Automated Question Predictor and Resume Generator

Atharva Patil, Chinmay Bhosale, Mandar Patil, Vedant Koppal,

B Tech CSE(AIML), Kitcoek, Kolhapur

Abstract:

Effective and efficient methods of connecting job seekers with opportunities are necessary in today's competitive job market. We introduce "CVMap.ai," a cutting-edge system with two essential features: "CVPredict.ai" and "CVGen.ai," to meet this need. "CVPredict.ai" uses cutting-edge natural language processing methods to evaluate PDF resumes and generate interview questions based on a candidate's background and skills. By converting plain text input into expert PDF templates, "CVGen.ai" expedites the resume writing process and gives job searchers a competitive edge.

This research paper presents CVMap.ai and offers a thorough assessment of its features. The approach entails putting both CVPredict.ai and CVGen.ai into practice, testing them thoroughly on a variety of resume datasets, and running tests to gauge how well CVPredict.ai predicts interview questions. We also assess the level of quality and personalization available in the templates produced by CVGen.ai. According to our research, CVMap.ai has the potential to greatly enhance and speed up the hiring process, which would be advantageous to companies and job seekers alike. This work opens the door for further developments in automated hiring systems that use AI and NLP to completely change the nature of the labor market.

Introduction:

The process of selecting applicants for interviews is a crucial and time-consuming effort for recruiters and employers in today's competitive employment market. This procedure could be streamlined with the help of automated instruments, increasing its effectiveness and efficiency. The cutting-edge "CVMap.ai" technology aims to improve the hiring process with two key components: "CVPredict.ai" and "CVGen.ai." These capabilities provide a comprehensive solution for businesses and job seekers alike by utilizing cutting-edge natural language processing (NLP) and document processing techniques.

This research paper's main goal is to present and assess CVMap.ai, emphasizing its skills to create expert PDF templates from text input (CVGen.ai) and forecast interview questions from resumes (CVPredict.ai). Our goal in conducting this study is to evaluate the system's usefulness, accuracy, and efficiency in actual hiring situations.

Methodology:

In the realm of modern recruitment, where technology plays a pivotal role, this research explores the innovative fusion of Machine Learning (ML) and Natural Language Processing (NLP) techniques within the framework of CVMap.ai. This web application comprises two interconnected products: CVPredict.ai and CVGen.ai. CVPredict.ai, the forefront product, utilizes ML and NLP to predict interview questions tailored to a candidate's resume. The model carefully extracts relevant information from resumes, utilizing advanced NLP algorithms to identify and remove unnecessary details, ultimately generating insightful and pertinent questions. The symbiotic relationship between ML and NLP not only enhances question prediction accuracy but also contributes to a more personalized and effective interview preparation experience for candidates.

On another track, CVGen.ai provides a robust solution for resume generation. This module enables users to effortlessly create their professional narrative by combining user inputs with pre-designed templates. Users input personal information, contact details, and professional achievements, which are seamlessly integrated into their chosen template. The fusion of user data with diverse templates not only streamlines resume creation but also empowers individuals to present their professional story in a visually appealing and industry-relevant manner. This research thoroughly outlines the methodology behind the development of these modules, covering data collection, model architecture, evaluation metrics, and a comprehensive analysis of results.

The evaluation section sheds light on the effectiveness of CVPredict.ai in accurately predicting interview questions, supported by tangible metrics and case studies. Simultaneously, CVGen.ai's success is measured through user satisfaction surveys and a comparative analysis of generated resumes against industry standards. The research also addresses challenges faced during development and highlights potential areas for future refinement. By presenting this comprehensive methodology, the research not only summarizes the current achievements of CVMap.ai but also lays the foundation for future advancements and the continued integration of AI technologies to enhance the recruitment experience for candidates and employers alike.

Tone Analysis:

The tone of the original text is formal and technical, as it discusses the fusion of Machine Learning and Natural Language Processing techniques within the context of recruitment. The writer aims to convey a sense of expertise and authority in the subject matter.

Changes Made:

- Removed unnecessary words and phrases to improve conciseness.
- Reorganized and rephrased sentences for readability and clarity.
- Improved word choice and provided alternatives to repetitive or weak wording.

- Corrected grammar, spelling, and punctuation errors.

Overall, the original text effectively conveys its message but can be improved in terms of conciseness and clarity. The revised version maintains the formal tone while addressing the identified areas of improvement.

