12-30-2022

Artificial Intelligence Enabled, Social Media Leveraging Job Matching System for Employers and Applicants

Vishnu Pendyala
San Jose State University, vishnu.pendyala@sjsu.edu

Nishtha Atrey
International Business Machines

Tina Aggarwal
Amazon.com, Inc.

Saumya Goyal
Playstation

Recommended Citation

Vishnu Pendyala, Nishtha Atrey, Tina Aggarwal, and Saumya Goyal. "Artificial Intelligence Enabled, Social Media Leveraging Job Matching System for Employers and Applicants" *Proceedings - 2022 International Conference on Recent Trends in Microelectronics, Automation, Computing and Communications Systems, ICMACC 2022* (2022). https://doi.org/10.1109/ICMACC54824.2022.10093323

Artificial Intelligence Enabled, Social Media Leveraging Job Matching System for Employers and Applicants

Vishnu Pendyala
Department of Applied Data Science
San Jose State University
San Jose, CA, USA
vishnu.pendyala@sjsu.edu

Nishtha Atrey
IBM
San Jose, CA, USA
nishtha.atrey@gmail.com

Tina Aggarwal
Amazon
San Francisco, USA
aggarwaltina.96@gmail.com

Saumya Goyal
Playstation
San Francisco, USA
saumya.goyal2903@gmail.com

***Abstract*—Social media is increasingly becoming a window to the user's personality. Hiring the right candidate is a formidable task for any organization and particularly in the highly competitive software industry. This paper presents a machine learning and natural language processing based system to leverage social media to assess job applicants for their suitability for a given job. We use LinkedIn profiles to assess the technical suitability and combine Twitter posts with them to assess the emotional intelligence of the applicant. The system thus indicates both the technical and soft skills perspective of the job applicants. The system can be used by both prospective employers and employees. Employers can use it to shortlist job applicants and prospective employees can use it to evaluate their chances, retrospect, and take any corrective action. The results from the created system are encouraging.**

***Index Terms*—Machine Learning, Regression, Hiring, Social Media, Natural Language Processing**

I. Introduction

Workplace diversity is of prime importance across all organizations. Candidates should not be judged based on their religion, gender, race, age, or other external factors. Using automated Artificial Intelligence based tools that can help eliminate or at least reduce these factors and conscious biases introduced by humans in the decision-making process is currently the need. Reducing human subjectivity in applicant screening is crucial to a fair and progressive job market. At the same time, care must be taken not to introduce bias through the datasets used and the algorithmic solutions designed. Employing this guiding principle is shown to improve the organization's growth as well. It is now a widely accepted fact that a more diverse work environment, in fact, boosts the company's growth. Using an Artificial Intelligence approach to hiring can help reduce human biases and increase diversity in the workplace. It is now common practice for hiring teams to use social media during the hiring process [1]. The work described in this paper helps automate the process to alleviate their workload.

While the number of candidates is increasing for every job posting, in many organizations, the methodologies employed for the hiring process largely remains the same. Identifying ideal candidates for any organization is a pivotal task for the growth of the organization. The technical expertise of the candidates needs to align with the job requirements. There is plenty of manual intervention needed in filtering out the candidates, even before the actual interview process begins. This implies substantial cost and time overhead per job opening. There is a need to find an efficient and automated solution to help the recruiting managers in generating a compact candidate set, who are then selected for the subsequent recruitment process. Since this is a prototype, the work described in this paper uses a reduced dataset in the ICT and Computer Science areas but can be easily generalized to hiring in other areas as well.

Securing trust in online social networks [2] [3]and evolving a truthful world wide web [4] [5] are important problems, implying that using social media for hiring has ethical implications. The current literature includes substantial work on ethical considerations in algorithmic hiring. For brevity, this paper does not delve into the ethical aspects of the work in detail. The objective of this work is to support the hiring process by providing an automated mechanism to shortlist candidates from a huge pile of applications. It is by no means meant to replace the due diligence required on part of the hiring team in the context of the legal, ethical, and social framework. The work described in this paper follows from our earlier conference presentation [6].

II. Literature Review

As part of the literature review, we investigated a few implementation techniques that aim to provide automated recruitment solutions. The literature available in this domain is massive. This section covers some of the literature that is closely related to the work done in this paper but for brevity reasons, not all. From what we observed, the popular approaches do not seem to provide a holistic solution for filtering out eligible candidates. We need to have a grading system based on different qualifying factors that have been indicated in the job portals. Hence, it did not seem that the suggested techniques in the existing literature we surveyed are

978-1-6654-9604-9/22/$31.00 ©2022 IEEE

the most ideal solutions to this problem. The work detailed in this paper attempts to address some of the gaps.

Machine learning has been used to predict the performance of an applicant during the hiring process and also during employment from a given dataset containing a number of features [7]. It has also been used to study under-reporting of discrimination during hiring with respect to gender differences [8]. Clustering using Partition Around Medoid (PAM) algorithm can be used to study the ”Quality of Hire” problem [9]. Clustering, Association rule mining, and natural language processing techniques have been used on resumes and job test results to help with hiring [10]. Artificial Intelligence techniques have been used to build an intelligent recruitment system [11].

