

Company To Candidate Mapper

1st Harsh Oza

Computer Engineering Dept.
St. Francis Institute of Technology
Mumbai, India
harshoza36@student.sfit.ac.in

2nd Darrel Noronha

Computer Engineering Dept.
St. Francis Institute of Technology
Mumbai, India
darrelnoronha@student.sfit.ac.in

3rd Shelton Jade Pinto

Computer Engineering Dept.
St. Francis Institute of Technology
Mumbai, India
sheltonpinto99@student.sfit.ac.in

4th Janhavi Parab

Computer Engineering Dept.
St. Francis Institute of Technology
Mumbai, India
janhavi.parab.d@student.sfit.ac.in

5th Prof. Dakshata Panchal

Computer Engineering Dept.
St. Francis Institute of Technology
Mumbai, India
dakshatapanchal@sfit.ac.in

Abstract—The problem of shortlisting and screening of candidates to find the best match in given time frame has always been a challenging task for recruiters in emerging as well as established companies. The paper proposes an intelligent system that could help in easing the recruitment process by mapping the resumes of the best candidates to the company's job description. The described algorithm is based on ranking candidates on the basis of a scoring system that will act as a mapper. Ranking of the candidates would be on the basis of certain parameters which will be extracted from the resume parser using Machine Learning(ML) and Natural Language Processing(NLP) Techniques which are the best match with the requirements in a specified job description. Also, recommending resumes to the company based on the previous selection of candidates for a similar job description using Recommender Systems is a part of such an intelligent system. The accuracy of the system was calculated by using Normalized Discounted Cumulative Gain accuracy metric. The significance of this project is to bridge the gap and the challenges faced in the recruitment process.

Keywords—Resume Recommendation, Resume Similarity, Resume Ranking, Natural Language Processing, Machine Learning, Online Recruitment

I. INTRODUCTION

In the field of recruitment, ranking the candidate is one of the most critical aspects of the hiring process. The client requires the best candidate and hence the best candidate should be ranked at the top. Also, every company has its own requirements pertaining to a particular job, so the best candidate for each client will be different. The current recruitment process is very tedious and time consuming. Nowadays a large number of candidates apply for the same job position in the same company. So, it's a very challenging task for company recruiters to go through each and every resume. The HR team requires more manpower to scrutinize the resumes of the candidates. So that motivated us to build an intelligent system that is more flexible, automated and time saving. We have to take into account various factors that would find the most suitable candidate for the company. So with Company to

Candidate mapper, we are proposing a model where we rank candidates based on a hybrid recommender system and map those candidates to the company for closing the jobs.

II. LITERATURE SURVEY

The rapid development of the online recruiting industry has resulted in an explosive increase in the number of resumes and business positions on the recruiting website. Businesses always have to spend a lot of time selecting resumes. As a result, the recruiting efficiency has decreased. The authors of [1] have implemented an efficient automatic user demand modeling algorithm with the help of genetic algorithms that match jobs and resumes. In [2], this paper proposed an automated system to extract required information from the unstructured resumes and convert them into structured format whereas in [7], a resume parser was built that parsed the unstructured resumes and transformed them into structured JavaScript Object Notation(JSON) format. The resumes were ranked with the help of NLP and ML based on the job description of the company. On the other hand, Smart Talents Recruiter [3] and EXPERT [9], attempt to develop a system based on Ontology to compare resume models against given job description to match the best candidates. Both of these models focused on screening of candidates using Ontology. The authors of [4] developed a ranking system based on the hiring pattern. The proposed system predicts ranking and sorting of resumes which simplifies the work of the Human Resource (HR) department. An Automated Resume Evaluation System using NLP was developed by the following paper [5] that divided the entire resume into three segments. The first segment has converted the unstructured resumes into structured format and the second segment extracted valuable entities and an identifier value was given. In the final segment, resumes were ranked based on the values assigned. In [6], a system was developed that automatically extracted entities using the spaCy Named Entity Recognition (NER) model from the resumes. A graph was generated to display scores of each and every resume. This helped recruiters to choose

the best candidates from the piles of resumes. The authors of [8] proposed a new method of automatic Curriculum Vitae (CV) segmentation and parsing. The CV segmentation has been done into three blocks Basic Information, Education and Work Experience. Job Post and Resume Classification (JRC) system [10], carried out the classification task by exploiting an integrated knowledge base. Unlike standard systems that plan to search globally within the entire house of resumes and job posts, JRC matches the resumes as per the given job description. The authors of [11] created a system that used different classifiers to classify all resumes according to the given job profile. The top candidates are ranked using Content Based Recommendation.

