

**CV ANALYSIS AND OPTIMIZING THE
RECRUITMENT PROCESS IN THE IT INDUSTRY
USING MACHINE LEARNING TECHNIQUES**

Final (Draft) Report

Edirimuni Sandani Zoysa

(IT20231200)

Bachelor of Science (Hons) Degree in Information Technology

Specializing in Data Science

Department of Information Technology

Faculty of Computing

Sri Lanka Institute of Information Technology

Sri Lanka

September 2023

CV ANALYSIS AND OPTIMIZING THE RECRUITMENT PROCESS IN THE IT INDUSTRY USING MACHINE LEARNING TECHNIQUES

Edirimuni Sandani Zoysa

(IT20231200)

Dissertation submitted in partial fulfillment of the requirements for the Bachelor of
Science (Hons) Degree in Information Technology Specializing in Data Science

Department of Information Technology

Faculty of Computing


Sri Lanka Institute of Information Technology

Sri Lanka

September 2023

DECLARATION

We declare that this is our own work, and this proposal does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or Institute of higher learning, and to the best of our knowledge and belief, it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Student ID	Signature
Zoysa E.S.	IT20231200	

The above candidate is carrying out research for the undergraduate Dissertation under my supervision.

.....

26.03.2023

Signature of the supervisor

Date

(Dr. Anuradha Karunasena)

ABSTRACT

The primary focus of this research is the information technology sector, with a specific emphasis on the recruitment of IT-related positions. This study aims to streamline the hiring process in organizations through the integration of machine learning techniques, data extraction techniques, and natural language processing. This research paper examines the process of using data extracted from GitHub and LinkedIn user profiles and Employee reference check feedback to aid in job recruitment. The study analyzes the benefits and limitations of collecting data from these sources and explores the relationship between the collected data and job performance. The research methodology involves data collection from a sample of job candidates' GitHub and LinkedIn profiles as well as reference feedback. The objective is to comprehensively evaluate a candidate's technical skills, professional skills, and preferences by extracting data from candidates generating a candidate profile. Results show that collecting data from these platforms can provide valuable insights into candidates' skills, experiences, and suitability for the job. However, limitations such as the potential for bias in the data and privacy concerns must also be considered. The findings of this study can assist HR professionals in enhancing their recruitment process by incorporating insights from GitHub and LinkedIn profiles.

Keywords: *GitHub, LinkedIn, Rest API, Machine Learning, reference-checking IT industry, user profile, professional, data visualization*

ACKNOWLEDGEMENT

I am pleased to acknowledge the efforts of everyone who has contributed to the preparation of this proposal report. Firstly, I would like to express my gratitude to my supervisor Dr.Anuradha Karunasena, and co-supervisor Dr.Lakmini Abheywardhana for providing me with valuable guidance and support throughout the process. I would also like to extend my appreciation to the research team for their dedication and hard work in conducting the necessary research and analysis.

I would like to acknowledge the support and cooperation of the participants who contributed their time and insights to the study. Lastly, I would like to thank all the stakeholders who have shown interest in this research and provided feedback to help improve its quality. The success of this report is a result of the collaborative efforts of everyone involved, and I am grateful for their contributions.

Sandani Zoysa.

Faculty of Computing,

Sri Lanka Institute of Information Technology.

TABLE OF CONTENT

DECLARATION.....	iii
ABSTRACT.....	iv
ACKNOWLEDGEMENT.....	v
TABLE OF CONTENT.....	vi
LIST OF TABLES.....	viii
LIST OF FIGURES.....	viii
LIST OF ABBREVIATIONS.....	ix
1 INTRODUCTION.....	1
1.1 CV Analysis and Optimizing the Recruitment Process.....	1
1.2 GitHub and LinkedIn.....	2
1.3 Employee Reference Checking.....	2
1.4 Area of Research.....	3
1.5 Component Overview.....	3
2 LITERATURE REVIEW.....	4
2.1 Background Study.....	4
2.2 Literature Survey.....	9
3 RESEARCH GAP.....	12
4 RESEARCH PROBLEM.....	14
5 OBJECTIVE.....	15
5.1 Main Objective.....	15
5.2 Sub Objectives.....	15
6 METHODOLOGY.....	16
6.1 System Architecture Diagram.....	16
6.2 LinkedIn Skills-Based Job Category Prediction.....	16
6.2.1 Model Development.....	16
6.2.2 Prediction.....	18
6.3 GitHub Programming Language Comparator.....	19
6.3.1 Language proficiency for a single candidate.....	19
6.3.2 Language proficiency comparator with peer candidates.....	21

6.4	Sentiment Analysis on Reference Checking	21
7	Commercialization aspect of the product	23
7.1	LinkedIn Skill-based Job Category Prediction.	23
7.2	GitHub Programming Language Comparator	23
7.3	Sentiment Analysis on Reference Checking	24
8	Tools and Technologies.....	24
9	Testing.....	25
9.1	LinkedIn-based job category prediction.....	25
9.2	GitHub Programming Language Proficiency	26
9.3	Sentiment Analysis.....	27
10	Results and Discussion.....	29
10.1	Results	29
10.1.1	LinkedIn-based Job Category Prediction	29
10.1.2	GitHub Programming Language Proficiency	30
10.1.3	Sentiment analysis on reference checking.....	31
10.2	Research Findings.....	31
10.2.1	LinkedIn skills-based job category prediction	31
10.2.2	GitHub Programming Language Proficiency	31
10.2.3	Sentiment analysis on candidate references	32
10.3	Discussion.....	32
10.3.1	LinkedIn skills-based Job Category Prediction	32
10.3.2	GitHub Programming Language Proficiency.	32
10.3.3	Sentiment analysis on candidate references	33
10.3.4	Future Work.....	33
11	Conclusion	34
12	REFERENCES.....	35
13	APPENDICES	xxxvii
13.1	Survey Questionnaire	xxxvii
13.2	Frontend Development	xxxviii
13.3	Turnitin Report.....	xli

LIST OF TABLES

Table 1: Research gap.....	13
Table 2: LinkedIn analysis test case 01	25
Table 3: Linked analysis test case 02.....	25
Table 4: GitHub analysis test case 03	26
Table 5: GitHub analysis test case 04	26
Table 6: GitHub analysis test case 05	27
Table 7:Sentiment analysis test case 06.....	27
Table 8: Sentiment analysis test case 07	28
Table 9: Job category prediction accuracy scores.....	29

LIST OF FIGURES

Figure 1 Social media recruitment in 2021	5
Figure 3:professional platform-survey	7
Figure 2 Current Day Professional Media Platforms	7
Figure 4 Digital Approach to showcase professional work	8
Figure 5 System Architecture Diagram	16
Figure 6: TfidfVectorizer function.....	17
Figure 7: Finding of Most 3 correlated words.	17
Figure 8: scrape LinkedIn skills function	19
Figure 9: fetch GitHub user data.....	19
Figure 10: calculate language proficiency.	20
Figure 11: calculate weighted language scores.	20
Figure 12: language proficiency percentage calculation.....	21
Figure 13: common language proficiency	21
Figure 14: calculate sentiment score.	22
Figure 15:calculate overall sentiment distribution.....	22
Figure 16: Box plot for accuracy	29
Figure 17: Heatmap for model accuracy.....	29
Figure 18: peer candidate comparison	30
Figure 19:Single candidate PLP visualization	30
Figure 20: bar chart visualization for comparison	30
Figure 21:sentiment results for recommendation letters.....	31
Figure 22: sentiment results for Google form responses.	31

LIST OF ABBREVIATIONS

DM	Data Mining
API	Application
NLP	Natural Language processing
ML	Machine learning
HR	Human Resource
PLP	Programming Language Proficiency
IT	Information Technology

1 INTRODUCTION

1.1 CV Analysis and Optimizing the Recruitment Process

The success and growth of an organization are heavily dependent on the selection of the right candidates. The traditional hiring process, which mainly consists of manual steps such as reviewing resumes and educational records, and evaluating technical and professional skills, is both time-consuming and inefficient. To effectively address the needs and expectations of employers, it is crucial to implement a more efficient and accurate method for evaluating a candidate's skills and abilities. Such a method would not only enhance the overall hiring process but also help to ensure that the most suitable candidates are selected for the available roles. It is widely recognized that individuals possess a unique combination of personality traits, resulting in diverse personalities. However, there is currently no universally accepted method for assessing these traits during the hiring process, which can present a challenge in accurately evaluating a candidate's potential fit with a company culture or role. This limitation highlights the need for a more comprehensive approach to personality assessment in the hiring process.

1.2 GitHub and LinkedIn



GitHub is a web-based platform that allows developers to store and manage their code repositories. It provides a collaborative environment for developers to work on projects, track changes to the code, and manage versions of their code. GitHub provides developers with several functions, including issue tracking, pull requests, code reviews, and project management tools, that facilitate collaboration on projects.



LinkedIn is the world's preeminent social network for professionals. Members create CVs and list their current and previous job roles, skills, and education. The business network is also a recruiting website, with businesses able to create profiles and list current vacancies.

(LinkedIn Usage and Revenue Statistics, n.d.)

1.3 Employee Reference Checking

A reference check is a process in which a hiring manager, employer, or recruiter reaches out to a candidate's previous employer to gather additional insights about the candidate's abilities and work history. The primary objective of a reference check is to confirm that the candidate possesses the necessary qualifications for the position as well as gaining a deeper understanding of a candidate's background, past experiences, and skill set.

1.4 Area of Research

A lot of research has been done to evaluate a candidate's professional skills through different aspects and views and aspects. Further, there has been lots of research done to extract data from LinkedIn and GitHub user profiles separately. There are limited numbers of research that has been done for employee reference checking. This research focuses on analyzing and evaluating a candidate's professional skills. Additionally, this research explores the effectiveness of using data from professional media platforms, such as LinkedIn and GitHub, to assess a candidate's professional skills. So, HR could investigate whether this method provides a more accurate representation of a candidate's skills than traditional methods, such as resumes and interviews, and whether it can help recruiters identify top talent more efficiently.

1.5 Component Overview

The focus of this research component is to develop a way to identify a candidate's professional skills using the candidate's LinkedIn and GitHub user-profiles and Employee reference checking. Although these sources provide basic information along with their UI, they are more focused on representing directly what the user has directly input to the platform. This component will provide a solution so that HR will be able to understand the summary inspection and representation of the candidate's professional skills through this embedded tool.

