。现有工具仅仅能区分服务类、货物类、工程类、其它类四大类别。

2、无法提供决策参考。标书中包含必不可少的报价部分，现有工具几乎不能给出有可靠依据的报价。

2.1.3 RAG 技术研究现状

RAG 技术最早由 Facebook AI Research 提出，随后在任务规划、推理检索、多模态等研究方向进一步发展。

Rewrite-Retrieve-Read 模型利用 LLM 的能力，通过重写模块和 LM 反馈机制来优化检索查询，并通过更新重写模型来提高任务性能^[7]。HyDE 通过关注生

成答案与真实文档之间的嵌入相似性，旨在提高检索的相关性^[8]。2024 年 7 月 2 日，微软开源 GraphRAG^[6]。GraphRAG 利用基于知识图谱的方法，从原文档导出知识图谱，聚类后生成社区摘要。将社区摘要对用户问题的响应汇总，作为对用户的最终响应。这一方法在生成答案的全面性、多样性都相较于传统 RAG 的基线有显著提升。

经过多年发展，RAG 技术的范式已经从朴素 RAG 发展到模块化 RAG，RAG 过程已经被抽象出各种专用子组件，如搜索模块、记忆模块、预测模块、任务适配器模块。^[5]

RAG 技术栈也越加丰富。LangChain 和 LLamaIndex 等大模型框架已经支持 RAG 技术的快速实现。目前也已经出现多个成熟的专用 RAG 框架，如 Haystack、RAGFlow。

第三章 智能标书撰写系统设计

3.1 需求分析

3.2 系统架构设计

系统架构主要分为三个层次：

1. 数据访问层：负责存储和管理企业的历史标书、招标文档、用户参考文段等数据资源。这一层包括方案库（知识图谱）、资历库和素材库（半结构化数据），为企业信息管理和标书内容生成提供数据支持。

2. 业务逻辑层：是系统的核心，负责标书内容的生成和处理。这一层包括 RAG（Retrieval-Augmented Generation）模块、标书内容生成模块（招标理解、目录生成、正文生成）、风格迁移模块（基于 prompt 和基于 LoRA）以及后期处理模块。

