Journal of Computer Science

Original Research Paper

A Recommendation-Based Contextual Model for Talent Acquisition

[1]Channabasamma A. and [2]Yeresime Suresh

[1]*Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India*
[2]*Department of Computer Science and Engineering, Ballari Institute of Technology and Management, Ballari, India*

Article history
Received: 04-03-2022
Revised: 21-06-2022
Accepted: 23-06-2022

Corresponding Author:
Channabasamma A.
Department of Computer Science and Engineering,
Gokaraju Rangaraju Institute of Engineering and Technology,
Hyderabad, India
Email: channu.ar@gmail.com

Abstract: It is important to assist the job seekers in selecting the perfect jobs, which suit candidates' current skills and career objectives. Given the job description and resumes in the unstructured form, choosing the best job manually is a tedious task, so there is a need for an automated system to deal with raw data. The extraction of structured information from applicant resumes is needed not only to support the automatic screening of candidates but also to efficiently route candidates to the corresponding occupational categories based on their respective skills. The primary objective of this article is to process and extract relevant information from the unstructured data, like resumes in the form of .pdf, .doc, using natural language processing. This study also proposes machine learning algorithms that exploit user context information to shortlist for the desired job role and also recommends alternative jobs to the candidates. Based on existing skills, new opportunities and possibilities will be introduced, which the candidate wouldn't have explored before. In addition, it also focuses on formalizing the problem of identifying the additional skills, taking into account the employee's existing skills. To exhibit the effectiveness of the proposed algorithms, various resumes have been passed and tried for different formats. The results obtained by the proposed method excel the traditional methods mathematically and practically.

Keywords: Talent Acquisition, Recommender System, Natural Language Processing, Eligible Candidates, Potential Candidates and Latent Candidates

Introduction

People can find jobs online using many websites such as Monster and Indeed.com. These web platforms have been providing services for more than a decade; helping job seekers and organizations by saving a lot of money and time. To automate the workflow, the candidate has to enter long forms about different entities though most of the information is already present in the candidate's resume. Long forms demotivate users and may introduce delay and also there is a tendency of skipping non-mandatory fields even though the information is available.

Traditional approaches to information retrieval may not be appropriate for users. The reason being a greater number of results returned to a job seeker requires a lot of time to analyze options. Usually, many potential resumes may be excluded from the search results or may not be examined due to the complexity of the database and stringent timelines. However, most people don't identify what is needed until it is shown. People may love a product, a movie, job opening- but may not be known that exists. Further, automatic parsing of resume documents is very challenging as resumes may contain partial sentences or full; may differ in the type of information, and order; conversion from different formats like.pdf, .doc, and .docx to text yields unexpected formats of information (Aluvalu and Jabbar, 2018; Deepa and Rajeswara, 2020).

To automate the recruitment process, the system should be independent of the order and type of information in the documents to parse effectively and efficiently. Taking into account the above-mentioned challenges, the primary objective of this article is to process and extract relevant information from the unstructured data, like resumes in the form of .pdf, .doc, etc., using natural language processing and shortlisting for the desired job role. This study also recommends alternative jobs to the candidates based on the skills to introduce new opportunities and possibilities, which the candidate wouldn't have explored before. Recommender systems are broadly recognized in several fields to suggest services, products, and information to latent customers (Faqihi *et al.*, 2020; Le *et al.*, 2022; Boorugu and Ramesh, 2020; Ramesh and Madhavi, 2019). A recommendation is an important aspect of an organization's workforce for creating and maintaining the right skilled profiles. Proposed job recommendation is primarily aimed at supporting the discovery of jobs that may interest the user as it automatically returns suitable and prospective jobs.

Literature Survey

The recommendation system is an eminent research area, which includes a lot of applications in the field of e-banking, e-health, and e-commerce (Aggarwal, 2016; Bobadilla *et al.*, 2013; Xavier Amatriain *et al.*, 2011; Shahab, 2019). Especially, in the field of e-recruiting, the role of a recommendation system is imperative (Kantipudi *et al.*, 2021). This section outlines the contributions of various researchers towards matching candidates with respective jobs.

Srivastava *et al*. (2018) focused on algorithms to provide training recommendations in an industrial setup. By keeping a record of the employee's training details and work experience, the authors formalized the problem of the next training recommendation. Past training data has been mined using several unsupervised algorithms to generate the next training recommendations. Though these algorithms beat several recommendation algorithms and also sequence mining algorithms, still there is a scope to enhance these algorithms to predict a sequence of training required for an employee's long-term career (Srivastava *et al.*, 2018).

Kumar *et al*. (2017) designed a method to explore the hiring pattern and intelligence behind it. To accommodate the known intelligence, machine learning methods are applied. This method offered a ranking system based on hiring patterns. Highly trained models were used to predict the ranking and sorting of resumes (Kumar *et al.*, 2017).

Belsare and Deshmukh (2018) presented the employment recommendation system, which uses various recommendation methods. Which are simple matching, collaborative and content-based filtering Belsare and Deshmukh (2018). Personalized recommendations can be generated using both content-based recommendation and collaborative filtering, hence these are more suitable for the job aspirants; however, collaborative filtering suffers from a cold start problem whereas with content-based recommendation too precise results might be created.

