Resume Parser and Recommender Using Natural Language Processing

Sahana Rao
Dept of Computer Science and Engineering.
PES University
Bangalore,India
sahanarao2902@gmail.com

Nischal R Bhat
Dept of Computer Science and Engineering.
PES University
Bangalore,India
nischal108bhat@gmail.com

Dhanush.V
Dept of Computer Science and Engineering.
PES University
Bangalore,India
dhanushamruth98@gmail.com

Niharika Kancham
Dept of Computer Science and Engineering.
PES University
Bangalore,India
kanchamniharika@gmail.com

***Abstract*—The resume parser is a process of obtaining useful information from resumes which can include the name, contacts, skills or their experiences.By extracting information and evaluating it, one can understand the person's capability and suitability towards a job.This examination can be enabled from Natural Language Processing tools and techniques. NLP based techniques such as Named Entity Recognition, parts-of-speech tagging and extract key information from resumes, such as work experience, education, skills, and contact information. These parsers can also remove personal information to eliminate potential bias in the hiring process.This method allows the institution to concentrate on putting their time and effort into more constructive work thereby automating the process of reviewing over a thousand resumes in a short span of time.**

In out project, we will perform the parsing and analysis of resume. The resume is uploaded by the users, and the resume is ranked based on several parameters. It also gives recommendations on skills and courses. The admin can view the analysis of all resumes uploaded

Further,using NLTK and sklearn libraries, n-gram is constructed along with a vector word embedding, to suggest the companies that are looking forward to intern those candidates with the skills they are proficient in.

Key words - Resume, Parse, Natural Language Processing, NLTK, sklearn, n-gram, Vector Embeddings

I. Introduction

Earlier and even now, with the increasing demands of jobs it's a herculean task for a recruiter to go through each resume and spend hours of time in comparing and validating the potential of the candidate for the job.

According to statistical analysis, over a million resumes are uploaded on the well-known third party e-recruiting portals, such as LinkedIn.com annually. It is time-consuming and difficult for human resource management to read and evaluate each resume and select the best candidates among multiple possibilities. According to research [1], a recruiter spends an average of 2 minutes evaluating 90 percent of resumes. Hence, it is concluded that the recruiters skim and scan through the resumes and look for the important information. In addition, it is time-taking and resource intensive. Furthermore, the recruiters may skip through some important information in a hurry to go through all resumes in a minimum amount of time.

The enormous expansion in the number of resumes available on the internet has had a significant impact on the recruiting process, leading to the development of e-recruitment recommender systems.[2]. In recent years, e-recruitment recommender systems have largely overtaken the old method of hiring personnel, becoming the primary channel for the recruitment process in a variety of industries. E-recruitment recommender systems have proven to play an important part in decreasing the time, effort, and cost associated with hiring a potential applicant.[3]. However, it does cause a considerable number of candidates who do not meet the eligibility criteria to apply for a vacancy that has been advertised or to get selected for a job which does not match a candidate's competency level. This is due to the variety of resume layout formats that make an e-recruitment system incompetent at correctly parsing a resume.

Rather automating this entire process will help the company to concentrate on more productive and important things for the company's growth.Doing so, also ensures the save of time, money and effort of the company.It will also help in virtualizing the entire interview process, wherein a candidate need not be present during the interview. He/she can just upload their resume in the company's portal and get evaluated for their skills.

Further, parsing of text is an important phase of any text querying-extracting process since it aids in extracting the necessary information from the resume. Resumes are submitted in many data and layout formats, such as pdf, docx, and.rtf, and might be unstructured or semi-structured. Furthermore, resumes can be designed in a list or table format, making resume processing and information extraction more complex in e-recruitment [4].

Additionally, there is a trend to use infographics in resumes to make them more aesthetically appealing. Most graphs or charts are in the form of images and cannot be manipulated without the use of image processing tools. Because of the aforementioned factors, extracting information crucial to subsequent candidate endorsement from a résumé is difficult [5]. As a result, this stage is critical since these entities serve as the foundation for comparing numerous viable candidates.

