Web Application for Screening Resume

Conference Paper · January 2019

DOI: 10.1109/ICNTE44896.2019.8945869

6 authors, including:

Sujit Amin
Fr. C. Rodrigues Institute of Technology
5 PUBLICATIONS **7** CITATIONS
SEE PROFILE

Nikita Jayakar
2 PUBLICATIONS **5** CITATIONS
SEE PROFILE

Sonia Sunny
FCRIT
1 PUBLICATION **3** CITATIONS
SEE PROFILE

DOI: 10.1109/ICNTE44896.2019.8945869

Citation:

S. Amin, N. Jayakar, S. Sunny, P. Babu, M. Kiruthika and A. Gurjar, "Web Application for Screening Resume," 2019 International Conference on Nascent Technologies in Engineering (ICNTE), Navi Mumbai, India, 2019, pp. 1-7, doi: 10.1109/ICNTE44896.2019.8945869.

Web Application for Screening Resume

Sujit Amin[1], Nikita Jayakar[2], Sonia Sunny[3], Pheba Babu[4], M.Kiruthika[5], Ambarish Gurjar[6]

[1,2,3,4,5]Department of Computer Engineering,
Fr. Conceicao Rodrigues Institute of Technology,
Vashi, Navi Mumbai, India.
[1]Email-id: mastersujitamin@gmail.com

[6]Benco Solutions
Waghle Estate, Thane,India

***Abstract*—This paper focuses majorly on the design of the web application which will be used to screen resumes (Curriculum Vitae) for a particular job posting. In the proposed system, a web application will encourage the job applicant candidates as well as the recruiters to use it for job applications and screening of resumes. Recruitment is a tedious process wherein the first task for any recruiter is to screen the resumes. The proposed web application is designed in such a way that job applicant as well as recruiters can use it with ease for applying for job openings and screening respectively. The recruiters from various companies can post the details of the job openings available in their respective companies. The interactive web application will allow the job applicants to submit their resume and apply for their job postings they may still be interested in. The resumes submitted by the candidates are then compared with the job profile requirement posted by the company recruiter by using techniques like machine learning and Natural Language Processing (NLP). Scores can then be given to the resumes and they can be ranked from highest match to lowest match. This ranking is made visible only to the company recruiter who is interested to select the best candidates from a large pool of candidates.**

Keywords—Natural Language Processing (NLP); NER (Named Entity Recognition); Section-based segmentation; NLTK (Natural Language Toolkit).

I. INTRODUCTION

Recruitment is a 200-billion-dollar business. It deals with hiring the best fit candidates having the relevant skills for a given job profile from an immensely large pool of candidates. If a company has any job opening for a position, scores of candidates mail their resumes to the company to apply for that opening.[1-4]

In the hiring process, the first task for any recruiter is to screen the resumes of all the job applicants. Any company having a job opening for a particular position will have their mail inboxes bombarded with thousands of emails from the aspiring job applicants every single day. Selecting the prospective candidates for that job position from a large pool of candidates for any recruiter is very tedious. It is an extremely daunting task for the recruiters of a company to manually go through thousands of resumes and select the most appropriate candidates for the job. Out of those thousands of resumes submitted to the company for the given job posting, about 75% of them do not showcase the relevant skills that are required for the job profile.[5-9] Due to this, the recruiters quite often find it really arduous to narrow down the most appropriate candidates from a large applicant pool.

In the recent years, there have been more than 50,000 e-recruitment sites which have been developed.[10-12] The developers of these online recruitment sites have used various approaches to identify the prospective candidates for a given job profile of a company. Some of these, have managed to employ classification techniques which will classify the candidate resumes into various categories for every job posting given by every company.[13-16] In these approaches, every candidate resume is tried to match with every given job posting on the recruitment site. The aim of these recruitment sites is to throw up the results to the candidate to which they are best fit into.[17] The techniques used by these sites have resulted in high accuracy and precision, but one of the major disadvantages is the factor of time complexity. If every candidate resume is matched with every other job posting given on the online recruitment site, the time complexity for acquiring the results is very high.[18-20]

The approach discussed in this paper is that by using machine learning to train the dataset for a particular type of job position. It is also proposed to use section-based segmentation for data extraction using Natural language Processing (NLP). In order improve the time efficiency of the web application, the candidate's resume will only be matched to those job openings where they are interested in and have applied to which will in turn reduce the time complexity.[21,22] Besides, the results of the resume matching of all the candidates who have applied for the job opening will be visible only to the recruiter of that particular company. This is done with the aim to aid the recruiters of any company from the long and tedious task of viewing and analyzing thousands of candidate resumes. In this intelligent-based approach, they will be given the option to view the candidate's resume as well as they will get

the results of the best candidates suitable for the required job position.

