

Ensemble Method for Student's Skills Evaluation System

Abstract— There is a critical need to redefine the evaluation process for IT students, transcending conventional academic metrics and embracing a holistic assessment that incorporates various parameters. The current evaluation criteria, particularly the stringent 60% academic benchmark, impose limitations by restricting opportunities for students and neglecting the recognition of practical skills. Traditional evaluation systems heavily rely on academic metrics, leading to the exclusion of deserving candidates due to the absence of a comprehensive assessment considering diverse parameters. This research proposes an Ensemble Method algorithm as a solution, introducing a holistic evaluation approach. The system provides a user-friendly interface for students and a secure dashboard for educators. The research contributes to a more inclusive evaluation process by considering certifications, programming skills, projects, and extracurricular activities such as sports, empowering students to showcase their complete skill set. The application leverages the Repoquestor API, OCR Technology, and Codility API in conjunction with the proposed algorithm, ensuring a comprehensive and well-rounded assessment of students' capabilities. The research demonstrates approximately 87.93% particularly beneficial for IT students seeking a more inclusive evaluation, educators and evaluators desiring a detailed analysis of student progress, and organizations aiming to recognize a diverse range of talents and skills in potential candidates.

Keywords—Optimal character recognition, API integration, Repoquestor API, Codility API, Web scraping.

I. INTRODUCTION

The assessment of students in the realm of Information Technology (IT) has historically leaned heavily on conventional academic metrics, providing a somewhat narrow perspective of their capabilities. In the swiftly evolving technological landscape of today, IT professionals are in demand to possess a versatile skill set that extends beyond textbook knowledge. This research paper introduces a groundbreaking web application designed to redefine the evaluation of IT students. It transcends traditional grading systems, offering a thorough and holistic assessment that takes into account various dimensions of a student's development. In the contemporary job market, academic achievements have held significant weight in evaluating job candidates. However, the common practice of establishing a minimum academic benchmark, often set at 60% throughout academic records, has created barriers for many highly talented individuals. This stringent benchmark, initially intended to gauge academic skills, frequently falls short in capturing the complete range of an individual's abilities. This research proposes an innovative web application that aims to revolutionize the assessment of IT students, providing a comprehensive and holistic evaluation that considers various aspects of a student's development. The application is designed to empower educators and evaluators with a deeper understanding of a student's potential, acknowledging that excellence surpasses mere grades. Our pioneering web application seeks to redefine the evaluation of IT students beyond traditional academic metrics.

Instead of adhering to the 60% rule, we propose a new system that considers multiple factors, including grades, certifications, projects, programming skills, and extracurricular activities like sports. This approach strives to offer a more complete and accurate depiction of each student's abilities, talents, and achievements in the professional world.

The primary objective of our research is to create a user-friendly web application that transforms the evaluation process. Going beyond traditional academic assessments, the application takes into account a diverse range of factors. It not only redefines how we measure the capabilities of IT students but also acts as a catalyst for their overall development. The interface enables students to input relevant data, such as certifications earned, programming skills acquired, projects completed, and participation in extracurricular activities like sports. Educators and evaluators can access this data through a secure dashboard, providing them with a powerful tool to visualize and analyze each student's progress. Utilizing the Repoquestor API for project assessment, OCR technology for certificate evaluation, and the Codility API for programming tests to assess students' performance and skills, our web application simplifies the identification and evaluation of students based on multiple factors, ensuring a fair and detailed assessment of their potential. Additionally, providing individual progress reports empowers students by giving them insights into their development, fostering a sense of ownership and motivation for continuous improvement. This approach not only enhances the overall evaluation process but also prepares students for the dynamic IT industry by exposing them to the data-driven decision-making prevalent in the field. In summary, our research endeavors to overcome the limitations of traditional academic evaluations by proposing a comprehensive system that values the diverse talents and skills of IT students. Through this innovative approach, we aim to contribute to a job market that recognizes the varied nature of individual potential, making it fairer and more dynamic.