3. 表示层：提供用户界面，使用户能够方便地输入招标文件、管理模板、查看生成的标书内容等。

系统包含以下模块

1. 企业信息管理模块：该模块负责存储管理企业的历史标书、方案库（知识图谱）和资历库素材库（半结构化数据），提供可视化管理页面。

2. 标书内容生成模块：该模块是系统的核心，负责根据招标文件的要求生成标书的框架和正文内容，包括招标理解、目录生成和正文生成三个子模块。

3. 风格迁移模块：该模块实现基于 prompt 的风格迁移技术，提供预定义的风格和用户自定义风格。

4. 决策参考模块：该模块提供成本定价模型、盈亏平衡定价和第一价格密封拍卖算法等决策，为用户提供决策参考。

5. 后期处理模块：该模块对生成的标书内容进行校对、格式调整和优化，以文件的格式输出。

3.3 企业信息管理

3.4 方案库（知识图谱）

基于知识图谱的方案库，作为将用户上传的文件解析为可以用于 RAG 的结构化数据的模块，需要实现文件解析和检索支持两方面的功能。我们需要从文件

中提取两部分内容，首先将文本原文分块并向量化存储，向量将作为全局索引用于检索，其次从文本中提取实体和关系的语义信息这两部分内容都将存储在知识图谱中。

文本解析 workflow 如图：

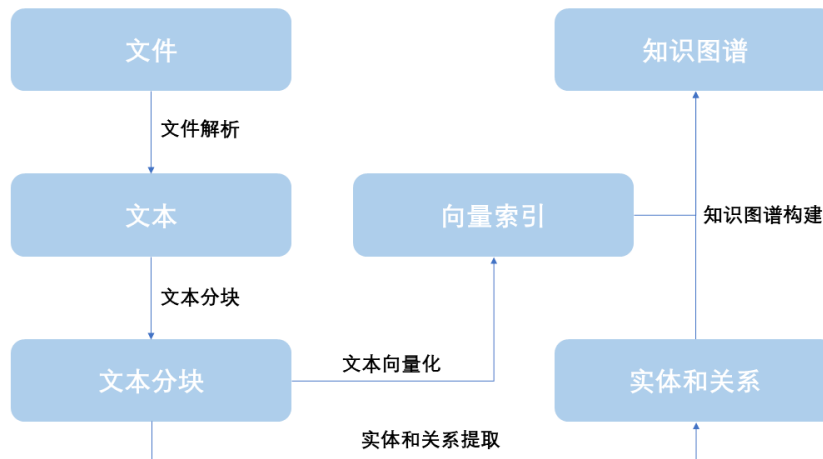


图 3-1: 上传的文件解析为文本后，分块进行实体和关系提取，合并相同实体后存储为知识图谱

文档解析：将用户上传的文件解析为文本。

文本分块：过长的文本会降低文本中早期信息的召回率^[9]，因此需要将文件提取的文本进一步分块。

实体和关系提取：利用 LLM 提取文本块中的实体和关系信息。

文本向量化：将文本块语义信息转化为向量信息，以便于后续检索。

知识图谱构建：将文本块、实体作为节点，关系作为边存储在知识图谱中。实体通过“FROM_CHUNK”关系连接到所属的文本块。文本块节点之间通过“NEXT_CHUNK”关系作为边连接。同一文件中不同文本块提取的实体和关系直接视为同一实体进行合并。

检索 workflow 如图：

语句转化：将用户输入的自然语言语句转化为知识图谱查询语句。

向量检索：从全局索引（即向量信息）中检索找到初始文本块。

图检索：根据初始文本块在相应的节点中查询邻居节点路径。

返回结果：将检索到的图输出。

知识图谱参数定义如下：

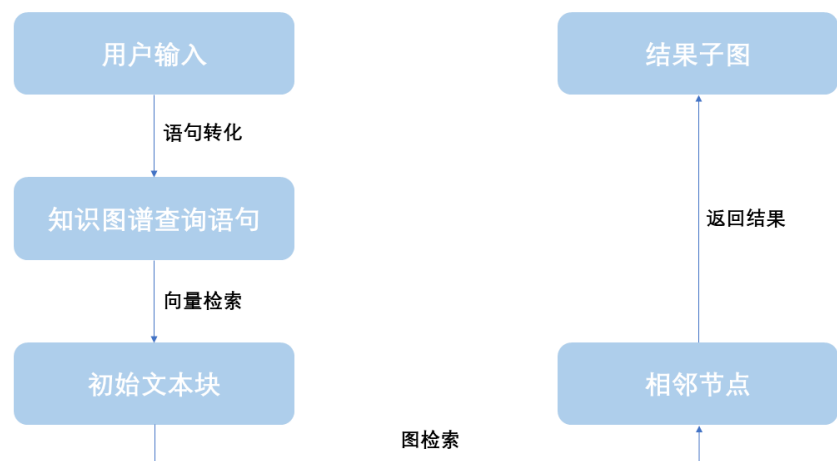


图 3-2: 将用户输入转化为查询语句，先通过向量检索得到初始文本块，再搜寻向量节点形成完整图

表 3-1: 节点定义

参数	参数类型	参数描述	备注
id	str		
label	str		
properties	dict[str,Any]		
embedding_properties	dict[str, list[float]] None		

表 3-2: 关系定义

参数	参数类型	参数描述	备注
start_node_id	str		
end_node_id	str		
type	str		
properties	dict[str,Any]		
embedding_properties	dict[str, list[float]] None		

