

# AI-Powered CV Enhancer & ATS Optimizer

Thiruvarankan M

October 2025

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	What This Project Does . . . . .	2
1.2	The Problem It Solves . . . . .	2
<b>2</b>	<b>How It Works - For Non-Technical Users</b>	<b>2</b>
2.1	The Simple Process . . . . .	2
2.2	Example Transformation . . . . .	3
<b>3</b>	<b>Technical Explanation</b>	<b>4</b>
3.1	Technology Stack . . . . .	4
3.2	Why CrewAI Makes This Easy . . . . .	4
3.3	Architecture Overview . . . . .	4
3.4	Code Structure . . . . .	5
3.5	How CrewAI Simplifies Development . . . . .	6
3.6	Key Implementation Details . . . . .	6
<b>4</b>	<b>Why GROQ AI?</b>	<b>7</b>
<b>5</b>	<b>Project Benefits</b>	<b>7</b>
5.1	For Users . . . . .	7
5.2	For Developers . . . . .	7
<b>6</b>	<b>Future Enhancements</b>	<b>7</b>
<b>7</b>	<b>Conclusion</b>	<b>8</b>

# 1 Introduction

## 1.1 What This Project Does

Imagine having a professional career coach who reads your CV and gives you specific advice like:

- "Your React.js e-commerce project should be listed first because it shows full-stack skills"
- "Change 'improved system performance' to 'improved system performance by 40%, reducing load time from 3s to 2.3s'"
- "Add these keywords: Docker, Kubernetes, microservices"

That's exactly what this AI tool does! It's like having an expert who knows what companies look for and helps you present your experience in the best possible way.

## 1.2 The Problem It Solves

### Common CV Problems

1. **Generic feedback:** "Make your CV better" (not helpful!)
2. **Manual work:** Spending hours guessing what to improve
3. **ATS rejection:** Your CV gets rejected by automated systems before humans even see it
4. **Wrong prioritization:** Not knowing which projects to highlight first

# 2 How It Works - For Non-Technical Users

## 2.1 The Simple Process

Think of it as a conversation between AI experts about your CV:

### 1. Step 1: Upload Your CV

- You upload your CV (PDF or Word document)
- Select your target industry (technology, finance, healthcare, etc.)
- Optionally paste the job description

### 2. Step 2: AI Analyst Reviews

- First AI "expert" reads your entire CV carefully
- Identifies your strongest achievements
- Spots sections that need improvement
- Decides which projects should be prioritized

### 3. Step 3: AI Writer Enhances

- Second AI "expert" takes the analysis
- Rewrites weak sections with powerful language
- Adds specific numbers and metrics
- Suggests keywords for your industry

#### 4. Step 4: Get Your Report

- Receive detailed feedback with specific examples
- See before/after comparisons
- Get actionable recommendations

## 2.2 Example Transformation

### BEFORE (Weak)

*"Worked on web development projects and improved system performance."*

### AFTER (Strong)

*"Led development of e-commerce platform serving 10,000+ users, improved system performance by 40%, reducing page load time from 3.2s to 1.9s using React.js and Node.js optimization techniques."*

## 3 Technical Explanation

### 3.1 Technology Stack

#### Core Technologies

- **CrewAI:** Multi-agent AI framework for orchestrating AI collaboration
- **GROQ AI:** Lightning-fast inference with Llama 3.1-8b model
- **Streamlit:** Web interface for user interaction
- **Python:** Core programming language
- **PDFPlumber & python-docx:** Document parsing libraries

### 3.2 Why CrewAI Makes This Easy

**CrewAI** is a framework that allows multiple AI agents to work together like a professional team. Here's why it's perfect for this project:

#### 1. Role-Based Agents

- Define specific roles: "CV Analyst" and "CV Writer"
- Each agent has expertise in their domain
- No need to build complex AI logic from scratch

