# 📊 AI事实核查系统 - 技术汇报文档

## 📋 项目概述

本项目是团队自主研发的AI驱动事实核查系统，专门用于检测和验证文本内容中的事实声明。系统已成功部署上线，具备完整的API服务能力，可为组织内部的内容审核、信息验证等业务场景提供技术支撑。

**项目价值**： - 🎯 **功能定位**：提供文本内容的事实声明提取和验证服务 - 📈 **应用场景**：适用于内容审核、信息验证等业务需求 - 🚀 **技术特点**：基于AI模型实现声明提取、搜索和验证的完整流程

## 🏗️ 系统架构

### 技术选型

| 组件 | 技术方案 | 选择理由 |
| --- | --- | --- |
| **AI模型** | Anthropic Claude | 业界领先的文本理解能力，支持中英文 |
| **搜索引擎** | Exa.ai | 专业的AI搜索，结果质量高 |
| **后端框架** | Node.js + Express | 开发效率高，部署便捷 |
| **部署方案** | Vercel云平台 | 零配置部署，自动扩容 |

### 核心流程

📝 输入文本 → 🔍 AI提取声明 → 🌐 搜索验证源 → 🧠 AI交叉验证 → 📊 输出结果

**处理流程**：支持文本内容的声明提取和验证，单个声明验证时间约2-5秒

## ⚡ 系统功能

### 1. 声明智能提取

**能力描述**：从文本中自动识别需要验证的事实性声明 - ✅ **智能识别**：区分事实陈述与主观观点 - ✅ **多语言处理**：同时支持中文、英文内容 - ✅ **批量处理**：单次可提取数十个声明

### 2. 信息源智能搜索

**能力描述**：为每个声明搜索相关的权威信息源 - ✅ **专业搜索**：使用AI搜索引擎获取高质量结果 - ✅ **实时获取**：抓取最新网络信息 - ✅ **质量筛选**：自动过滤低质量信息源

### 3. 交叉验证分析

**能力描述**：对比声明与信息源，给出验证结论 - ✅ **深度分析**：综合多个信息源进行判断 - ✅ **置信度评分**：提供量化的可信度指标 - ✅ **错误修正**：对虚假信息提供正确版本

### 4. 可视化报告

**能力描述**：生成易读的验证报告 - ✅ **结果汇总**：清晰展示验证结果 - ✅ **详细分析**：提供验证依据和推理过程 - ✅ **批量导出**：支持多种格式的结果输出

## 🌟 技术特点

### 1. 多AI模型分工处理

**实现方式**：使用不同AI模型处理不同任务 - 💡 **声明提取**：使用Claude Haiku模型进行文本分析和声明识别 - 💡 **事实验证**：使用Claude Sonnet模型进行深度推理和验证 - 💡 **任务优化**：根据不同处理阶段选择合适的模型

### 2. 错误处理机制

**实现方式**：针对AI服务的不稳定性设计容错方案 - 💡 **格式修复**：自动处理AI返回的JSON格式异常问题 - 💡 **异常捕获**：完善的try-catch错误处理流程 - 💡 **降级方案**：API调用失败时的备用处理逻辑

### 3. 中英文内容处理

**实现方式**：针对中英文内容的差异化处理  
- 💡 **语言检测**：根据文本特征自动识别语言类型 - 💡 **搜索优化**：中英文使用不同的搜索关键词策略 - 💡 **本地化**：保持输出结果与输入语言的一致性

## 🎯 开发中的技术挑战

### 1. AI模型响应一致性

**挑战**：AI模型偶尔返回不符合预期格式的响应 **解决方案**：使用Zod Schema进行数据验证，并实现自动格式修复逻辑处理异常情况

### 2. 搜索结果相关性

**挑战**：网络搜索结果质量不稳定，需要筛选有效信息 **解决方案**：根据中英文特点构建不同的搜索查询，并对结果进行基础过滤

### 3. 多语言内容处理

**挑战**：中英文内容的理解和处理方式存在差异 **解决方案**：在搜索和验证阶段分别针对不同语言调整处理策略

### 4. API调用频率控制

**挑战**：外部AI API存在调用频率限制 **解决方案**：实现基础的请求限流和错误重试机制

## 📊 项目现状

### 系统指标

| 功能模块 | 实现状态 | 性能表现 |
| --- | --- | --- |
| **声明提取** | ✅ 已完成 | 单次处理约2-3秒 |
| **信息搜索** | ✅ 已完成 | 返回10条相关结果 |
| **事实验证** | ✅ 已完成 | 综合分析约3-5秒 |
| **批量处理** | ✅ 已完成 | 支持最多10个声明 |

### 技术成果

* ✅ **完整API服务**：4个核心API端点已部署上线
* ✅ **多语言支持**：支持中文和英文内容处理
* ✅ **错误处理**：具备基本的异常处理和重试机制
* ✅ **客户端工具**：提供JavaScript SDK便于集成

## 🚀 快速试用指南

### 在线体验

**API地址**：https://hallubacken-5t6wwxo9o-fraps-projects.vercel.app

### 组员试用步骤

# 方式1：直接API调用测试  
curl -X POST https://hallubacken-5t6wwxo9o-fraps-projects.vercel.app/api/claims/extract \  
 -H "Content-Type: application/json" \  
 -d '{"content": "测试文本内容"}'  
  
# 方式2：使用测试脚本  
npm run test:basic  
  
# 方式3：本地部署（完整功能）  
npm install && npm start

### 集成示例

团队提供了完整的客户端SDK，支持快速集成到现有项目中。

## 🔮 后续改进方向

### 功能优化

1. **性能提升**：优化API调用逻辑，减少响应时间
2. **错误处理**：完善异常情况的处理和用户提示
3. **结果展示**：改进验证结果的展示格式和详细程度

### 功能扩展

1. **批量处理**：支持更大规模的文档批量处理
2. **缓存机制**：添加结果缓存减少重复计算
3. **用户界面**：考虑开发简单的Web界面便于使用

### 技术改进

1. **代码优化**：重构部分代码提高可维护性
2. **测试完善**：增加更多边界情况的测试用例
3. **文档补充**：完善API文档和使用说明

## 📋 总结

本项目实现了基于AI的事实核查系统，具备声明提取、信息搜索、事实验证等核心功能。系统采用Node.js构建，集成Anthropic Claude和Exa.ai服务，已成功部署并可正常使用。

**项目特点**： - ✅ 功能完整，包含完整的事实核查流程 - ✅ 技术实用，基于成熟的AI服务和框架  
- ✅ 部署简便，支持一键部署到云平台 - ✅ 易于集成，提供标准REST API接口

*文档最后更新：2025年7月 | 项目状态：已完成并上线*