The main bottleneck in generating information retrieval systems is to stabilize the available expertise of the text analysis methods and the capability to evaluate the model for quick interpretation. Prokhorov and Safronov (2019) provided an outline of natural language processing and machine learning approaches, which are very much essential to modern information systems to search and filter text documents (Prokhorov and Safronov, 2019; Deepa and Rajeswara, 2017).

Rodriguez and Chavez (2019) identified the required key attributes of profile in the process of job matching. This study implemented a prominent feature selection approach that extracts the most cogent attributes required for job matching.

Conventional recommender systems focused on either finding the best match between a candidate's preferences and the job description(content-based) or finding candidates with the same desires (collaborative filtering) (Ricci *et al.*, 2015; Jalili *et al.*, 2018). However, information regarding the user environment (contextual knowledge) can be used to amend the candidate's initial desires (Adomavicius and Tuzhilin, 2011; Lappas, 2020; Raj *et al.*, 2022). Several candidates prefer jobs closer to their skills, but some candidates seek jobs far beyond their skills. The recommendations for improving their lacking skills will help them in achieving their dream job.

Materials and Methods

This article proposes two algorithms viz, JOB_DJ and JOB_Alternate for shortlisting a resume for the desired job role and alternate job recommendations respectively. It also identifies additional skills required by other candidates; keeping it as side information, the designed method not only improves the procedure of resume shortlisting but also can perform recommendations for new experts or skills.

The Basic Idea

Generally, it can be observed that most of the skills are commonly present in every resume. An individual resume may include special distinct skills which differentiate it from other resumes in the set. The basic idea here is that identification of special skills from each resume will reduce the time required to select an appropriate resume when compared to the time required to look for complete information.

Skill-Based Talent Acquisition

Talent acquisition is an approach to finding specialists for a position that requires a very specific skillset. This article explores the skill section of the candidate's resume and develops a strategy to identify potential resumes with special skills. This approach uses a blend of Natural Language Processing (NLP) and other sub-recommendations to suggest jobs to candidates. In our previous work, using an advanced library of NLP, Spa Cy, an attempt has been made to capture candidates' special skills required for data scientist job selection (Suresh and Manusha Reddy, 2021). In this article, the work is extended to focus on shortlisting candidates for any desired jobs and generating alternative job recommendations that the candidate is most likely to select or interact with. The dataset includes seven resumes, which are in different formats and layouts.

Organization of Special Skills Knowledge

The main bottleneck is to measure the quality of content in the resume's skills section. This study extends the concept of special skills to develop a technique for selecting appropriate resumes. Each part of the skill information can be organized into "skill_type" and their corresponding "skill_values". For example, 'programming-languages' is a skill_type, and 'Java, Python' is skill_values. From the resume's skills information, two types of skillsets can be extracted using NLP. The first is the Skill-type-Desired Jobst (SDJ) and the second is Skill-type-Alternate Jobs (SAJb).

Approach to Build SDJ and SAJb

The approach to building data frame SDJ includes several steps: First, Information from the documents can be extracted using one of the Python libraries, PyPDF2, it includes extracting the text information, cleaning it, and then exporting it easily readable text files. Next, to program the model to process and analyze such a huge amount of data, NLP is used. The NLP's Spa Cy library has the Matcher tool, which can be used to specify custom rules for phrase matching. The process of the matcher tool involves: First, defining the required patterns, it involves creating a dictionary-1 with the set of skill types required for the desired job role. Next, these patterns are added to the Matcher tool and finally, the Matcher tool parses the resume documents to extract the skill values for the skill types mentioned in the dictionary-1. Once the phrases are found, the count of the respective skill type in the data frame is updated. It is important to assist the job seekers in selecting other alternative jobs, which suit with his/her current skills and career desires. Dictionary-2, skill types required for other alternative jobs are constructed. Considering the dictionary-2, SAJb is constructed; the approach of constructing SAJb is exactly similar to the SDJ.

Organization of Candidates Based on Skills

After extracting the count of skill values to all skill types in the data-sets, the next part of the process is to well organize the candidates' resumes. Based on the count of the skill values, candidates can be categorized into one of the three categories: Eligible candidates, who have perfect skills required for the desired job role. Potential candidates, who have near potential skills. Latent candidates, other candidates who are not skilled enough and may require additional skills.

Setting Skills Threshold Value

The count of skill values of SDJ is compared against the threshold, T_{SDJ} in function JOB_DJ. It helps to cluster the candidates into eligible candidates, potential candidates, and latent candidates. The objective of clustering the candidates is to improve the efficiency of HR analytics in meeting the need for talent acquisition. T_{SDJ} could be chosen effectively by considering the number of skills to eliminate candidates with common skills. The value of the threshold can be increased gradually to analyze the total number of eligible candidates' resumes. If the number of eligible candidates is shown to be decreased, the threshold value can be reduced and monitored. From the experiments, it is noticed that by decreasing the threshold value, the number of candidates with common skills would increase. Another Threshold, T_{SAJb}, is used much similar to T_{SDJ}, to recommend alternative jobs in function JOB_Alternate.