2 LITERATURE REVIEW

2.1 Background Study

In today's digital age, companies are increasingly relying on automated online job recruitment processes to screen potential candidates quickly and efficiently. More importantly in today's competitive job market, recruiters are shifting their approach to hiring by assessing candidates beyond the confines of their CVs. This transformation is driven by the need for a deeper understanding of potential hires. CV offers a glimpse of qualifications and experiences, but recruiters now seek a more comprehensive perspective. By delving further, recruiters can validate skills, evaluate cultural fit, and gauge a candidate's adaptability and soft skills, all of which are vital in modern workplaces. This holistic evaluation not only promotes diversity and inclusion but also ensures that candidates align with the organization's values and can thrive in ever-evolving roles. Ultimately, it equips recruiters to make more informed and strategic hiring decisions, benefiting both the candidates and the organizations they join. As part of this process, candidate profiles on social media and professional networking sites such as GitHub and LinkedIn are being analyzed to extract valuable information about their professional qualifications. Employee reference checking is being analyzed as it is a valuable tool for employers to make well-informed hiring decisions, reduce risks associated with new hires, and build a workforce that aligns with the company's goals and values. It contributes to a more effective and successful hiring process.

The purpose of this research is to explore the effectiveness of extracting metadata from Professional networking sites like GitHub and LinkedIn and Reference checking then using this information to represent their professional qualifications. The study will focus on the use of automated tools to collect and analyze data from candidate profiles and the potential benefits of using this approach in the recruitment process.

Traditionally, job recruitment processes involved manually reviewing resumes and conducting interviews to determine a candidate's qualifications. However, with the advent of social media and professional networking sites, recruiters now have access to a wealth of information about candidates that can be used to supplement traditional recruitment methods. Which is called social media recruitment.



Figure 1 Social media recruitment in 2021

A survey done by
“ISmartRecruiters” in 2021
<https://www.ismartrecruit.com/blog-social-media-recruiting-practices>

GitHub and LinkedIn are two popular platforms used by professionals to showcase their work experience, skills, and qualifications. GitHub is a code hosting platform that allows IT developers to collaborate on projects and share code. LinkedIn is a professional networking site that allows users to showcase their work experience, skills and connect with other professionals. Automated tools can be used to extract metadata from candidate profiles on these platforms, such as programming languages used, projects completed, and endorsements from colleagues. This information can be used to represent a candidate's professional qualifications and can provide valuable insights into their skills and experience.

Advocates believe that reference checks are a crucial step in the hiring process. They see it as a way to make sure the person is right for the job. It's like a final check to be sure. They also believe that it's not just about getting a "yes" or "no" from past employers. Instead, it's about learning what the person is good at and what they may need help within their new job. It's like trying to understand them better so they can do well in their new role.

The utilization of professional networking platforms, such as LinkedIn and GitHub, has become increasingly prevalent in the hiring process. Employers are now leveraging these platforms to assess a candidate's professional skills and evaluate their potential fit for a role. In today's highly competitive job market, being able to identify top talent efficiently and accurately has become critical for organizations to maintain a competitive edge.

Extracting data from LinkedIn and GitHub profiles provides a wealth of information about a candidate's skills and experience. LinkedIn profiles, for example, provide details about a candidate's skills, education, work history, endorsements, and recommendations. Meanwhile, GitHub provides insight into a candidate's coding abilities, including the programming languages they know, the projects they have worked on, and the contributions they have made to open-source projects. By doing a proper employee reference check recruiters can get a deeper understanding as well as a candidate's personality, soft skills, and cultural fit within the organization.

While the use of professional networking sites in the hiring process has its advantages, it also presents some challenges. For example, recruiters may struggle to analyze large amounts of data and extract meaningful insights. Additionally, there may be ethical considerations related to the use of personal data and privacy concerns that need to be addressed.

This research aims to explore the effectiveness of extracting data from LinkedIn, GitHub user profiles, and employee reference checking feedback to evaluate a candidate's professional skills. Specifically, the paper will investigate whether this approach provides a more accurate representation of a candidate's skills than traditional methods, such as resumes and interviews. Additionally, the paper will discuss the ethical considerations of using personal data for hiring decisions and explore ways to ensure that the data collected is used fairly and transparently. By providing insights into this topic, this paper aims to help organizations improve their recruitment processes and identify top talent more efficiently.

7. What is/are the professional platforms you currently using to showcase you professional work?

111 responses

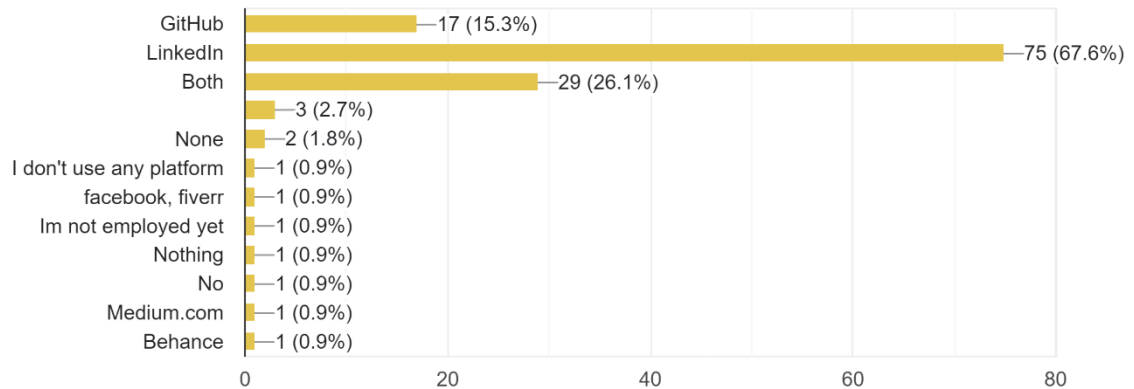


Figure 2: professional platform-survey

Figure 2 shows that most people are using professional social platforms to showcase the professional work they have done. Further, this figure shows people are more likely to in GitHub (15.3%) and LinkedIn (67.6%) compared to other platforms. Some of them are using both platforms which represent 29.1%

According to a study done, based on its global advertising audience reach numbers, LinkedIn had at least **900.2 million** members around the world in January 2023. It suggests that **16.0%** of all people aged 18 and above around the world have an account on LinkedIn today. (LinkedIn Statistics and Trends, n.d.) And when it comes to GitHub The total number of developers on GitHub now stands at more than 73 million, with more than 16 million users added in 2021 alone. According to figures released in last year's October report, this number will increase to 100 million by 2025. The number of first-time contributors increased significantly in the last year to more than 3 million, the highest number of new contributions since 2015. (GitHub Global Growth, n.d.)

9. What is your main reason for using GitHub/LinkedIn to showcase your professional skills?

111 responses

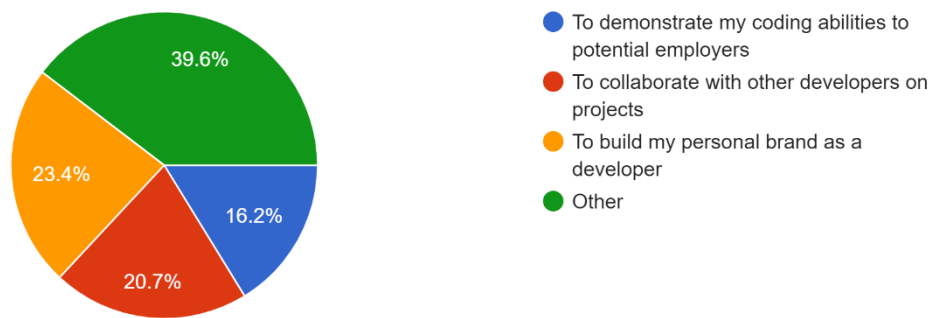


Figure 4 Digital Approach to showcase professional work.

Figure 2.2 shows what motivates job seekers to use professional platforms to showcase their work achievements. From the figure, it is visible that 23.4% of the people use these platforms to build their personality brand as a developer and 20.7% are using them to collaborate with other developers on projects.

2.2 Literature Survey

In [1] Junalux Chalidabhongse and others discuss how tedious the selection process that the HR department has to perform in every organization. Also, they discuss the importance of introducing a novel method for the manual selection process. Further, traditional hiring processes rely heavily on resumes and interviews, these methods may not be enough to fully evaluate a candidate's professional skills. In this literature survey, we aim to critically analyze the existing literature on the use of LinkedIn, GitHub profiles, and Employee reference checking in evaluating professional skills during the candidate hiring process. Specifically, we will review the literature on the types of skills that can be assessed through these platforms, the methodologies used to evaluate skills, and the effectiveness of using these platforms to identify top talent. Through this literature survey, we hope to provide insights into the current state of knowledge on this topic and identify the best practices for using these platforms in the candidate hiring process.

In recent years, the IT industry has experienced a surge in recruitment, resulting in a growing need for recruiters and talent scouts to find suitable technical candidates for preliminary assessments. However, they are facing difficulties in identifying potential candidates solely based on resumes.

Research done in 2022 [2] discusses how using novel methods to inspect public GitHub repositories can help manage the influx of candidate profiles during the recruitment process. The manual process of reviewing repositories is both time-consuming and prone to inaccuracies, highlighting the need for a more efficient tool. The paper explores the use of GitHub REST API to gather information on potential candidates from their user profiles. They have come up with a solution as a skill assessment through REST APIs and the importance of code repositories. Simply, it is a web application developed using the REACT framework and it has a search box when a GitHub username is entered it will fetch user details and repositories for the respected user and display them after creating search queries.

In [3] it has proposed a pre-screening solution to screen the applicants for the position of Software Engineer where candidates are screened based on GitHub profile, LinkedIn, and other sources. From GitHub, they have extracted repositories, commits, and respective programming languages directly to report using GitHub Users API, Repository API, and

Organizations API using a two-factor authentication. Further, they have facilitated to report extracted summary of a CV.

Another research was done in 2016 [4] for social coding platforms like GitHub, software developers can display their skills and expertise in specific areas of software development by evaluating their source code contributions. This allows them to showcase their experience and competencies across different projects and programming languages to the community and potential employers. To determine the level of contribution, the quantity and continuity of the developer's commits are taken into consideration across various isolated projects over time. By assessing these factors, a clear view of the developer's capabilities can be gained, allowing them to effectively demonstrate their skills and experience to potential employers. They have used Google's big query service through which they retrieved data related to GitHub. They have chosen to keep every project feature like name, email, blog, created date, URL, location, no of public repositories, etc. Fragkiskos Chatziasimidis and others have used the Apriori algorithm for data mining through the *Weka* tool to identify frequent item sets. Then apply the k-means algorithm to discretize feature values. From the results they have shown that a large number of projects have just one download and very few have a lot of number of downloads including distribution of languages, open issues, number of followers, and number of followers.