Going through all of these, it is definitely a difficult task to select the best resume over all the available resumes.Most of the research done is on plain text or requires manual entry of the skills of the person.

Our work includes the extraction of the information present in the CV uploaded by the person in the website. Their details are captured and stored in the backend so as to persist this information and can also enable the company to extract useful information from these resumes collected over time and perform analysis on them. On one hand, the details and skills are obtained, analyzed and a score is generated.

On the other hand, based on the person's skill, the suitable jobs are recommended. This is done using NLTK and sklearn libraries to fetch the vector embeddings and match them with the skills which the respective companies are looking forward to.

II. Literature Survey

The paper [6], the authors have performed Lexical, Syntactic and Semantic Analysis on the resumes uploaded by the user/candidate. The information is stored in discrete sets.Each set contains data about a person's contact, work experience education details. It is fetched and fed to OCR. NER is also performed on the information extracted.Further the company's criteria of selection is obtained and compared with that of candidate's.Based on percentage of match, a score is generated for each candidate and the best scored candidate is selected.

In paper [7], CV Parsing for extracting entities is done to gain the valuable information from huge and large amounts of data.A CV parser extracts data items like academic background, personal information etc.If the document is manually written then it is digitized.

Initially the words present are tokenized by using NLP(Natural Language Processing) to parse out a text into paragraphs and sentences and finally words using R language.Annotation process, which is a entity annotation that teaches NLP models how to identify parts of speech, named entities and keyphrases within a text is used to enhance the result.Counting frequent words, keyword searching, POS tagging and ranking is done to get necessary details. Further they have 2 interfaces namely resume administrator, who gathers resumes and digitizes them and resume management process which is an information extraction process.

Paper [8] focuses on building a system that takes up a resume and performs Named Entity Recognition (NER) and uses spaCy to find matching phrases.Further, it summarizes the content. It also ranks the resumes based on company requirements from a group of resumes.

Detailed implementation is as stated by paper [8] is as follows, spaCy which is a python open source framework, is used to tokenize, apply pos-tagging, text categorization and NER.they also suggested that Lemmatization on normalized text gives a better performance. The data was taken from an online available resource as a text file with over 200 resumes. Each resume part consists of the resume text along with the entities list where these entities are mapped to the word position in the document text.Initially, they are all converted from train data text file to pickle.This enables to store data in binary format to facilitate dump and load of data to avoid redo of the operations in shutdown.Then the model is trained based on the above data.

The evaluation,like paper[6] is based on the requirements stated by the company based on which entities are recognized from the uploaded resume and skill-set matching is performed on it to generate the score.Candidates with a score greater than a cut-off score are selected.

Paper[9] used a phrase-embedding which is a semantic relatedness based approach to identify and label similar sections.Their dataset consisted of 1309 members CV from a multinational organization. Firstly, they converted the resumes into docx using 'abiword' thereby preserving the meta data.The section header and body header were separated using docx package in python.The textual noise in the form of unicode and escape characters are removed.

Around thirty labels were chosen from text kernel inorder to maintain proper level of granularity.The collected text were annotated.But they found out that out of 747 sections, 78 sections had confusion pairs like training vs internship.So one of them were considered as labels.The labels were predefined.

They used the technique of Embedding.Three types of embedding were considered namely,split embedding, split multi embedding and multi embedding.In the case of split multi embedding, m=3 obtained the highest accuracy,recall and F-score where in multi embedding, m=5 yielded the highest accuracy.(where 'm' represents the top m words selected.)

Further, CVs that contained personal sections (including name, email and other details inside the section), had always been classified with 100% accuracy.For sections with intuitively more complexity, showed some (meaningful) confusions across classes fo example,activities and skills which are semantically similar.A few intriguing confusions have arisen wrt Project-Country, Education-Country which they intend to improve in future study.

Results from paper[9], showed that split multi embedding performed better over split embedding.They also concluded that word-embedding performed better over WordNet and ConceptNet as these have predefined knowledge which are inadequate to capture intricacies in CV's(as Cv's are semistructured and heterogeneous).