II. LITERATURE REVIEW

The research centers around the development of a web system tailored for individuals with academic scores below 60% above this threshold. The web system evaluates students comprehensively, taking into account certifications, programming skills, projects, and sports achievements. By leveraging these parameters, the application offers a holistic understanding of a student's capabilities, identifying strengths and areas for improvement. This novel web application revolutionizes the evaluation of IT students, moving beyond traditional grading systems. Students input their skills through a user-friendly interface and undergo assessments. Teachers monitor progress through a secure dashboard. Using APIs and algorithms, the web system provides a thorough evaluation, generating progress reports for students. The goal is to empower students, offering a complete picture of their skills and fostering excellence in the real world, redefining student evaluation for visualizing and analyzing progress. The application employs APIs and various algorithms to identify

and assess students based on different parameters, and students can access their own progress reports.

In related studies, various innovative approaches have been proposed to enhance IT education and evaluation. "Repo Quester" assesses the quality metrics of software projects on GitHub, employing nonverbal cues [1]. Another study focuses on extracting short text from chart pictures and papers, outperforming previous methods [2]. An automated grading system evaluates programming assignments based on source code analysis [3]. The increasing demand for IT expertise globally is emphasized, with a focus on skills like word processing and spreadsheet usage [4]. Student projects, crucial for promoting critical thinking, have been a significant focus in educational institutions [5]. A proposed code-quality awareness system utilizes continuous integration and Git-driven technologies for team project management in programming [6]. A study investigates differences in test item creation between highly skilled and lowly skilled engineers, suggesting a novel approach to test item development [7]. An assignment on web-based programming for autonomous scoring systems, utilizing the JUnit test Reflect API, has demonstrated success in university programming courses [8]. The Monash SE courses emphasize the idea that software systems are composed of systems of signals with meanings and applications in particular professional settings [9]. In summary, these studies collectively contribute to reshaping the evaluation and education landscape in the field of Information Technology, emphasizing practical skills, project-based learning, and innovative assessment methods.

III. ARCHITECTURE

The comprehensive evaluation system for students across four key parameters: certification, programming skills, projects, and sports achievements. The system's architecture is depicted in three figures, corresponding to each evaluation parameter.

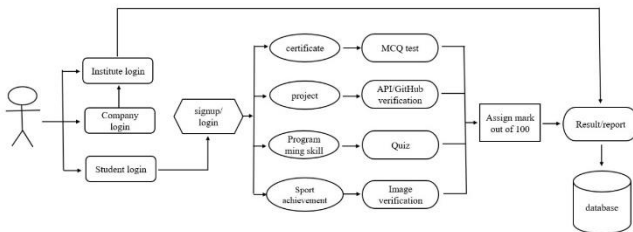


Fig. 1. System Architecture

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

A. Certificate evaluation :

1) upload and validation of Certification: Students directly upload certification documents to the web application, seamlessly incorporating certification records into the evaluation process. After uploading a certification, students take a test assessing their understanding and proficiency in the associated skills. Marks are assigned based on their performance in this evaluation.

2) Test Integration : Integrating a test module within the web application that aligns with the skills certified in the uploaded document. Implement an automated evaluation system that assesses the test results objectively.

B. Project evaluation :

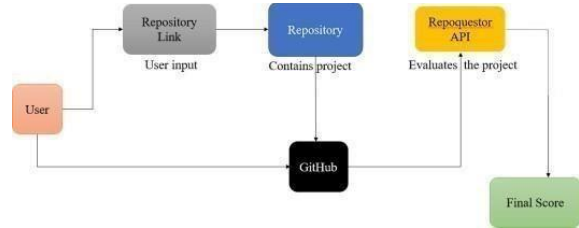


Fig. 2. Repoquester architecture

1) Automated Evaluation with RepoQuester: Students are required to upload their project on GitHub for evaluation. This platform allows for a centralized and accessible repository of all projects. A tool designed for assessing project quality and complexity, involves predefined criteria. RepoQuester objectively evaluates projects, considering factors like code structure, documentation, functionality, and overall complexity, ensuring a comprehensive assessment beyond surface-level considerations.