3.5 资历库

资历库主要存储企业的资质证书、业绩证明、合同等信息。这些信息可以帮助企业在投标过程中展示自身的实力和信誉，提高中标的机会。

3.6 标书内容生成

3.6.1 识别招标需求

调用 `deepseek api`，返回招标需求该模块包括两类功能，一类是由用户上传招标文件，系统自动识别招标需求；另一类是用户手动输入招标需求。对于第一类功能，系统会调用 `deepseek api`，将招标文件中的文本内容进行分析和处理，提取出关键的招标需求信息，并将其存储在系统中。对于第二类功能，用户可以手动输入招标需求，系统会将其保存为结构化数据，以便后续使用。该模块的设计包括以下几个步骤：1. 用户输入：用户上传招标文件或手动输入招标需求。2. 招标需求识别：系统调用 `deepseek api`，对招标文件进行分析和处理，提取出关键的招标需求信息。3. 数据存储：将识别出的招标需求信息存储在系统中，以便后续使用。4. 用户反馈：用户对识别出的招标需求信息进行审核和修改，并提供反馈信息，以便于后续的招标需求识别和处理。该模块的设计旨在提高招标需求识别的准确性和效率，以满足企业投标的需求。通过招标需求识别模块，用户可以快速获取招标文件中的关键信息，从而为后续的标书撰写提供支持。

3.6.2 生成标书目录

根据招标需求和评分标准，生成“章-节-小节”固定格式的目录，每次生成一章该模块主要负责根据招标需求和评分标准，生成“章-节-小节”固定格式的目录。用户可以根据招标文件中的要求，选择生成的目录格式和内容。系统会根据用户的选择，自动生成相应的目录，并将其保存为结构化数据，以便后续使用。该模块的设计包括以下几个步骤：1. 用户输入：用户根据招标文件中的要求，选择生成的目录格式和内容。2. 目录生成：系统根据用户的选择，自动生成相应的目录，并将其保存为结构化数据。3. 数据存储：将生成的目录保存为结构化数据，以便后续使用。4. 用户反馈：用户对生成的目录进行审核和修改，并提供反馈信息，以便于后续的目录生成和处理。该模块的设计旨在提高标书目录生成的准确性和效率，以满足企业投标的需求。通过生成标书目录模块，用户可以快速获取招标文件中的关键信息，从而为后续的标书撰写提供支持。

3.6.3 生成标书正文

根据招标需求和评分标准，生成“章-节-小节”固定格式的正文该模块主要负责根据招标需求和评分标准，生成“章-节-小节”固定格式的正文。用户可以根据招标文件中的要求，选择生成的正文格式和内容。系统会根据用户的选择，

自动生成相应的正文，并将其保存为结构化数据，以便后续使用。该模块的设计包括以下几个步骤：1. **** 用户输入 ****：用户根据招标文件中的要求，选择生成的正文格式和内容。2. **** 正文生成 ****：系统根据用户的选择，自动生成相应的正文，并将其保存为结构化数据。3. **** 数据存储 ****：将生成的正文保存为结构化数据，以便后续使用。4. **** 用户反馈 ****：用户对生成的正文进行审核和修改，并提供反馈信息，以便于后续的正文生成和处理。该模块的设计旨在提高标书正文生成的准确性和效率，以满足企业投标的需求。通过生成标书正文模块，用户可以快速获取招标文件中的关键信息，从而为后续的标书撰写提供支持。在生成正文时，能够根据知识图谱中的信息，自动生成相关的内容，并将其嵌入到正文中。这样可以提高标书正文的专业性和准确性，使其更符合招标方的要求。

3.7 风格迁移

风格迁移模块主要负责根据用户的需求和招标方的要求，对生成的标书内容进行风格迁移。该模块支持两种风格迁移技术：基于 **prompt** 的风格迁移和基于 **LoRA** 的风格迁移。基于 **prompt** 的风格迁移：该方法通过在输入文本中添加特定的提示信息，引导生成模型生成符合目标风格的文本。用户可以根据招标方的要求，提供一些示例文本或描述性提示，以指导生成模型进行风格迁移。基于 **LoRA** 的风格迁移：该方法通过对预训练模型进行微调，使其能够生成符合目标风格的文本。用户可以根据招标方的要求，提供一些示例文本，以指导模型进行微调。**LoRA (Low-Rank Adaptation)** 是一种高效的模型微调方法，通过在预训练模型中添加低秩矩阵来实现参数的高效调整，从而实现风格迁移。该模块的设计包括以下几个步骤：1. **用户输入**：用户根据招标方的要求，提供一些示例文本或描述性提示，以指导生成模型进行风格迁移。2. **风格迁移**：根据用户输入的信息，选择合适的风格迁移方法（基于 **prompt** 或基于 **LoRA**），对生成的标书内容进行风格迁移。3. **输出结果**：将经过风格迁移的标书内容输出，供用户进行审核和修改。4. **用户反馈**：用户对生成的标书内容进行审核和修改，并提供反馈信息，以便于后续的风格迁移和模型微调。该模块的设计旨在提高标书内容的个性化和专业化程度，以满足不同招标方的要求。通过风格迁移模块，用户可以根据具体项目需求定制标书内容的风格、格式等，从而提高中标率和竞争力。

