Candidate Selection for the Interview using GitHub Profile and User Analysis for the Position of Software Engineer

R.G.U.S.Gajanayake
Department of Information Technology
Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
us97gajanayake@gmail.com

M.H.M.Hiras
Department of Information Technology
Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
hirasharis@gmail.com

P.I.N.Gunathunga
Department of Computer Systems Engineering
Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
nilankaisith@gmail.com

E.G.Janith Supun
Department of Information Technology
Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
janithsupun2@gmail.com

Anuradha Karunasenna
Department of Information Technology
Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
anuradha.k@sliit.lk

Pradeepa Bandara
Department of Information Technology
Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
pradeepa.b@sliit.lk

2020 2nd International Conference on Advancements in Computing (ICAC) | 978-1-7281-8412-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICAC51239.2020.9357279

***Abstract*—Selecting the most suitable candidates for interviews is an important process for organizations that can affect their overall work performance. Typically, recruiters check Curriculum Vitae (CV), shortlist them and call candidates for interviews which have been the way of recruiting new employees for a long time. To minimize the time spent on the above process, pre-screening mechanisms are nowadays implemented by organizations. However, those mechanisms need sufficient information to evaluate the candidate. For example, in case of a software engineer, the recruiters are interested on the programming ability, academic performance as well as personality traits of potential candidates. In this research, a pre-screening solution is proposed to screen the applicants for the post of Software Engineer where candidates are screen based on an initial call transcript, GitHub profile, LinkedIn profile, CV, Academic transcript and, Recommendation letters. This approach extracts textual features of different dimensions based on Natural Language Processing to identify the Big Five personality traits, CV and GitHub insights, candidate's skills, background, and capabilities from Recommendation letters as well as programming skills and knowledge from Academic transcript and Linked Profile. The results obtained from the different areas are presented and shown that the selected supervised machine learning algorithms and techniques can be used to evaluate the best possible candidates.**

Keywords— Candidate Selection, Big five personality, GitHub, Recommendation, LinkedIn, Curriculum Vitae, Questionnaire, Academic Transcript, Machine Learning Algorithms

I. Introduction

Organizations have been using the same method for recruiting new employees for a long period. They ask for CV from candidates, then shortlist them and call for interviews. This method is not always very effective for the recruitment process when it comes to hire for a high demanding job, since the information in the CV may not be very accurate and CV does not correctly reflect the personality of the candidate on which the organization may also be interested it. As a result, it is not possible to select the ideal candidate only by looking at the CV [1].

With the technical advancement in the business world, businesses as well as software companies need Software Engineers to maintain their IT infrastructure and software developments that promise to maintain the business systems [2]. Therefore, the demand for the Software Engineering in the industry is increasing, as a result companies need to spend many resources into the screening process to decide and select the ideal candidates from the pool of candidates and then shortlist them for the interview.

Unlike other job roles in the industry, recruiters expect candidates for Software Engineering positions to have a wide range of abilities such as self-learning, problem solving etc. [3]. Therefore, it is not possible to evaluate the candidates using the CV only. Under such circumstances analyzing the GitHub profile, academic transcript, recommendation letter, LinkedIn Profile and personality of the candidate would reveal important insights about the candidate. Using the above method, the handful of best candidates for the position of Software Engineer interview can be selected [4].

Existing researchers have proposed several creative solutions for pre-screening candidates for different positions. However, these research have evaluated the candidates from limited number of dimensions. The purpose of this research is to develop a comprehensive screening solution to for candidates applying for the position of software engineer which add more value for the screening process by assisting organizations to evaluate candidates from multiple dimensions. The proposed system is a candidate selection model which analyze the candidates using GitHub profile and CV, recommendation and questionnaires, academic transcript and LinkedIn profile, and personality prediction of the candidate. Overall, a feedback is given as a result to analyze the ideal candidate for the position of Software Engineer.

II. Literature Review

There are various researchers done in recent years regarding the candidate analysis for interviews using different approaches. For example, Majumder et al, [5] forwarded a concept for determining personality traits of people from text using deep learning-based methods which predicts the presence and absence of the big five personality traits which are Openness, Conscientiousness, Extraversion, Agreeableness and Neuroticism, where deep convoluted neural network is used to build the classification model.