Operation of the JOB_DJ and JOB_Alternate

This article proposes an algorithm JOB_DJ to better serve the latent candidates who miss an opportunity due to the lack of a few required skills. The major goals of the algorithm are: First shortlisting the candidates for the Desired job role. Second, based on existing (near potential) skills, candidates will be recommended for alternative job positions. Third, identifying and providing recommendations to improve their lacking skills to apply for the desired job role.

It takes SDJ as input, the count of each skill type in the SDJ constitutes a major part; which is compared against the threshold, T_{SDJ}. Due to a lack of few 'near potential skills', the candidates shouldn't lose opportunities. With this objective, the JOB_DJ incorporates another function JOB_Alternate. It attempts to identify the skills that potential candidates possess and maps with the skills required for alternative job roles. This function takes SAJb as input. The values of this data frame are compared against another threshold, T_{SAJb} to recommend alternate jobs based on existing skills. Furthermore, the approach is also extended to identify the additional skills required by latent candidates. Thereby recommendations can be generated on these skills to improve profiles. Let *Ci* be an individual candidate and Si be an individual skill type in SDJ/SAJb. The detailed steps involved are shown in JOB_DJ and JOB_Alternate:

```
Sub program function of JOB_Alternate {
for each Ci in SAJb
        Li = []
        for each Si in SAJb
                count2←SAJb[Ci , Si ]
                if (count2>TSAJb)
                        Li←append si if
```

```
    if (len (Li>0))
            AltJb←append Li return
            AltJb /* Holds the list of alternative
            jobs to be recommended for potential
            candidates. */
    else

            Write (No alt job)
Return
}
```

Sub program function of JOB_DJ {

```
for each Ci in SDJ
Li= []    //holds the list of skill types
          for each Si in SDJ count←
          SDJ [Ci, Si] if(count<TSDJ)
                    Li←append Si // identifying
                lacking skills
          if (len(Li)==0)
                Write (Shortlisted: Yes) else
                 Write (Shortlisted: No)
                 Write (Feedback: There is a scope to
          improvement in Li)
                AltLi←append Ci /*
                Holds list of candidates not shortlisted
          for desired job role. Which can be further
          considered for recommending alternate jobs. */
if (ci = = AltLi) // Not shortlisted and looking for
alternative job positions
            AltJb =JOB_Alternate()
            Write (Recommended for Alternative jobs,
            Altjb)
else
            Write (Not applicable because you are
shortlisted for desired job role: Congratulations)
}
```

Case Study

To gather insights from industry experts on IT demand, a survey has been made. Survey questionaries include Q1. As per your knowledge, what is the most required factor to be considered for an IT job? Q2. Do you feel the data scientist job is the highest-paid job in the IT industry? Q3. Can you please list out a few other highest-paying jobs? Q4. To what extent do you feel the following skills (Table 1) are most required for the data scientist role, please rate the following: (1: Low, 10: High). As per the survey responses it has been observed that special skills are the most contributing factors to IT jobs and the highest paid job is the data scientist.

A data scientist is a proficient person, who knows how to discover meaningful information from raw data. To become a data scientist, one should possess the right set of skills related to various underlying fields of statistics and computer science. However, the list of skills mentioned in the Table 1 got the highest score from survey responses. Henceforth, to validate the proposed algorithms, this article focused on extracting the special skills required for the data scientist role (Desired job role) as a case study.

As per industry experts' opinions, besides the data scientist role, many other alternative jobs are also in demand nowadays. So, it is important to assist the job seekers in selecting the perfect jobs, which suit with his/her current skills and career desires. Focusing on a few, for example, software developers, web developers, system engineers, quality assurance engineers, power programmers, etc., dictionary-2 (Table 2) is constructed. It includes a set of skill types required for other alternative jobs (respective skills required for these alternative jobs also gathered from industry experts).

Required sample patterns for data scientist job roles are defined in dictionary-1 and patterns for other alternative jobs are defined in dictionary 2 (Table 1 and 2 respectively). Results of a count of each skill type for the data scientist role are plotted using Matplotlib as shown in Fig. 1. The graph has been plotted by considering the count of skill values against X- the axis and candidates against the Y-axis.

The detailed roadmap of how JOB_DJ and JOB_Alternate are applied to the data scientist job role and other alternative job roles is shown in Fig. 2. It includes several steps, the steps followed are:

1. The knowledge base, Dictionary-1 is created, which includes the special skill set required for the data scientist role (Table 1)
2. Another knowledge base, Dictionary-2 is created, which includes the other alternative job roles and respective skill types required. It can be considered for generating alternative job recommendations (Table 2)
3. Resumes are parsed by Spa Cy, an advanced library of NLP
4. The NLP Spa Cy library has the Matcher tool, which parses the whole resume to match the respective phrases in the Dictionary-1
5. For each of the phrases found in the data frame, the count of the respective skill type in the SDJ will be updated
6. Once the SDJ is built, JOB_DJ is used to match candidates' skills with skill types of data scientist job role by comparing with the threshold, TSDJ
7. If the candidate's skill values meet the threshold value, TSDJ, the candidate is assumed to be eligible and gets shortlisted for the data scientist role. Else it identifies the lacking skills and will be recommended to the candidate so that the candidate can improve
8. Step 3 will be repeated to look for the phrases in the dictionary-2
9. For each of the phrases found, the count of the respective skill type in the SAJb will be updated
10. Once the SAJb is built, JOB_Alternate is used to match candidates' skills with skill types of alternate jobs specified in the dictionary-2