III. METHODOLOGY

The Aim of the project is to find the best suitable match for a candidate against the company's job description in a given time frame. To reach this desired goal the proposed work model will be i) Reducing Human Intervention and automating screening of candidates. ii) Applying Candidate Scoring Algorithm to find proficient candidates from a pool of candidates. iii) The System fine tunes the search based on the choices of the previous candidates saving massive amounts of time and effort. The entire system is divided into three modules as follows :

- Recruiter Mapping
- Resume Parser
- Candidate Ranking Algorithm

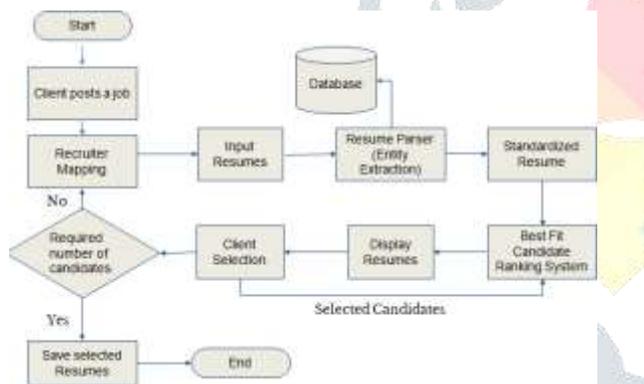


Fig. 1. Company To Candidate Mapper Architecture.

Dataset Description : The data used to build and train the models for this system was around 30000 resumes. These resumes were in different file formats of .docx, .doc and .pdf. Each of the resumes had their own structure, making the whole dataset in an unstructured format. Even though the resumes are unstructured they have some similar sections namely personal section, work experience section, education section, achievement section, certification section, projects section and summary section. So to collectively create a structured format, this resume data was then converted to .pdf file format first and then to .jpeg format and that was used to annotate different resume sections mentioned before in Amazon Web Services(AWS) SageMaker and DataTurks. The annotated

resumes were then saved into a COCO dataset format which has details about the image including the filename,height,width and image id, for all image the annotations will be saved which contain the area, bounding box information and the class category.

A. Recruiter Mapping

A company will have a number of requirements to fill for which they have to hire best candidates in a specific period of time. The type of requirements can be :

- Backfills (For company members that resigned.)
- RampUps (New positions)

Once the Requirements of the company are decided with the joining date of the job, they will need some number of resumes from recruiters to assess the candidates and give them job offers if they fit in the system as per the job description. Recruiter mapping algorithm is designed to map the best recruiters who can close the requirements in a minimum amount of time and also focuses on giving a fair share to all the recruiters signed up with the system.

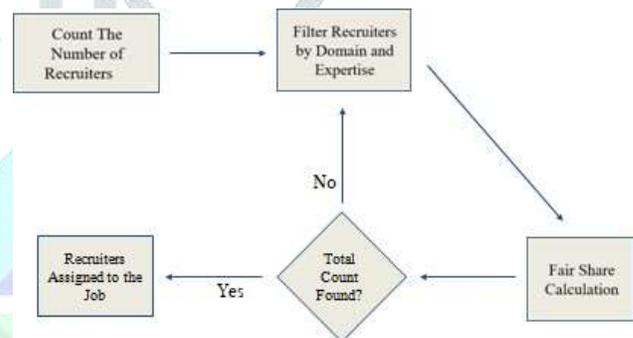


Fig. 2. Recruiter Mapping Flow.

To assign recruiters to a given company, the first part will be to calculate the number of recruiters required. Once the number of recruiters required are calculated, the algorithm will be ranking recruiters from the database as per their domain and expertise.