In [5] it has a study on “*newcomer*” candidates to explore new users to the GitHub platform. Research has further tracked and characterized the initial contributions of 208 newcomer candidates in GitHub using a mixed-methods approach, analyzed whether candidates who are new to a particular coding community engage in social coding practices, and examined the types of contributions they make and the projects they focus on.

Another research was done in 2021 [6] as a feasibility study using the use of Facebook data to enhance the recruitment system's performance and determine a candidate's personality through social network analysis. This highlights the significant role of social media platforms like Facebook and LinkedIn in people's lives, where they tend to spend a lot of time posting updates and uploading certifications. The system is designed to verify the identities of individuals by analyzing data from LinkedIn, Facebook, or both. This is achieved by data scraping and a machine learning algorithm that utilizes a previously trained dataset with SVM.

In [7] Brandon Gibson and others analyze the data of LinkedIn in 2021. It used LinkedIn's overly invasive API to scrape a massive amount of personal information data. By linking this information with other API sources, the attacker was able to generate a comprehensive list of data, which was then sold illicitly over the internet. This research investigates the impact of a specific event and proposes possible strategies to mitigate its effects. The suggested countermeasures include implementing adequate authorization and authentication procedures, limiting data scraping activities, and utilizing anomaly detection techniques.

Another study was done [8] in 2021 as An examination of different social media platforms reveals that the substantial volume of data generated by users on these platforms is a valuable source of unstructured information that can be used to study public opinion and gauge people's sentiments. This data can help capture views on social events and corporate strategies. The data is collected from different services and programs by utilizing official APIs and scraping tools. To extract valuable insights about individual users' interests, usage patterns, and media reach, machine learning and image processing algorithms are applied to the gathered data. These algorithms help in analyzing the data, thereby providing valuable insights, and enabling businesses and researchers to understand the users' preferences and behaviors on a deeper level.

R.T.R Jayasekara and others have proposed a framework in a paper [9] to automate the pre-screening of Software Engineer candidates. They have used platforms like social media, GitHub, LinkedIn, and an open-ended questionnaire to predict the big five traits. Here they use a Technical Skill Usage Analysis Model (TSUAM) using GitHub extracted data. It is an API-based data extraction to assess candidate technical skills as a skills usage percentage that is chosen by the recruiter. Also, they use the Personality Prediction Model (PPM) to evaluate candidate personality traits using LinkedIn scraped data. For this purpose, they have used a third-party API called "*Proxycurl*" to extract the 'about' and 'headline' sections of the LinkedIn profile of a user. Pre-processing and normalizing the scraped data using the Doc2Vec model. Finally, a predictor that contains five models related to each Big Five Trait. In [3] Gajanayake and others analyzed LinkedIn profiles with academic transcripts in the process of selecting a candidate for the position of Software Engineering. They have scraped technical skills in the LinkedIn profile using an automated script created with selenium. And then train a

model to match the required programming languages for the software engineering position.

When it comes to reference checking R.G.U.S.Gajanayake and others have utilized recommendation letters to analyze candidate skills, backgrounds, and capabilities before the pre-screening process. They have built a machine learning model to identify the competencies as highly recommend, recommend, neutral, and weakly recommended using the identified levels.

3 RESEARCH GAP

According to the literature survey done, the following issues were found as research gaps,

- Limited system tool only to extract raw data.
- Systems use either LinkedIn or GitHub separately.
- Existing systems lack analysis of LinkedIn data.
- Employee job category prediction before screening.
- Not reliable and automated employee reference checking.

So, this research is done to provide a solution to these identified gaps by developing a system with the ability to showcase candidates' professional skills from GitHub, LinkedIn user profile data, and employee reference check feedback.

When we consider existing implementations in this area:

- In Research A [8], the authors have presented user profiling using text and images for topic modeling, sentiment labeling, and image labeling from Facebook, Instagram, and Twitter.
- In Research B [6], the authors have proposed a personality prediction by scraping data from digital footprints and classifying them using a machine learning approach that previously trained datasets using SVM.
- In Research C [3], the authors have proposed a system where data extracted from the CV and the programming Languages obtained from GitHub will be further

analyzed and then displayed as the candidate’s insights.

- In Research D [2] , the authors have proposed to evaluate the students by providing them with the assignments on GitHub and asking them to upload their solutions on it so that they can view their code easily.
- In [9], they have proposed a method to assess the technical skills of a candidate through GitHub data giving a score and assessing the LinkedIn scraped “header” and “about” section for predicting the personality traits of a candidate.

Feature Research	Based GitHub	Based on LinkedIn	Evaluate Profession al skills	Sentiment analysis on reference checking	LinkedIn- based Job Category Prediction
Research A [8]	✗	✓	✗	✗	✗
Research B [6]	✗	✓	✗	✗	✗
Research C [3]	✓	✗	✓	✗	✗
Research D [2]	✗	✗	✓	✗	✗
Research E [9]	✓	✓	✓	✗	✗
Proposed system	✓	✓	✓	✓	✓

Table 1: Research gap

4 RESEARCH PROBLEM

The research question for this study is as follows:

- Are CVs reliable? Can we go beyond the CV and assess a candidate?
 - CVs are typically self-reported documents that are created by candidates. It should not be considered as the sole indicator of a candidate's qualifications or suitability for a job.
 - This research will focus on identifying a candidate's true identity by analyzing their professional media accounts.
- Can we optimize the recruitment process without human interference?
 - Automation can handle repetitive and time-consuming tasks allowing human recruiters to focus on more strategic and value-added activities, such as interviewing, relationship-building, and decision-making.
 - This research will deliver candidates' professional preferences depending on posts and repositories shared on GitHub and LinkedIn by doing the topic modeling. It will further help in the interview process.
- Can we go for better decision-making with trending technologies?

Research problems can be conveyed when there is an incoming flood of possible candidates for position or vacant, finding what is the effectiveness of using automated tools to extract metadata from candidate profiles on GitHub and LinkedIn and analyzing employee reference feedback to represent their professional qualifications in an automated online job recruitment process.

Therefore, it is necessary to develop a feature where HR would be able to find the most suitable candidate productively with less time and more matching. This will enable an HR person to reduce human interference and speed up the manual process by referring to this tool even if there are flood of candidates, which is the case in many situations.

Here are some reasons to understand the need to automate your recruitment process.

1. Save time and money.

2. Reach out to a wider pool of candidates.
3. Keep the recruitment process organized.
4. Make better hiring decisions.
5. Leverage social site recruiting.
6. Remove bias and build diversity.

5 OBJECTIVE

5.1 Main Objective

The main objective of this component is to allow a HR system to evaluate and confirm candidates beyond the CV with less human interference as time time-saving and reliable solution in the recruitment process.

5.2 Sub Objectives

The following sub-objectives should be fulfilled to achieve the specified objective.

- Extracting the content of the candidate's LinkedIn user profile.
This objective aims to analyze the extracted candidate LinkedIn profile data using the LinkedIn REST API and then summarize the skill set mentioned in a user profile. This will give the opportunity to understand overall skills through LinkedIn.
- Extracting content of the candidate's GitHub user profile.
This objective aims to analyze the extracted candidate GitHub profile data using the GitHub REST API and then summarize the repositories. Programming languages, followers, repository counts, and commits. This will give the opportunity to understand professionalism through GitHub.
- Analyzing employee references.
This objective aims to do an effective and time-consuming analysis of proper employee reference checking.
- A machine learning approach to predict the job category of a candidate.
- Proper visualization methods to graphically represent the necessary information insightfully.

6 METHODOLOGY

6.1 System Architecture Diagram

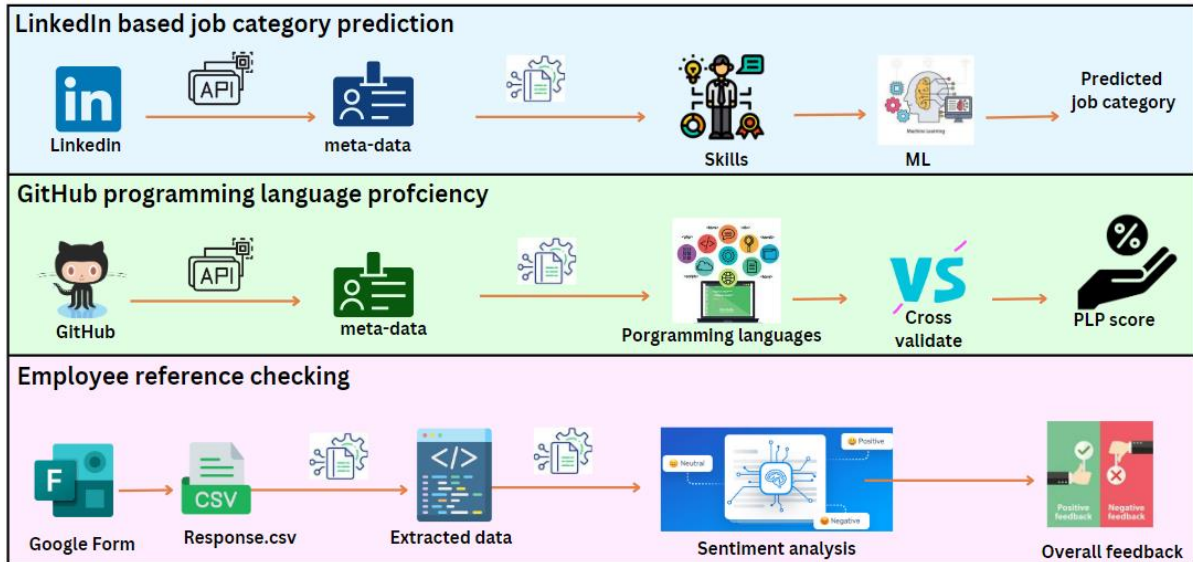


Figure 5 System Architecture Diagram

6.2 LinkedIn Skills-Based Job Category Prediction

6.2.1 Model Development

Dataset - This dataset, vital for our IT job research, was methodically constructed by combining data from Kaggle, job descriptions, advertisements, vacancies, and IT recruiter validation. And it is crucial for the research on IT job categories, roles, and skills. Through careful data cleaning and integration, it has established a solid research foundation while upholding data privacy and reliability. Beyond the research, this dataset offers valuable insights for the broader IT community, facilitating workforce planning and industry analysis.

A multi-class text classification model under supervised machine learning was used to predict the matching or classifying of job candidates into specific job categories.

Text Processing

This involves converting the text into numerical vectors through the Term Frequency-Inverse Document Frequency (TF-IDF) method. This technique allows us to assess the significance of individual words within the entire corpus. To achieve this, we first ensure uniformity by converting all text to lowercase and eliminating punctuation. Subsequently, we calculate the importance of each word based on its frequency within the text. In essence, TF-IDF transforms our text data into a format that the classification algorithm can comprehend, highlighting the importance of words in the broader context of the text collection. The `TfidfVectorizer` function was used with the parameters below.

- `min_df` - Exclude words that appear in fewer than '`min_df`' documents.
- `Sublinear_tf` – Scale the frequency in logarithmic scale when it is true.
- `Stop_words`- remove pre-defined English words.

```
tfidf = TfidfVectorizer(sublinear_tf=True, min_df=5,
                        ngram_range=(1, 2), # consider both unigrams and bigrams
                        stop_words='english')
# Transform each skill into a vector
features = tfidf.fit_transform(df1.clean_skills).toarray()
labels = df1.category_id
```

Figure 6: `TfidfVectorizer` function

Then identify the terms that exhibit the highest correlation with each of the specified product categories. This process involves pinpointing words or phrases that are strongly associated with each category, shedding light on the key features or attributes that distinguish them from one another.

```
# Finding the three most correlated terms with each of the job categories
N = 3 # top 3 most correlated terms
for category, category_id in sorted(category_to_id.items()):
    features_chi2 = chi2(features, labels == category_id)
    # computes the chi-squared statistic and p-values for each feature against the sel
    indices = np.argsort(features_chi2[0])
    feature_names = np.array(tfidf.get_feature_names_out())[indices] # chi-squared scc

    unigrams = [v for v in feature_names if len(v.split(' ')) == 1]
    bigrams = [v for v in feature_names if len(v.split(' ')) == 2]
    print("n==> %s:" %(category))
    print(" * Most Correlated Unigrams are: %s" %(', '.join(unigrams[-N:])))
    print(" * Most Correlated Bigrams are: %s" %(', '.join(bigrams[-N:])))
```

Figure 7: Finding of Most 3 correlated words.

Explore Text Multi-Class Classification Models

Classification models that were used were,

- Random Forest
- Linear Support Vector Machine
- Multinomial Naive Bayes
- Logistic Regression.
- XGBoost Classifier

Next divided the data into two subsets: a training set and a test set allocating 75% of the data for training purposes, and the remaining 25% will be used for testing. In this split, the 'required skills' column will serve as our input data (X), while the 'Job Category' column will be our output (Y). This division allows us to train our model on a substantial portion of the data and then assess its performance on unseen data to evaluate its effectiveness in predicting job categories based on candidate LinkedIn-based skills.

All five models were kept using in list iterating the list for each model to get mean accuracy and standard deviation to compare the performance of each model. For the best-performing model cross-validation and hyperparameter tuning were done to tweak model performance for optimal results.

Compare Text Classification Model performance.

Compare the Mean Accuracy and the Standard Deviation for each of the five text classification algorithms. Then identified the best performing model to train model multi-class classification tasks.

Text Classification Model Evaluation - Trained the model using the best-performing model to evaluate and check unseen data.

6.2.2 Prediction

LinkedIn profile data was extracted using a third-party paid API service called 'ProxyCurl' to retrieve candidate skills added to the LinkedIn profile under "**skills**". The candidate's LinkedIn profile must be public to extract the skills for prediction. This process was done using the function below.

```

def scrape_linkedin_skills(linkedin_profile_url):
    api_key = 'MxVw1MuCI00hrmsugxwLjA'
    api_endpoint = 'https://nubela.co/proxycurl/api/v2/linkedin'
    headers = {'Authorization': 'Bearer ' + api_key}

    response = requests.get(api_endpoint,
                             params={'url': linkedin_profile_url, 'skills': 'include'},
                             headers=headers)

    profile_data = response.json()

    if 'skills' in profile_data:
        return profile_data['skills']
    else:
        return []

```

Figure 8: Scrape LinkedIn skills function

Finally, run the prediction for the skills scraped through the above function.

6.3 GitHub Programming Language Comparator

6.3.1 Language proficiency for a single candidate

- GitHub user profile data was used to evaluate Programming language proficiency in this research.
- GitHub REST API with PyGitHub modules was used to fetch data from candidate profiles using a personal access token to authenticate with GitHub.
- Then fetched the user's repositories belonging to the user with a GET request. `‘users/{username}/repos’`

```

# Fetch user information
current_username = "Maldeniya99"
current_user = g.get_user(current_username)

```

Figure 9: Fetch GitHub user data

- Iterate through the repositories returned in the response and generate a repository object

for each repository by using a loop.

- Retrieve the repository language proficiency using the `'get_languages()'` method which returns the dictionary where the keys are languages used in the repository and the values are the number of bytes written in that language. If there are languages present, it will iterate over the dictionary of languages and their respective line counts and print them. If no languages are detected, it will print "Languages: None" for that repository.
- Calculated the language proficiency by iterating over all the user's repositories and retrieving the languages used in each repository by accumulating the line counts for each language.

```
# Calculate language proficiency
language_proficiency = {}
repository_count = user.public_repos

for repo in user.get_repos():
    languages = repo.get_languages()
    for language, value in languages.items():
        if language in language_proficiency:
            language_proficiency[language] += value
        else:
            language_proficiency[language] = value
```

Figure 10: Calculate language proficiency.

- Next calculate the weighted language proficiency scores by multiplying the language proficiency value by the reciprocal of the user's repository count.

```
# Calculate weighted language proficiency scores
weighted_scores = {}

for language, value in language_proficiency.items():
    weighted_score = value * (1 / repository_count)
    weighted_scores[language] = weighted_score
```

Figure 11: Calculate weighted language scores.

- Then the total score by summing the weighted score.
- Normalized the weighted scores by dividing each score by the total score and multiplying

by 100 to convert them to percentages. The normalized percentage scores are stored in the **percentage_scores** dictionary.

```
# Normalize and convert scores to percentages
total_score = sum(weighted_scores.values())

percentage_scores = {
    language: (score / total_score) * 100
    for language, score in weighted_scores.items()
}
```

Figure 12: Language proficiency percentage calculation

6.3.2 Language proficiency comparator with peer candidates

- In the comparator, it stores the weighted scores in separate dictionaries for each user and then compares the language proficiency for common languages assuming that a user with more repositories may have a broader range of language proficiency.

```
# Compare language proficiency
common_languages = set(current_user_language_proficiency.keys()) & set(other_user_language_proficiency.keys())
```

Figure 13: Common language proficiency

6.4 Sentiment Analysis on Reference Checking

- A pre-created Google form was sent to referees mentioned in the CV. This Google form was created surveying how to do a proper and reliable reference checking on candidates from internet resources and in recruiters' view covering all the aspects of a candidate when it comes to reference checking.
- Google Form Link here - <https://forms.gle/1HCVJ1F7efaM2vhh8>
- This Google form was sent to a referee to answer the questions related to candidate reference checking.
- Responds will saved as CSV files.
- “pandas” and library were used to read CSV files and pre-process the text.
- “SentimentIntensityAnalyzer” from VADER sentiment analysis tool to perform sentiment on the text.
- ‘SentimentIntensityAnalyzer’ from NLTK calculates the sentiment score which is a compound value that ranges from -1 (negative sentiment) to 1 (positive sentiment).
- CSV data was read into a DataFrame and then sentiment analysis was performed on all

the columns except the first column as it contains the Timestamp of the Google form.

- Then sentiment scores were calculated for each response column referring to “SentimentIntensityAnalyzer” function.

```
# Step 2: Perform sentiment analysis
sid = SentimentIntensityAnalyzer()

# Calculate sentiment scores for each column
sentiment_scores = []
for column in columns_to_analyze:
    column_scores = [sid.polarity_scores(str(response)) for response in data[column]]
    sentiment_scores.extend(column_scores)
```

Figure 14: Calculate sentiment score.

- Finally, visualize the sentiment distribution across all columns collectively in a single pie chart.

```
# Calculate overall sentiment distribution
labels = ['Positive', 'Negative', 'Neutral']
sentiment_distribution = [0, 0, 0]

for score in sentiment_scores:
    max_sentiment = max(score, key=score.get)

    if max_sentiment == 'pos':
        sentiment_distribution[0] += 1
    elif max_sentiment == 'neg':
        sentiment_distribution[1] += 1
    else:
        sentiment_distribution[2] += 1
```

Figure 15: calculate overall sentiment distribution.

- This function is designed to retrieve the latest entry in a CSV file, with the assumption that it represents the most recent candidate’s referee who has submitted their information through a Google Form.

7 Commercialization aspect of the product

Contemporary recruitment automation systems exhibit a limitation in their ability to comprehensively evaluate applicants beyond the information provided in their CVs. This component is engineered to address this deficiency by offering an integrated and efficient solution for assessing candidates through their GitHub and LinkedIn profiles, as well as streamlining the process of conducting reference checks. Its implementation enhances the overall effectiveness of candidate evaluation while saving valuable time in the recruitment process.

The service can be offered at various tiers, with different pricing and features based on the volume of data extraction and level of customization required by the client. The software tool can also be marketed as a value-added service to complement existing recruitment software solutions.

This software tool will,

- Reach a wider audience with various backgrounds.
- It is modern, easy, inexpensive, and targeted.
- Gain more information about candidates.
- This component will have two versions.
- Free version - basic features
- Premium version - advanced features
- The target audience would be HR in the IT industry.

7.1 LinkedIn Skill-based Job Category Prediction.

The dataset, comprising job categories, job roles, and required skills, can be tailored to align with the specific preferences and needs of the hiring company. This customization allows for the generation of more effective predictions, which can vary based on the unique criteria and specifications of each company.

7.2 GitHub Programming Language Comparator

In the GitHub language comparator hiring companies can decide what batch size to compare at once based on their requirements and preferences.

7.3 Sentiment Analysis on Reference Checking

The Google form which is sent to referees can be customized according to the requirements of the recruiters.

8 Tools and Technologies

Python was selected as the primary programming language for the development of this system due to its versatility and the availability of a wide range of libraries. These libraries proved invaluable in achieving the predefined objectives of the project.