Hiring may not be effective if the hired employee stays away from work for various personal reasons. Researchers [12] tackled the problem of predicting absenteeism of prospective employees using various Machine Learning models and achieve around 90% accuracy with the models. Factors contributing to the acceptance of Artificial Intelligence models such as the ones used in this paper are analyzed in [13]. In [14], an automated recruitment technique is proposed that asks the applying candidates to enter their Twitter handle along with their resume. While the resume is parsed for understanding the technical standing of the candidate, the Twitter profile can help the recruiter understand the candidate's emotional intelligence. Machine learning techniques have been suggested to generate a personality predictor. This balance of both technical knowledge and the emotional quotient will help generate a potential candidate list with higher employee retention.

Authors [15] proposed a recruitment system that uses a combination of social networking sites (Twitter, Facebook), code repositories (Github) and coding platforms (SPOJ) to generate job-preparedness criteria that will help the recruiters identify the potential candidates with aptly aligned technical and social expertise. This paper proposes the use of openly available data to better gauge the candidate's potential. This will help the organizations get a wholesome view of the employees.

Researchers [16] discuss the ways of automation of subprocesses of recruiting, such as personnel scoring and content analysis. They also discuss the importance of the initial analysis of the resume and the selection of the employee based on the need of the organization. In large corporate organizations, firms with high turnover and companies with a permanent personnel reserve, there is the problem of analyzing many resumes during the recruitment cycle. In this paper, the solution to the issue is proposed to be an automated resume analysis with sorting along with integration with the overall recruitment system adopted in the organization.

Emotional intelligence is pivotal in determining candidates' suitability for different corporate positions. Emotional intelligence has a valuable impact on the overall performance of not just the applying candidate, but also the team spirit of the organization that they will be joining. In [17], the significance of mean differences in the work performances of the executives was determined. The dimensions covered are work performance quality, amount of effort put to complete a task, job speed, care in handling the company, ability to handle different jobs, initiative on the job and other such factors. The resulting difference in the mean scores of lower and higher emotional intelligence executive groups on all the dimensions of work performance was in favor of the high emotional intelligence group of executives.

Researchers [18] proposed an analysis of the personality traits of the potential candidates to help identify the right candidates, as well as improve employee retention. The produced candidates' listing combines character features automatically extracted from each candidate's social media activity, in this case, Facebook and Twitter. This is assessed against actual personality traits manually evaluated by human recruiters. The paper employs the technique of asking the applicants to log-in to the system with their Facebook or Twitter credentials. This will help get access to a large amount of information, such as status updates, interaction with other users and interests indicated by “likes” or side projects. This information can demonstrate their strengths and passions in a way that the employment history can't always do.

Compared to the existing solutions, we believe that our solution is a relatively more comprehensive technical approach to the problem of job matching, incorporating the computation of employability and emotional intelligence indicators from multiple sources.

III. Methodology

The system is divided into the following modules:

Recruitment Module: For this module, we created a web portal for recruiters to look for suitable candidates for a job posting. The recruiter is expected to enter the job description and the country for which they are looking for candidates. Following this requirements entry, the recruiter is prompted with a list of the top 5 candidates for the job. The machine learning algorithms return the top 5 candidates from the dataset to the recruiter based on the relevancy scores, keywords, relevant education, and experience of the candidates.

User Module: The user module is meant for individuals who want to know their general employability score and emotional quotient indicator. The module is divided into two parts:

- Emotional Quotient Indicator: This module is for calculating the general emotional quotient indicator of a candidate. Sentiment Analysis is performed on the Twitter data of the candidate, along with the summary and interests section of their LinkedIn profiles to calculate the emotional quotient indicator that is returned to the candidate.
- Employability Score: This module predicts the general employability score of a candidate. The candidate score for each candidate in the dataset is first calculated using a formula. A manual correction is then applied to the score by quick visual inspection of the profile, where

applicable. Since the number of profiles is huge, initial automatic calculation of the score helps in most cases and the next step of quick browsing and manual correction helps in generating the machine learning model more accurately. Multiple Linear Regression is used to predict the scores for new candidates, based on the keywords in their profile, their education and work experience.

Job Application module: The job seekers are required to submit their resume and the URLs for social media websites such as LinkedIn, Twitter, and GitHub. The system extracts all necessary information like background, skills, code contributions, contributions to discussion forums and provides it to the technical expertise calculator. The job seekers also provide their Twitter handle and the system extracts relevant data from their posts and other activity on the social media platform. This data is fed into the emotional quotient calculator.

Technical Expertise Calculator: This module takes in the extracted data from the different social media handles and calculates the technical expertise of the applicant. It considers the information in Table I while giving it a score/rank.

TABLE I
FEATURES USED FOR EXPERTISE CALCULATION

Latest Degree	All Degrees Awarded
Universities Attended	Education Description
Years of Experience	Experience Description
Experience Skills	Interests and Events Attended
Group Memberships	Honors Received
Industry Experience	Skills and Specialties Listed

Emotional Intelligence Quotient Indicator: This module takes in the extracted data from the different social media handles and calculates the emotional quotient indicator of the applicant. The data from the different social media platforms are used for performing lexical analysis and predicting personality and estimating their emotional aptitude.