The section 2 briefly describes the existing systems available. After that, section 3 describes the entire design of the system which has been proposed to solve this problem. Section 4 includes a bird's eye view of the implementation of this ongoing project and section 5 is the conclusion and the future scope.

II. Literature Review

There have been over 50000 online recruitment sites which ask the job applicant candidates to submit their resumes on their website. In some of these websites, classification techniques for screening the resumes are not even employed. It is the job of the company recruiter to go through all the candidate resumes manually. This task is unassumingly daunting for the recruiters to select the most capable candidates for the subsequent rounds of the hiring process. Meanwhile, some recruitment sites have implemented the intelligent concept of automatically rating or classifying the resumes given by the candidates for a particular job position. Some of these websites or web applications are Indeed, Monster.com, Adecco.com, Top resume, Ideal etc. The description of some of these websites including their advantages and disadvantages has been given below in detail.

If we discuss one of the case study websites, Indeed, resumes can be uploaded by the job applicants on their profile. This opens up the avenue for the prospective job seekers to apply to various job openings in various companies. Initially, this happened to be a good approach since the recruiters did not feel the pressure to manually go through each and every resume. This is mainly because the job applicants had a very skewed number as compared to today's burgeoning number of job applicants. Going through the resumes of all the candidates has become a living nightmare for the already overburdened recruiters. They have to spend a lot of their energy and precious time to go through each and every candidate resume for selecting only the most appropriate candidates for the subsequent rounds. On top of this, the already frustrated recruiters learn that about 75% of the resumes submitted for the job opening have totally irrelevant skills for the given job description. Nevertheless, this website is still very popular where people post their resumes with the hope of getting selected for the subsequent rounds in the hiring process.

Coming to another case study, a website called Top Resume has employed the usage of techniques like Natural Language Processing to analyze a prospective job seeker's resume. Here, the task of the candidate is to only upload their resume on the portal. With the help of Natural language Processing, only the text data is extracted from the resume and the strength of the candidate's profile is displayed in terms of percentage. Additional attributes, such as the percentage of the skills the candidate has according to the education, certifications, courses and work experience of the candidate are also disclosed to the candidate itself.
No provision has been made for any job applicant to apply for a particular job opening on this website, nor does this website have the provision of providing the recruiter with a rank list of all the resumes according to the relevant skills for the particular job position.
There are many other web applications available in the literature providing mostly similar features.

III. Proposed System

As an attempt to overcome some of the disadvantages of the existing system, the proposed web application aids in reducing the workload of the recruiters.

1. Overall Working Model:

In the first step, the system accepts the resume from the job applicant and performs text extraction on it (see Fig 3.1). Since it is not possible to perform operations on a file in pdf format, it is first converted to text. Text is then fed to the pre-trained model that takes only that data which is relevant to the system. After this step, the data extracted will be displayed to the job applicant in editable format. The system allows the job applicant to edit information in this text before performing any operation on it. Once the job applicant is satisfied and submits the data, it can be sent for comparison with the job description requirements which is already passed through pre-trained model and the requirements like number of year of experience,qualifications, etc. is already known. Now, the job data(after passing through pre-trained model) is compared with the resume data(after passing through pre-trained model) after which it can calculate the score based on the formula.
Finally, the web application will display the candidate profile to the recruiter with other applicants for same profile in a rank list based on their score.