DOI: 10.3844/jcssp.2022.612.621

11. If the candidate's skill values meet the threshold value, TSAjb, the candidate is assumed to be potential and alternative jobs will be recommended
12. Else, in addition to this, the model also identifies the other candidates who don't possess the required skills; recommendations can be generated to improve on these skills

Table 1: Dictionary-1: Skill set for SDJ (Data scientist)

Statistics	Machine learning	Deep learning	R language	Python language	NLP	Data engineering
Statistics statistical	Linear regression	Neural network	R	PYTHON	Nlp	Aws
Models	Logistic regression	Keras theano	Ggplot	Flask django	Topic	ec2
Statistical	K means random	Face detection	Shiny	Pandas	Modeling	Amazon
Modeling	Forest	Neural network	Cran dplyr	Numpy	Lda named	Redshift s3
Probability normal	XGBOOST Svm	Convolutional	Tidyr	Scikitlearn	Entity	Docker
Distribution	Naive bayes Pca	Neural network	Lubridate	Sklearn	Recognition	Kubernetes
Poisson	Decision trees	(cnn)	Knitr	Matplotlib	Pos tagging	Scala
Distribution	SVD ensemble	Recurrent		Scipy bokeh	Word2vec	Teradata
Survival models	Models boltzman	Neural		Statsmodel	Word	Google big
Hypothesis testing	Machine	Network (RNN)			Embedding	Query aws
Bayesian inference		Object			Spacy	Lambda aws
Factor analysis		Detextion yolo			Gensim nltk	Emr hive
Forecasting		Gpu cuda			Doc2vec	Hadoop
Markov chain		Tensorflow			Applications	Sql
		ISTM gan			Cbow bag od	
		Opencv			Words skip	
					Gram bert	
					Chat bot	

Table 2: Dictionary-2: Skill set for SDJ (Data scientist)

Software developer	System engineer	Power programmer	Quality assurance engineer	Android developer	Web developer
Machine learning	Assembly language	Scala	IOT	Java	HTML
and artificial	Programming	Akka	Block chain	Android SDk	CSS
Intelligence	Microsoft office	Play	RFID	Android	Analytical
Cloud computing	Oracle	Java/JEE	Artificial	Studio	Skills
Software testing	Java	Spring boot	Intelligence	XML	Back end
Docker and	SQL	Cloud foundry	Ranorex	Kotlin	basics
kubernete	Statistical methods	Docker	Test Plant eggplant	Object-oriented	Javascript
DevOps	GAMP	ReactJS	Robot Framework	oriented	Search
Python	Cisco networking	Angular JS	Watir	fundamentals	Engine
java	C++	Express JS	Unified Functional	SQL	Optimization
C	Algebra	Node.js	Testing (UFT)	Material	Jquery
C++	Computer networks	Mongo DB	Katalon Studio	Design	Graphic
PHP	GxP	Cassandra	Selenium	language	Design
C#	Linux	Hadoop HDFS		Python	Applications
				Node.js	
				.Net	