2) Marks Assignment Based on Evaluation: Marks are assigned to each project based on the evaluation conducted by RepoQuester. The automated nature of this process ensures consistency and objectivity in the assessment.

C. Programming skills evaluation :

1) Programming Challenges Test: Students engage in a targeted test aligned with their chosen programming skills, solving challenges to showcase practical proficiency. Marks are assigned based on individual performance.

2) Test Integration : Integrating Codility API into the web application, allowing the platform to fetch and evaluate coding challenges dynamically.

D. Sports evaluation:

1) OCR Technology Verification: The sports evaluation process by uploading certificates detailing their sports achievements. The level of sports certification is verified through OCR (Optical Character Recognition) technology, ensuring accuracy and authenticity. This step enhances the credibility of the sports parameter.

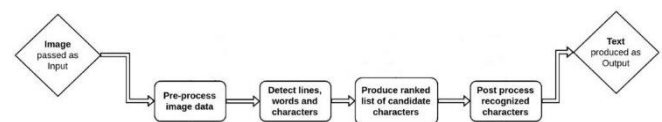


Fig. 2. Working of OCR

2) Working of OCR (Optimal character recognition) Technology:

1. Image Acquisition: This is where the physical document, be it a scanned page, a photo of a sign, or even a screenshot, enters the picture. Scanners or digital cameras capture the image, converting it into a digital format.

2. Preprocessing: Raw scanned images aren't perfect. Pre-processing steps like de-skewing (tilting correction), binarization (black and white conversion), and noise reduction are done to create a cleaner image for better character recognition.
3. Text Recognition: This is where the real magic happens! The OCR engine analyzes the preprocessed image pixel by pixel, identifying patterns and shapes. AI algorithms, particularly machine learning, play a crucial role here. They compare the image features to their database of known characters, fonts, and even language patterns.

E. Database Creation and Management :

- 1) SQLite Setup: Initialize an SQLite database to store student data, including academic records, certifications, project details, programming skills, extracurricular activities, and sports achievements.
- 2) Data Insertion: Developing a mechanisms by incorporating SQL INSERT statements for accurate record storage to insert student data into respective tables.

IV. METHODOLOGY AND WORKFLOW

- 1) In the Projects we are constructing a scoring system that evaluates the quality, complexity, and impact of the student's projects.
- 2) In the Certifications points will be assigned to different certificates according to their relevance to the student's field of study or future career and according to their marks that achieve in test.
- 3) In the Programming Skills Design coding challenges or test that assess the student's programming proficiency.
- 4) In Sports Certifications additional marks will be assign according to certificate levels using OCR also known as (optimal character recognition).
- 5) Then we calculate the final score based on the above mentioned four parameters and assign weightage according to their importance.

In our proposed method we use an Ensemble method for the evaluation of students based on their curricular, co-curricular, and extracurricular activities, our proposed method we use a combination of diverse models to make predictions. Here's an Algorithm for ensemble method for Student Evaluation:

Algorithm: Ensemble Student Evaluation
Input:
- StudentData: Data containing information about the student's curricular, co-curricular, and extracurricular activities.
Output:
- OverallEvaluation: A numerical score representing the overall evaluation of the student.
Procedure:
1. Initialize weights for different activities: - curricularWeight = 0.6 - cocurricularWeight = 0.3 - extracurricularWeight = 0.1 2. Train base models for different aspects: curricularModel = TrainCurricularModel(trainingDataCurricular) cocurricularModel = TrainCocurricularModel(trainingDataCocurricular) extracurricularModel=TrainExtracurricularModel(trainingDataExtracurricular) 3. Make predictions using individual models: curricularPrediction = curricularModel.Predict(StudentData) cocurricularPrediction = cocurricularModel.Predict(StudentData) extracurricularPrediction = extracurricularModel.Predict(StudentData) 4. Combine predictions with weights to get the overall evaluation: OverallEvaluation = (curricularPrediction * curricularWeight) + (cocurricularPrediction * cocurricularWeight) + (extracurricularPrediction * extracurricularWeight) 5. Return OverallEvaluation as the final result.
End Algorithm