Bhannarai and Doungsa-ard [6] used a Personality Test and KNN Classification technique to identify personality traits of individuals where it each apply big five personality models which is known as OCEAN model to predict how people are selected for the agile methodology of Software Project Managers[7].

As GitHub has become a widely used version control system, there are many individual and organizational repositories being created and updated every day. Since GitHub has a widely used open source repositories in its database, many research has been conducted to analyze GitHub profiles using GitHub user API [8] to get the insights of candidates. For example, Giri et al. [9] has conducted a research to get insights such as the programming languages used by candidates to do the projects, the number of public repositories available in each language, the number of stars the repository has, and the number of followers the candidate possess. Furthermore, there are research conducted in relation to the CV analysis. Kopparapu [10] identified important features such as personal details, education and skills from the CVs using Natural Language Processing techniques (NLP).

According to P. P. Shelke et al. [11] research has been done on candidate's feature extraction and categorization for unstructured text document which inspects various mechanisms for feature mining and the part of speech tagging is used for feature labelling for candidate phrases. Savidu Amarakoon and Amitha Caldera [12] research has been done mainly focusing on finding right document from large collection of unstructured documents using text mining techniques and then identify the common characteristics among them.

In a research conducted by Dai et al. [13] popular professional online social network (OSN) are explored using scraping and clustering techniques. Use of NLP to classify the educational qualifications and clustering of the professional qualifications of the collected profiles is done in the above research to gain some insights about candidates. Sage et al. [14] implementation of extracting information from documents which contains tables using OCR engine. The table field values extracted, tokenized and then trained from recurrent neural network.

Most of the research has evaluated the candidates in a limited number of dimensions whereas more insights could be obtained by evaluating information available of candidates from multiple dimensions. To address the above research gap, this paper proposes a system which adds more value to the interviewing process by screening the candidates using information available from GitHub, CV, LinkedIn, Recommendation Letter and Personality Traits.

III. Methodology

This section includes the detailed description about the methodology, processes, techniques and mechanisms used in the proposed work. The description includes how the software development is carried out, materials and data needed and how they are collected.

The proposed system determines the candidate's statistics by analyzing their user profiles using different techniques and methodologies. The proposed work consists of 4 major components. They are 1) GitHub Profile and Curriculum Vitae Analysis, 2) Big Five Personality Traits Analysis, 3) Questionnaire and Recommendation Letter Analysis, 4) LinkedIn and Academic Transcript Analysis.

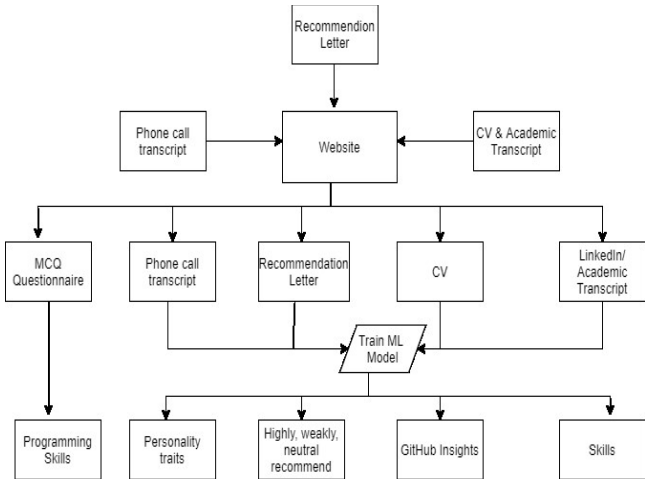


Fig. 1. High level system diagram

A. Big Five Personality Prediction from Phone Call Transcript

To obtain the personality traits of candidates, a transcript of a phone call is used in this research. To predict the big five personality of a candidate from phone call transcript, Amazon Transcribed [15] service is used to analyze the recorded audio call to a text document which uses advanced machine learning techniques. To identify candidate's personality, selected four questions are put forwarded to all the candidates and answers to those questions are added to the dataset. The questions forwarded to the candidates are as follows:

- What experiences do you possess to perform in this position?
- Why do you think of leaving your current organization?
- What are the challenges that you are looking for this position in your organization?
- What is your career life in five years?