3.8 决策参考

该模块基于大模型调用工具技术，结合招标文件中的信息和历史数据，为用户提供决策参考。主要包括以下几个功能：1. **成本定价模型**：根据招标文件中的

要求和历史数据，自动生成成本定价模型，帮助用户进行投标定价。2. 盈亏平衡定价：根据招标文件中的要求和历史数据，自动生成盈亏平衡定价模型，帮助用户进行投标定价。

3.9 后期处理

qwe

第四章 系统实现与实验

4.1 系统开发环境与工具

4.1.1 系统环境

- 操作系统: Ubuntu 22.04
- Python 版本: 3.8
- 依赖库: deepseek、fastapi、uvicorn、pydantic、pymongo、neo4j

4.1.2 开发工具

- IDE: VSCode
- 版本控制: Git
- 文档编写: Markdown

4.1.3 系统框架

- 前端框架: Vue.js
- 后端框架: FastAPI
- 数据存储: MongoDB、neo4j

4.2 系统功能实现

4.2.1 方案库 (知识图谱)

使用 neo4j 存储知识图谱。neo4j 是一个图数据库, 使用 `cipher` 语句进行数据存储和查询。

后端使用 `neo4j-graphrag` 库和 `neo4j` 库进行知识图谱操作。`neo4j` 库用于连接到 neo4j 进行基本数据操作。`neo4j-graphrag` 库是 neo4j 提供的知识图谱框架库, 用于实现文件解析和检索支持。文件解析过程中, 文本向量化使用阿里百炼 `text-embedding-v3` 模型, 向量维度为 1024 维, 实体和关系提取使用 `deepseek-chat` 模型。

前端使用阿里 `g6` 图可视化引擎。该引擎提供了图的绘制、布局、分析、交互、动画等基础的图可视化能力。

后端 api 定义:

- POST /api/upload:

测试样例: (待办) ”The son of Duke Leto Atreides and the Lady Jessica, Paul is the heir of House ” ”Atreides, an aristocratic family that rules the planet Caladan.”

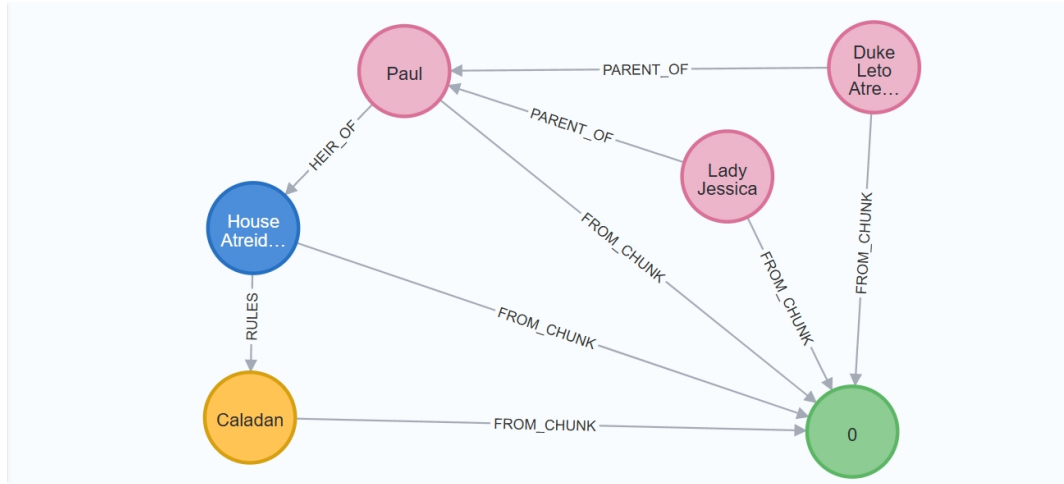


图 4-1: 文段生成的知识图谱, 0 节点为文本块

4.2.2 招标需求识别模块

定义招标需求的结构化数据模型, 使用 deepseek api 进行招标需求识别。定义 Chapter、Section、Subsection 数据模型, 使用 pydantic 进行数据验证。利用 langchain 结构化输出模型, 将大模型输出转化为结构化数据。后端 api 使用 fastapi 实现, 前端使用 axios 进行数据交互。api 定义:

- POST /api/identify: 识别招标需求
- GET /api/chapters: 获取章节列表
- GET /api/sections: 获取节列表
- GET /api/subsections: 获取小节列表

4.2.3 标书目录生成模块

使用 deepseek-chat 模型生成目录。

- 目录生成: 根据招标需求和评分标准, 生成 “章-节-小节” 固定格式的目录。
- 目录保存: 将生成的目录保存为结构化数据, 以便后续使用。
- 目录审核: 用户对生成的目录进行审核和修改, 并提供反馈信息。

4.2.4 标书正文生成模块

使用方案库后端作检索增强，使用 deepseek-chat 模型生成正文。

- 目录生成：根据招标需求和评分标准，生成“章-节-小节”固定格式的目录。
- 目录保存：将生成的目录保存为结构化数据，以便后续使用。
- 目录审核：用户对生成的目录进行审核和修改，并提供反馈信息。

4.2.5 风格迁移模块

在生成正文时，根据用户提供的示例文本或描述性提示，选择合适的风格迁移方法（基于 prompt 或基于 LoRA），对生成的标书内容进行风格迁移。api 定义：

- POST /api/style-transfer: 风格迁移
- GET /api/styles: 获取可用的风格列表

4.2.6 决策参考模块

在标书生成报价部分，使用成本定价模型、盈亏平衡定价和第一价格密封拍卖算法等决策参考功能，辅助用户进行投标决策。

4.2.7 后期处理模块

对生成的标书内容进行校对、格式调整和优化，确保标书的准确性和专业性。

4.2.8 模板管理模块

支持用户自定义和保存标书模板，以便于未来的标书撰写。

4.3 系统测试

4.3.1 测试方法

4.3.2 测试结果

第五章 总结与展望

未来优化点：使用 graphRAG 的全局-社区-摘要模式尝试多种文本块节点的连接和存储方式知识图谱融合

参考文献

- [1] 芮丽梅, 石国虎. 新形势下招标投标现状与发展趋势 [J]. 招标采购管理, 2021, (07): 11-12.
- [2] 智研咨询. 2025-2031 年中国招标采购行业发展模式分析及未来前景规划报告 [EB/OL]. <https://www.chyxx.com/research/202110/980866.html>, 2024.
- [3] 中研普华产业研究院. 2025-2030 年招投标市场深度分析及发展趋势研究咨询报告 [EB/OL]. <https://www.chinairn.com/news/20250220/103457473.shtml>, 2024.
- [4] 程建宁. AI 技术在招标投标活动中的应用及展望 [J]. 招标采购管理, 2023, (07): 61-63.
- [5] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, Haofen Wang. Retrieval-Augmented Generation for Large Language Models: A Survey [J/OL]. arXiv:2312.10997 [cs.CL]
- [6] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, Jonathan Larson. From Local to Global: A Graph RAG Approach to Query-Focused Summarization [J/OL]. arXiv:2404.16130 [cs.CL]
- [7] Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, Nan Duan. Query Rewriting for Retrieval-Augmented Large Language Models [J/OL]. arXiv:2305.14283 [cs.CL]
- [8] Luyu Gao, Xueguang Ma, Jimmy Lin, Jamie Callan. Precise Zero-Shot Dense Retrieval without Relevance Labels [J/OL]. arXiv:2212.10496 [cs.IR]
- [9] Yuri Kuratov, Aydar Bulatov, Petr Anokhin, Dmitry Sorokin, Artyom Sorokin, Mikhail Burtsev. In Search of Needles in a 11M Haystack: Recurrent Memory Finds What LLMs Miss [J/OL]. arXiv:2402.10790 [cs.CL]

附 录

致 谢

暂时不写