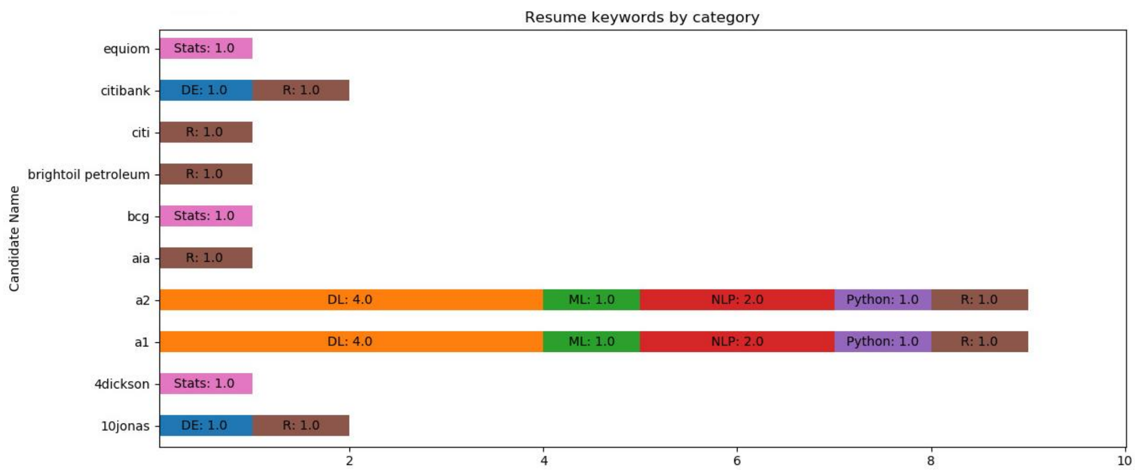


Fig. 1: SDJ (data scientist): The count of skill values for each skill type

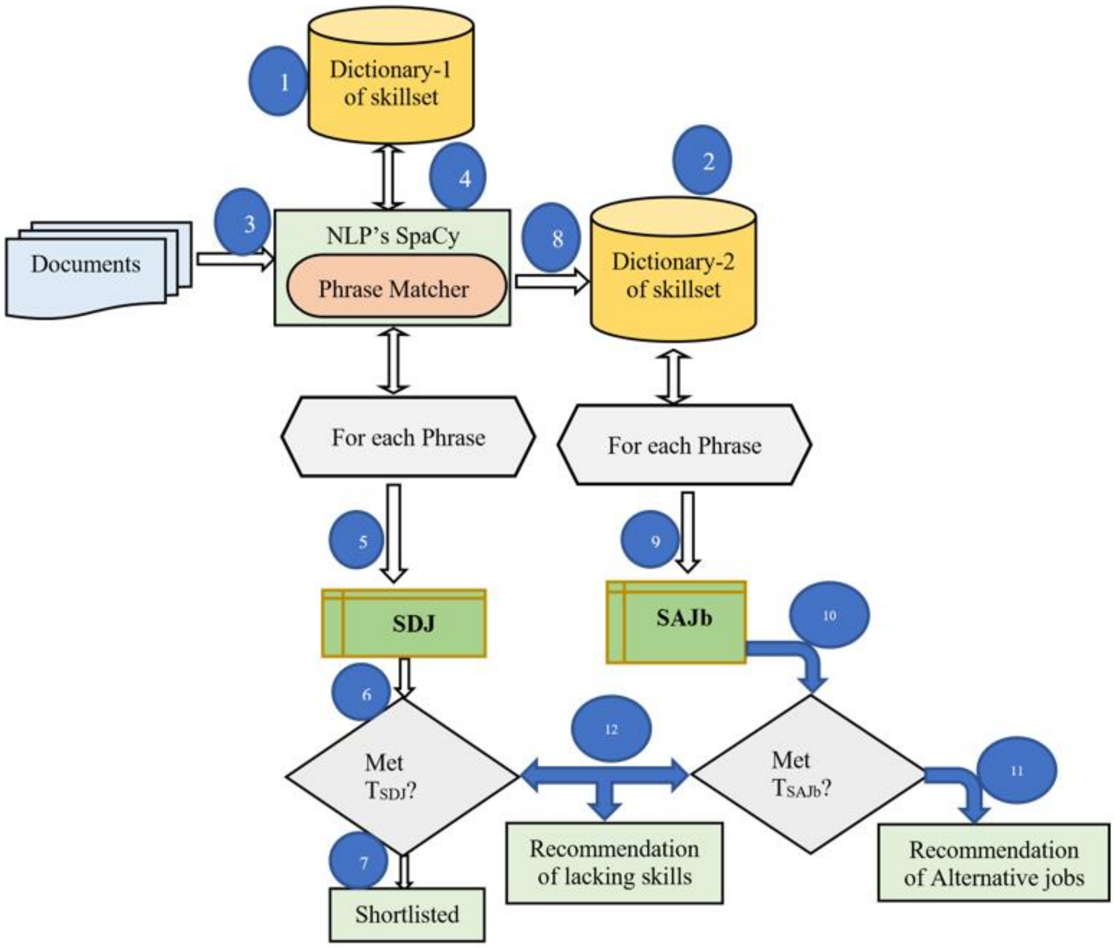


Fig. 2: High-level view of skill selection for resume shortlisting

Results and Discussion

On inputting SDJ and SAJb to JOB_DJ and JOB_Alternate respectively, another data frame (CSV) is obtained; which meets the objectives mentioned in the above algorithms. For effective visualization, Tableau is used by considering this CSV file as input. Tableau is one of the data-visualization tools, used to show the objectives of this study in an effective manner. The respective candidate's status can be viewed in Fig. 3.

In addition to shortlisting the candidates, the model also focuses on:

1. Based on the candidate's existing skills, alternative jobs can be recommended so that the potential candidate shouldn't lose an opportunity (Fig. 4)
2. Another main contribution of this article is to identify and recommend additional skills required by other latent candidates who are not skilled enough (Fig. 5). This helps candidates in improving their profile to meet career goals

A tree map chart has been used to describe all the candidates' details like their status, feedback, and alternative jobs (Fig. 6) Treemapping is a data visualization technique under Tableau that is used to display hierarchical data.

The primary objective of this article is to shortlist the candidates for the data scientist role. In this article, an objective is to eliminate candidates with common skills, therefore, the Threshold (TSDJ) value is set to two. Among seven input candidate resumes, it can be observed that only one candidate is known to be eligible and shortlisted for the data scientist role; one candidate is known to be potential, alternate jobs can be recommended; and for the other five latent candidates, feedback can be generated to improve upon the skills which candidate doesn't possess. The value of the threshold can be increased gradually to analyze the total number of eligible candidates' resumes.