Big Five Personality Model is used to predict the personality traits of candidates using the data collected. The Big Five Model also known as OCEAN Model [16] consists of:

1. Openness - Curious, intelligent, imaginative. High scorers will, in general, be creative and advanced in taste and acknowledge assorted perspectives, thoughts, and encounters.
2. Conscientiousness - Responsible, organized, persevering. Such individuals are greatly dependable and will, in general, be high achievers, diligent, employees and organizers.
3. Extraversion - Outgoing, amicable, assertive. Cordial and vivacious, outgoing people draw motivation from social circumstances.
4. Agreeableness - Cooperative, helpful, nurturing. Individuals who score high in suitability are harmony attendants who are by and large hopeful and trusting of others.

5. Neuroticism – Anxious, insecure, sensitive. Neurotics are testy, tense, and effectively tipped into encountering contrary feeling.

Text Pre-processing - Text preprocessing is done to remove the unwanted characters like punctuation marks and spaces except the numerical and ASCII letters from the call transcript. The corpus is converted to lower cases which is a most common data cleaning steps to process. Tokenization, stop word removal are done in text pre-processing step. The data preprocessed are stored in a pickle file for further use.

Feature Extraction - Features from pre-processed text were extracted to convert the collection of text in to TfidVectorizer feature vector using bag of words method. The data is split into training and testing dataset where 30% is for testing and the other 70% for training.

Model Selection - Several supervised machine learning classification algorithms such as Support Vector Machine, Logistic Regression, Naïve Bayes, Decision Tree and Random Forest classifiers were used to select the best algorithm to train and evaluate the model. Classification reports and confusion matrix are generated to evaluate the performance of the models. The model with high accuracy is chosen as the best model/algorithm to test our future text prediction. Logistic regression is used as the best model that gives a high accuracy because it is a very efficient algorithm for text data that performs well when the dataset is linearly separable as well as it is less prone for overfitting for low dimensional dataset. Finally, a predictor was developed with a visualization to predict the big five personality traits of the text (candidate).

Deploy Machine Learning Model – Machine learning model is deployed on flask Health check server by creating REST API. By sending the response the logs are created to list down the status details in a log file.

B. Analyzing the Curriculum Vitae (CV) and GitHub

The purpose of this component is to evaluate the CV and GitHub profile of potential candidates to gain insights. Most of the Curriculum Vitae sent by candidates are in either pdf or word format. If the document updated is in the word format it will be converted to PDF format and the "PDFMiner" python library is used to extract the text from the pdf document. The Extracted document will be converted to readable ASCII format. The readable document then will be undergoing NLP Techniques. Here the "NLTK" python library is used, the NLTK steps are, the document is tokenized, tagged and chunked to segments. Chunking data in to segments as above simplifies the extraction process of data for machine learning algorithms. Machine learning algorithms such as Naïve-Bayes classifier, Logistic Regression, and Random Forest were used to evaluate the best model and Naïve Bayes model is considered as the best model since it performs better compare to other models. The main advantage of the algorithm is that it requires less training data as well as it performs well in multiclass prediction for categorical input features compared to numerical features.

GitHub Users API - is used to track a user's coding contributions, giving a view about user's programming language expertise, projects he/she's worked on and the organization a potential employer has been associated with.

Repositories API - This includes repositories owned by the authenticated user, repositories where the authenticated user is a collaborator, and repositories that the authenticated user has accessed to through an organization membership. Each repository denotes a project - from which a list of programming languages can be extracted.

Organizations API - This API can be used to fetch a list of organizations of which the user is a member of. The description field of every organization assesses the core competence of the organization, thereby helping the system classify the potential employer.

By using the regular expression to the processed data, the GitHub username can be identified. And by using the username, it is possible to get the users contribution from GitHub. The steps to get GitHub user information are, Obtain the candidate's GitHub username which is a unique identifier of every user from the CV then Using that username an API request to the GitHub is made, finally the request is responded with JSON type meta information For authentication purpose, OAuth two-factor mechanism is used.

The features extracted from the CV and the programming Languages obtained from the GitHub will be further analyzed and displayed as the candidate's insights.

C. Analyzing the Recommendation Letter and MCQ Questionnaire

Recommendation letters are used to analyze the candidate's skills, background, and capabilities before the initial screening process of the interview. Academic competence, work competence, social competence, and personal competence are identified from the recommendation letters to label as highly recommend, recommend, neutral and weakly recommend using the identified levels of competences.