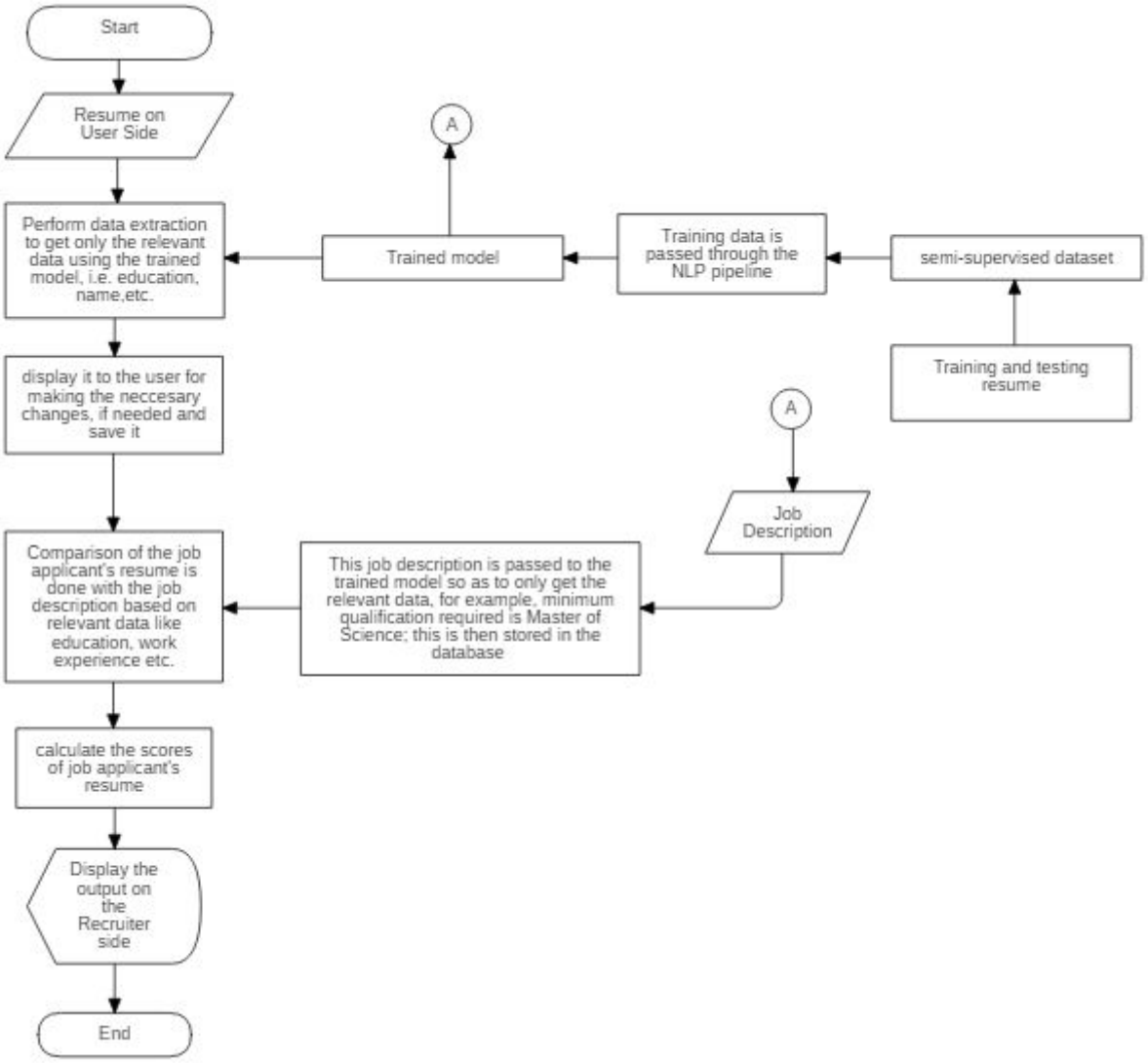

Fig. 3.1. Activity diagram showing how the overall system works

The proposed web application for screening and rating the resumes according to the job requirement posted by a company recruiter has various modules. These modules come under 3 parts, which are as follows:-

1. Job Applicant Side
2. Server Side
3. Recruiter Side

A. Job Applicant Side

In this part, the job applicant side or the client side design is discussed. It contains the following modules, namely, 'Accepting resume as input' and 'Data extraction module'.

1) Accepting resume as input

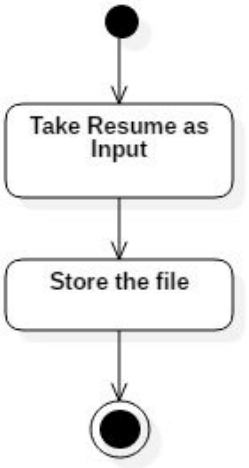

Fig. 3.2. Activity diagram for accepting resume as input

The proposed web application will have 2 types of users:- job applicant side and recruiter side. The company recruiter will have a login on the website. The recruiter will be able to post job openings available in their company. The job applicant will also have a login which will allow him to upload his/her resume as input. This input resume accepted in the format of a .pdf file will then be stored in the database. Since MySQL or any other primary SQL cannot store pdf files directly in the database, base64 encoding will be used to store this pdf resume file. Fig.3.2 shows the flow of the input resume acceptance and storage in the database.

