Jobby: AI-Powered Job Hunting Assistant

Mohamed Sonbol[1], Omar Ismail[2], Abdelrahman Elkhayat[3], and Umutcan Dogan[4]

[1]Department of Computer Science

July 30, 2025

Abstract

The process of finding a job today is more time-consuming than ever. Applicants often need to submit a large number of applications, and it doesn't help that each company or employer typically asks for a motivation or cover letter. Many also use Applicant Tracking Systems (ATS) that automatically filter CVs based on keywords. Additionally, different platforms list different job postings, making the search even more fragmented. We developed this project to address these problems and to try to streamline the job application process.

The system is composed of several integrated modules. The frontend is built using Streamlit, providing an interactive interface. On the backend, web scraping for jobs is handled using Selenium, while document processing modules allow users to upload and parse their CVs. The AI integration is powered by Gemini 2.0 Flash.

Overall, the system performs adequately. It successfully scrapes job listings and filters them according to user preferences. The CV assistant parses the user's CV correctly, offers advice for improvement, fills in missing information, and can generate a new CV based on a selected template. The motivation letter module produces letters relevant to the job posting and incorporates information about the company. However, the system faces several limitations and challenges, which are discussed in detail later in this report.

Keywords: AI, job application, Gemini, LLM, job scraping

1 Introduction

The current job market has become increasingly competitive and time-consuming for job seekers. Applicants often need to apply to a large number of positions across multiple job platforms, many of which lack comprehensive filtering options. In addition, employers frequently require tailored motivation or cover letters for each application and often rely on Applicant Tracking Systems (ATS) to screen CVs based on specific keywords or criteria. This makes the job application process both tedious and demanding.

The Recent development in artificial intelligence and natural language processing presents an opportunity to address these issues and try to streamline the job-hunting process.

Jobby aims to simplify the job application process by offering an AI-powered assistant that performs three main tasks:

- Scrapes and filters job postings.
- Creates or refines user CVs interactively.
- Generates tailored cover letters for each job.

Using Streamlit for the front-end and Gemini API for intelligent text generation, Jobby enables job seekers to focus on applying rather than manually editing documents and searching job boards.

2 Materials & Methods

2.1 System Architecture and Development Environment

Jobby was developed using Python 3.8+, following a modular architecture with distinct frontend, backend, and AI integration components. Dependency management was handled using `pip` via a `requirements.txt` file.

2.2 Frontend Implementation

The user interface was built using **Streamlit 1.28.0**, a Python framework designed for rapid development of data applications. The frontend includes four primary pages:

1. **Main Interface** – Serves as the application entry point, providing navigation to other modules.

[1] https://github.com/osayed159/Jobby

2. **CV Generator** – Handles template selection, AI-guided CV creation, and PDF export.
3. **Job Search** – Enables job scraping from LinkedIn and Xing with user-defined filters.
4. **Motivation Letter Generator** – Generates personalized cover letters using CV data, job descriptions, and enriched company information.

The frontend uses Streamlit's session state to maintain continuity across interactions with the AI assistant.

2.3 Document Processing and File Handling

Document processing primarily targets PDF files. The following library is used for text extraction:

- **PyPDF2 3.0.0** – Extracts text content from uploaded CVs in PDF format.

2.4 AI Integration and Language Model

The system integrates Google's **Gemini 2.0 Flash** experimental model through the `google-generativeai 0.3.0` API, using a custom `generate_gemini_response()` function to interface with the model.

The **CV Assistant** prompts the model to assist users in creating and refining their CVs, maintaining a conversation history for continuity and context.

2.5 Web Scraping Infrastructure

Automated job data collection is performed using **Selenium 4.0.0** alongside **undetected-chromedriver 3.5.0** to bypass bot detection mechanisms. The scraping architecture consists of:

- **Base Driver Class** – Handles core browser automation and setup.
- **LinkedIn Scraper** – Automates job search and extraction from LinkedIn.
- **Xing Scraper** – Performs similar functionality for Xing.

Job listings are extracted using a combination of CSS selectors and XPath expressions to identify relevant HTML elements.