Data Cleaning - Data is cleaned to remove the unnecessary parts of the texts to a standard format for further analysis. Common data cleaning steps like remove special characters, single characters and multiple spaces, lowercase letters are done.

Tokenization - Split text into smaller pieces, tokens using NLTK word_tokenize. The most common tokenization process is breaking the text into words, which is used in the implementation.

Stop Word Removal - Filter out words that have very less meaning like "is, am, are" etc. because it doesn't add much meaning for the machine to process.

Feature Extraction – Document term matrix is created using TfidfVectorizer feature vector using bag of words methods. Data is split into 70% training and 30% for testing.

Normalization [17] - Stemming and lemmatization are performed for normalizing the text data where porters stemming is used for stemming whereas wordNet is used for lemmatizing.

Exploratory Data Analysis - Main trends of the dataset are summarized and figured out using visualization. The word cloud plot for each class labels are visualized to find the most common words for each class labels.

Model Selection - Several supervised machine learning algorithms such as Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), Random Forest and a simple neural network in Scikit-learn are used to determine the best accuracy. Mainly the SVM is used as the model selection to test the future unknown predictions since it gives the highest

accuracy score among all the above models and it is used in many text classification problems and also it works well for relatively small datasets.

Deploy Machine Learning Model - Implemented machine learning model to deploy on flask using REST API.

In MCQ programming questions analysis, 10 random questions per each programming language is taken. Candidate should enroll with the system to take these questions. He/she can select one or more languages that he/she prefers. So that the marks will be calculated for each language he takes. The question generation is implemented through backend and call the API to view the Questions to candidates. This is done only to analyze the programming skill of the candidate and this is done only to support the proposed work.

D. Analyzing the LinkedIn Profile and Academic Transcript

The system only focused on candidate's skills in LinkedIn. Scrap data from LinkedIn profiles is tough, since LinkedIn maintains very strict privacy policy and security measures. LinkedIn still trying to prevent any scrapping activities by using their anti-scraping technology which adds captcha puzzles, changing login page and their html element ids. The automated script is created using selenium to login to the LinkedIn as a user and then navigate to candidates profiles. Using this method is the only way to bypass login without facing LinkedIn policy restrictions.

Scrapping - After logged into the account candidate profile URL should provide to navigate and scroll down to "Skills & Endorsement" section of the page. Then scrap the section by selecting the html id tag.

Text Pre-processing - Initially the scraped data is in html format, then clean and preprocess the text to create usable datasets.

Train the Model - Using the created neural network model which categorize available skills in profile of each individual candidate after input the datasets and train the model under supervised learning which compares available skills of the candidate are matching with required skills to get an accurate output which predicts if the candidate has the required programming skills for software engineering position.

Academic Transcript contains the academic performance of the candidate over the years, breakdown to semesters including module names and their respective grades in tabular format which can be changed from one institute to another. This analysis is focused on modules which includes programming languages and their grades to identify candidates programming skills and other technical modules, soft skills which fulfill the job requirements. Using tesseract.js optical character recognition (OCR) [18] and tokenization technique is used to collect words, which separate and recognize the module names. Then clean the extracted text by removing unnecessary characters and numbers to filter the relevant module names and their grades with the help of NLTK. Then categorize them using machine learning algorithms like Naïve-Bayes classifier. Finally using labeled data from the created dataset to identify the knowledge level and skills (programming skills, soft skills) of each candidate.

IV. Results And Discussions

This section describes the results and discussions of the research applied in four main components with their classification models.

A. Result Analysis of Big Five Personality Prediction from Phone Call Transcript

In this section, the results of the classification models that has been developed will be discussed by comparing the accuracy score of classified scores based on personality traits.

When given a text input, for example the selected answers for the questions, the system will predict whether the five personality traits of the candidate are present (1) or absent (0) for that piece of text. It helps the recruiters to gain a deep understanding about the candidate. Therefore, it is convenient for recruiters to find the qualified candidate the organization is looking for over the other candidates in the initial screening process.

Fig. 2. Predictions of the unknown personality text

The hyper parameters have been tuned according to the classifier. The tuned hyper parameters for SVM is kernel='linear', for Logistic Regression is solver='newton-cg' and Random Forest is n_estimators=100.