The following fields are also considered for calculating the emotional quotient indicator of the candidate:

- Profile Summary
- Interests
- Twitter Handle

Candidate Ranking Algorithm: This module calculates each candidate's relevance score to the job and ranks the candidates based on their respective scores. The eligibility scores thus generated reflects how well the candidate fits a particular job. This is done through features that may be represented as Boolean values set to 1 in case of the presence of a particular trait or as numerical values where the number indicates the score of a candidate for that particular feature. This is then fed into the machine learning algorithm which constructs the ranking model. The ranking model then uses the scoring function to determine the candidate's rank based on the feature vector of the candidate.

A. Computing Infrastructure

For the implementation of this work we made use of Jupyter notebooks, Google Colab and the High-Performance Computing (HPC) system provided by the Charles W. Davidson College of Engineering at San Jose State University. The following computing infrastructure has been used to perform all the computations required in our approaches:

- Computing system with powerful multi-core and multi-socket servers, high performance storage, GPUs, and large amounts of memory.
- Compute partition contains all compute nodes (128GB RAM, no GPU)
- GPU partition contains all general GPU programming capable nodes (256GB RAM, NVIDIA P100 GPU)
- Several queues, or node partitions used for submitting jobs.

B. Dataset

The dataset used for this project is a collection of the JSON objects extracted using the LinkedIn API and contains detailed profiles of people available on the platform. We found this dataset in the form of a JSON dump online. Prior to preprocessing, the size of the dataset was approximately 10 GB. It contained profile details of over 2 million LinkedIn users and around 20 attributes per user in each JSON object per user. We also collected some job descriptions from the Internet.

C. Dataset description

The dataset originally contained 20 attributes for every LinkedIn profile in the JSON object. This JSON dump was then refined to create new attributes which are used in the Machine Learning models. The attribute selection and construction follow a typical feature engineering task for a machine learning project and are explained below, starting with a description of the original attributes in the dataset:

- EDUCATION_DESC: Contains information related to the education of the person such as Degrees, Majors, University etc.
- EVENT_DATA: Contains information related to events attended, name of the event, and title held when attending the event.
- EXP_YEARS_RANGE: The years of experience for each person categorized into 8-year ranges to make the dataset easier to deal with.
- LATEST_DEGREE: Contains the last degree obtained by the person.
- DEGREES: Contains all the degrees obtained by the person.
- UNIVERSITY: Contains a list of names of universities where the person studied.
- EXPERIENCE_DESC: Contains information related to the work experience of the person like the Company worked for, the Title of the position, the year joined the company and a description of the tasks and projects completed during their stay at the company.
- EXPERIENCE_YEARS: The number of years of work experience from the start of the career.
- GROUP_MEMBER: Contains information related to all the groups the person is a member of.

- GROUP_AFFILIATION: Contains information related to all the groups the person is affiliated with.
- HONORS: Contains any honors that the person might have received.
- INDUSTRY: Name of the industry the person belongs to.
- INTERESTS: Contains a list of the interests the person has.
- LOCALITY: Indicates the locality where the person is from, or where the person currently resides.
- COUNTRY: Indicates the country where the person is from, or where the person currently resides.
- NAME: Name of the person.
- SKILLS: Consists of a list of skills the person has.
- SPECIALITIES: Consists of a list of skills the person considers to be their specialties.
- SUMMARY: It contains a paragraph or points that the person has mentioned on their LinkedIn profile to give an overview of their background.
- URL: Gives a URL to the LinkedIn profile of the user.

All these columns consist of textual data which is later converted into numerical information and stored in their respective score columns. For instance, for the LATEST_DEGREE_SCORE column, the score is calculated based on the Latest Degree obtained by the person. The higher the degree, the higher the score. Similarly, for EXP_YEARS_RANGE_SCORE, the score was calculated based on the range the number of years of experience of the person lies in. The higher the range, the higher the score.

For some columns like DEGREES_SCORE, UNIVERSITY_SCORE, HONORS_SCORE, INTERESTS_SCORE, if any data was present in their columns, a score of 1 is awarded, otherwise 0. For other columns like EDUCATION_DESC_SCORE, EVENT_DATA_SCORE, EXPERIENCE_DESC_SCORE, EXPERIENCE_SKILLS_SCORE, GROUP_MEMBER_SCORE, GROUP_AFFILIATION_SCORE, SKILLS_SCORE, SPECIALITIES_SCORE, SUMMARY_SCORE, the score is calculated by counting the number of computer-related keywords that could be found in these columns from the keyword corpus curated by the team.

- CANDIDATE_SCORE is a score calculated for the candidate using a method created by the authors and the scores calculated for each feature. The score is computed from the following attributes, encoded as numbers: (a) Latest Degree (b) Experience in years (c) All Degrees (d) University Education_Description (e) Event Data (f) Experience_Description (g) Skills (h) Honors (i) Group Affiliation (j) Interests (k) Industry Skills (l) Specialities and (m) Summary. This score is scaled and made to lie between 1 and 10. It gives the general Employability rank of the person based on the information present in their LinkedIn profile.
- ALL_DATA: This column is a cleaned, comma separated combination of multiple columns namely EDUCATION_DESC, EXP_YEARS_RANGE, LATEST_DEGREE, DEGREES, EXPERIENCE_DESC, EXPERIENCE_SKILLS, INDUSTRY, INDUSTRY, INTERESTS, COUNTRY, SKILLS, SPECIALITIES, SUMMARY. The purpose of this column is to provide important keywords present in a candidate's profile which is used along with the keywords present in the job description for computing the TF_IDF vector.
- YEAR_RANGE: In this column the EXPERIENCE_YEARS column has been split into ranges of two years each. Making it easier to get predictions on the basis of similar work experience.
- TWEETS: This column consists of tweets from the candidate that have been fetched via the Twitter API. This column is further used to determine the Emotional Quotient Indicator of the candidate.
- POLARITY: This column consists of the score generated after the sentiment analysis of the SUMMARY, INTERESTS and TWEETS column. The score originally lies between - 1 to 1. -1 being completely negative, 0 being neutral and 1 being completely positive. This score is scaled from 0 to 1 using MinMax Scaler and then converted to a factor of 10 for further processing.

D. Data Preprocessing

Before feeding the data into any Machine Learning model, it needs to be preprocessed. This step improves the accuracy of the model and helps in obtaining better results. Data Preprocessing includes the removal of redundancy, unnecessary symbols, outliers, noise, dimensionality reduction and data normalization. Once the data has been preprocessed and the quality of the dataset has been improved, it is ready to be processed by the Machine Learning model. The dataset being used in this project had the entire profile of each user dumped as a JSON object. We used the first 2 million entries for this project and converted them into a dataframe for preprocessing and cleaning.

The steps followed to preprocess the dataset are:

- Irrelevant features as determined by visual inspection and human intuition such as gender, race, and language familiarity are simply dropped, ensuring not to introduce any bias. Since this dataset is just a raw dump of web crawling, it had plenty of other extraneous information such as numerical id.
- Special characters in the features like Summary, Experience, Education, Events are removed.
- The dataset had information also in multiple languages other than English. For the purpose of this project, only the rows containing English characters were kept and the rest were dropped.
- The data also consisted of people from various industries, not necessarily related to Computers and Information Technology. But for this work we dropped the rest and only kept entries that were related to Computers, Software, Hardware, Information Technology, and related fields.

- A list consisting of over 2500 words related to Computers, Computer Programming Languages, Tools and technologies was also curated to identify the keywords in the dataset.
- Calculated the number of years of work experience the person has from the information collected from the Experience section.
- Used Python packages and regular expressions to find out the country of the person from the information mentioned in the Locality column.

The final dataset after the preprocessing contained 190,773 entries.

IV. Experiments and Results

For this work, various machine learning algorithms are used to predict the top candidates based on the input provided by the recruiter. The approaches and computation details are discussed below.

A. Document Vectorizer

In view of the length of texts involved, we determined that TF-IDF is better suited to the problem than other language models like BERT [19], which do not work well with long texts [20]. The results with TF-IDF seemed satisfactory. As explained later, an instance of the effectiveness of the TF-IDF approach can be seen in Fig. 1. The Term Frequency-Inverse Document Frequency is a weighting metric used for creating document vectors from textual data. TF-IDF defines weights for documents and helps create a vector space using those weights, which can then be used to find similar documents. More important words from the textual data can be found and given higher weights after using the TF-IDF metric.

Term Frequency (TF) is the number of times a particular word or "term" appears in a document divided by the total number of words in the document.

Inverse Document Frequency (IDF) is the logarithm of the number of documents divided by the number of documents that contain the word w. It determines a weight that prioritizes the relative rarity of words across all documents in the corpus.

TF-IDF (term frequency-inverse document frequency) is a statistical measurement for evaluating the relevance of a word in a document that is part of a collection of documents or corpus. This is done by multiplying how many times a word appears in a document, the term frequency and the inverse document frequency of the word across a set of documents. In simple words TF-IDF is TF multiplied by IDF.

Steps that were followed to implement the TF-IDF document vectorizer:

1) A new column called ALL_DATA is first created. This column is a cleaned, preprocessed combination of multiple columns like EDUCATION_DESC, LATEST_DEGREE, DEGREES, EXPERIENCE_DESC, EXPERIENCE_SKILLS, INDUSTRY, INTERESTS, COUNTRY, SKILLS, SPECIALITIES, SUMMARY. The purpose of this column is to provide important keywords present in a candidate's profile which is used along with the keywords present in the job description for computing the TF_IDF vector.
2) The job description from the user is added at the end of the dataframe before computing the TF-IDF vector.
3) Once the job description node is added, a TF-IDF vector is created using the TfidfVectorizer() method. The TF-IDF vectorizer can be used to find out the importance of a word relative to other words in the text.
4) A matrix containing words from the job description and word count for each word in each document DF (document frequencies or local weights for all words in each document) can be observed.

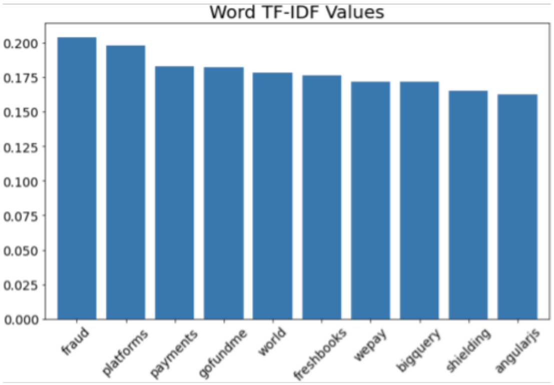


Fig. 1. TF-IDF values of a few words in the profiles matching the job description used

The TF-IDF values in Fig. 1 give insights into the word usage in the best matching candidate profiles. The job description used in this instance relates to financial software development. Therefore, the relevant profiles fetched from the dataset demonstrate experience in fraud detection and other financial applications.