DOI: 10.3844/jcssp.2022.612.621

Table 3: The performance of JOB_DJ and JOB_Alternate

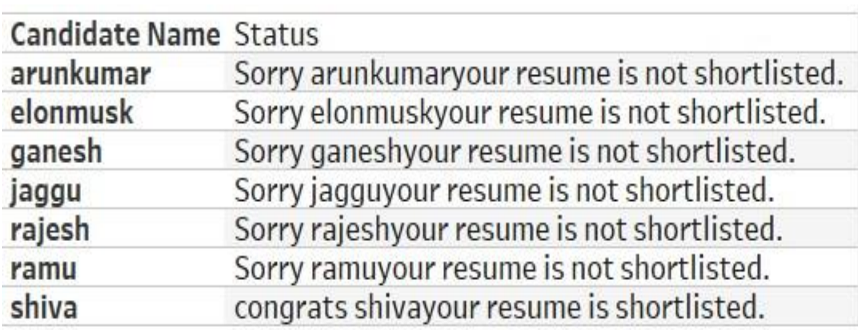
Input TSDJ and TSAJB		Algorithm	TP	FP	TN	FN	Accuracy
7	2	JOB_DJ	2	1	2	2	0.57
	2	JOB_Alternate					
30	2	JOB_DJ	8	7	6	9	0.46
	1	JOB_Alternate					
1096	1	JOB_DJ					
	1	JOB_Alternate	279	153	446	218	0.66

Table 4: The performance of different resume recommendation systems

System #	# Shortlisted resumes	Accuracy
System-1	51	51/67 = 0.761
System-2	53	53/67 = 0.791
System-3	48	48/67 = 0.716
Proposed system (Using JOB_DJ)	56	56/67 = 0.835

Table 5: Comparison of JOB_DJ with traditional systems

Criteria	JOB_DJ	Traditional systems
Unstructured data (.pdf, .jpeg, .doc etc.)	Yes	No
Utilization of existing skills	Yes	Partially
Feedback to improve lacking skills	Yes	No
Alternate job Recommendations	Yes	Yes



Candidate Name	Status
arunkumar	Sorry arunkumaryour resume is not shortlisted.
elonmusk	Sorry elonmuskyour resume is not shortlisted.
ganesh	Sorry ganeshyour resume is not shortlisted.
jaggu	Sorry jagguyour resume is not shortlisted.
rajesh	Sorry rajeshyour resume is not shortlisted.
ramu	Sorry ramuyour resume is not shortlisted.
shiva	congrats shivayour resume is shortlisted.

Fig. 3: Candidates' status

If the number of eligible candidates is shown to be decreased, the threshold value can be reduced and monitored. It can be noticed that by decreasing the value of the threshold, the candidates with common skills would increase. The value of the threshold can be gradually changed as per the requirement of Human Resources.

To exhibit the effectiveness of the proposed algorithms, various numbers of resumes have been passed and tried for a different levels of thresholds. The performance of both JOB_DJ and JOB_Alternate algorithms has been shown in Table 3. The performance of the model is evaluated by comparing the results obtained by JOB_DJ with the results of manual inspection. With the manual verification, it is confirmed that out of 100 candidates, only 67 candidates are having the minimum required skills to be shortlisted. To predict the accuracy, the same set of resumes was given as input to different resume recommendation systems (System-1, System-2, and System-3), and the prediction is recorded in Table 2. Figure 7 shows the graphical representation.

The performance of the proposed system using JOB_DJ is compared with the results of the most preferable recommender systems in the literature survey. The results obtained by the proposed method excel the traditional methods mathematically and practically (Table 3 to 5).

The article proposes algorithms that can be applied to any of the job roles given the dictionary of skillset for the respective jobs; taking one as the desired one and the other as alternative job roles. It focuses mainly on the extraction of information from resumes. The proposed system parses resumes effectively and efficiently as the system is independent of the order and format of the documents; thereby reducing the burden of entering long forms by the candidate.

DOI: 10.3844/jcssp.2022.612.621

Candidate Name	Alternative Job Recommendation
arunkumar	Sorry we didnot find any alternative jobs
elonmusk	Sorry we didnot find any alternative jobs
ganesh	Sorry we didnot find any alternative jobs
jaggu	Sorry we didnot find any alternative jobs
rajesh	['AndDev', 'PP', 'QA', 'SD', 'WebDev']
ramu	Sorry we didnot find any alternative jobs
shiva	Not applicable because your resume is shortlised for Data Science Role

Fig. 4: Recommendations for alternative jobs

Candidate Name	feedback
arunkumar	('There is scope of improvement in', ['DE', 'DL', 'ML', 'NLP', 'Python', 'R', 'Stats'])
elonmusk	('There is scope of improvement in', ['DE', 'DL', 'ML', 'NLP', 'Python', 'R', 'Stats'])
ganesh	('There is scope of improvement in', ['DE', 'DL', 'ML', 'NLP', 'Python', 'R', 'Stats'])
jaggu	('There is scope of improvement in', ['DE', 'DL', 'ML', 'NLP', 'Python', 'R', 'Stats'])
rajesh	('There is scope of improvement in', ['DL', 'ML', 'NLP', 'Python', 'R', 'Stats'])
ramu	('There is scope of improvement in', ['DE', 'DL', 'ML', 'NLP', 'Python', 'R', 'Stats'])
shiva	Your techinical skills are good . Wait for the further updates about interview

