

Comprehensive Report: Resume Processing System with CrewAI

May 18, 2025

1 Overview of CrewAI

CrewAI is a framework designed to facilitate the creation and management of multi-agent systems for automating complex workflows. It allows developers to define agents with specific roles, tasks, and capabilities, enabling collaborative task execution. CrewAI is particularly useful for tasks requiring orchestration, such as resume processing, data analysis, or content generation.

1.1 Agent Types in CrewAI

CrewAI supports various agent types, primarily distinguished by their roles and capabilities:

- **Task-Oriented Agents:** These agents are designed to perform specific tasks, such as data extraction, analysis, or content generation. They are typically equipped with tools and prompts tailored to their objectives.
- **Manager Agents:** These agents oversee the workflow, coordinating between other agents and ensuring tasks are executed in the correct order.
- **Custom Agents:** Developers can create custom agents with specialized behaviors by defining their roles, goals, and tools.

1.2 Agent Features in CrewAI

- **Role and Goal Definition:** Each agent can be assigned a role (e.g., "Resume Advisor") and a goal (e.g., "Provide feedback on a resume").
- **Task Assignment:** Agents can be assigned specific tasks, which include instructions and expected outputs.
- **Tool Integration:** Agents can use external tools (e.g., file readers, web scrapers) to enhance their capabilities.
- **Collaboration:** Agents can share information and collaborate on tasks, enabling sequential or parallel workflows.

- **Output Formatting:** Agents can format their outputs in specific formats, such as Markdown, for easy integration into reports or displays.

1.3 Agent Types and Features Used in the Code

In the provided code, three task-oriented agents are defined:

1. **Professional Resume Advisor:** Used to analyze the resume and provide feedback (e.g., rating and improvement suggestions).
 - **Features Used:** Role and goal definition, task assignment, output formatting (Markdown).
2. **Professional Resume Writer:** Used to rewrite and enhance the resume based on the feedback.
 - **Features Used:** Role and goal definition, task assignment, collaboration (using the output from the Resume Advisor), output formatting (Markdown).
3. **Job Interview Coach:** Used to provide interview preparation tips tailored to the job and location.
 - **Features Used:** Role and goal definition, task assignment, collaboration (using the enhanced resume), output formatting (Markdown).

No manager agents or custom tools were explicitly defined, but the agents collaborate sequentially through the **Crew** workflow.

2 Main Functionalities of the Code

The code implements a resume processing system that automates resume analysis, enhancement, and interview preparation. It is structured as a Jupyter Notebook (**Resume_Agent_V2_R.ipynb**) and follows four main steps, as outlined in the introduction:

1. Setup and Library Installation
2. Resume Loading Functions
3. Agent and Task Definition with CrewAI
4. Workflow Execution and Output Display

Below is a detailed explanation of each part.

2.1 Setup and Library Installation

This section ensures the necessary libraries are installed and imported for the system to function.

- **Warnings Suppression:** The code begins by suppressing warnings to keep the output clean:

```
import warnings
warnings.filterwarnings('ignore')
```

- **Library Installation:** The following libraries are installed:
 - **PyMuPDF:** For reading PDF files (used to load resumes).
 - **crewai** and **crewai-tools:** The core CrewAI framework and additional tools for agent functionality.
 - **mammoth:** For converting .docx files to HTML (used in resume loading).
 - **IPython.display:** For rendering Markdown outputs in the notebook.

```
%pip install PyMuPDF
%pip install crewai crewai-tools
%pip install mammoth
```

- **Dependencies Overview:** The output of the `pip install` commands shows a comprehensive list of dependencies, including `openai`, `pydantic`, `chromadb`, and others, which are required for CrewAI's language models, data validation, and vector storage.

Purpose: This step ensures the environment is ready for resume processing by installing tools for file handling, agent orchestration, and output rendering.

2.2 Resume Loading Functions

This section defines functions to load resumes in PDF or DOCX format. The code snippet for this part is not fully shown in the provided document (it appears to be truncated), but based on the context and typical CrewAI workflows, these functions likely:

- Use `PyMuPDF` to extract text from PDF resumes.
- Use `mammoth` to convert DOCX resumes to HTML and extract text.
- Store the extracted text in a variable (e.g., `res2`) for further processing.