B. K-Nearest Neighbor Algorithm Implementation

Similarity plays an important role in Machine Learning [21]. The K-Nearest Neighbor (KNN) Algorithm is used to determine similar documents, in this case the candidate profiles similar to the job description. This algorithm constructs a document plot in the vector space from the documents supplied to it. Documents that have the least distance among themselves are considered to be the most similar. This algorithm can be used to generate recommendations based on textual data.

The distance between two vector representations of the documents is calculated using the Euclidean distance formula. Euclidean distance is Minkowski distance shown in the equation (1) when p = 1. This approach is used to suggest the top five candidates based on their position in the K-nearest neighbor plot relative to the data supplied by the recruiter. It must be noted that the K-NN algorithm is used for determining the 'K' nearest neighbors and not in the usual classification or regression contexts.

Minkowski distance of order p,

$$D_i = \sqrt[p]{\sum_{i=1}^{N} |u_i - v_i|^p} \tag{1}$$

where u_i and v_i are the feature vectors of two text documents.

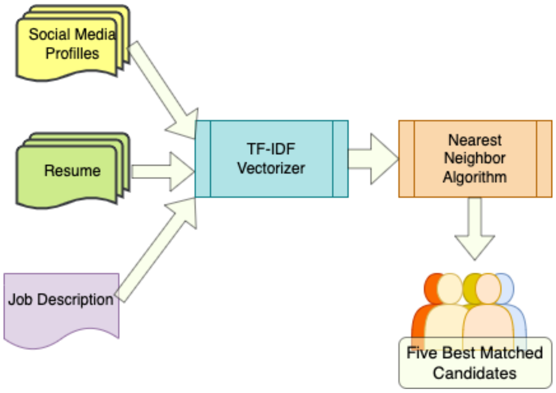


Fig. 2. Selecting the best matches using the nearest neighbor algorithm

As illustrated in Fig. 2, the following steps detail the use of the Nearest Neighbor algorithm to generate recommendations:

1) The TF-IDF matrix generated by the TF-IDF algorithm is passed to the KNN algorithm.
2) The KNN algorithm runs on this matrix to produce a vector space of documents. Each document is a vector in the multidimensional space where each dimension is a unique word in the corpus. For our experiments, if the corresponding resume is not available, we relied only on LinkedIn social media profile because LinkedIn profiles are usually comprehensive and resumes are often generated from LinkedIn profiles.
3) K is a hyperparameter and is set to 5, so that the algorithm returns the five best matching profiles. In this case, it is assumed that only 5 best suited candidates will be called for an interview.
4) From the plot, the five document vectors that are nearest to the job description vector are determined.
5) Since the vectors represent the candidate profiles, the five vectors with the least distance from the job description nodes are provided as the output suggesting that those candidates are the ideal matches to the requirements being input by the recruiter.
6) Those five candidates are then shown as the final output to the recruiter.

C. Multiple Linear Regression Implementation

Linear regression is a statistical model that considers the linear relation between two (Simple Linear Regression) or more (Multiple Linear Regression) variables. In the case of two variables, one of them is a dependent variable and another one is an independent variable. Linear relationship means that when the value of the independent variable(s) increases, the value of the dependent variable also increases proportionately.

Using Multiple Linear Regression, the General Employability Score of a potential candidate can be calculated. The following steps are performed to find the Employability Score for each candidate:

1) First, the Candidate Score for each person in the dataset is assigned using criteria that took into consideration the scores calculated for each column or feature.
2) The calculated score is scaled using MinMaxScaler to lie between 0 and 1.
3) It is then multiplied by 10 and rounded off to 2 digits.
4) A number of such scores are manually inspected for glaring mismatches and corrections applied.
5) When a new candidate entered their data on the portal, their information is converted to numerical scores for each feature and their Employability score is calculated using Multiple Linear Regression. The process is illustrated in Fig. 3a.

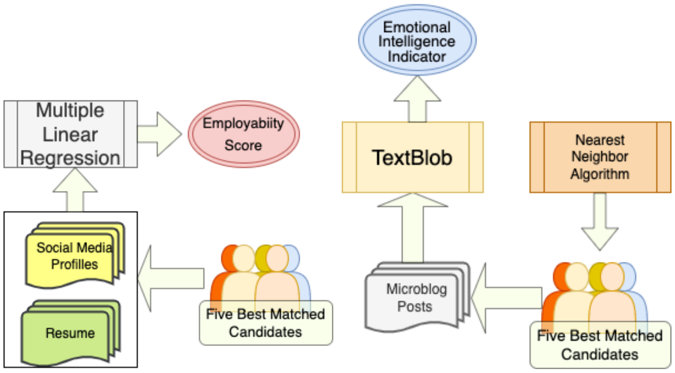

Multiple Linear Regression
Employabiity Score
Emotional Intelligence Indicator
TextBlob
Nearest Neighbor Algorithm
Social Media Profiles
Resume
Five Best Matched Candidates
Microblog Posts
Five Best Matched Candidates

Fig. 3. (a) Computation of the employability score using Multiple Linear Regression (b) Computation of the emotional intelligence indicator

D. Sentiment Analysis Implementation

The emotional quotient indicator has been computed from the sentiment analysis of the SUMMARY, INTERESTS and TWEETS columns. Sentiment analysis is the process of determining the emotional tonality behind a text or a series of words.