The below table 1 summarizes the accuracy scores obtained by 4 different classifiers. The highest accuracy model for Openness is Logistic Regression, for Conscientiousness it is Logistic regression, for Extraversion it is SVM and Logistic Regression, for Agreeableness it is Logistic Regression and Random Forest, and Neuroticism has a slightly equal accuracy for SVM and Logistic Regression. So, it can be concluded that the model with highest accuracy for all the five traits is Logistic Regression with 0.68, 0.56, 0.57, 0.55 and 0.63 respectively.

Since this is a binary classification, the classification reports and the confusion matrix are used to evaluate the performance of the model.

So, by referring the table 1, overall, it can be stated that the Logistic Regression classifier with bag of words approach outperform the rest of the classifiers with the highest accuracy. As a result, Logistic Regression is used to predict the unknown text of the candidate.

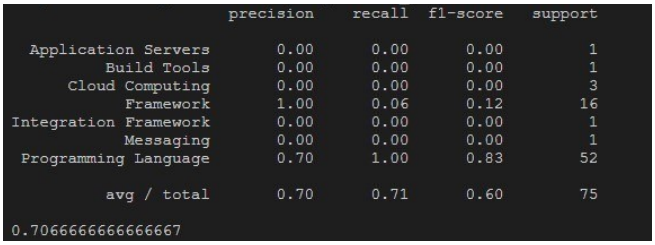
TABLE I. Accuracy Scores For Each Traits

Classifiers / ML Algorithms	OPN	CON	EXT	AGR	NEU
SVM	0.66	0.55	0.55	0.53	0.62

Decision Tree	0.53	0.51	0.51	0.53	0.47
Naïve Bayes	0.50	0.55	0.49	0.51	0.51
Logistic Regression	**0.68**	**0.56**	**0.57**	**0.55**	**0.63**
Random Forest	0.64	0.53	0.56	0.55	0.50

B. Result Analysis of CV and GitHub

From the candidate CVs, the details of the candidate are extracted. The details like name, e-mail address, GitHub username, GPA, work experiences, programming languages are extracted from the CV.



	precision	recall	f1-score	support
Application Servers	0.00	0.00	0.00	1
Build Tools	0.00	0.00	0.00	1
Cloud Computing	0.00	0.00	0.00	3
Framework	1.00	0.06	0.12	16
Integration Framework	0.00	0.00	0.00	1
Messaging	0.00	0.00	0.00	1
Programming Language	0.70	1.00	0.83	52
avg / total	0.70	0.71	0.60	75

0.7066666666666667

Fig. 3. Classification report and accuracy score obtained from the CV

From the GitHub, the details like GitHub repository, committed works and programming languages are extracted for analysis. Then the results obtained from CV and the GitHub insights are analyzed to determine the score of the candidate. The test result of the GitHub and CV analysis is shown in the figure 4 and 5 respectively.

Fig. 4. Results obtained from GitHub – Repos, commits

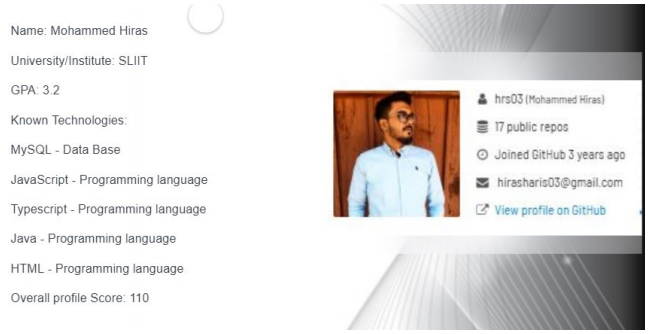


Fig. 5. Results obtained from CV

C. Results Analysis of Recommendation and Questionnaire

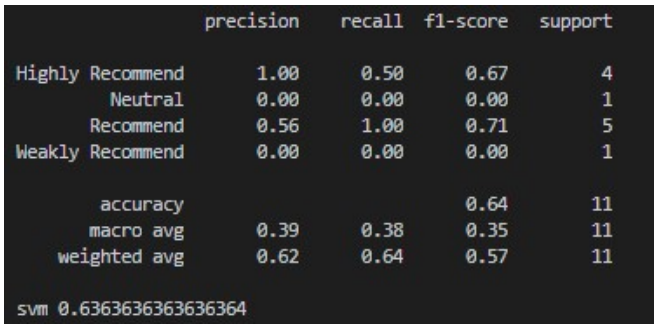
The results of the classification models that has been developed is shown below by comparing the accuracy score for labels in this section. The figure 6 visualizes all the words in the text corpus using the word cloud plot. The frequently used common words are highlighted with a large font size.

