Project Proposal: Job Jedi

Yulin Zhao
yulin6@illinois.edu
University of Illinois Urbana-Champaign
Champaign, United States

Kaimeng Zhu
kaimeng2@illinois.edu
University of Illinois Urbana-Champaign
Champaign, United States

1 Introduction

Are you spending excessive time crafting customized resumes for each job application? Look no further, as Job Jedi is here to assist you! Our innovative project leverages the capabilities of GPT to generate highly relevant resumes based on job descriptions, and seamlessly fills out job applications. Users input their project experience, skills, and work history, and Job Jedi extracts and summarizes the most pertinent information. With a customized resume format, Job Jedi generates an optimized resume and automatically uploads it, while also filling in all the necessary fields in the job application. Say goodbye to the hassle of manual resume customization and application form filling with Job Jedi by your side!

2 Significance

Job seekers often face the challenge of repetitive resume revisions and application form filling, which can be time-consuming and tedious. To address this issue, Job Jedi will integrate the powerful language model, ChatGPT [Brown et al. 2020], and provide a convenient browser extension that automates both resume optimization and application form filling. With just a simple click, Job Jedi will take care of most of the steps, utilizing the optimized resume and automatically filling in the required information. Job Jedi aims to become a public Chrome extension that helps individuals reduce the complexity of job applications. The effectiveness of Job Jedi will be measured and evaluated based on the relevance of the generated resume and the completeness of the auto-filling feature. We also welcome feedback and testing from our users to further enhance the capabilities of the Job Jedi extension.

3 Implementation

3.1 Resume Generation

Job Jedi will generate an optimized resume for each job. Users will be prompted to upload a background file that includes additional projects and detailed information about their project and work experience. Based on the job description, our language model (such as ChatGPT) will identify the most relevant projects and refine their descriptions to better align with the job requirements. Subsequently, Job Jedi will generate a professionally formatted PDF resume file incorporating these optimized descriptions. This part will be implemented in Python and deployed on AWS EC2.

3.2 Auto Filling

With the optimized resume, Job Jedi will automatically fill out the application. Job Jedi will automatically detect the required fields on the webpage and accurately fill in the corresponding information. This eliminates the need for laborious manual data entry, saving valuable time for our users. This part will be implemented in JavaScript and developed to be a Chrome extension.

4 Limitation

As an important note, our program utilizes ChatGPT, which may require uploading user data. This could raise potential data privacy concerns in accordance with OpenAI's policy. Additionally, it's important to note that the performance of the generated resume is dependent on the capabilities of GPT. Moreover, there may be associated costs with the commercial use of GPT. We are committed to maintaining transparency and ensuring the utmost security of user data while also managing expectations regarding the limitations and costs associated with using GPT for resume generation.

5 Tasks Assignment and Timeline

Here are all the parts with the timeline of this project:

1. extraction of the job description (by Apr. 20)
2. user interface of extension (by Apr. 20)
3. generates an optimized resume using ChatGPT (by Apr. 20)
4. extraction of individual questions on the application website (by Apr. 27)
5. send those questions to GPT and get GPT to answer (by Apr. 27)
6. fill in the answer from GPT and fill it back to the website (by Apr. 27)
7. generate reports and presentations (by May 4)

Yulin Zhao will work on the back-end (task 3, 5) and Kaimeng Zhu will work on the front-end (task 1, 2, 4, 6).

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL]