

****

学术翻译&

多语言摘要助手v1.0

学术翻译 & 多语言摘要助手 PRD

# 1. 背景与目标

## 1.1 背景（Problem Statement）

在科研和学术学习中，学生和研究人员需要快速理解大量英文或中文论文内容，提取关键信息和术语。然而，人工阅读耗时长、效率低，同时跨语言阅读也增加理解难度。因此，需要一个轻量化的工具，能够快速生成摘要和关键术语，并支持多语言翻译。

## 1.2 产品定位

一款基于大模型 API 的学术辅助工具，提供 PDF 文档上传、自动摘要生成、多语言翻译及关键术语提取功能，面向高校学生、科研人员和科研助理，帮助用户快速获取论文核心信息。

## 1.3 目标（v1.0，1 周可交付）

* 用户可上传 PDF（可选文本）文档
* 系统自动提取正文内容
* 用户可选择中文或英文输出
* 输出包含：摘要 + 关键术语（中英文）
* 支持一键导出 Word 文档

## 1.4 范围（In / Out）

**In Scope:**

* PDF 上传及文本提取（可选文本 PDF）
* 中文/英文摘要生成
* 关键术语提取
* Word 文件导出
* 简单前端 UI（Streamlit）

**Out of Scope:**

* OCR 扫描 PDF
* 图片、公式或图表解析
* 多用户管理、云存储
* 高级排版或论文引用管理

## 1.5 关键假设与限制

* 用户上传 PDF 为可选文本文件
* 系统依赖 DashScope（通义千问）API 稳定性
* v1.0 仅单文件处理，不支持批量
* 文件大小及页数有限制（建议 <50MB，前 5 页为重点内容）

# 2. 用户需求（用户故事）

## 2.1 用户画像

* **高校学生**：需要快速理解课程相关论文
* **科研人员**：需要高效整理文献，提取术语和摘要
* **科研助理**：需辅助导师进行文献梳理与翻译

## 2.2 用户故事

1. 作为学生，我希望上传论文 PDF 后快速获得摘要，以节省阅读时间。
2. 作为科研人员，我希望输出中英文摘要和关键术语，以便整理文献笔记。
3. 作为科研助理，我希望一键导出 Word 文件，方便分享给团队。

## 2.3 用户需求优先级（MoSCoW）

* **Must have:** PDF 上传、文本提取、摘要生成、关键术语提取、Word 导出
* **Should have:** 多语言选择（中文/英文）、前端 UI 简洁显示结果
* **Could have:** 结果复制/分享功能
* **Won’t have:** 批量处理、OCR 扫描 PDF、云端账号管理

## 2.4 非功能性需求（v1.0）

* 响应时间 ≤ 20 秒（处理前 5 页 PDF）
* 输出结果格式清晰，兼容 Word 2007+
* 系统易于部署在本地或轻量服务器

# 3. 产品功能设计

## 3.1 功能概览

|  |  |
| --- | --- |
| 功能模块 | 描述 |
| 文件上传 | 用户上传 PDF 文档 |
| 内容解析 | 提取正文文本 |
| 总结生成 | 调用 DashScope API 生成摘要和术语 |
| 结果展示 | 页面显示摘要及术语，提供 Word 下载 |

## 3.2 核心功能模块

### **3.2.1 文件上传模块**

* 支持 PDF 文件类型
* 文件大小限制提示
* 显示文件名及页数

### **3.2.2 内容解析模块**

* 使用 pdfplumber 提取可选文本
* 支持前 5 页文本提取
* 提取文本长度显示

### **3.2.3 总结生成模块**

* 用户选择输出语言（中文/英文）
* 调用 DashScope API 生成摘要与关键术语
* 控制返回文本长度（max\_output\_tokens）

### **3.2.4 结果展示模块**

* 页面显示生成摘要与关键术语
* 支持一键导出 Word 文件
* 提示处理状态和错误信息

## 3.3 非功能性需求

### **3.3.1 性能要求**

* 单文档处理时间 ≤ 20 秒
* 页面加载时间 ≤ 3 秒

### **3.3.2 易用性要求**

* 简洁明了的上传按钮和语言选择
* 摘要和术语清晰展示

### **3.3.3 安全性要求**

* 本地运行，PDF 不上传至云端
* API Key 安全配置，不硬编码

### **3.3.4 兼容性要求**

* 浏览器支持 Chrome、Edge、Firefox
* Word 文件兼容 2007+

# 4. 系统架构设计

### **4.1 架构设计思路**

轻量级 Web 应用，前端使用 Streamlit，后端调用 DashScope API，实现快速交互和摘要生成。

### **4.2 系统总体架构**

用户浏览器

│

▼

Streamlit 前端

│

▼

PDF 文本解析模块(pdfplumber)

│

▼

DashScope API 调用（摘要生成）

│

▼

Word 导出 & 页面显示结果

### **4.3 技术选型**

* **前端**：Streamlit
* **文本解析**：pdfplumber
* **AI 接口**：DashScope (通义千问)
* **文件导出**：python-docx
* **开发语言**：Python 3.10+

### **4.4 模块交互与流程**

1. 用户上传 PDF
2. 系统解析 PDF 文本
3. 用户选择语言并点击生成
4. 调用 DashScope API，生成摘要与术语
5. 页面显示结果并可导出 Word

### **4.5 稳定性与扩展性设计**

* 支持单文件处理，避免多任务阻塞
* 后续可扩展 OCR 支持扫描 PDF
* 可增加批量处理、云存储功能

# 5. 迭代与规划

### **5.1 短期迭代（v1.1 ~ v1.2）**

* 增加批量 PDF 上传功能
* 支持 OCR 扫描 PDF
* 优化前端 UI 展示

### **5.2 中期迭代（v2.0）**

* 多用户管理及云端存储
* 提供 API 接口给第三方系统调用
* 支持论文公式和图表识别

### **5.3 长期发展愿景**

* 打造完整学术助手平台
* 支持多语言文献、学术笔记自动整理
* AI 驱动的文献推荐和学术写作辅助