2) Data extraction module

This module deals with the extraction of data from the resume given as input by the job applicants. This data extraction is done using section-based segmentation with Natural Language Processing.

This module will only extract the data relevant to the job posting from the resume of the job applicant. An unstructured resume of the job applicant is taken. The entire document is then parsed. After this, all the unimportant words such as 'this', 'for', 'is' etc. which is also called as stop words are removed. After this, the remaining data is then tokenized. Only the tokens containing the relevant data pertaining to the required entities for the job posting are then taken into consideration and a text file is generated of the same. Fig.3.3 gives the diagrammatic explanation of the above process.

The next part in the data extraction module is to display the extracted relevant important data to the job applicant. This sub-module is used to cross-check whether the job applicant is satisfied with the data which is extracted from their resume. The extracted entities will be passed to the front-end using flask to the job applicant in the form of an editable text file. They are allowed to make any necessary changes in the text file in case any information was not extracted by the data extraction module. Once the job applicant is satisfied with the changes being made in their text file, they can submit it to the website. This text file will then be stored in the database. Also, this text file will be forwarded to the score calculator module. Fig.3.4 shows the detailed flow of the module discussed above.

B. Server Side

In this part of the proposed system, the design of the server side is discussed. It contains the modules, namely, 'The training data module' and 'Converting the job description into required format module'.

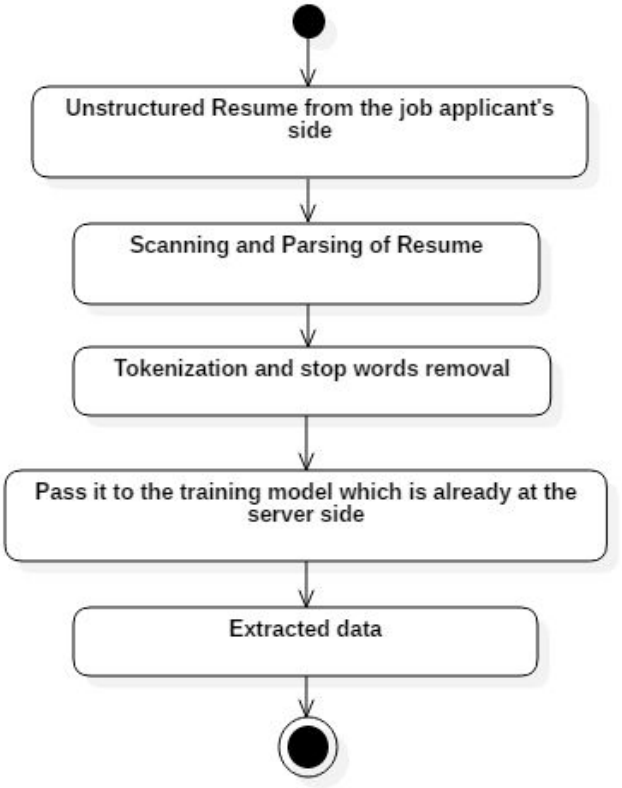

Fig. 3.3. Activity diagram showing how data extraction happens for the resume input