Fig. 6. Word cloud in the text corpus

For the model prediction SVM, Random Forest, Logistic Regression, Naïve Bayes, and a simple neural network MLP classifier were used. The hyper parameters for these classifiers are tuned to get a better result. The model with highest accuracy is chosen as the best classifier for our recommendation prediction. The comparison of classifiers used with the accuracy score is shown in the table 2. SVM shows the highest accuracy among other classifiers with 0.64 accuracy score. So, SVM is used to predict the unknown recommendation letters of the candidate as recommend, highly recommend, neutral and weekly recommend.

TABLE II. ACCURACY SCORES FOR EACH CLASSIFIERS

ML Classifiers	**SVM**	Random Forest	Naïve Bayes	MLP	Logistic Regression
Accuracy	**0.64**	0.37	0.55	0.47	0.55



	precision	recall	f1-score	support
Highly Recommend	1.00	0.50	0.67	4
Neutral	0.00	0.00	0.00	1
Recommend	0.56	1.00	0.71	5
Weakly Recommend	0.00	0.00	0.00	1
accuracy			0.64	11
macro avg	0.39	0.38	0.35	11
weighted avg	0.62	0.64	0.57	11

svm 0.6363636363636364

Fig. 7. Classification Report and accuracy obtained from SVM Classidfier

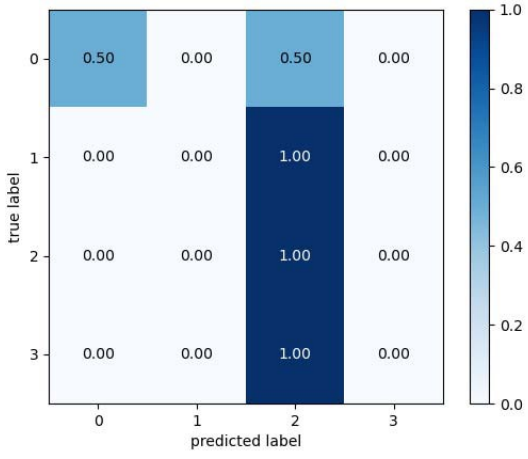


Fig. 8. Confusion Matrix obtained from SVM

D. Result Analysis LinkedIn and Academic Transcript

The LinkedIn analysis prediction results are based on total number of matching skills of candidate's technical and soft skills in the LinkedIn profile with required skills of the job position. This may change due to different requirements of different companies for the same position. Higher the number of matching skills of the candidate has the better chance for selecting to the job position. After training the neural network model, it gives the output as 1 or 0 which represents either selected or rejected. Figure 9 shows the accuracy score obtained from the model is shown below.

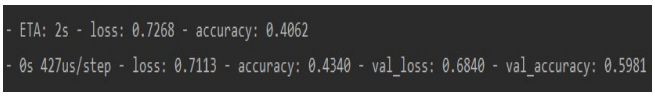


Fig. 9. Accuracy score obtained from LinkedIn

After identifying the relevant modules to the job requirements and their grades from the academic transcript, we classify those modules as technical skills and soft skills using machine learning. The candidates who has the most technical skills relevant to the job requirements will be selected to a comparison of their grades for the respective modules. After the comparison of grades, candidates who have higher grades greater than C+ for the module will be selected and others will be rejected. That would be the output of academic transcript analysis.

V. Conclusion And Future Works

The proposed system assists to evaluate the candidate, seeking employment as a 'Software Engineer' by analyzing their academic transcript, CV, recommendation letter, phone call transcript, GitHub and LinkedIn profile. It is an optimal solution which helps recruiters to select the most suitable candidates for the role of Software Engineer using less resources and time thus makes recruitment process accurate and efficient. In addition, it provides support to candidates where they can evaluate their Skills. In future, companies will experience the improvements in recruitment process, after the proposed system comes to live. The authors of the paper believe that the proposed system will satisfy the candidate's and recruiters requirements and will improve the effectiveness and efficiency of the recruitment process. In the future, this system will be used in relation to other IT related job opportunities and by considering all the four aspects to give an overall mark to prioritize each candidate which make recruiters easy to make decision. Also, further improvements will be done by implementing and validating the video call interview process for selecting candidates.