Fig. 5: Identifying and recommending required skills

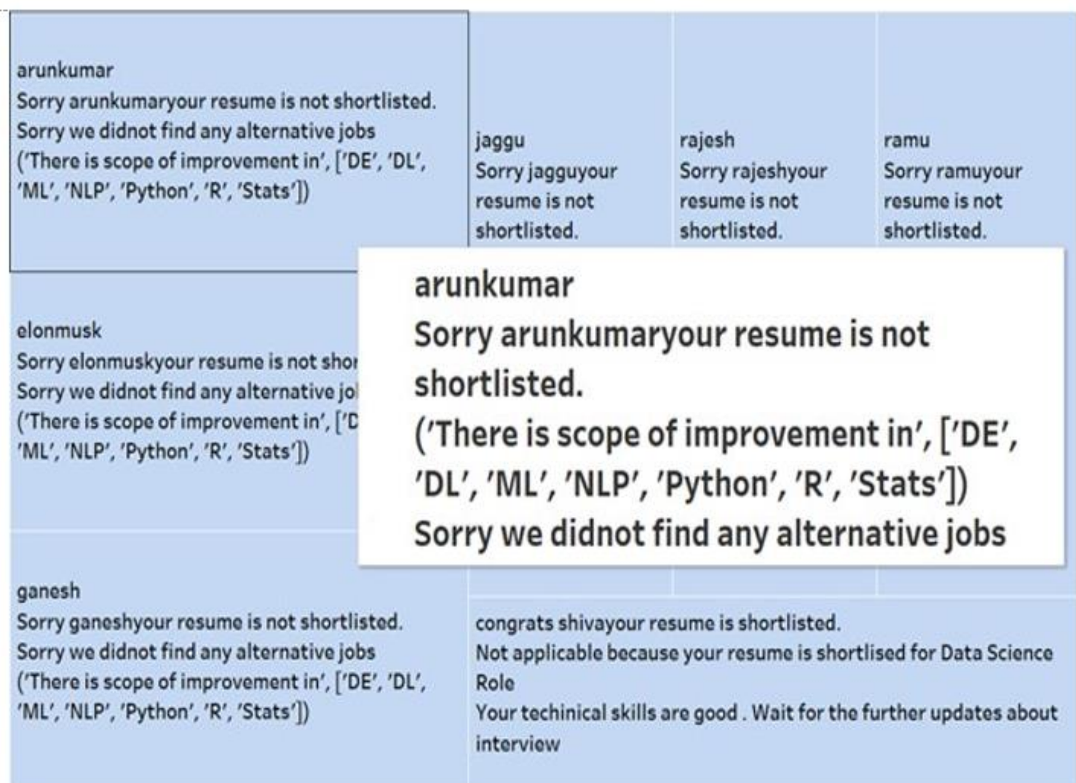


Fig. 6: Visualizing the individual candidate's summary

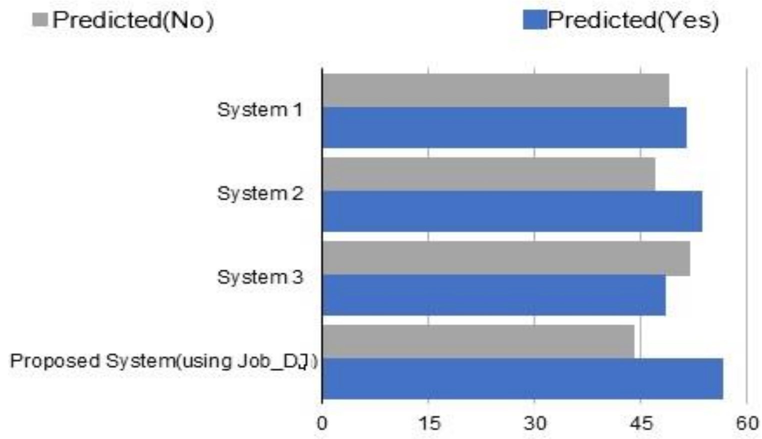


Fig. 7: Performance of different resume recommendation systems

Conclusion

In this study, two algorithms have been proposed with the objective of skill matching with the desired job role and alternate job recommendations by incorporating raw data (.pdf, .doc). Further, it incorporates the feedback to improve the algorithm performance along with identifying the lacking skills as side information. It has been proved that adding user contextual information in the JOB_Alternate algorithm (such as the skills that an employee currently possesses, to recommend alternative jobs) generally improves the prediction accuracy for talent acquisition. The proposed system parses resumes effectively and efficiently as the system is independent of the order and format of the documents; thereby reducing the burden of entering long forms by a candidate. To enhance the accuracy of the system's recommendation results, the shortlisting process can be further enhanced to drill down from the data scientist role to include sub-roles in the future. Further, it focuses on a two-way recommendation system. In this study, an approach was developed by exploring only the skill section to identify potential resumes. Future work can be extended to explore more contextual information about the employee in other sections of the resume.

Acknowledgment

We are thankful to the people who gave us a comments on the work, to the people who participated in data collection and questionnaire distribution.

Author's Contributions

Channabasamma A: Contributed in drafting the article and contributed to conception and design, or acquisition of data, and analysis and interpretation of data.