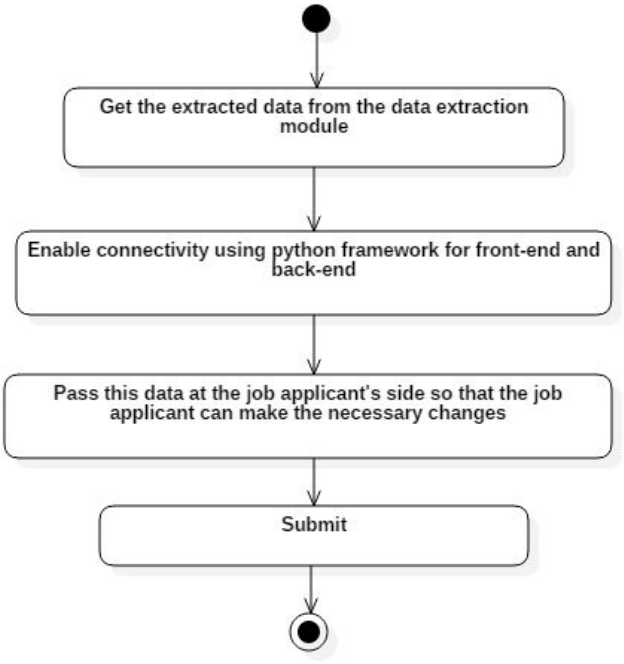

Fig. 3.4. Activity diagram to get the extracted part of data

1) Training data module

This module in the web application deals with training the dataset for a particular type of job posting, e.g., senior web developer. Fig. 3.6 shows the flow in which the dataset is collected and trained. First of all, resumes related to the particular type of job posting are collected. Then the relevant text of all the resumes is converted to JSON file by manually uploading the ZIP file of all these resumes on a website portal called dataturks.com.[23] It is possible to highlight only the selected text which needs to be converted into the JSON format on this portal. After converting the text annotations in pdf to JSON format, the JSON file is then passed through the NLP pipeline. The model is then trained using this NLP pipeline. It can be trained by using SpaCy which is an NLP framework. SpaCy is a framework which is trained for general data and not for specific datasets such as that of a resume. Hence, it is essential to tailor-make this SpaCy framework according to the proposed system's needs. This can be done by making changes to the NER (Named Entity Recognition) for this model so that the entities such as college name, skills, degree, work experience etc. can be correctly identified for the raw dataset. Now, the dataset is converted to JSON format on dataturks website and the output is then passed to the NLP SpaCy pipeline to obtain the trained model. In this approach, semi-supervised learning is used to label the important data in the ZIP file of pdf of resumes rather than manually typing each and every word for creating the dataset.

This dataset is then split into 2 parts:- training data and testing data and these parts are passed through the SpaCy pipeline.

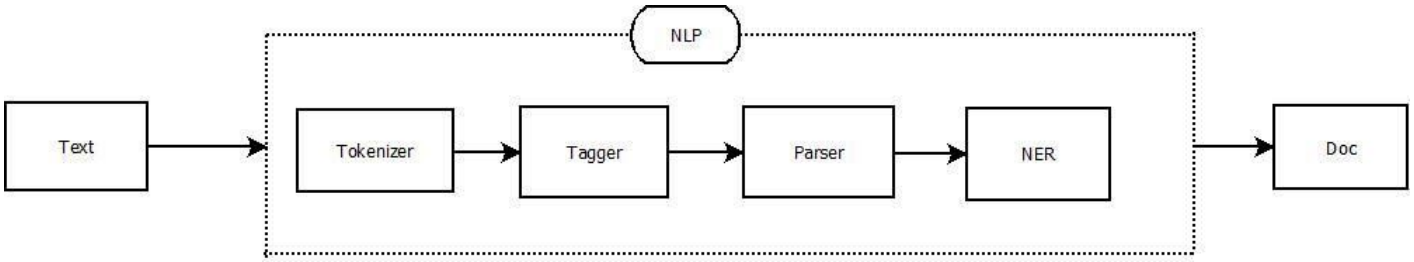

Fig. 3.5. The NLP pipeline for training dataset

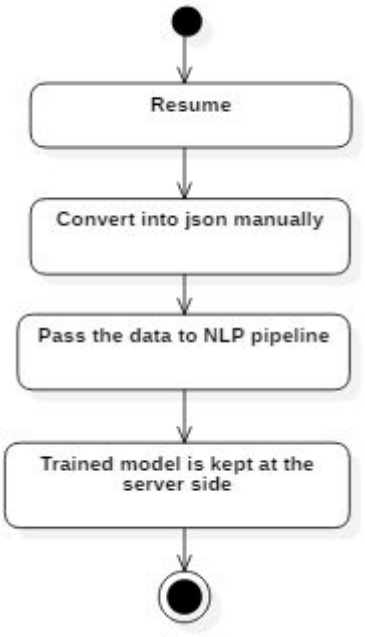

Fig. 3.6. Activity diagram for training the dataset

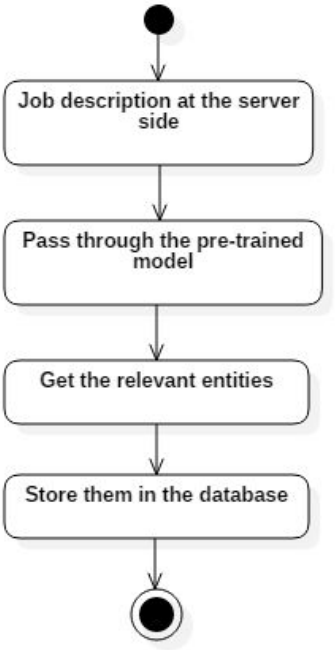

Fig. 3.7. Activity diagram showing job description is converted to required format

2) Converting job description into required format

This module deals with getting the job description or posting in the required text file for comparison with the candidate resumes text file so that the candidate resumes can be scored. For this to happen, the job posting is passed through the trained model relevant to the job position.