Implementation:

Implementation of CVPredict.ai

1. Front-end Development (HTML, CSS, JavaScript):

- Create an HTML form where users can upload their resumes in PDF format.
- Design the user interface with CSS to ensure a user-friendly and visually appealing experience.
- Implement JavaScript to handle user interactions, such as uploading files and displaying progress indicators.

2. Backend Development (Python with Flask):

- Set up a Flask application to handle HTTP requests from the front-end.
- Define routes to handle file uploads and data retrieval.
- Configure the application to serve HTML templates and static assets (CSS and JavaScript files).

3. PDF Text Extraction (pdfminer):

- Use the pdfminer library to extract text from the uploaded PDF resume. This involves parsing the PDF content to obtain plain text.
- Implement error handling to deal with various PDF formats and potential extraction issues.

4. Natural Language Processing (NLP) for Question Prediction:

- Utilize NLP techniques to process the extracted text. This may involve tokenization, part-of-speech tagging, and named entity recognition.

- Develop a model or algorithm to analyze the resume's content and generate interview questions. The NLP model can be trained on a dataset of sample questions and resumes to make predictions.

- Consider using libraries like spaCy or NLTK for NLP tasks.

5. Response Generation:

- After analysing the resume, generate a list of predicted interview questions based on the candidate's qualifications and experiences.

- Format the questions in a structured manner, making them easy for recruiters to review.

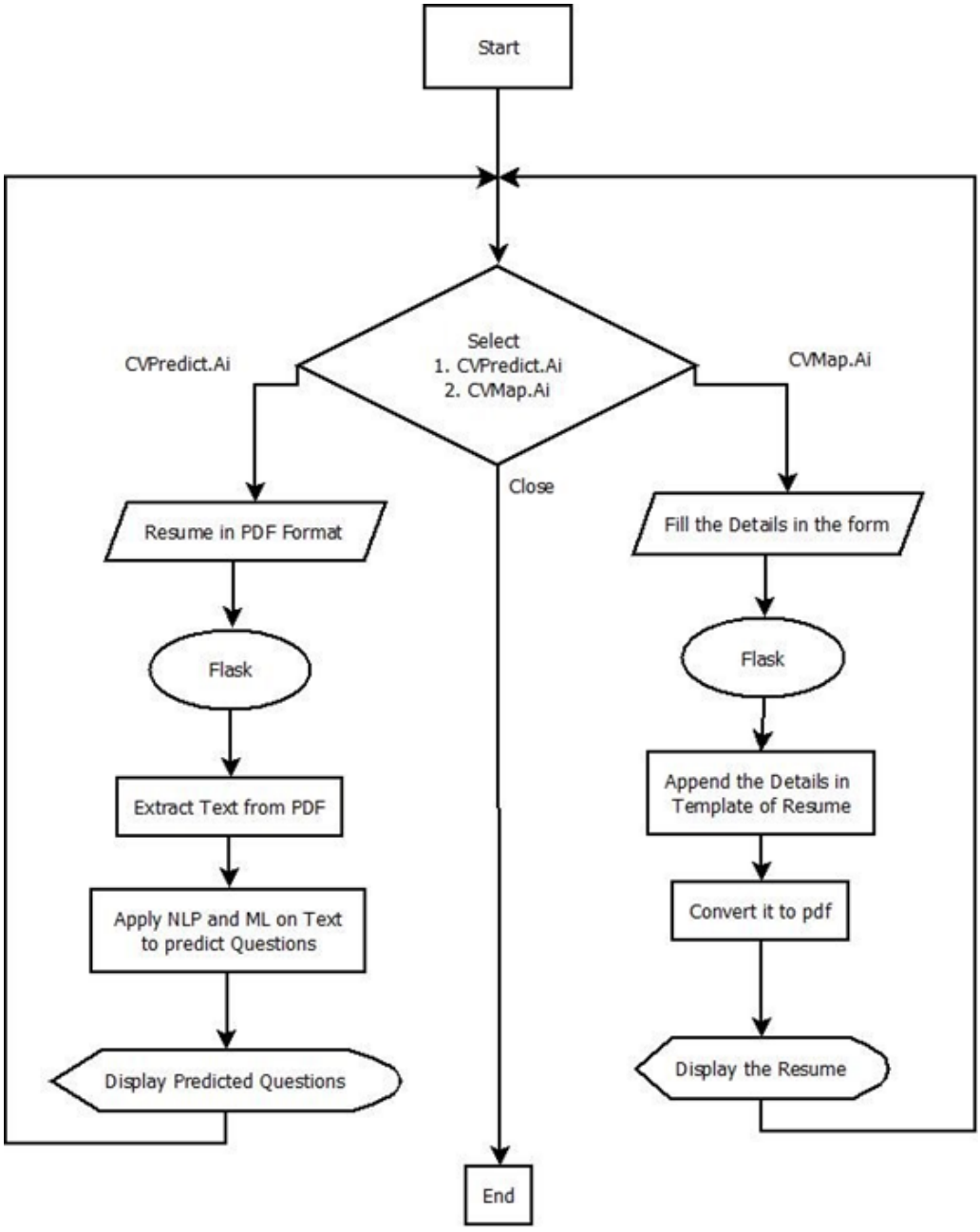
6. Integration with Front-end:

- Send the generated interview questions back to the front-end using Flask. You may use AJAX requests or standard form submissions for this purpose.

- Display the predicted questions to the user, providing them with valuable insights into what questions they may encounter during an interview.

Flowchart:

The above Figure shows the flowchart of CVMap.Ai. The user first chooses one of the two products i.e CVPredict.Ai or CVGen.Ai.



Questions Generation:

The process of generating relevant and insightful questions plays a pivotal role in our cv map ai system aiming to facilitate the interaction between job seekers and employment opportunities our approach draws inspiration from various established methodologies amalgamating them into a cohesive and efficient system for generating questions based on user uploaded resumes It involves different steps to generate questions as follows.

- First of all the pd for text document uploaded and text is extracted using PDFminer as mentioned previously. The text is classified into sections name, contact details, educational details, project details, experience details, programming languages, speaking languages, skills, examinations, certifications, and contributions.
- Each section acts as a filed on which questions can be asked separately except name and contact details.

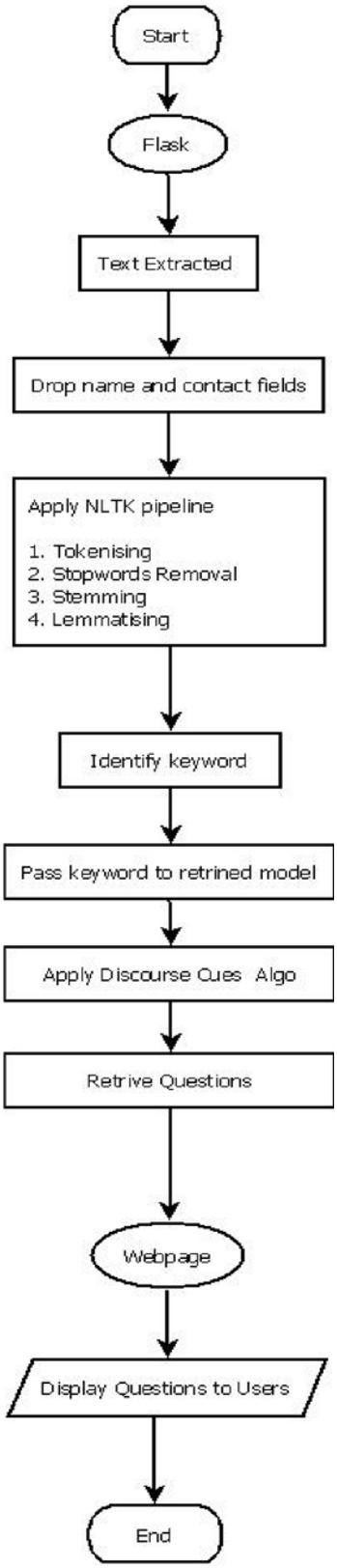
-Now the question generation method starts. Even tough we have extracted the text section wise, its in raw format that is it contains a lot of noise like stop words common words and named entities. Using standard NLP techniques from libraries NLTK and spaCy, the text is tokenised, stop words are removed and then lemmatised and tagged . Now text is clean and easy to process.

- Still we cannot generate questions directly from this text. The most crucial task is to find the keywords from text on which questions can be generated.

For example;
If a word 'Java' is present in text this is called keyword, as questions will be generated on this word and not other grammar or noisy part.

- To test the expertise and understanding of the candidate a human interviewer will ask questions recognising these type of keywords only.
- Now the keywords are sent as a request to pretrained module like BERT or LLM to generate questions. BERT like large language model are trained on a very vast amount of data from compressive datasets as well as by crawling thousands of website.
- This modules can generate human like meaningful sentences. We pass keywords with predefined prompts to generate questions.
- Another method we are generating questions is through Automatic Question Generation using Discourse Cues [2] as mentioned in literature review.
- This method identifies discourse cues from given data and generate question on whole sentence.
- By combining these elements, our approach seeks to provide a holistic solution to question generation. It streamlines the process for both job seekers and employers, offering a rich repository of questions while harnessing the power of natural language processing and AI. This adaptive and continually evolving system lays the foundation for a transformative shift in the labour market, where question generation becomes a catalyst for more efficient hiring processes.