Yeresime Suresh: Contributed in reviewing it critically for significant intellectual content and gave final approval of the version to be submitted.

Ethics

This article is an original research work. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. *In recommender systems handbook* (pp. 217-253). Springer, *Boston*, MA. https://link.springer.com/chapter/10.1007/978-0-387-85820-3_7

Aggarwal, C. C. (2016). "An introduction to recommender systems" in recommender systems, springer Cham Heidelberg New York Dordrecht London. pp, 1-28.

Aluvalu, R., & Jabbar, M. A. (2018). Handling data analytics on unstructured data using MongoDB. DOI.org/10.1049/cp.2018.1409

Amatriain, X., Jaimes, A., Oliver, N., & Pujol J. M. (2011). "Data mining methods for recommender systems" in recommender systems handbook, *Springer New York Dordrecht Heidelberg London*, pp. 39-67.

Belsare, R. G., & Deshmukh, D. V. (2018). Employment recommendation system using matching, Collaborative filtering and content-based recommendation. *International Journal Computer Applied Technology Res*, 7(6), 215-220.

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, 46, 109-132. DOI.org/10.1016/j.knosys.2013.03.012

Boorugu, R., & Ramesh, G. (2020, July). A survey on NLP-based text summarization for summarizing product reviews. In 2020 2nd International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 352-356). *IEEE*. DOI.org/10.1109/ICIRCA48905.2020.9183355

Deepa, A., & Rajeswara, R. R. (2017). An Eigen character Technique for Offline-Tamil Handwritten Character Recognition. In Proceedings of the First International Conference on Intelligent Computing and Communication (pp. 495-505). *Springer, Singapore*.

Deepa, A., & Rajeswara, R. R. (2020). A novel nearest interest point classifier for offline Tamil handwritten character recognition. *Pattern Analysis and Applications*, 23(1), pp, 199-212.

Faqihi, B., Daoudi, N., Hilal, I., & Ajhoun, R. (2020). Pedagogical resources indexation based on ontology in an intelligent recommendation system for content production in a d-learning environment. *Journal of Computer Science*, 16(7), 936-949. DOI.ORG/10.3844/JCSSP.2020.936.949

Jalili, M., Ahmadian, S., Izadi, M., Moradi, P., & Salehi, M. (2018). Evaluating collaborative filtering recommender algorithms: A survey. *IEEE Access*, 6, 74003-74024. DOI.org/10.1109/ACCESS.2018.2883742

Kantipudi, M. V. V., Moses, C. J., Aluvalu, R., & Goud, G. T. (2021). Impact of COVID-19 on Indian Higher Education. Library Philosophy and Practice, 4992, 1-11.

Kumar, A., Pandey, A., & Kaushik, S. (2017, January). Machine learning methods for solving complex ranking and sorting issues in human resourcing. In 2017 IEEE 7th International Advance Computing *Conference* (IACC) (pp. 43-47). *IEEE*. DOI.org10.1109/IACC.2017.0024.

Lappas, T. (2020). Mining career paths from large resume databases: Evidence from IT professionals. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(3), 1-38. DOI.org/10.1145/3379984

Le, Q., Vu, S. & Le, A. (2022). A comparative analysis of various approaches for incorporating contextual information into recommender systems. *Journal of Computer Science*, 18(3),187-203. DOI.ORG/10.3844/JCSSP.2022.187.203

Prokhorov, S., & Safronov, V. (2019, September). Ai for ai: What NLP techniques help researchers find the right articles on NLP. In 2019 International Conference on Artificial Intelligence: Applications and Innovations (IC-AIAI) (pp. 76-765). *IEEE*. DOI.org/10.1109/ICAIAI48757.2019.00023.

Raj, H., Harsh, K., Khattri, P. & Haider, T. U. (2022). ”Knowledge-based System of Indian Culture using Ontology with Customized Named Entity Recognition”, *Journal of Computer Science*, 18(3), 172-186. DOI.ORG/10.3844/JCSSP.2022.172.186

Ramesh, G., & Madhavi, K. (2019). summarizing product reviews using nlp-based text summarization. *International Journal of Scientific and Technology Research*.

Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender systems: Introduction and challenges. In recommender systems handbook (pp. 1-34). Springer, *Boston*, MA. DOI.org/10.1007/978-1-4899-7637-6_1

Rodriguez, L. G., & Chavez, E. P. (2019, February). Feature selection for job matching application using profile matching model. In 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS) (pp. 263-266). *IEEE*., DOI.org/10.1109/CCOMS.2019.8821682

Shahab, S. (2019). Next Level: A course recommender system based on career interests. https://scholarworks.sjsu.edu/etd_projects/684/

Srivastava, R., Palshikar, G. K., Chaurasia, S., & Dixit, A. (2018). What’s next? a recommendation system for industrial training. *Data Science and Engineering*, 3(3), 232-247. DOI.ORG/10.1007/S41019-018-0076-2.

Suresh, Y., & Manusha Reddy, A. (2021). A contextual model for information extraction in resume analytics using NLP’s spacy. In Inventive Computation and Information Technologies (pp. 395-404). *Springer, Singapore*. DOI.ORG/10.1007/978-981-33-4305-4_30.