- **PyGitHub** - allows developers to interact with the GitHub platform programmatically.
- **Matplotlib** - library for creating static, animated, and interactive visualizations in various formats.
- **Io** - provides tools for working with input and output streams. It includes classes like **BytesIO** and **StringIO** for handling binary and text data as file-like objects.
- **Base64** - commonly used for representing binary data, such as images or binary files, as ASCII text.
- **NLTK** - working with human language data (text). It provides easy-to-use interfaces to over 50 corpora and lexical resources.
- **Pickle** - module in Python that provides a way to serialize and deserialize Python objects.
- **VADER** - It is a lexicon and rule-based sentiment analysis tool specifically designed for analyzing text sentiment in the English language.
- **TfidfVectorizer** - class in Python's sci-kit-learn library (sklearn) that is used for converting a collection of raw documents (text data) into a numerical feature matrix based on the TF-IDF (Term Frequency-Inverse Document Frequency) representation.
- **chi2** - function to perform chi-squared tests for feature selection in classification tasks.

9 Testing

9.1 LinkedIn-based job category prediction.

Table 2: LinkedIn analysis test case 01

Test Case	01
Description	Testing LinkedIn skills-based job category classification for famous personality
Input	<pre># Example LinkedIn profile URL linkedin_profile_url = 'https://www.linkedin.com/in/chanuxbro/'</pre>
Expected Outcome	Executive Leadership / Infrastructure and Operations
Actual output	<pre># Example LinkedIn profile URL linkedin_profile_url = 'https://www.linkedin.com/in/chanuxbro/' # Call the function to get skills skills = scrape_linkedin_skills(linkedin_profile_url) predicted_category = model.predict(fitted_vectorizer.transform(skills)) result = f"Predicted Job Category: {predicted_category[0]}" print(result)</pre> <p>Predicted Job Category: Infrastructure and Operations</p>
Result	Pass

Table 3: Linked analysis test case 02

Test Case	02
Description	Testing LinkedIn skills-based job category classification for a team member
	<pre># Example LinkedIn profile URL linkedin_profile_url = 'https://www.linkedin.com/in/maleesha-de-silva-37a26a221/'</pre>
Expected Outcome	Software Engineering
Actual output	<pre># Example LinkedIn profile URL linkedin_profile_url = 'https://www.linkedin.com/in/maleesha-de-silva-37a26a221/' # Call the function to get skills skills = scrape_linkedin_skills(linkedin_profile_url) predicted_category = model.predict(fitted_vectorizer.transform(skills)) result = f"Predicted Job Category: {predicted_category[0]}" print(result)</pre> <p>✓ 0.8s</p> <p>Predicted Job Category: Software Development and Engineering</p>
Result	Pass

9.2 GitHub Programming Language Proficiency

Table 4: GitHub analysis test case 03

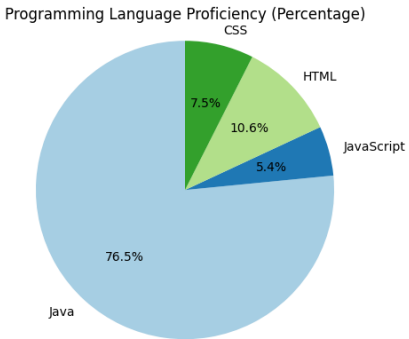
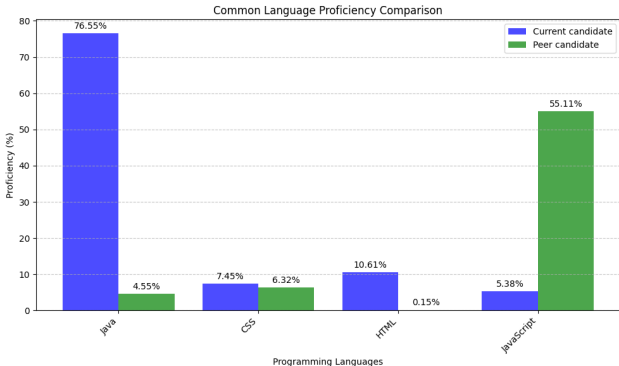
Test Case	01
Description	Testing programming language proficiency of student in SLIIT
Input	<pre># Fetch user information username = "Maldeniya99"</pre>
Expected Outcome	Proficient in Java
Actual output	 <p>Programming Language Proficiency (Percentage):</p> <ul style="list-style-type: none"> - Java: 76.55% - JavaScript: 5.38% - HTML: 10.61% - CSS: 7.45%
Result	Pass

Table 5: GitHub analysis test case 04

Test Case	02
Description	Testing programming language proficiency of student in SLIIT
Input	<pre># Fetch user information username = "Maldeniya12345"</pre>
Expected Outcome	There should not be a user account under the name: Maldeniya12345
Actual output	<pre>1] ✓ 0.7s · User 'Maldeniya12345' not found.</pre>
Result	Pass

Table 6: GitHub analysis test case 05

Test Case	03															
Description	Testing Programming language proficiency for two peer candidates.															
Input	<pre># Fetch user information current_username = "Maldeniya99" current_user = g.get_user(current_username) other_username = "IT20207854" other_user = g.get_user(other_username)</pre>															
Expected Outcome	Common languages – Java, CSS, HTML, Javascript															
Actual output	 <table><caption>Common Language Proficiency Comparison</caption><thead><tr><th>Programming Languages</th><th>Current candidate</th><th>Peer candidate</th></tr></thead><tbody><tr><td>Java</td><td>76.55%</td><td>4.55%</td></tr><tr><td>CSS</td><td>7.45%</td><td>6.32%</td></tr><tr><td>HTML</td><td>10.61%</td><td>0.15%</td></tr><tr><td>Javascript</td><td>5.38%</td><td>55.11%</td></tr></tbody></table>	Programming Languages	Current candidate	Peer candidate	Java	76.55%	4.55%	CSS	7.45%	6.32%	HTML	10.61%	0.15%	Javascript	5.38%	55.11%
Programming Languages	Current candidate	Peer candidate														
Java	76.55%	4.55%														
CSS	7.45%	6.32%														
HTML	10.61%	0.15%														
Javascript	5.38%	55.11%														
Result	Pass															

9.3 Sentiment Analysis

Table 7:Sentiment analysis test case 06

Test Case	01
Description	Testing and applying sentiment analysis truly filled form.
Input	<p>A CSV file containing referee responses.</p> <pre># Read CSV file into a DataFrame csv_file_path = 'Responses.csv' data = pd.read_csv(csv_file_path)</pre>
Expected Outcome	There should be real-world output with a mix of all sentiments.

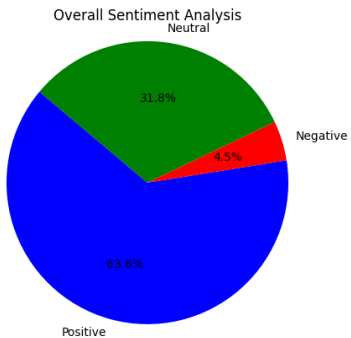
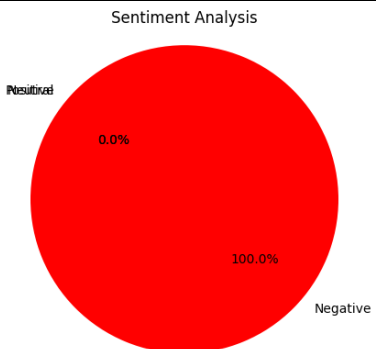
Actual output	 <p>Overall Sentiment Analysis</p> <table border="1"> <thead> <tr> <th>Sentiment</th> <th>Percentage</th> </tr> </thead> <tbody> <tr> <td>Positive</td> <td>63.6%</td> </tr> <tr> <td>Neutral</td> <td>31.8%</td> </tr> <tr> <td>Negative</td> <td>4.5%</td> </tr> </tbody> </table>	Sentiment	Percentage	Positive	63.6%	Neutral	31.8%	Negative	4.5%
Sentiment	Percentage								
Positive	63.6%								
Neutral	31.8%								
Negative	4.5%								
Result	Pass								

Table 8: Sentiment analysis test case 07

Test Case	02						
Description	Testing and applying sentiment analysis purposely filling the Google form with negative.						
Input	A CSV file containing the referee's responses (intentionally negatively filled) <pre># Read CSV file into a DataFrame csv_file_path = 'Responses.csv' data = pd.read_csv(csv_file_path)</pre>						
Expected Outcome	Negative Sentiment should be the highest one.						
Actual output	 <p>Sentiment Analysis</p> <table border="1"> <thead> <tr> <th>Sentiment</th> <th>Percentage</th> </tr> </thead> <tbody> <tr> <td>Positive</td> <td>0.0%</td> </tr> <tr> <td>Negative</td> <td>100.0%</td> </tr> </tbody> </table>	Sentiment	Percentage	Positive	0.0%	Negative	100.0%
Sentiment	Percentage						
Positive	0.0%						
Negative	100.0%						
Result	Pass						

10 Results and Discussion

10.1 Results

10.1.1 LinkedIn-based Job Category Prediction

To predict the job category five models were tested for accuracy to select the best performing classification model. These models took the skills as input, it analyzed the skills, and identified the job category. Further, the hyperparameters for these classifiers are tuned to get better after cross-validation for each. The following tables show the accuracy of each model.

Model	RandomForest Classifier	LinearSVC	MultinomialNB	Logistic Regression	XGB Classifier
Mean Accuracy	0.5790	0.6638	0.5505	0.5790	0.4371
SD	0.1199	0.1314	0.1094	0.1432	0.0963

Table 9: Job category prediction accuracy scores.