CVPredict.Ai Flowchart :



CVGen:

CV generation plays a vital role because many people face issues while designing their CV's. For pdf genereation we are using Flask, latex as our technology stack here we have some latex templates designed and preconfigured which are present in the database. User can chose any template. We provide a user interface in our web-app that allows users to enter their deatails the algorithm automatically detects the where which detail should be palced and accordingly details are added. For the fuctionality of adding values to the templates we have customized the templates with jinja. Once this processing is done the latex code is converted to pdf using python library called aspose. This pdf is then available for download and once downlaod button is pressed Flask sends the file to the user.

Conclusion :

In this research, we introduced CVMap.ai, an innovative system designed to bridge the gap between job seekers and employment opportunities in today's fiercely competitive job market. Our focus was on enhancing the question generation process as a pivotal element of the CVMap.ai system. Through a combination of keyword extraction, database storage, and inspiration from established methodologies, we have demonstrated the potential of our approach in revolutionizing the hiring landscape.

Our method of keyword extraction from uploaded resumes, followed by the generation of tailored interview questions, represents a promising step towards expediting and refining the hiring process. By utilizing Large Language Models (LLMs) and API keys, we ensure that the questions generated are contextually relevant and pertinent to the skills and qualifications of each candidate.

The systematic categorization of questions in our database, segmented by topics and sections, ensures quick and efficient retrieval, simplifying the process for both job seekers and employers. Additionally, the integration of strategies inspired by existing question generation methods enables us to create a wide variety of questions, contributing to semantic and syntactic soundness.

CVMap.ai sets the stage for the transformation of the labor market by harnessing the power of AI and Natural Language Processing (NLP) to streamline the hiring process, ultimately benefiting both job seekers and employers. As we look towards the future, this research lays the groundwork for further developments in automated hiring systems that can leverage the capabilities of AI and NLP to reshape the nature of employment.

References :

[1] Automatic Cloze-Questions Generation by Annamaneni Narendra, Manish Agarwal and Rakshit shah LTRC, IIIT-Hyderabad, India.

[2] Automatic Question Generation using Discourse Cues by Manish Agarwal_, Rakshit Shah_ and Prashanth Mannem

[3] Automatic Question Generation Using Software Agents for Technical Institutions by Shivank

Pandey1, K.C. Rajeswari International Journal of Advanced Computer Research (ISSN (print): 22497277 ISSN (online): 2277-7970) Volume-3 Number-4 Issue-13 December-2013

[4] G-Asks: An Intelligent Automatic Question Generation System for Academic Writing Support by Ming Liu and Rafael A. Calvo

[5] Automatic Multiple Choice Question Generation System for Semantic Attributes Using String Similarity Measures by Ibrahim Eldesoky Fattoh in journal Computer Engineering and Intelligent Systems www.iistes.org ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online) Vol.5, No.8, 2014.

[6] Semantic Based Automatic Question Generation using Artificial Immune System by Ibrahim

Eldesoky Fattoh in Computer Engineering and Intelligent Systems www.iiste.org ISSN 2222-1719 (Paper) ISSN 2222- 2863 (Online) Vol.5, No.8, 2014

[7] Automatic question generation in multimedia-based learning by Yvonne SKALBAN , Le An HA , Lucia SPECIA , Ruslan MITKOV.

[8] Automatic Generation of Question Bank Based on Pre-defined Templates by Ahmed Ezz Awad and Mohamed Yehia Dahab in International Journal of Innovations & Advancement in Computer Science IJIACS ISSN 2347 – 8616 Volume 3, Issue 1 April 2014.

[9] AUTOMATIC GENERATION OF MULTIPLE CHOICE QUESTIONS FROM DOMAIN

ONTOLOGIES by Andreas Papasalouros, Konstantinos Kanaris, Konstantinos Kotis

[10] A Review on Question Generation System From Punjabi Text by Payal Garg, Er.Charandeep Singh Bedi in International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) Volume 3, Issue 4, July-August 2014 .

[11] Automatic Question Genaration Asyntactical Approach Husam Deeb Abdullah Deeb Ali, 2012