Relevant entities as per the trained model will then be extracted from the job posting uploaded by the recruiter. A text file will be made of the same and this will ultimately be stored in the database.

C. Recruiter Side

The recruiter side part contains the module- 'Calculation of scores for the resume given as input'.

1) Calculation of scores for the resume given as input

This module is the last module of this proposed system. It deals with the calculation of score for a candidate's resume according to the job posting they have applied. According to the score each candidate resume receives, a rank list will be made with the candidate receiving a higher score placed higher as compared to the candidate receiving a lower score. The scores as well as the rank list will only be visible to the recruiter and not to the candidates. The recruiter can then make an informed decision on when to select for the next round of the hiring process.

The job description text file is retrieved from the database. After that, the relevant entities of the candidate resume text file as well as the job description text file are then compared and a score is assigned to the candidate based on the following formula:-

$$S = \frac{|Sr|}{|RSj|} * 50\% + \frac{|Er|}{|REj|} * 20\% + \frac{|Xr|}{|RXj|} * 20\% + \frac{|\sum Rw|}{|\sum Cw|} * 10\% \quad \text{..... (1) [22]}$$

Where,

- S: relevance score assigned between a job post and a resume
- Sr: set of candidate's skills
- RSj: required job skills by the job post
- Er: set of concepts that describes the candidate's educational information
- REj: required education by the job post
- Xr: candidate's experience
- RXj: required work experience by the job post
- Yw: total number of employment years
- Cw: number of companies the applicant has worked in

The rank list based on the candidate resume scores in equation (1) will then be displayed to the recruiter of the company only.

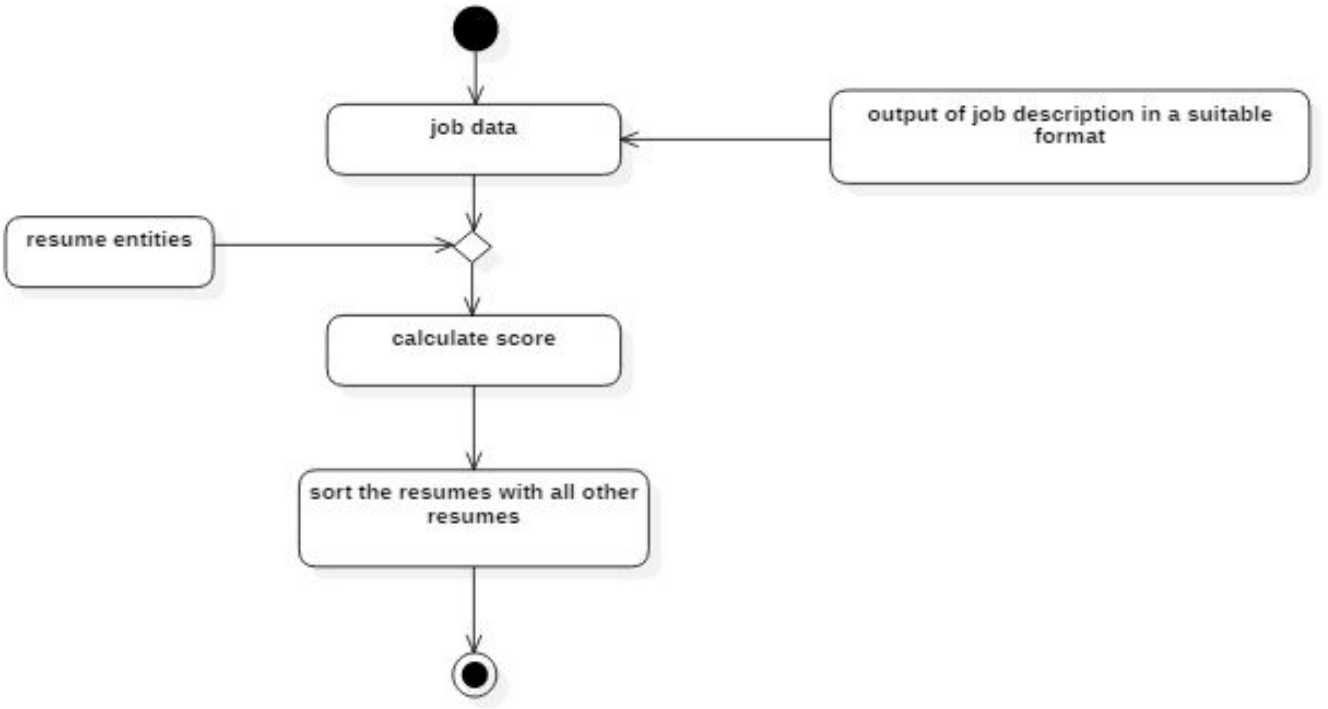

Fig. 3.8. Activity diagram showing comparison of candidate resume with job description