2.6 Intelligent Job Filtering System

The **Agent** class is responsible for intelligent job filtering based on user preferences. The filtering process involves:

1. **Batch Processing** – Jobs are analyzed in configurable batches (default: 5 per batch) to optimize API usage.
2. **Preference Matching** – Job descriptions are compared to user-defined preferences using NLP.
3. **Response Parsing** – The model's responses are parsed to extract reasoning for matches and summarized job descriptions.

2.7 CV Template System and PDF Generation

The CV generation module uses a **JSON-based template structure** supporting multiple sections:

- Basic information (name, contact details, location)
- Education history (institutions, degrees, dates)
- Work experience (roles, responsibilities, dates)
- Skills (grouped by category with proficiency levels)
- Projects (descriptions and technologies used)
- Awards and certifications

PDF generation is handled by the `ResumeDriver` class, which interacts with the **resumake.io** service. It uploads the structured JSON data and automatically downloads the generated CV in PDF format.

2.8 Company Information Enrichment

To enrich the motivation letter with relevant company information, the system uses:

- **googlesearch-python 1.2.3** – For issuing automated search queries.
- **BeautifulSoup 4.12.0** – For parsing and extracting content from retrieved HTML pages.

Information is gathered from:

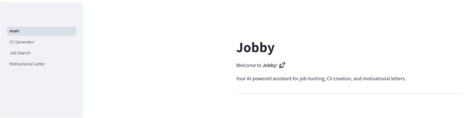
- LinkedIn company pages
- Official company websites
- General overview and description sources

The collected data is then summarized and incorporated into the motivation letter generation process to improve personalization and relevance.

3 Results

3.1 Application Interface and User Experience

The *Jobby* application successfully delivers a multi-page web interface accessible via a local Streamlit server. The main dashboard offers navigation to three core functionalities: CV generation, job search, and motivational letter creation.



Jobby

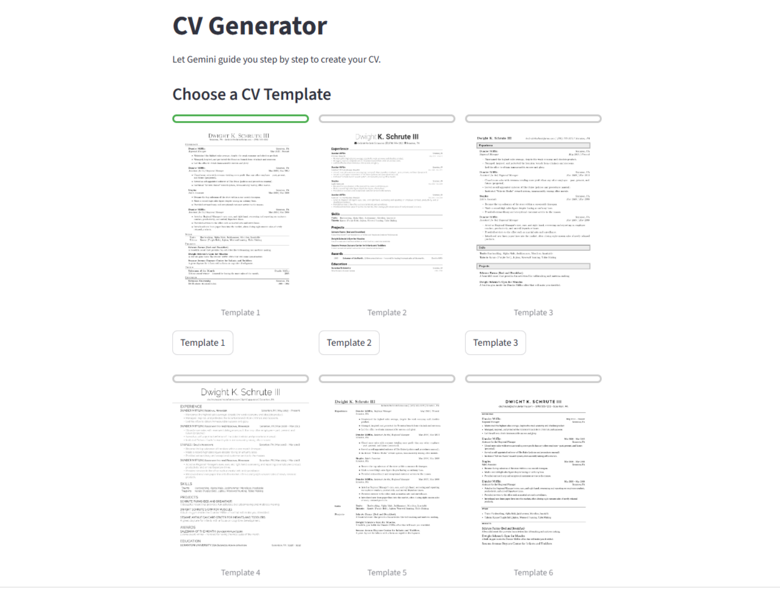
Welcome to **Jobby**! 🚀

Your AI-powered assistant for job hunting, CV creation, and motivational letters.

Figure 1: Main Jobby application interface showing navigation options

3.2 CV Generation System Performance

The CV generation module processes user inputs through an AI-guided conversation flow. The system supports nine distinct CV templates, each visually represented via preview images arranged in a responsive grid layout.


CV Generator

Let Gemini guide you step by step to create your CV.

Choose a CV Template

Figure 2: CV template selection interface with visual previews

The system produces valid JSON structures conforming to a predefined schema. Generated CVs include all standard sections:

- Basic information
- Education
- Work experience
- Skills
- Projects
- Awards

PDF generation via `resumake.io` results in professionally formatted documents with consistent styling.