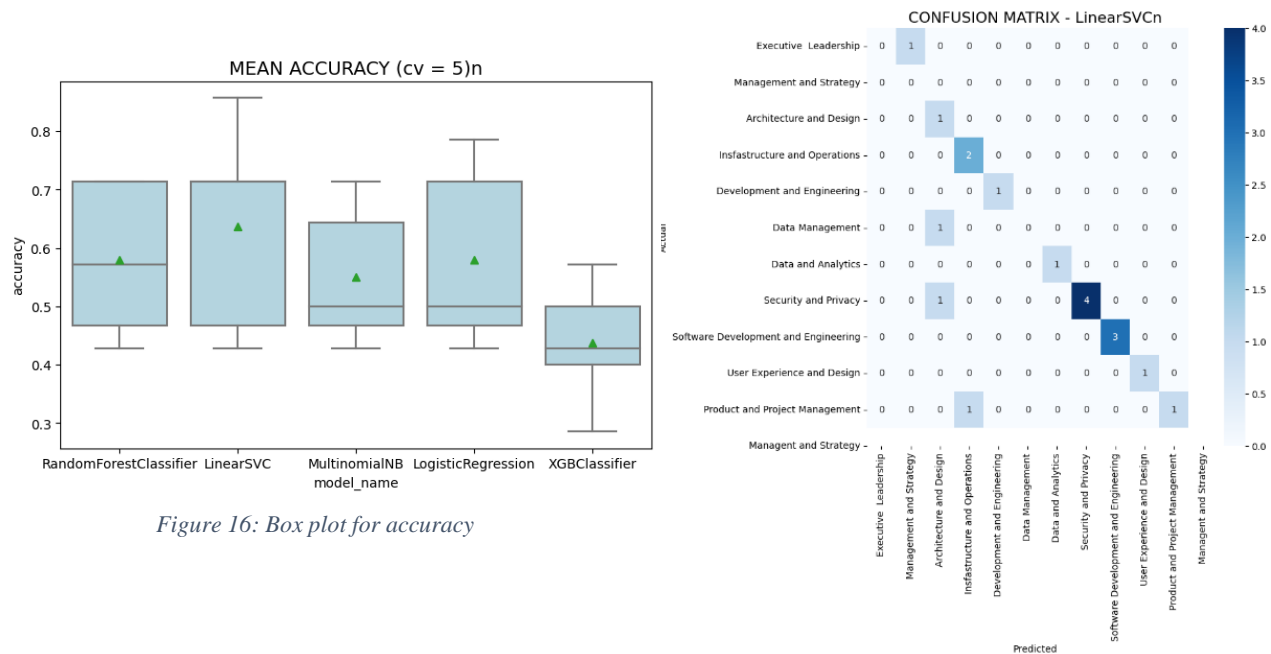


Figure 16: Box plot for accuracy

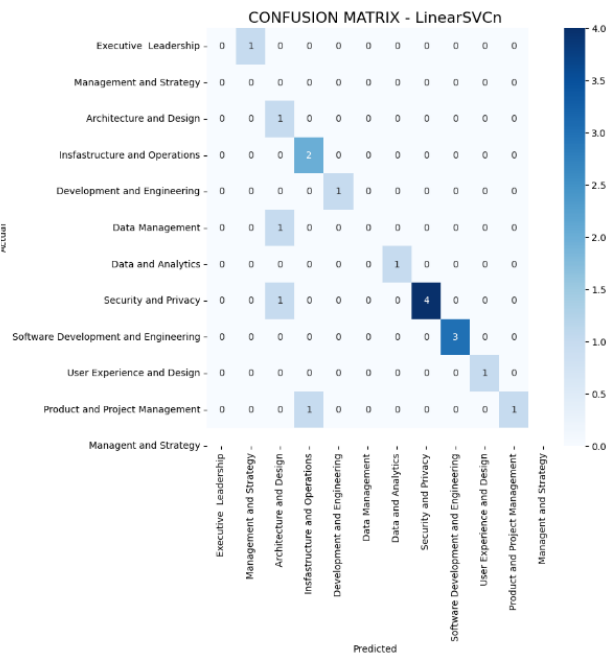


Figure 17: Heatmap for model accuracy

10.1.2 GitHub Programming Language Proficiency

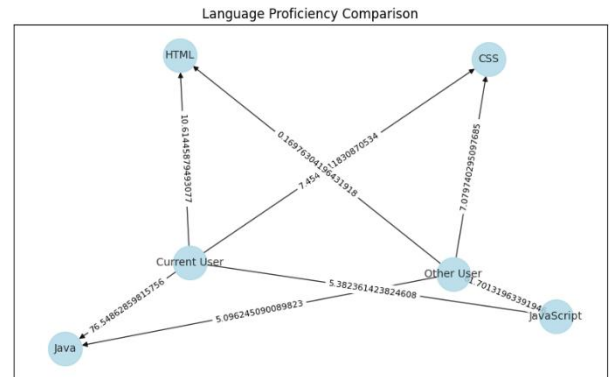
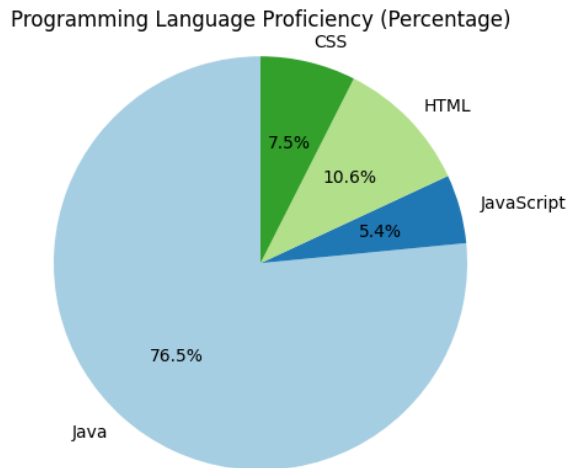


Figure 18: peer candidate comparison

Figure 19: Single candidate PLP visualization

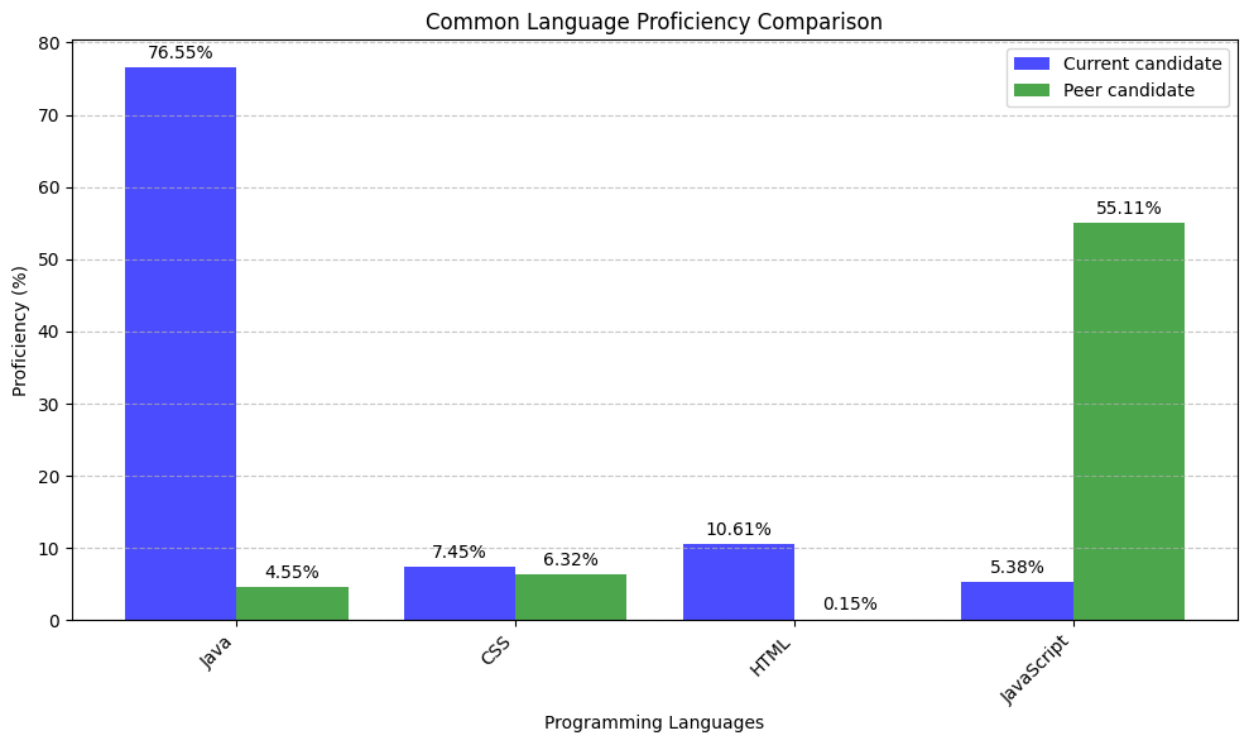


Figure 20: bar chart visualization for comparison

10.1.3 Sentiment analysis on reference checking

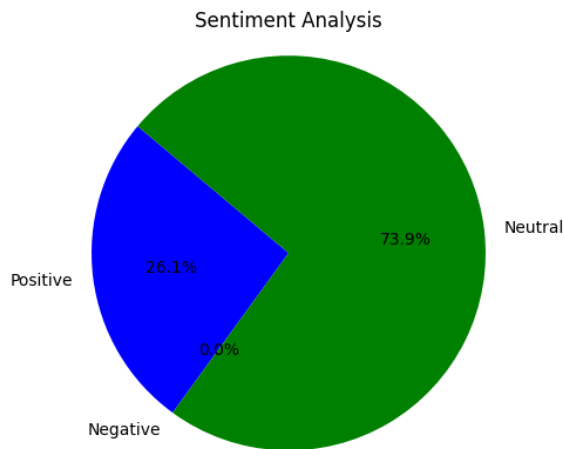


Figure 21: sentiment results for recommendation letters

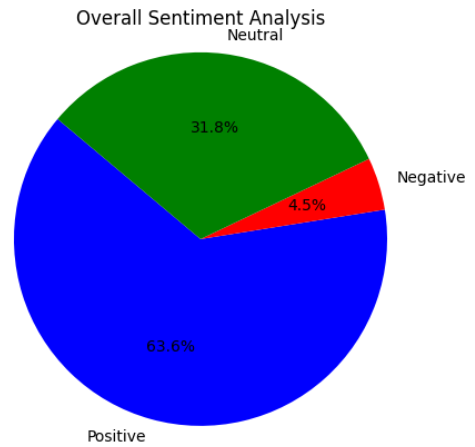


Figure 22: sentiment results for google form responses.

10.2 Research Findings

10.2.1 LinkedIn skills-based job category prediction

According to the results of accuracy testing, among all five models Linear Support Vector Classifier outperforms the best one giving a mean accuracy of 66.38%. This result was delivered after following a fivefold cross-validation and hyperparameter tuning for each model.

When it comes to LinkedIn scraping LinkedIn is generally not allowed according to LinkedIn's User Agreement and robots.txt file. LinkedIn has implemented measures to prevent scraping and unauthorized access to its platform, and it takes measures to protect the privacy and security of its users' data. However, LinkedIn allows a third-party API called ProxyCurl for limited scraping in LinkedIn.

10.2.2 GitHub Programming Language Proficiency

This analysis often involves tracking the number of repositories, commits, and contributors for each language. It allows to extract these data by using GitHub API's. it has been able to reliable calculations on the overall language proficiency of a GitHub user considering all the repositories.

10.2.3 Sentiment analysis on candidate references

Sentiment analysis on recommendation letters and reference letters gives overall highly positive feedback while sentiment on Google form responses done reference checking gives mixed sentiments regarding the candidate.