For creating a fair share, recruiters ranked will be placed in 5 different priority classes as per the expertise in the form $Priority\ Class1 > Class2 > Class3 > Class4 > Class5$. If all the recruiters of Class1 are occupied then we move to Class2. We keep a count of the number of jobs assigned to each recruiter where maximum 3 jobs can be assigned to a recruiter. So, if a recruiter has 3 jobs, others get a chance and that recruiter will be ignored until he/she is back to less than 3 jobs. For this, we maintain a status for recruiters namely Accepted, Rejected, Offered, Ignored. If a recruiter ignores a job once, he/she is degraded to a lower class. Eg. Class1 recruiters will be shifted to Class2 and if he/she ignores twice he will be removed from priority queue and added to the rejected list.

Finally, once the pool of recruiters is formed, all these recruiters will be sent a request to accept the job, if some recruiters reject a job and the required recruiter count is not satisfied, the algorithm will re run itself until the count is satisfied. Once, the required number of recruiters are mapped, they can submit resumes to the system which would then be used to rank the best candidates.

B. Resume Parser

A resume parser is a compiler or interpreter that converts the unstructured form of data into a structured form. It is a component that automatically segregates the information into various fields and parameters like personal information, educational qualification, work experience, skills, achievements, projects, certifications, etc. The resume parser is actually a pre-step for the Candidate Ranking Algorithm as the data required is processed and obtained in a structured format. The parser encompasses the use of LibreOffice for the given input resumes so as to convert them from various file formats to PDF. These are then subjected to Object Detection using RetinNet as backend and resnet18 as encoder. The sections of the resumes, for example, work experience details, educational qualification, skills and competencies etc are object detected and obtained in JPEG format. Now, from these images, text is extracted using PyTesseract thereby, performing Optical Character Recognition on them. Once the text for each entity is extracted, a trained NER model is applied to extract actual data in the entities. The data is then stored in the database which is now structured and serves as an input to the Candidate Ranking Algorithm and the parameters for the same will be discussed in section C. The parsing of resumes is required once the recruiters submit resumes to the system who are mapped for the job opening posted by the client.

C. Best Fit Candidate Ranking

This algorithm is the heart of the whole system. In the field of recruitment, ranking the candidate is one of the most critical aspects of recruitment. The client requires the best candidate hence the best candidate should be ranked at the top. Also, every company is different so the best candidate for each client will be different. On the other side, it is also difficult to go through a large set of candidates and select those that best match the requirements of the company. As a resume comprises various sections, it becomes a tedious process to manually filter out the best candidates based on their information present on the resume. Hence we have to take into account various factors besides the performance of the candidate.

Just as skills or experience is not enough to filter and provide the best candidates, it is required to consider various parameters along with these such that we have a cumulation of other important parameters from the data obtained through the parsed resumes. This will in turn solve the problem of using manual filters to find the candidates.

If the Client requirements are not met i.e. if the Client needs more candidates then the data of the previously selected

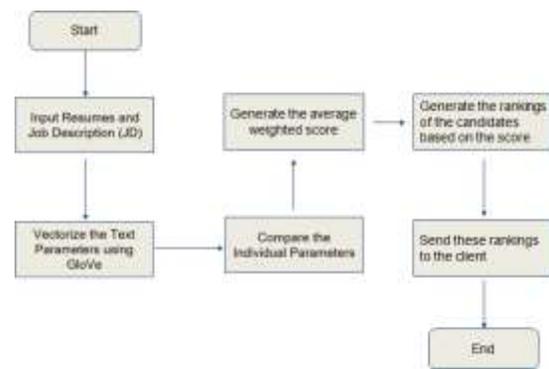


Fig. 3. Best Fit Algorithm Architecture.

candidates is also considered to rank all the other candidates in the database. This will help in providing similar candidates to the previously selected candidates along with the job description. The flow of the ranking algorithm is given in Fig. 2.

Parameters Considered with their significance:

- (i) Skills : Skills have the most weightage and are an important factor that decides the capability of a candidate for a job. Skills can be divided into Technical, Soft Skills and Behavioural Skills. Each of these have their own significance and helps to identify the candidate if he/she is fit for the company environment and the job.
- (ii) Years of Experience : Every Company requires candidates that have a minimum experience required for that job role and hence, it serves as the second most valuable parameter while ranking. The experience in the field will signify how much knowledge the candidate actually has. It goes hand in hand with the Work information to confirm that the knowledge gained in those years is relevant or not.
- (iii) Educational Qualification : It is a benchmark set for a given job opening. By understanding the qualification, it is made sure that the candidate at least has some amount of knowledge as per the degree.
- (iv) Job Title : If a candidate has worked under a similar job title, it adds on weightage to the candidate score proving direct suitability for the job.
- (v) Work Information : The previous job experience of the candidate has information about the work done and responsibilities undertaken which can be easily compared with the roles and responsibilities mentioned in the job description such that a better score would be generated for a candidate already having that experience.