IV. Implementation Details

This is an ongoing project and some part of the proposed system has been implemented till now.

Training model for NLP pipeline implementation screenshots

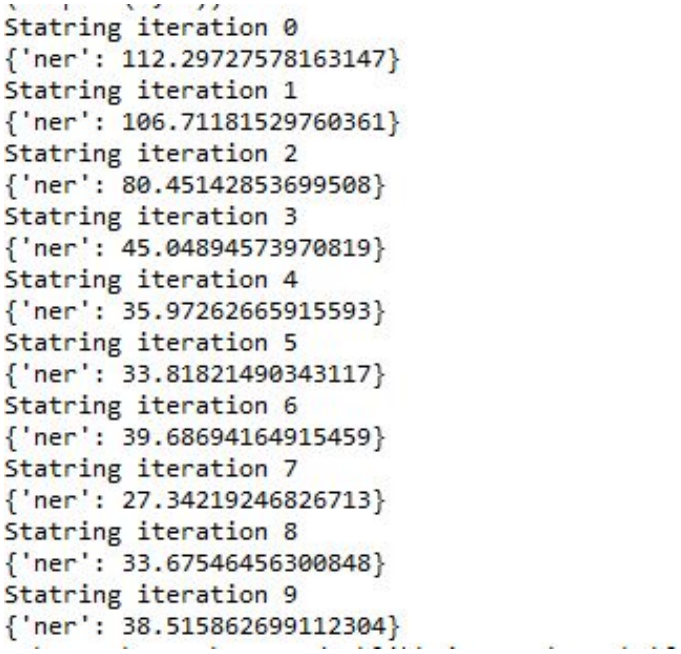

Fig 4.1. Training model screenshot 1

Accuracy of trained model:
For the current implementation, the accuracy is calculated using a model of 220 resumes out of which 20 resumes are designated for testing and 200 resumes are reserved for training the model. System gave the expected accuracy.

For scanning and parsing of resume screenshots, (see Fig 4.2):

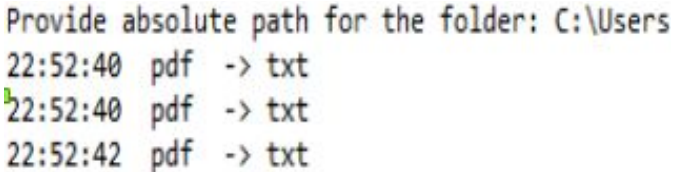

Fig 4.2. Scanning and Parsing of resume screenshot

Here, the system takes the input resume and passes it through NLTK(natural language toolkit) of which scanned and parsed text data is obtained which is fed to next module. Basic front end is developed.

V. Conclusion

The proposed system is currently under implementation and uses semi-supervised learning to achieve high accuracy. This system will definitely aid the recruiters to filter out the most prospective candidates based on their resumes for further rounds in the hiring process. It will ease the burden of the recruiters and they will not have to manually view each and every resume of the large pool of candidates.

References

[1] E Faliagka, L Iliadis, I Karydis, M Rigou, S Sioutas, A Tsakalidis, and G Tzimas, " On-line consistent ranking on e-recruitment: seeking the truth behind a well-formed CV," The Artificial Intelligence Review, 42(3), 515, 2014.

[2] A Kmail, M Maree, M Belkhatir, and S Alhashmi "An Automatic Online Recruitment System based on Exploiting Multiple Semantic Resources and Concept-relatedness Measures," Proceedings of the EEE 27th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 620-627, 2015.

[3] J Chen, Z Niu, H Fu, "A Novel Knowledge Extraction Framework for Resumes Based on Text Classifier," Proceedings of the International Conference on Web-Age Information Management. Springer International Publishing, pp. 540-543, 2015.

[4] C Hauff, G Gousios, "Matching GitHub developer profiles to job advertisements." Proceedings of the 12th Working Conf. on Mining Software Repositories, pp. 362-366, 2015.