10.3 Discussion

10.3.1 LinkedIn skills-based Job Category Prediction

Initially, it was proposed to scrape and extract LinkedIn data of a particular user including comments, and recommendations given with the purpose of predicting the job category with candidate behavioral patterns on LinkedIn. Because currently most of the job recruiting is proceeding through LinkedIn Profiles. So, people tend to maintain their LinkedIn profile carefully and accurately up to their professional standards. However, LinkedIn has implemented measures to prevent scraping and unauthorized access to its platform, and they take measures to protect the privacy and security of their users' data. In LinkedIn user profile scraping, 'ProxyCurl' is a service that provides web scraping and data extraction capabilities, particularly for websites that might block or restrict access to scrapers. In the paper [9] they have scraped "about" and "heading" for personality prediction. In this research "skill" section was scraped to predict the job category.

When it comes to the dataset utilized, currently a dataset is available at Kaggle which includes all the job categories in the world and has few numbers of records for IT job categories skills extracted from LinkedIn. Thus, a dataset was created recently collecting data and information on IT job categories, job roles, and skills required. No records should be increased with a wide range to get high accuracy for better predictions.

10.3.2 GitHub Programming Language Proficiency.

When it comes to the IT industry, Programming Language Proficiency holds an important place where a candidate's technical skills are evaluated. This component provides a time-saving solution without manual interference. A single candidate can be evaluated solely for his/her overall candidate programming language proficiency. This was done assuming the candidate has maintained a reliable, not cloned account. Necessary steps were taken to verify the ownership to some extent. Commit history and number of followers were analyzed before

calculating the proficiencies. Further technical skills extracted from the GitHub will be cross-checked with CV extracted technical skills for further verification of candidate true skills evaluation. This component also allows to compare multiple candidates at once while past research [3], [9] focuses on one candidate's technical skills evaluation rather than a comparison.

10.3.3 Sentiment analysis on candidate references

Initially, it was proposed to evaluate candidate references by checking through recommendation letters and service letters.

Recommendation letters, while valuable in assessing an applicant's qualifications, may not always provide a statistically reliable measure of an applicant's overall professional standing. It is important to recognize that when an application consists solely of highly positive recommendation letters, this may not necessarily indicate unanimous acclaim from all colleagues. Rather, such positive endorsements could potentially be attributed to selecting interpersonal relationships or friendships within the professional network.

There should be a way for sought to ensure a more accurate and well-rounded evaluation of their suitability for a given role or opportunity. So, the above methodology was taken into consideration for more accurate reference checking.

10.3.4 Future Work

In future work, it is imperative to focus on refining the machine learning algorithms used for LinkedIn skills-based job category prediction, enhancing their accuracy and adaptability to evolving skillsets and job requirements. Additionally, exploring ways to further automate the reference-checking process and integrating it seamlessly into the recruitment workflow would be beneficial. Furthermore, addressing ethical concerns, such as bias mitigation in automated recruitment processes, and continuously staying updated with legal and privacy regulations will be critical as these technologies continue to advance. Additionally, conducting comprehensive user testing and feedback collection from both recruiters and candidates will be instrumental in refining and fine-tuning the entire recruitment automation system to ensure its effectiveness and user-friendliness.

11 Conclusion

In conclusion, this research project has endeavored to address the critical challenges in the contemporary job recruitment process by introducing a comprehensive and innovative framework comprising three distinct components. Firstly, the utilization of GitHub analysis has provided a valuable means of assessing candidates' overall programming language proficiency, enabling recruiters to make more informed decisions regarding technical competencies. This component not only saves valuable time but also ensures that candidates' capabilities align closely with the specific technical requirements of the job, reducing the chances of mismatches in skillsets.

Secondly, the development of a LinkedIn skills-based job category prediction model, employing machine learning techniques, has demonstrated the potential to streamline the initial screening phase and enhance the accuracy of matching candidates with job opportunities. By automatically categorizing candidates based on their skills, recruiters can quickly identify those who are most suitable for a given role, further reducing manual effort and improving the efficiency of the recruitment process.

Lastly, the integration of sentiment analysis within the reference-checking process via a pre-created Google form has offered a novel approach to assessing candidate qualities and character beyond traditional CV evaluations. This not only adds a layer of objectivity to the reference-checking process but also helps identify candidates who are not only technically proficient but also possess the soft skills and qualities required to thrive in the organization's culture.

Together, these components represent a significant step towards optimizing and automating the job recruitment process, providing recruiters with a timesaving and more comprehensive solution to evaluate candidates, ultimately contributing to more efficient and informed hiring decisions. As technology continues to evolve, this research paves the way for a future where job recruitment can be conducted with greater precision and efficiency, benefiting both employers and candidates alike. However, it is essential to remain mindful of ethical considerations and potential biases in the automated processes, ensuring fairness and transparency in the hiring process.

12 REFERENCES

GitHub Global Growth. (n.d.). Retrieved from Techmonitor:

<https://techmonitor.ai/technology/software/github-users-microsoft-thomas-dohmke>

LinkedIn Statistics and Trends. (n.d.). Retrieved from DATAREPORTAL:

<https://datareportal.com/essential-linkedin-stats>

LinkedIn Usage and Revenue Statistics. (n.d.). Retrieved from Business of Apps:

<https://www.businessofapps.com/data/linkedin-statistics/>

- [1] J. Chalidabhongse, N. Jirapokakul, and R. Chutivisarn, "Facilitating Job Recruitment Process Through Job Application Support System," in *2006 IEEE International Conference on Management of Innovation and Technology*, Singapore, China: IEEE, Jun. 2006, pp. 111–115. doi: 10.1109/ICMIT.2006.262244.
- [2] S. Gupta, B. Gupta, and S. Gupta, "A Novel Method for Technical Candidate Assessment using Github Repository Inspection Automation," in *2022 International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India: IEEE, Jan. 2022, pp. 1–5. doi: 10.1109/ICCCI54379.2022.9740986.
- [3] R. G. U. S. Gajanayake, M. H. M. Hiras, P. I. N. Gunathunga, E. G. Janith Supun, A. Karunasenna, and P. Bandara, "Candidate Selection for the Interview using GitHub Profile and User Analysis for the Position of Software Engineer," in *2020 2nd International Conference on Advancements in Computing (ICAC)*, Malabe, Sri Lanka: IEEE, Dec. 2020, pp. 168–173. doi: 10.1109/ICAC51239.2020.9357279.
- [4] F. Chatziasimidis and I. Stamelos, "Data collection and analysis of GitHub repositories and users," in *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*, Corfu, Greece: IEEE, Jul. 2015, pp. 1–6. doi: 10.1109/IISA.2015.7388026.
- [5] I. Rehman, D. Wang, R. G. Kula, T. Ishio, and K. Matsumoto, "Newcomer Candidate: Characterizing Contributions of a Novice Developer to GitHub," in *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, Adelaide, Australia: IEEE, Sep. 2020, pp. 855–855. doi: 10.1109/ICSME46990.2020.00110.
- [6] S. M. Patil, R. Singh, P. Patil, and N. Pathare, "Personality prediction using Digital footprints," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India: IEEE, May 2021, pp. 1736–1742. doi: 10.1109/ICICCS51141.2021.9432380.
- [7] B. Gibson, S. Townes, D. Lewis, and S. Bhunia, "Vulnerability in Massive API Scraping: 2021 LinkedIn Data Breach," in *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA: IEEE, Dec. 2021, pp. 777–782. doi: 10.1109/CSCI54926.2021.00191.
- [8] H. K. S M, S. Hegde, S. G, S. M, S. R, and S. L. N, "User Interest Prediction based on Social Network Profile with Machine Learning," in *2021 6th International Conference for Convergence in Technology (I2CT)*, Maharashtra, India: IEEE, Apr. 2021, pp. 1–6. doi: 10.1109/I2CT51068.2021.9418126.

- [9] R. T. R. Jayasekara, K. A. N. D. Kudarachchi, K. G. S. S. K. Kariyawasam, D. Rajapaksha, S. L. Jayasinghe, and S. Thelijjagoda, “DevFlair: A Framework to Automate the Pre-screening Process of Software Engineering Job Candidates,” in *2022 4th International Conference on Advancements in Computing (ICAC)*, Colombo, Sri Lanka: IEEE, Dec. 2022, pp. 288–293. doi: 10.1109/ICAC57685.2022.10025337.

13 APPENDICES

13.1 Survey Questionnaire

QuestionsResponses111Settings

7. What is/are the professional platforms you currently using to showcase you professional work? *

☐ GitHub

☐ LinkedIn

☐ Both

☐ Other...