The following steps are used to compute the final score:

1) The three columns are cleaned and preprocessed.
2) The text from SUMMARY and INTERESTS from the LinkedIn profile is passed to TextBlob, which is a Natural Language Processing based library. The text is analyzed by the library for sentiment and a score called “polarity” is generated.
3) Using the Twitter handles provided in the LinkedIn profiles, a separate data frame is constructed for the Tweets by every user that are fetched from the Twitter API. We limited the number of tweets collected using the API to 200, which is fairly sufficient and reflective of the information we seek to gather from them. The tweets are then passed to the TextBlob library and a polarity score is generated for each tweet and stored in

the POLARITY column of the tweets data frame. The average of this column is calculated.

4) The polarity score lies in the range of -1 to 1. -1 being completely negative, 0 being neutral and 1 being completely positive. Further, the average of the tweets polarity and summary, interests polarity is calculated.
5) This score is then scaled using the MinMaxScaler so that the score lies between 0 and 1. Once the score has been adjusted to this range, it is converted into a factor of 10 so that the final score that is returned to the user is out of 10.

The process is illustrated in Fig. 3b.

E. Performance evaluation

For evaluating the model, the dataset is randomly split into training and testing partitions. The result of the testing partition is used to evaluate the overall performance of the model.

K-fold Cross-validation The dataset is split into k (=6) numbers of partitions. The model is trained based on (k-1) partitions and one of the partitions is then used as the test set. This process is repeated using the (k-1) partitions as the training set and some other partitions as the test set, such that each partition gets the chance of acting as the test set. The final performance of the model is found by calculating the average of all the test results.

Supplied test set The model is also evaluated based on an externally supplied test set. This is then used to judge the performance of the model based on expectations.

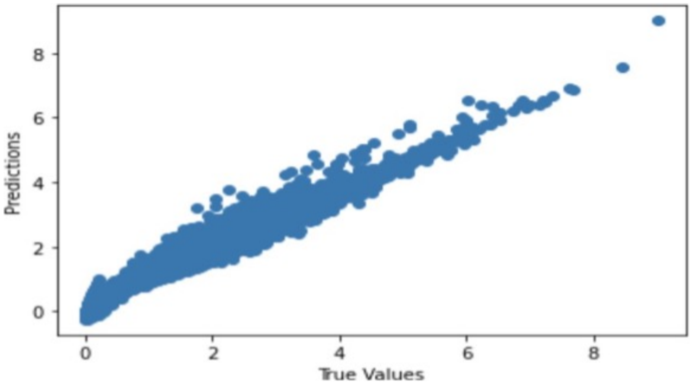
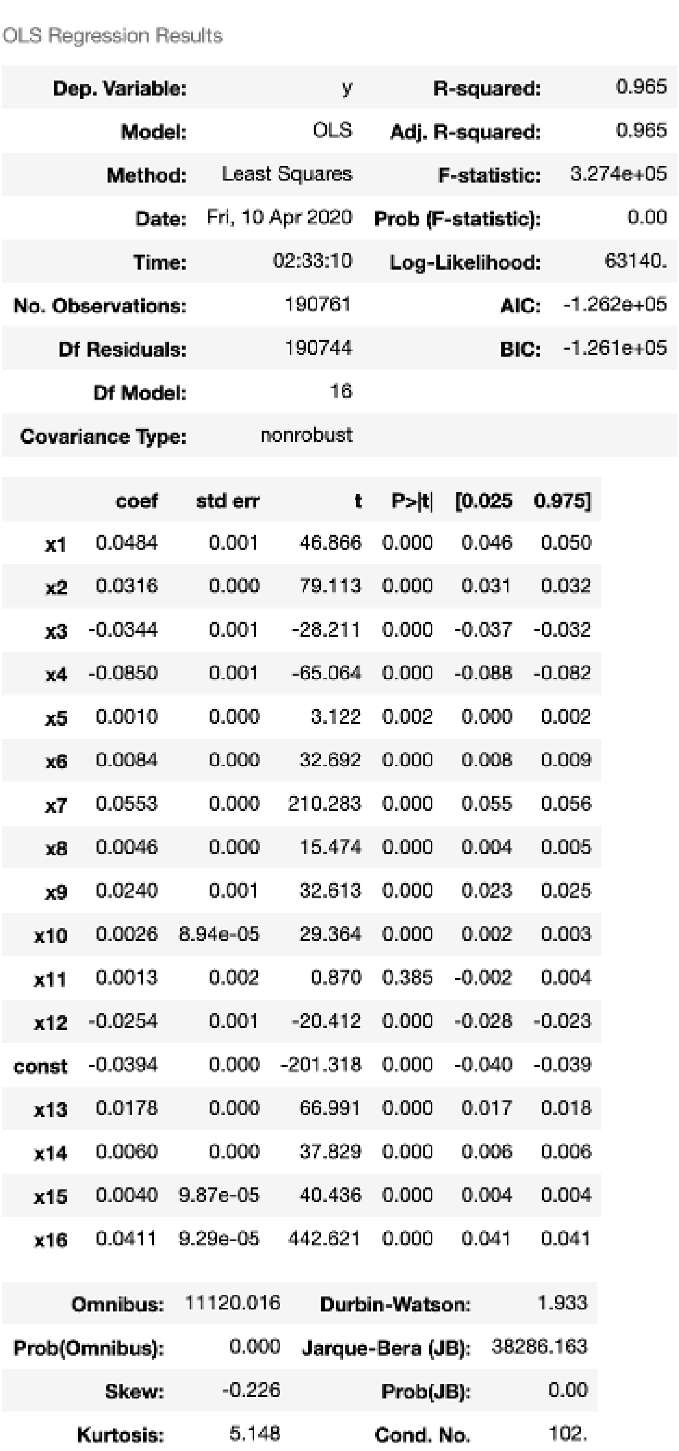