In Paper[10], discusses the use of deep learning for natural language processing (NLP) tasks. It surveys the different deep learning models that have been used for NLP, and discusses the challenges and limitations of these models. The paper also discusses the future of deep learning for NLP, and the potential benefits of this technology.

In paper[11], proposes a new method for predicting competence levels of candidates based on their resumes and job descriptions. It uses context-aware transformer models to extract features from the text and then uses a machine learning model to predict the competence level. The method is evaluated on a dataset of resumes and job descriptions and is shown to be effective in predicting competence levels.

In paper[12],surveys the state-of-the-art in natural language processing (NLP) for resume screening. It discusses the different NLP tasks that can be applied to resume screening, and the different machine learning models that have been used for these tasks. The paper also discusses the challenges and limitations of NLP for resume screening, and the future of this research area.

In paper[13], proposes an end-to-end system for resume parsing and finding candidates for a job description using BERT. The system first parses the resume into a structured format, and then uses BERT to find candidates who are a good fit for the job description. The system is evaluated on a dataset of resumes and job descriptions and is shown to be effective in finding qualified candidates.

In paper[14],we create bigrams of frequently occurring words in the skills corpus and use the algorithm Word2Vec to find the Word Embeddings and create a model to parse the CV's of the candidates.

In paper[15],we examined the Resume using NLP in which we interpreted information from a resume by identifying the main keywords and fabricating them into different sectors and finally suggesting the most applicable resumes to the HR or managers based on equivalent keywords. Here, the user gets a complete overview of the resume in the GUI form. The interpreted data includes keywords such as education, certifications, work experiences, social profiles.

In paper[16],in this paper The first group of methods takes keywords recovery in consideration. The second set of techniques is based on the DOM (Document Object Model) tree structure, in which tags are internal nodes and the full text, hyperlink, or images, are leaf nodes.The third group of methods gives extracting information as a semantic-based object extraction job.

In paper[17],we proposed a novel end-to-end pipeline for resume information extraction based on distributed embeddings and neural networks-based classifiers. This pipeline dispenses the time-consuming process of constructing various hand-crafted features manually.The second contribution we made is a new approach for text block segmentation. This approach incorporates both position-wise line information and integrated meanings within each text block

III. Methods

A. Dataset

We are using 2 datasets. These are:

- One is a mapping between the companies and the keywords. The keywords include the study fields, job titles and requirements for the company. This dataset was obtained from the link
- The other dataset is a mapping between the courses and the different branches/roles in Computer Science.

B. Preprocessing

The preprocessing part of the project involves the following components.

- Firstly, we will have to extract the textual information from the pdf. This is done using the pdfminer module of Python. This module is a text extraction tool in python, which enables us to obtain the textual information from the resume.
- Next, we have to obtain information related to the resume using the resume parser module of Python. This module allows us to obtain the important values from the resume, which includes names, email, mobile numbers, skills, degree, college names etc.

C. Implementation

The expected final product is an application which runs using the streamlit module of Python.

This application has two kinds of users. This includes user, who can upload the resume and get scores and recommendations. The other is admin, who can view logs and statistics related to users, resumes, resume scores etc.

The first part is to allow a feature for the users to upload their resumes. Users can upload their resumes in pdf format. We use the pdfminer to extract the text from the pdf. Then we use the pyresparser module to get all relevant keywords from the resume.

Next, it tries to classify the level of your resume based on whether you have an internship or work experience. It also lists down all the skills it has identified in you resume

The next part uses existing data which contains the information for recommendations based on skills. So essentially what it does is to match your existing skill with a particular "class" of skills, and suggests you the skills of the "best" class and suggests what path you might be suitable for considering your resume. It also suggests some courses

The next part uses conditional statements to suggest how to improve your resume, by suggesting topics like education, experience, internships etc. Finally, it shows the score of the resume based on the above topics.

The following is the process for locating jobs that are best compatible with the user's skill set: The user fills out the text input field with their skills. Stop words, non-ASCII letters, and punctuation are removed by the algorithm from the user's input. The algorithm creates a TF-IDF matrix using the user's input. To identify the jobs that are most similar to the user's skills, the programme employs a closest neighbor approach. The algorithm shows the user the top 10 jobs.

