Takneek Submission – KSHTRIYAS [Hall II]

Website Link – **http://172.177.226.93/**

[Admin Credentials - {"username": "admin", "password": "prathamsahu"}]

RESUMATE MINER

1. EXTRACTING TEXT OUT OF PDF DOCUMENT

A. pdfminer.six API -

i. We utilized the **pdfminer API**, a powerful tool designed for extracting textual information from PDF documents. With this API, we were able to successfully retrieve all textual content from a resume-based PDF document.

ii. It also formatted all the tables in a formatted way providing structured view of the information, allowing for efficient processing and analysis.

B. PyPDF2 API -

i. There was one drawback for this API, that it was not able to parse the links from the document properly. To overcome that we used **PyPDF2 API**, to extract all the hyperlinks present within the document.

2. PARSING THE RESUME

A. Extracting Name -

i. We have used the pre-trained English Language Model ('en_core_web_sm') using the **spacy** library to detect the name present in the resume using a custom pattern.

ii. The pattern that we have used matches sequences of one to three consecutive proper nouns (PROPN) in the text. (which assumes the fact that name in a sane resume occurs in the starting)

B. Extracting relevant Contact Details -

i. We utilized regular expression (**regex**) pattern matching to extract pertinent contact information, such as email addresses and phone numbers, from the resume.

ii. Pattern for detecting mobile no. ->
`r"\b(?:\+?\d{1,3}[-.\s]?)?\(?\d{3}\)?[-.\s]?\d{3}[-.\s]?\d{4}\b"`

iii. Pattern for detecting email address ->
`r"([^@|\s]+@[^@]+\.[^@|\s]+)"`

C. Extracting Important Links -

i. We utilized regular expression (**regex**) pattern matching to extract all the important profile links such as GitHub profile, LinkedIn profile.

ii. LinkedIn Profile ->
`r"(http(s)?:\/\/)?([\w]+\.)?linkedin\.com\/(pub|in|profile"`

iii. GitHub Profile ->
`r"(http(s?):\/\/)?(www\.)?github\.([a-z])+\/([A-Za-z0-9]{1,})+\/?$"`

D. Extracting Relevant Skills -

i. Now, to extract all the relevant skills of a candidate from the resume we are using a skills database (which basically consists of thousands of skills and it's complexity score, which we will be going to use in future in our analytics part).

ii. It basically splits our resume data into tokens (using **NLTK**) and checks for each tokens in our database. If it matches, then it is added to the skill set of the indivisual, also keeping the check on duplicates as well.

E. Extracting Other Important Info -

i. We have also extracted other important sections of the resume such as Academic Qualifications, Achievements and Awards, Relevant Courses, Position of Responsibilities, Projects.

ii. It was pretty difficult to extract these section with precision and that too accurately according to our JSON. Below are the techniques that we tried to parse these data.

1. *Basics* - Used multiple regular expression pattern matching and NLP techniques using spacy to detect and parse data from the above section. But, again it didn't went well with all or most of the resumes in our testing phase.
2. *OpenAI's API for Developers* –
   So, we tried using Large Language Models with pre-trained universal datasets to accurately detect and extract these fields with precision.
   In our parser, we have made use of **gpt3.5-turbo** model to extract the required data.

3. DATA FORMAT

```
{
      "name": ,
      "phone_number": ,
      "email": ,
      "github_profile_link": ,
      "linkedin_profile_link": ,
      "field_of_study": ,
      "education": [{
            "degree": ,
            "institute": ,
            "year": ,
            "gpa": ,
      }],
      "achievements": ,
      "skills": [],
      "relevant_courses": [],
      "projects": [{
            "name": ,
            "organisation": ,
            "timeline": ,
            "brief_description": ,
            "project_link": ,
      }],
      "position_of_responsibilities": [{
            "position": ,
            "organisation": ,
            "tenure": ,
            "brief_description": ,
      }],
      "summary": ,
}
```

4. RESUME ANALYTICS

A. **Scoring of Resume -**

i. We have done the scoring of resume based on three parameters :

1. *Completeness Score* -> Scored the resumes based on completeness of information, i.e., it checks that are all the crucial information fields such as such as personal information, work experience, educational qualifications are included in our resume as specified in our pre-defined JSON format.
   It assists the company in the initial screening of candidates by highlighting those who have provided comprehensive and well-structured information in their resumes.
2. *Skill Score* -> It provides a comprehensive assessment of a candidate's qualifications by taking into account not only the number of skills listed but also the significance of those skills in the context of the job market or industry requirements.
   We've used the complexity score from our skill_database to calculate it.

$$s_{score} = \left(\sum_{i=1}^{n} s_i\right) \times \log(n)$$

   This score helps employers and recruiters identify candidates who possess the most valuable and applicable skills for a given role.
3. *Academic Score* -> Scores the resumes based on their previous academic records, after all it also plays a great role in determining the ability of the candidate.
   We've used the GPA / Percentage of the most recent academic degree / institution the candidate have pursued in order to determine this score.

ii. The final resume score is calculated based on the above 3 scores. We have given 20% weightage to completeness of resume, 40% of weightage to the skill score and rest of it to academic score.

$$score = 0.2 * c_{score} + 0.4 * s_{score} + 0.4 * a_{score}$$

B. **Site Analytics –**

i. On our website we have displayed all the analytics separately, so that the recruiter can compare every resume based on those scores.

ii. Also, there is an additional feature of incorporating score of technical test from the admin, i.e. recruiter's side.

iii. The recruiter could also filter out candidates based on the skills that they require and the academic record of the candidate or based on the resume or technical test score. Precise filtering options have been taken care of for this purpose.