Fig. 4. Scatter-plot of the True Values vs Predicted Values for the Linear Regression Model

Accuracy scores We also evaluated the model by calculating the accuracy score based on RMSE values. A scatter-plot of the True Values vs Predicted Values for the Linear Regression Model is shown in Fig. 4 and the results from the regression runs are shown in Fig. 5.

Using the system to run the experiments Following steps briefly describe the use of the system.



OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.965
Model:	OLS	**Adj. R-squared:**	0.965
Method:	Least Squares	**F-statistic:**	3.274e+05
Date:	Fri, 10 Apr 2020	**Prob (F-statistic):**	0.00
Time:	02:33:10	**Log-Likelihood:**	63140.
No. Observations:	190761	**AIC:**	-1.262e+05
Df Residuals:	190744	**BIC:**	-1.261e+05
Df Model:	16		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
x1	0.0484	0.001	46.866	0.000	0.046	0.050
x2	0.0316	0.000	79.113	0.000	0.031	0.032
x3	-0.0344	0.001	-28.211	0.000	-0.037	-0.032
x4	-0.0850	0.001	-65.064	0.000	-0.088	-0.082
x5	0.0010	0.000	3.122	0.002	0.000	0.002
x6	0.0084	0.000	32.692	0.000	0.008	0.009
x7	0.0553	0.000	210.283	0.000	0.055	0.056
x8	0.0046	0.000	15.474	0.000	0.004	0.005
x9	0.0240	0.001	32.613	0.000	0.023	0.025
x10	0.0026	8.94e-05	29.364	0.000	0.002	0.003
x11	0.0013	0.002	0.870	0.385	-0.002	0.004
x12	-0.0254	0.001	-20.412	0.000	-0.028	-0.023
const	-0.0394	0.000	-201.318	0.000	-0.040	-0.039
x13	0.0178	0.000	66.991	0.000	0.017	0.018
x14	0.0060	0.000	37.829	0.000	0.006	0.006
x15	0.0040	9.87e-05	40.436	0.000	0.004	0.004
x16	0.0411	9.29e-05	442.621	0.000	0.041	0.041

Omnibus:	11120.016	**Durbin-Watson:**	1.933
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	38286.163
Skew:	-0.226	**Prob(JB):**	0.00
Kurtosis:	5.148	**Cond. No.**	102.

Fig. 5. Ordinary Least Squares Regression results

- The automated recruitment system designed for this paper provides the end user with a web portal, where they could log in as either the candidate or an employer. For brevity, we are skipping the implementation details of the front-end and other components not considered the core aspects of the system.
- Assuming the role of an employer, one needs to enter in the job description for which they are looking for potential candidates.
- Having a detailed job description with all the technology

specifications, unique skill set requirements, educational qualifications and such would help with generating a more comprehensive output.

- Based on the job requirements, the employer would now get a list of all the potential candidates who could be suitable for the job.
- In the case of candidate login, they need to key-in values for fields such as Degrees earned, university details, education description, experiences that showcase the candidate's potential, technical skills and so on. Based on the input details, the user is provided with a candidate score, which indicates the employability quotient of the candidate.

V. Conclusion and Future Directions

Social media posts are more spontaneous than a thoroughly planned resume and other job application artifacts. The work in this paper analyzes social media for technical and behavioral clues to a job applicant to help in the hiring process. The system can also help the applicants to evaluate their chances and retrospect. In future work, we plan to provide a more reliable list of the candidates by implementing some more robust language and machine learning models. News coverage also provides insights into the activities a job applicant is capable of. We plan to include searched news items also into the evaluation metrics of a prospective employee. Explainability is an important requirement in this domain, a direction we intend to pursue in the future. Data from social media is inherently heterogeneous. A future direction is to search for algorithms to normalize the data in some way so that certain classes of users are not disadvantaged. Additionally, rather than have the user key in their social media details, we plan to implement an ID generation technique that could help identify all the social media handles which are pertinent to the candidate's profile. Addressing bias in the models [22] used for hiring is another major area of research. Auditing the tools [23] of the like described in this paper is a future direction as well.

VI. Acknowledgment

The authors thank Madhu Prasanna Kalluraya for contributing to this work and San Jose State University for providing the infrastructure for doing the experiments.