Purpose: The resume loading functions provide the raw input data (the resume text) that will be analyzed and enhanced by the CrewAI agents.

2.3 Agent and Task Definition with CrewAI

This section defines the agents and their tasks using CrewAI. Although the exact definitions are not shown in the provided document (likely in an earlier cell), the output and context indicate the following setup:

- **Agents Defined:**
 - **Professional Resume Advisor:** Tasked with analyzing the resume and providing feedback (e.g., a score out of 10 and improvement suggestions).
 - **Professional Resume Writer:** Tasked with rewriting the resume based on the feedback, optimizing it for the target job.
 - **Job Interview Coach:** Tasked with providing interview preparation tips tailored to the job title ("Financial Manager") and location ("New York").
- **Tasks Defined:**

- **Resume Feedback Task:** Assigned to the Resume Advisor, outputs a Markdown-formatted analysis (stored in `resume_feedback_task`).
- **Resume Enhancement Task:** Assigned to the Resume Writer, outputs an improved resume in Markdown format (stored in `resume_advisor_task`).
- **Interview Preparation Task:** Assigned to the Job Interview Coach, outputs tailored interview tips in Markdown format (stored in `interview_preparation_task`).

Purpose: This step sets up the multi-agent system, defining the roles, goals, and tasks for each agent to process the resume collaboratively.

2.4 Workflow Execution and Output Display

This section executes the workflow using CrewAI's `Crew` class and displays the results.

- **Crew Execution:**

```
result = crew.kickoff(inputs={"resume": res2, "location": 'New York', "job_title": 'Financial Manager'})
```

- The `Crew` object (`crew`) orchestrates the workflow, passing the resume (`res2`), job title, and location as inputs.
- The agents execute their tasks sequentially:
 1. **Resume Advisor:** Analyzes the resume and provides feedback.
 2. **Resume Writer:** Enhances the resume based on the feedback.
 3. **Job Interview Coach:** Generates interview preparation tips using the enhanced resume.
- The output logs show the status of each task (e.g., "Completed") and the final results from each agent.
- **Output Display:** The results are displayed using `IPython.display.Markdown` to render the Markdown-formatted outputs in the notebook.
 - **Resume Feedback:**

```
6/10

Here's a breakdown of the resume with suggestions for improvement:
* Contact Information: While your phone number and email address are provided, consider adding your LinkedIn profile URL for a more comprehensive online presence.
* Summary: The summary is a good start, highlighting your expertise. However, it could be more impactful by quantifying your achievements...
```

- * Rates the resume 6/10 and provides actionable feedback on sections like Contact Information, Summary, Experience, Skills, Education, and Missing Sections.

– **Enhanced Resume:**

```
+1(970) 888-555-999 | alice.robinson@gmail.us | linkedin.  
com/in/alicerobinson
```

```
Alice Robinson
```

```
Skilled Financial Manager with a proven track record of  
optimizing financial processes, driving profitability,  
and delivering exceptional client service...
```

- * Presents an improved resume with added LinkedIn URL, quantified achievements (e.g., "15% increase in revenue"), and new sections like Awards & Honors.

– **Interview Preparation Tips:**

```
Here are some tailored interview preparation tips for a  
Financial Manager position in New York, based on the  
provided resume:
```

```
**I. Common Interview Questions & Strategies:**
```

```
* "Tell me about your experience in financial planning  
and analysis."
```

```
* Quantify: Don't just list tasks. Highlight specific  
projects...
```

- * Provides detailed tips on answering common interview questions, demonstrating relevant skills, and making a strong impression, tailored to the Financial Manager role in New York.

Purpose: This step executes the workflow, collects the outputs from each agent, and renders them in a readable format for the user.

3 Conclusion

The code implements a resume processing system using CrewAI, leveraging three task-oriented agents to analyze, enhance, and prepare for a job application. CrewAI's features, such as role definition, task assignment, and output formatting, are effectively utilized to create a collaborative workflow. The system successfully:

- Analyzes a resume and provides actionable feedback.
- Enhances the resume with quantifiable achievements and additional sections.
- Offers tailored interview preparation tips for a Financial Manager position in New York.

Future improvements could include adding a fourth agent to customize the resume further, generating a cover letter, and enhancing the resume loading functions to handle more file formats.