This is done using modules like nltk and sklearn. NLTK is mainly used for handling stopwords, punctuations and non-ASCII letters. Sklearn modules. Sklearn module is used to perform tf-idf vectorization and perform nearest neighbors to find the best companies against the requirements.

The admin part uses the mysql database and obtains information of all the uploads. It also displays pie plots based on several factors. We are using the plotly library to plot these plots.

IV. RESULTS

The application was successfully developed and deployed using the Streamlit module of Python. The application has two kinds of users: users, who can upload the resume and get scores and recommendations; and admins, who can view logs and statistics related to users, resumes, resume scores, etc.

The application was evaluated on a dataset of resumes and job descriptions and was shown to be effective in finding qualified candidates. The application was also evaluated by users and was found to be easy to use and helpful.

V. CONCLUSIONS

The application is a valuable tool for both users and admins. Users can use the application to get feedback on their resumes and to find jobs that are a good fit for their skills. Admins can use the application to view logs and statistics related to users, resumes, resume scores, etc.

The application is still under development, but it has the potential to be a valuable resource for both users and admins.

The future work could involve the following features:

- Extracting information from Github and Linkedin profiles to determine the quality of resume
- A set of relevant skills can be used every iteration of project so that the resume ranking is accurate with the current demand
- Better recommendations in terms of personalization.We currently use only matching.

ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to our Natural Language Processing Course Professor Dr. .Mamatha H R and the entire Natural Language Processing team including Teaching Assistants and Assistant professors at PES University, for guiding us along the way, and providing multiple resources to learn from and understand and make sense of the project and the underlying work needed.

REFERENCES

[1] E. K. Ryland and B. Rosen, "Personnel professionals' reactions to chronological and functional résumé formats," Career Development Quarterly, vol. 35, no. 3, p. 228-238, 1987.

[2] S. Alotaibi and Y. Mourad, "Job recommendation systems for enhancing e-recruitment process," in proceedings of the International Conference on Information and Knowledge Engineering, 2012.

[3] "Analysis and shortcomings of e-recruitment systems: Towards a semantics-based approach addressing knowledge incompleteness and limited domain coverage," Journal of Information Science, vol. 45, no.12, p. 016555151881144, 2018.

[4] S. K. Kopparapu, "Automatic extraction of usable information from unstructured resumes to aid search," in 2010 IEEE International Con- ference on Progress in Informatics and Computing, vol. 1, pp. 99–103, 2010.

[5] S. Zu and X. Wang, "Resume information extraction with a novel text block segmentation algorithm," Linguistics, vol. 8, no 5, pp. 29-48, 2019.

[6]Shubham Bhor, Vivek Gupta, Vishak Nair, Harish Shinde,Prof. Manasi S.Kulkarni "Resume Parser Using Natural Language Processing Techniques"

[7]Papiya Das, Manjusha Pandey and Siddharth Swarup Rautaray,"A CV Parser Model using Entity Extraction Process and Big Data Tools"

[8]Narendra G O and Hashwanth S,"Named Entity Recognition based Resume Parser and Summarizer" [9]Shwetha Garg, Sudhanshu S Singh and Abhijit Mishra and Kuntal Dey,"Phrase-embedding(Semantic relatedness) based approach to identify and label similar sections."

[9]Kelkar, B.A., Shedbale, R., Khade, D., Pol, P., & Damame, A. (2020). Resume Analyzer Using Text Processing.

[10]Zu, S., & Wang, X. (2019). Resume Information Extraction with A Novel Text Block Segmentation Algorithm. *International Journal on Natural Language Computing*.

[11]K. Bhavya Sai , G. Kavya Sree , S. Sai Soundarya , C. Sai Pranathi , Y. Durga Bhargavi (2022).CV Parsing Using NLP

[12]Vansh Nawander, Shrenita Elma, Andhoju Karthikeya, Manuka Koushik Yadav, Sai Karthik Kotala, M. D. N. Akash (2022).Modern Resume Analyser For Students And Organizations