References

[1] Elizabeth C Alexander, Deanna RD Mader, and Fred H Mader. Using social media during the hiring process: A comparison between recruiters and job seekers. *Journal of Global Scholars of Marketing Science*, 29(1):78–87, 2019.

[2] Vishnu S Pendyala. Securing trust in online social networks. In *International Conference On Secure Knowledge Management In Artificial Intelligence Era*, pages 194–201. Springer, 2019.

[3] Vishnu S Pendyala, Yuhong Liu, and Silvia M Figueira. A framework for detecting injected influence attacks on microblog websites using change detection techniques. *Development Engineering*, 3:218–233, 2018.

[4] Vishnu S Pendyala. Evolving a truthful humanitarian world wide web. 2018.

[5] Vishnu S Pendyala and Silvia Figueira. Towards a truthful world wide web from a humanitarian perspective. In *2015 IEEE Global Humanitarian Technology Conference (GHTC)*, pages 137–143. IEEE, 2015.

[6] Vishnu S Pendyala, Nishtha Atrey, Tina Aggarwal, and Saumya Goyal. Enhanced algorithmic job matching based on a comprehensive candidate profile using nlp and machine learning. In *2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService)*, pages 183–184. IEEE, 2022.

[7] Ali A Mahmoud, Tahani AL Shawabkeh, Walid A Salameh, and Ibrahim Al Amro. Performance predicting in hiring process and performance appraisals using machine learning. In *2019 10th International Conference on Information and Communication Systems (ICICS)*, pages 110–115. IEEE, 2019.

[8] Jaehong Yoon, Ji-Hwan Kim, Yeonseung Chung, Jinsu Park, Glorian Sorensen, and Seung-Sup Kim. Gender differences in under-reporting hiring discrimination in korea: a machine learning approach. *Epidemiology and health*, 43, 2021.

[9] Sateesh Shet and Binesh Nair. Quality of hire: expanding the multi-level fit employee selection using machine learning. *International Journal of Organizational Analysis*, 2022.

[10] Neha Sharma, Rigzen Bhutia, Vandana Sardar, Abraham P George, and Farhan Ahmed. Novel hiring process using machine learning and natural language processing. In *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, pages 1–6. IEEE, 2021.

[11] Said Achchab and Yassine Khallouk Temsamani. Use of artificial intelligence in human resource management:"application of machine learning algorithms to an intelligent recruitment system". In *Advances in Deep Learning, Artificial Intelligence and Robotics*, pages 203–215. Springer, 2022.

[12] Gopal Nath, Antoine Harfouche, Austin Coursey, Krishna K Saha, Srikanth Prabhu, and Saptarshi Sengupta. Integration of a machine learning model into a decision support tool to predict absenteeism at work of prospective employees. *arXiv preprint arXiv:2202.03577*, 2022.

[13] Vanessa Laurim, Selin Arpaci, Barbara Prommegger, and Helmut Krcmar. Computer, whom should i hire?–acceptance criteria for artificial intelligence in the recruitment process. In *Proceedings of the 54th Hawaii International Conference on System Sciences*, page 5495, 2021.

[14] Vishnu M Menon and HA Rahulnath. A novel approach to evaluate and rank candidates in a recruitment process by estimating emotional intelligence through social media data. In *2016 International Conference on Next Generation Intelligent Systems (ICNGIS)*, pages 1–6. IEEE, 2016.

[15] Animesh Giri, Abhiram Ravikumar, Sneha Mote, and Rahul Bharadwaj. Vritthi-a theoretical framework for it recruitment based on machine learning techniques applied over twitter, linkedin, spoj and github profiles. In *2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)*, pages 1–7. IEEE, 2016.

[16] Maria V Belova and Anton B Zhernakov. Modern methods of resume processing in recruiting information systems. In *2018 XVII Russian Scientific and Practical Conference on Planning and Teaching Engineering Staff for the Industrial and Economic Complex of the Region (PTES)*, pages 19–22. IEEE, 2018.

[17] Reza Gharoie Ahangar and Ali Alijani Rooshan. Emotional intelligence as determinant/predictor of work performance among executives. In *2010 International Conference on Financial Theory and Engineering*, pages 147–150. IEEE, 2010.

[18] Evanthia Faliagka, Maria Rigou, and Spiros Sirmakessis. An e-recruitment system exploiting candidates' social presence. In *International Conference on Web Engineering*, pages 153–162. Springer, 2015.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[20] Ming Ding, Chang Zhou, Hongxia Yang, and Jie Tang. Cogltx: Applying bert to long texts. *Advances in Neural Information Processing Systems*, 33:12792–12804, 2020.

[21] Vishnu Pendyala and Rakesh Amireddy. Enhancing the cognition and efficacy of machine learning through similarity. *SN Computer Science*, 3(6):1–14, 2022.

[22] Elisabeth K Kelan. Algorithmic inclusion: Shaping artificial intelligence in hiring. In *Academy of Management Proceedings*, volume 2021, page 11338. Academy of Management Briarcliff Manor, NY 10510, 2021.

[23] Mona Sloane, Emanuel Moss, and Rumman Chowdhury. A silicon valley love triangle: Hiring algorithms, pseudo-science, and the quest for auditability. *Patterns*, 3(2):100425, 2022.