The weighted average score is calculated based on the significance of parameters discussed by HR professionals and are given as follows inside brackets: Technical skills (5), Non-Technical skills comprising of both behavioural and soft skills (4), Years of Experience (4), Qualification (3), Job Title (3) and Work Information (1).

The parameter Years of Experience is numeric and Qualification is encoded from text to numeric using a value assigned dictionary for all the available qualifications. The score for these two parameters is considered by error of difference method whereas for the remaining parameters which are text-based, vectorization is necessary to convert them into vector format such that their contextual occurrence meaning is not lost. The vectorization is carried out using GloVe i.e. Global Vectors for Word Representation. GloVe enforces the word vectors to capture sub-linear relationships in vector spaces. It is a count-based model and relies on the word co-occurrences thereby working well on word analogy tasks as well as similarity measurement. It incorporates global statistics to obtain word vectors. GloVe proved to be faster in representing vector representation of large amounts of text than Word2vec, ELMo and BERT as a result of which it was incorporated into the system.

The similarity between the vector representation of textual data present in the resumes and the job description is calculated using Cosine Similarity. Cosine Similarity is a similarity measure that finds the cosine angle of the two vectors at a time. The vectors that are similar to each other have a higher similarity score tending towards 1 whereas the vectors that are dissimilar to each other have a lesser similarity score tending towards 0. It makes it possible for a longer document of texts to be compared with a short document of texts. The benefit being that it would not skew the results.

Once the similarity score is calculated, total score is found out using the weighted average method as discussed earlier. Similarly for each resume, scores will be generated and finally sorted on the basis of the scores ranked from highest to lowest match for the job which would be then displayed to the client. If the client selected the candidates and job openings are still open, the new set of resumes would be compared with both the Job description as well as the previously selected resumes such that the user preference is considered and used in the ranking.

IV. RESULTS AND ANALYSIS

The accuracy of the ranked candidates is measured using Normalized Discounted Cumulative Gain (NDCG) Ranking. The NDCG ranking metric will value the items high up in the ranking list thereby keeping in mind the relevant positions of the items. The rankings generated by the system are then compared based on the rankings given by the Human Resources(HR) representatives in the NDCG ranking metric. Equation (1) shows the NDCG ranking formula where DCG_p is the Calculated Discounted Cumulative Gain of the ranked list generated by the system and $IDCG_p$ is Ideal Discounted Cumulative Gain given by the HR professionals.

$$nNDCG_p = \frac{DCG_p}{IDCG_p} \quad (1)$$

The system was tested for a job description created by the client with a pool of 11 resumes submitted by the recruiters. The ranked list by the system and HR representatives created an ideal rankings for that given pool. The accuracy generated by the NDCG metric was 75.2%. After this, the client selected 2 resumes and the ranking list was regenerated by using recommender systems. The ideal ranking for the new list was generated by the HR representatives again. The accuracy was found to be improved to 79.1%. So, the system learns the client fingerprint and improves the accuracy for the ranking list.

Hence as a machine learning problem, the better the data the accurate the rankings will be which is an advantage of our system as we consider parameters separately and then find the weighted average score for each resume against the job description.

V. CONCLUSION AND FUTURE WORK

The most critical part of this intelligent system is the ranking algorithm. The candidate pool sent by the recruiters will be in the form of .pdf, .docx, or other file extension based resumes. These resumes will be then parsed and sent as an input with the job description hand in hand to the candidate ranking algorithm.

For ranking, a weighted average scoring based and Content Based Filtering method that compared contents of each resume to the job description was used. For text based parameters, a pre-trained GloVe model was used for vectorization and these vectors were then compared using cosine similarity metric. Error of difference method was used for other encoded parameters. The ranking accuracy was checked using NDCG i.e. Normalized Discounted Cumulative Gain method. Collaborative Filtering method is used to recommend resumes based on the previous selection of candidates.