References

[1] R. W. Zukowski, "Planning for effective technical recruiting," in IEEE Transactions on Engineering Management, vol. EM-12, no. 1, pp. 22-28, March 1965

[2] "The Importance of Information Technology In Business Today", BUSINESS 2 COMMUNITY, 2015. [Online]. Available: https://www.business2community.com/tech-gadgets/importance-information-technology-business-today-01393380. [Accessed: 06-Sep- 2020].

[3] J. Finlay, "What to expect as a software developer", Medium, 2018. [Online]. Available: https://medium.com/@JackWFinlay/what-to-expect-as-a-software-developer-e38b99055e9. [Accessed: 06- Sep- 2020].

[4] D. Bortz, "Top software engineer skills for today's job market," Monster Career Advice. [Online]. Available: https://www.monster.com/career-advice/article/software-engineer-skills. [Accessed: 06-Sep-2020].

[5] N. Majumder, S. Poria, A. Gelbukh and E. Cambria, "Deep Learning-Based Document Modeling for Personality Detection from Text," in IEEE Intelligent Systems, vol. 32, no. 2, pp. 74-79, Mar.-Apr. 2017

[6] R. Bhannarai and C. Doungsa-ard, "Agile person identification through personality test and kNN classification technique," 2016 2nd International Conference on Science in Information Technology (ICSITech), Balikpapan, 2016, pp. 215-219, doi: 10.1109/ICSITech.2016.7852636.

[7] "The Five-Factor Model", Personalityresearch.org. [Online]. Available: http://www.personalityresearch.org/papers/popkins.html. [Accessed: 06- Sep- 2020].

[8] "GitHub API v3," GitHub Developer. [Online]. Available: https://developer.github.com/v3/. [Accessed: 18-Sep-2020].

[9] A. Giri, A. Ravikumar, S. Mote and R. Bharadwaj, "Vritthi - a theoretical framework for IT recruitment based on machine learning techniques applied over Twitter, LinkedIn, SPOJ and GitHub profiles," 2016 International Conference on Data Mining and Advanced Computing (SAPIENCE), Ernakulam, 2016, pp. 1-7.

[10] S. K. Kopparapu, "Automatic extraction of usable information from unstructured resumes to aid search," 2010 IEEE International Conference on Progress in Informatics and Computing, Shanghai, 2010, pp. 99-103

[11] P. P. Shelke and A. A. Pardeshi, "Review on Candidate Feature Extraction and Categorization for Unstructured Text Document," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2020, pp. 88-92, doi: 10.1109/ICCMC48092.2020.ICCMC-00017.

[12] Amarakoon and A. Caldera, "Text mining: Finding right documents from large collection of unstructured documents," The 3rd International Conference on Data Mining and Intelligent Information Technology Applications, Macao, 2011, pp. 5-10.

[13] Dai, K., Nespereira, C., Vilas, A. and Redondo, R., n.d. *Scraping And Clustering Techniques For The Characterization Of Linkedin Profiles*. [online] arXiv.org.

[14] C. Sage, A. Aussem, H. Elghazel, V. Eglin and J. Espinas, "Recurrent Neural Network Approach for Table Field Extraction in Business Documents," 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 2019, pp. 1308-1313, doi: 10.1109/ICDAR.2019.00211.

[15] "How Amazon Transcribe Works - Amazon Transcribe", Docs.aws.amazon.com. [Online]. Available: https://docs.aws.amazon.com/transcribe/latest/dg/how-it-works.html. [Accessed: 06- Sep- 2020].

[16] P. S. Dandannavar, S. R. Mangalwede and P. M. Kulkarni, "Social Media Text - A Source for Personality Prediction," 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), Belgaum, India, 2018, pp. 62-65, doi: 10.1109/CTEMS.2018.8769304.

[17] Toman, Michal, Roman Tesar, and Karel Jezek. "Influence of word normalization on text classification." Proceedings of InSciT 4 (2006): 354-358.

[18] F. Kboubi, A. H. Chabi and M. B. Ahmed, "Table recognition evaluation and combination methods," Eighth International Conference on Document Analysis and Recognition (ICDAR'05), Seoul, South Korea, 2005, pp. 1237-1241 Vol. 2, doi: 10.1109/ICDAR.2005.224.