3.3 Job Scraping and Filtering Results

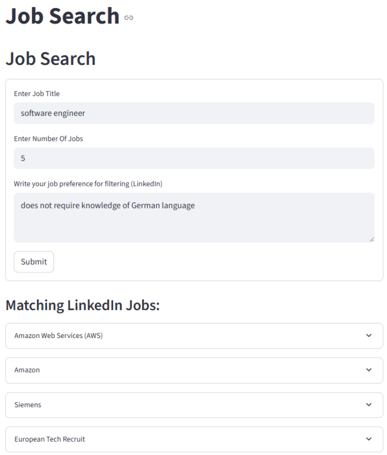
The web scraping system extracts job listings from both LinkedIn and Xing platforms. The LinkedIn scraper navigates multiple result pages and correctly extracts job details.

The Xing integration also successfully retrieves job postings, effectively handling dynamic content.

3.4 AI-Powered Job Filtering Effectiveness

The intelligent job filtering module processes scraped job data in configurable batches. Key results include:

- **Batch processing efficiency:** 5 jobs per API call
- **Relevance matching:** High accuracy in categorizing jobs based on user preferences
- **Summary generation:** Bullet-point summaries for each relevant job



Job Search

Job Search

Enter Job Title: software engineer

Enter Number Of Jobs: 5

Write your job preference for filtering (LinkedIn): does not require knowledge of German language

Matching LinkedIn Jobs:

- Amazon Web Services (AWS)
- Amazon
- Siemens
- European Tech Recruit

Figure 3: Filtered job results with AI-generated summaries

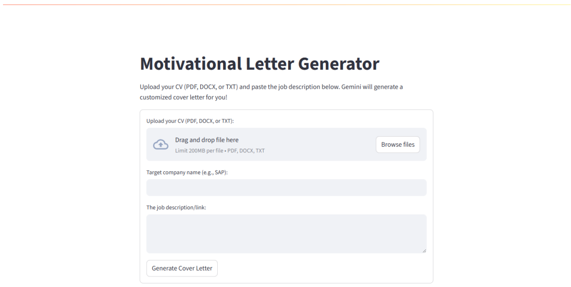
The system accurately extracts key attributes, including:

- Responsibilities and requirements
- Location and language expectations

3.5 Motivational Letter Generation Output

The motivational letter module generates personalized letters using:

- CV content analysis
- Job description requirement extraction
- Company research from web sources
- Contextual composition using AI



Motivational Letter Generator

Upload your CV (PDF, DOCX, or TXT) and paste the job description below. Gemini will generate a customized cover letter for you!

Upload your CV (PDF, DOCX, or TXT):

Drag and drop file here

Limit 200MB per file • PDF, DOCX, TXT

Target company name (e.g., SAP):

The job description/link:

Figure 4: Motivational letter generation interface with file upload and job description input

Generated letters follow professional structure and maintain coherent narrative flow, linking user qualifications to job expectations.

3.6 Output Quality and Format Compliance

CV Output:

- Consistent formatting across templates
- Complete section population
- Proper date and contact formatting
- Skill grouping by category

Job Search Output:

- Accurate job title and company name extraction
- Full job description preservation
- Functional direct links to job postings
- Expandable UI elements for filtered results

4 Discussion

Most of the project's technical details and system behavior have been covered in the Materials & Methods and Results sections. This section focuses on current limitations and implementation challenges.

4.1 System Limitations and Usability Challenges

While Jobby demonstrates core functionality across job search, CV generation, and motivation letter creation, several limitations affect the overall user experience:

- **Fragmented Workflow**: Currently, users must manually copy job descriptions between modules (e.g., from job search to CV or letter generation), which disrupts the flow.
- **Scraping Setup**: The Selenium-based scraping approach requires launching a visible Chrome window and manual login credentials for LinkedIn, which may be inconvenient or off-putting to users.
- **API Quota Constraints**: The Gemini API has request limits, which may prevent some users from generating a CV from scratch if too many requests are made during a session.

5 Conclusion

This project demonstrates the potential of combining AI-driven language models with automation tools to streamline the modern job application process. By integrating job scraping, intelligent filtering, CV generation, and tailored motivation letter creation into a single platform, Jobby offers users a practical assistant to reduce the time and effort typically required for job hunting.

While the system is functional and showcases the effectiveness of large language models for personal document generation, several usability and technical limitations remain. Improving workflow continuity, enhancing user experience during scraping, and addressing API limitations are key areas for future development.

Overall, Jobby serves as a promising prototype of how AI can augment real-world productivity tasks, especially in areas where personalization and efficiency are both critical.