The Front End or User Interface(UI) is developed using MERN stack. The backend is made up using Application Programming Interfaces(APIs) in the FLASK framework of python programming language. With that, all the APIs are deployed in Amazon Web Services(AWS) Lambda service.

The future scope of the project can be divided into two levels. The first level is improvement in the candidate ranking system by adding automated assessments. These assessments can be of multiple forms like a bot taking virtual interviews on call or on video, automated psychometric tests, coding tests and aptitude training tests. Once this level is improved we can move on to the next level. In this level we will try to build more management tools for the clients, such as all the HR related tasks after recruitment can be done using the platform. Candidate Ranking method can also be used in higher universities to admit candidates for Specialization courses.

ACKNOWLEDGMENT

We take the opportunity to thank all those people who have helped and guided us through this project and make this experience worthwhile for us. We wish to sincerely thank our

company LineupX, Mumbai for giving us this opportunity for making an outhouse project in the Final Year of Engineering. We would also like to thank Prof. Dakshata Panchal for guiding us throughout the project and encouraging us to explore this domain.

REFERENCES

- [1] Z. Wang, X. Tang and D. Chen, "A Resume Recommendation Model for Online Recruitment," 2015 11th International Conference on Semantics, Knowledge and Grids (SKG), Beijing, 2015, pp. 256-259, doi: 10.1109/SKG.2015.31.
- [2] Sayed Zainul Abideen Mohd Sadiq, Juneja Afzal Ayub, Gunduka Rakesh Narsayya, Momin Adnan Ayyas, Prof. Khan Tabrez Mohd. Tahir, "Intelligent Hiring with Resume Parser and Ranking using Natural Language Processing and Machine Learning," IRJET Volume 4 Issue 4, April, 2016.
- [3] A. Mohamed, W. Bagawathinathan, U. Iqbal, S. Shamrath and A. Jayakody, "Smart Talents Recruiter - Resume Ranking and Recommendation System," 2018 IEEE International Conference on Information and Automation for Sustainability (ICIAFS), Colombo, Sri Lanka, 2018, pp. 1-5, doi: 10.1109/ICIAFS.2018.8913392.
- [4] A. Kumar, A. Pandey and S. Kaushik, "Machine Learning Methods for Solving Complex Ranking and Sorting Issues in Human Resourcing," 2017 IEEE 7th International Advance Computing Conference (IACC), Hyderabad, 2017, pp. 43-47, doi: 10.1109/IACC.2017.0024.
- [5] R. Nimbekar, Y. Patil, R. Prabhu and S. Mulla, "Automated Resume Evaluation System using NLP," 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, 2019, pp. 1-4, doi: 10.1109/ICAC347590.2019.9036842.
- [6] Dr. K.Satheesh, A.Jahnavi, L. Iswarya, K.Ayesha, G. Bhanusekhar, K.Hanisha, "Resume Ranking based on Job Description using SpaCy NER model", IRJET Volume 7 Issue 5, May 2020.
- [7] Sanyal, Satyaki Hazra, Souvik Ghosh, Neelanjana Adhikary, Soumyashree, "Resume Parser with Natural Language Processing", 2017, 10.13140/RG.2.2.11709.05607.
- [8] M. Tikhonova and A. Gavrishchuk, "NLP methods for automatic candidate's CV segmentation," 2019 International Conference on Engineering and Telecommunication (EnT), Dolgoprudny, Russia, 2019, pp. 1-5, doi: 10.1109/EnT47717.2019.9030535.
- [9] V. Senthil kumaran Annamalai, Sankar, "Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping EXPERT", 2013, International Journal of Metadata, Semantics and Ontologies. 8. 56-64. 10.1504/IJMSO.2013.054184.
- [10] A. Zaroor, M. Maree and M. Sabha, "JRC: A Job Post and Resume Classification System for Online Recruitment," 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), Boston, MA, USA, 2017, pp. 780-787, doi: 10.1109/ICTAI.2017.00123.
- [11] Roy, Pradeep Chowdhary, Sarabjeet Bhatia, Rocky, "A Machine Learning approach for automation of Resume Recommendation system. Procedia Computer Science", 2020, 167. 2318-2327. 10.1016/j.procs.2020.03.284.