+

↔

Tt

AA

▶

≡

8. How frequently do you update your GitHub/LinkedIn profiles with new projects or code samples? *

☐ Daily

☐ Weekly

☐ Monthly

☐ Rarely

☐ Never

9. What is your main reason for using GitHub/LinkedIn to showcase your professional skills? *

☐ To demonstrate my coding abilities to potential employers

☐ To collaborate with other developers on projects

☐ To build my personal brand as a developer

☐ Other

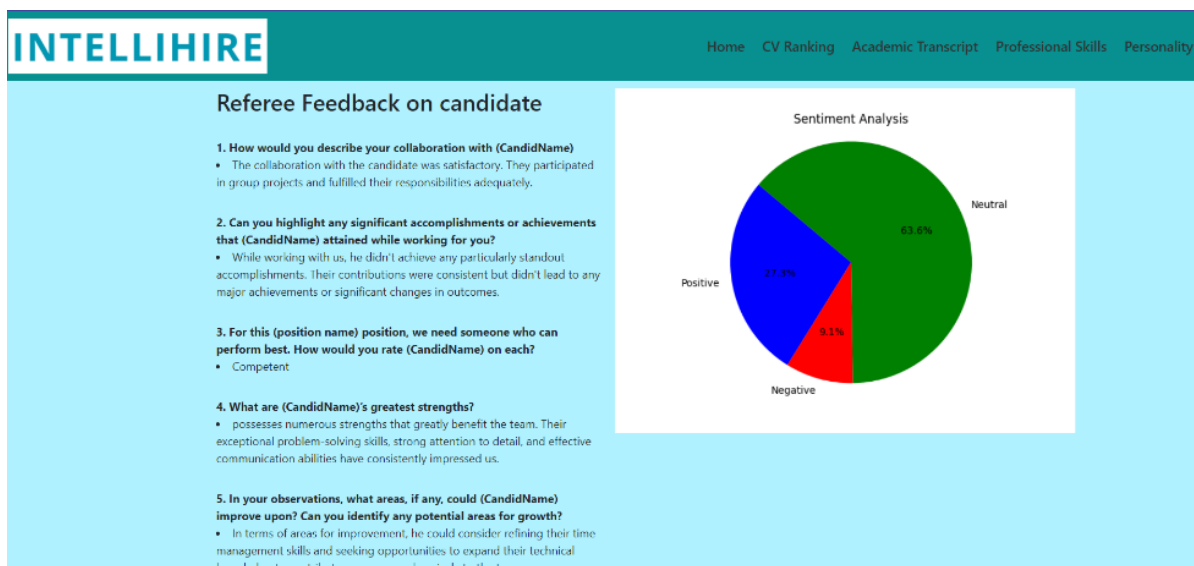
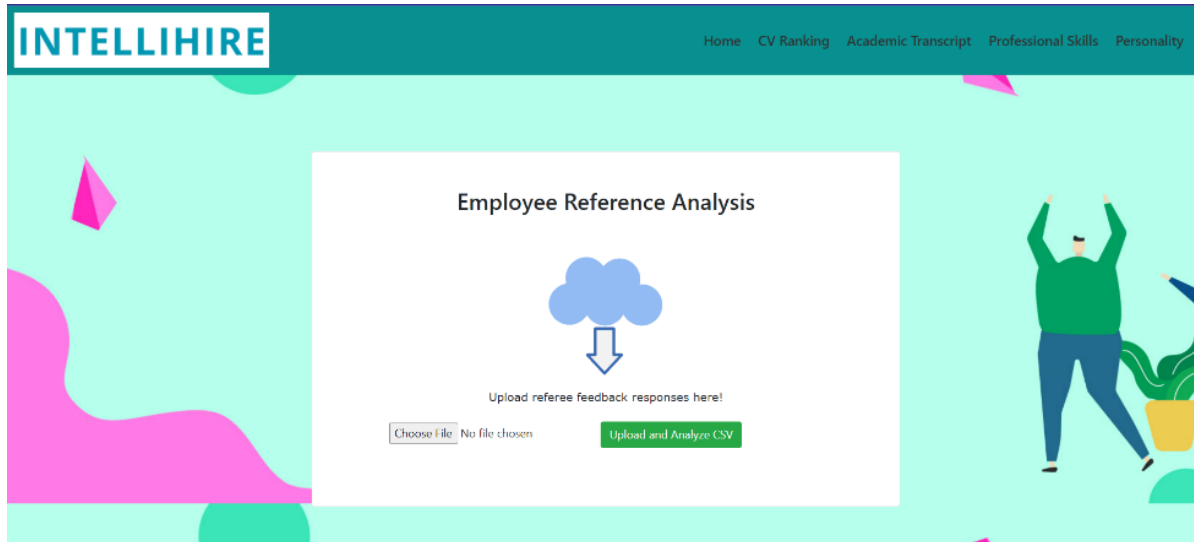
AA

▶

≡

13.2 Frontend Development

- PastPort – Employee Reference Evaluator



- GitHub Polygraph – PLP (Programming Language Proficiency)
 - PLP

INTELLIHIRE Home CV Ranking Academic Transcript Professional Skills Personality

Github Language Proficiency Comparison

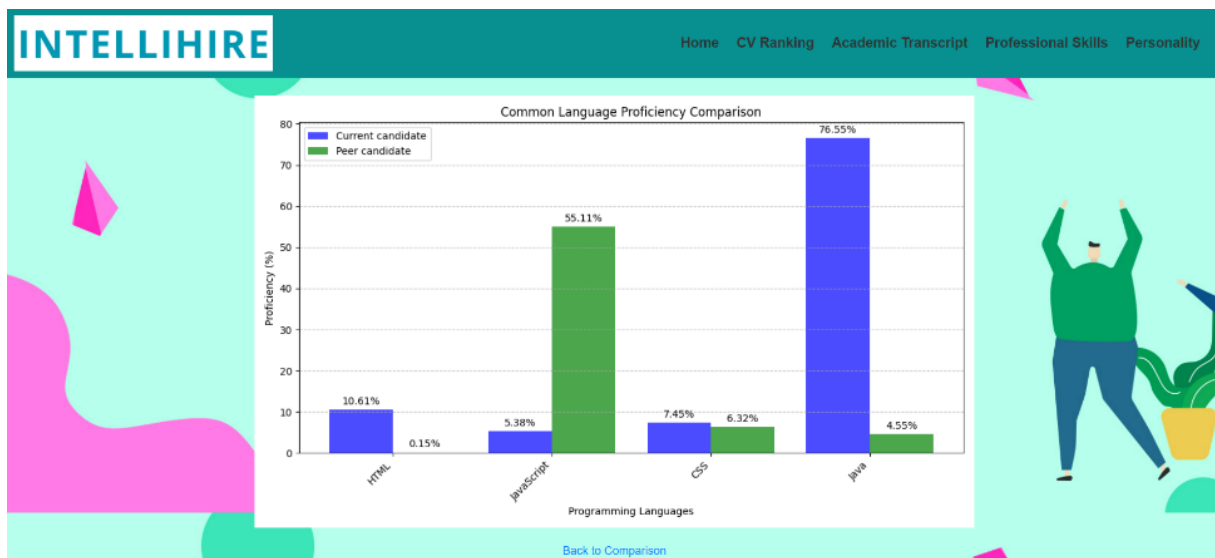
Enter the candidates' GitHub usernames here!

Current User:
Maldeniya99

Other User:
IT20207854

Compare

ABOUT **QUICK LINKS**




- PLP Comparator

INTELLIHIRE


[Home](#) [CV Ranking](#) [Academic Transcript](#) [Professional Skills](#) [Personality](#)

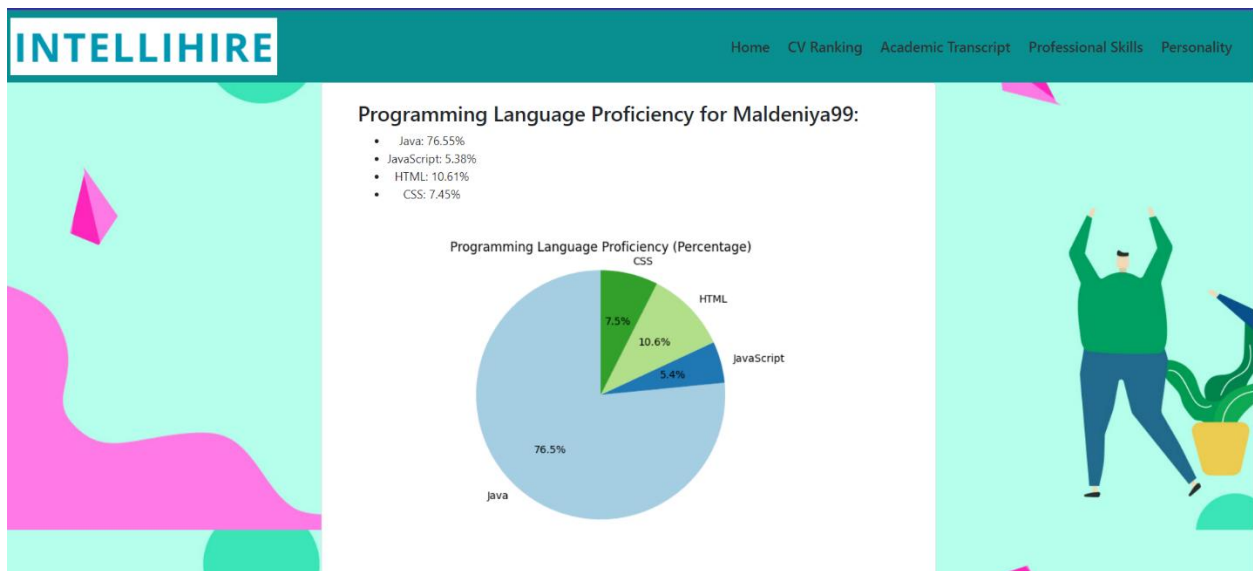
Github Language Proficiency



Enter the candidate GitHub username here!

Calculate





13.3 Turnitin Report

Turnitin Originality Report

Document Viewer

Processed on: 11-Sep-2023 16:53 +0530
ID: 2084937706
Word Count: 8029
Submitted: 6

final report By Sandani Zoysa

Similarity Index	Similarity by Source
11%	Internet Sources: 7% Publications: 5% Student Papers: 5%

include quotedinclude bibliographyexclude small matchesmode: quickview (classic) reportprintdownload

1% match (Internet from 18-Jul-2023)
<https://www.coursehero.com/file/202995964/2020-138-GroupDocumentdocx/>

1% match (Internet from 30-Aug-2022)
<https://techmonitor.ai/technology/software/github-users-microsoft-thomas-dohmke>

1% match (Sambhav Gupta, Bhoomi Gupta, Sachin Gupta. "A Novel Method for Technical Candidate Assessment using Github Repository Inspection Automation", 2022 International Conference on Computer Communication and Informatics (ICCCI), 2022)
[Sambhav Gupta, Bhoomi Gupta, Sachin Gupta. "A Novel Method for Technical Candidate Assessment using Github Repository Inspection Automation", 2022 International Conference on Computer Communication and Informatics \(ICCCI\), 2022](#)

1% match (R.G.U.S. Gajanayake, M.H.M. Hiras, P.I.N. Gunathunga, E.G. Janith Supun, Anuradha Karunasenna, Pradeepa Bandara. "Candidate Selection for the Interview using GitHub Profile and User Analysis for the Position of Software Engineer", 2020 2nd International Conference on Advancements in Computing (ICAC), 2020)
[R.G.U.S. Gajanayake, M.H.M. Hiras, P.I.N. Gunathunga, E.G. Janith Supun, Anuradha Karunasenna, Pradeepa Bandara. "Candidate Selection for the Interview using GitHub Profile and User Analysis for the Position of Software Engineer", 2020 2nd International Conference on Advancements in Computing \(ICAC\), 2020](#)

1% match (Shridhar Hegde, Santosh G. Shivakumar M, Srihari R, Shree Lakshmi N. "User Interest Prediction based on Social Network Profile with Machine Learning", 2021 6th International Conference for Convergence in Technology (I2CT), 2021)
[Shridhar Hegde, Santosh G. Shivakumar M, Srihari R, Shree Lakshmi N. "User Interest Prediction based on Social Network Profile with Machine Learning", 2021 6th International Conference for Convergence in Technology \(I2CT\), 2021](#)

1% match (Internet from 05-Jan-2022)
https://www.3ayachting.com/price-of-usage_3713/

<1% match (Internet from 15-Jul-2023)
<https://www.coursehero.com/file/202996093/2020-138pdf/>