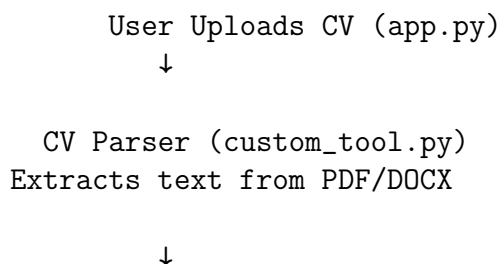
#### 2. Sequential Processing

- Analyst reviews first, writer enhances second
- Automatic data passing between agents
- Maintains context throughout the process

#### 3. Simple Configuration

- Define agents in YAML files (human-readable)
- Set goals and expected outputs
- CrewAI handles the orchestration

### 3.3 Architecture Overview



## CrewAI Multi-Agent System

Agent 1: CV Analyst

- Analyzes content
- Identifies strengths
- Prioritizes projects

↓

Agent 2: CV Writer

- Enhances weak sections
- Adds metrics & keywords
- Provides recommendations

↓

Enhanced CV Report

- Specific feedback
- Before/After examples
- ATS optimization

### 3.4 Code Structure

*# Agent Definition (agents.yaml)*

```
cv_analyst:
  role: CV Analyst
  goal: Analyze CV and identify improvements
  backstory: Expert CV analyst providing
           specific feedback

cv_writer:
  role: CV Enhancement Writer
  goal: Transform weak sections into strong content
  backstory: Professional CV writer creating
           impactful achievements
```

*# Task Definition (tasks.yaml)*

```
analyze_task:
  description: Analyze CV content, identify
               strengths, weaknesses, and priority
  expected_output: Analysis report with specific
                  feedback
  agent: cv_analyst

enhance_task:
  description: Enhance weak sections with
               quantified achievements
  expected_output: Enhanced recommendations
  agent: cv_writer
```

## 3.5 How CrewAI Simplifies Development

### Without CrewAI (Complex)

- Manual prompt engineering for each step
- Complex state management between AI calls
- Custom orchestration logic
- Error handling for each AI interaction
- 500+ lines of complex code

### With CrewAI (Simple)

- Define agents in YAML (20 lines)
- Define tasks in YAML (20 lines)
- CrewAI handles orchestration automatically
- Built-in error handling and retry logic
- **Total: 220 lines of clean code**

## 3.6 Key Implementation Details

### 1. Document Parsing

```
class CVParseTool(BaseTool):
    def _run(self, cv_file_path: str) -> str:
        if ext == '.pdf':
            # Extract text from PDF
        elif ext == '.docx':
            # Extract text from Word
        return extracted_text
```

### 2. Agent Creation

```
@agent
def cv_analyst(self) -> Agent:
    return Agent(
        config=self.agents_config['cv_analyst'],
        verbose=True
    )
```

### 3. Task Execution

```
result = CvEnhancer().crew().kickoff(inputs={
    'parsed_cv': cv_text,
    'job_description': job_desc,
    'target_industry': industry
})
```

## 4 Why GROQ AI?

- **Speed:** 10x faster than traditional AI APIs
- **Free:** No cost for API usage during development
- **Quality:** Llama 3.1-8b provides excellent results
- **Simplicity:** Drop-in replacement for other AI providers

## 5 Project Benefits

### 5.1 For Users

- Save hours of manual CV editing
- Get specific, actionable feedback
- Improve ATS compatibility
- Stand out to recruiters

### 5.2 For Developers

- Ultra-minimal codebase (220 lines)
- Easy to understand and modify
- No complex AI orchestration needed
- Fast and free inference

## 6 Future Enhancements

1. Real-time job description matching
2. Cover letter generation
3. LinkedIn profile optimization
4. Interview preparation tips
5. Multi-language support

## 7 Conclusion

This project demonstrates how modern AI frameworks like **CrewAI** make complex tasks simple. What would traditionally require hundreds of lines of intricate code and manual AI orchestration is reduced to just 220 lines of clean, maintainable code.

The key insight: **Let AI agents collaborate like human experts**, with each agent focusing on their specialized task. CrewAI handles the complexity, allowing us to focus on delivering value to users.

### Key Takeaway

**Minimal code + Maximum impact = Modern AI development**