[5] T Schmitt, P Caillou, M Sebag, "Matching Jobs and Resumes: a Deep Collaborative Filtering Task," Proc. of the 2nd Global Conf. on Artificial Intelligence, pp.1-14, 2016.

[6] S Mehta, R Pimplikar, A Singh, LR Varshney and K. Visweswariah, "Efficient multifaceted screening of job applicants," Proceedings of the 16th International Conference on Extending Database Technology. ACM, pp. 661–671, 2013.

[7] S Al-Otaibi and M Ykhlef, "Job Recommendation Systems for Enhancing E-recruitment Process", in Proceedings of the International Conference on Information and Knowledge Engineering (IKE), Las Vegas Nevada, USA, pp. 433-439, 2012.

[8] The International Association of Employment Web Sites (IAEWS), avaliable from: http://www.icmaonline.org/international-association-of-employment-web-sites, Date Visited: June 20, 2017.

[9] A Kmail, M Maree, and M Belkhatir, "MatchingSem: Online recruitment system based on multiple semantic resources," Proceedings of the 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), IEEE, pp. 2654-2659, 2015.

[10] W Hong, S Zheng, H Wang, J Shi, "A Job Recommender System Based on User Clustering," Journal of Computers, vol. 8(8), pp. 1960-1967, 2013.

[11] V.S Kumaran, and A Sankar, "Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping EXPERT," Int. J. Metadata Semantics. Ontologies, vol. 8(1), pp. 56-64, 2013.

[12] R Kessler, N Béchet, JM Torres-Moreno, M Roche and M. El-Bèze, "Job Offer Management: How Improve the Ranking of Candidates", in Foundations of Intelligent Systems, J. Rauch, et al., Editors. Springer Berlin Heidelberg, pp. 431-441, 2009.

[13] K Yu, G Guan, and M Zhou, "Resume information extraction with cascaded hybrid model." Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 499–506, 2005.

[14] F Javed, Q Luo, M McNair, F Jacob, M. Zhao, and TS. Kang, "Carotene: A Job Title Classification System for the Online Recruitment Domain," Proceedings of the IEEE First International Conference on Big Data Computing Service and Applications (BigDataService), pp. 286- 293, 2015.

[15] R Kessler, N Béchet, M Roche, J. M Torres-Moreno, and M El-Bèze, "A hybrid approach to managing job offers and candidates," Information Processing & Management, 48(6), 1124-1135, 2012.

[16] J.Martinez-Gil, A.L. Paoletti, and K.D. Schewe, "A smart approach for matching, learning and querying information from the human resources

domain," In East European Conference on Advances in Databases and Information Systems, Springer International Publishing, pp. 157-167, 2016.

[17] M Fazel-Zarandi and M S Fox, "Semantic matchmaking for job recruitment an ontology based hybrid approach," In Proceedings of the 3rd International Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web at the 8th International Semantic Web Conference, Washington D. C., USA, 2010.

[18] S Clyde, J Zhang, and CC Yao, "An object-oriented implementation of an adaptive classification of job openings," Proceedings of the 11th Conference on Artificial Intelligence for Applications, IEEE, pp. 9-16, 1995.

[19] R Kessler, J Torres-Moreno, and M El-Bèze, "E-Gen: automatic job offer processing system for human resources," in Proceedings of the artificial intelligence 6th Mexican international conference on Advances in artificial intelligence, Springer-Verlag: Aguascalientes, Mexico, pp. 985-995, 2007.

[20] G.A Miller, "WordNet: a lexical database for English," Comm. ACM, vol. 38(11), pp. 39-41, 1995.

[21] J Hoffart, FM Suchanek, K Berberich, E. Lewis-Kelham, G. De Melo, and G. Weikum, "YAGO2: exploring and querying world knowledge in time, space, context, and many languages", in Proceedings of the 20th international conference companion on World Wide Web, ACM: Hyderabad, India, pp. 229-232, 2011.

[22] A. Zaroor, M. Maree, and M. Sabha, "A Hybrid Approach to Conceptual Classification and Ranking of Resumes and Their Corresponding Job Posts," In: Czarnowski I., Howlett R., Jain L. (eds) Intelligent Decision Technologies 2017. IDT 2017. Smart Innovation, Systems and Technologies, vol 72. Springer, Cham.

[23] Best online platform for your ML annotation needs, an article which is available on the website dataturks.com