The **TrainCurricularModel**, **TrainCocurricularModel**, and **TrainExtracurricularModel** functions are responsible for training machine learning models specific to each aspect of evaluation. The Predict function is used to obtain predictions from the trained models based on the input student data. The proposed algorithm is designed to combine these predictions with percentage weights to generate an overall evaluation scorefor the student.

V. RESULT

About Repo-Quester RepoQuester is a tool developed to assess the quality metrics of software projects on GitHub using Ensemble Student Evaluation. The tool introduces two new metrics, Releases and Pull Requests Ration, to evaluate software projects.

Final score will be generated out of 100% including following parameters

1. Certification Evaluation: Fig. 4 shows the Certification pageas a tangible proof of an individual's knowledge and expertise in a specific domain. Certifications often involve rigorous testing and assessment, ensuring a certain level of competency. Including certification evaluation in the overall score recognizes the importance of formal education and standardizedassessments.

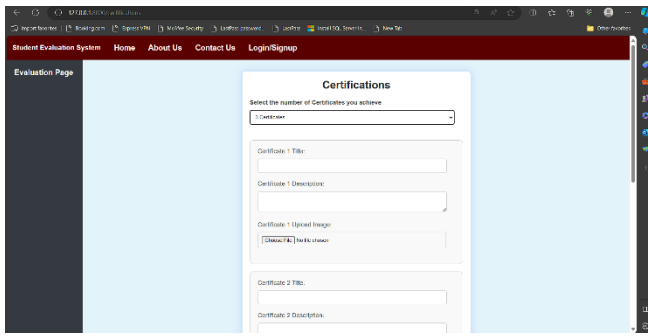


Fig. 2. Certificate page

2. Project Evaluation: Projects reflect practical application of theoretical knowledge, providing insights into an individual's problem-solving and implementation skills. Project evaluation assesses the ability to translate theoretical concepts into real-world solutions. Emphasizing project evaluation in the final score acknowledges the significance of hands-on experience.

3. Programming Skill Evaluation: In today's technology-driven world, programming skills are fundamental in various fields. Programming skill evaluation measures the ability to write efficient, error-free code and solve complex problems algorithmically. Including programming skill evaluation recognizes the demand for technical proficiency and problem-solving capabilities in the modern workforce.

4. Sport Evaluation: Incorporating sport evaluation recognizes the holistic development of individuals, including physical fitness, teamwork, and discipline. Participation in sports fosters qualities such as leadership, resilience, and time management.

In fig.4 and fig. 5 we used the OCR technology in the sportscertification.

```
from PIL import Image
import pytesseract

pytesseract.pytesseract_tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract.exe'

def ocr(image_path):
    # Open the image file
    img = Image.open(image_path)

    # Use pytesseract to do OCR on the image
    text = pytesseract.image_to_string(img)

    return text

# Example usage
image_path = 'path/to/your/image.png'
result = ocr(image_path)
print("OCR Result:")
print(result)
```

Fig. 5. OCR Image scanning

Image Input: OCR takes an image containing text as input. Preprocessing: The image may be preprocessed to enhance clarity (resize, grayscale, etc.). Text Extraction: OCR algorithms analyze the image to extract text characters. Character Recognition: Each character is recognized using pattern matching.

Text Output: The recognized text is outputted, allowing for further processing or display.

The certification analysis done using various score like reading, writing, technical, quantity and aptitude score basis. Following are the correlation found using Ensemble Student Evaluation in terms of 1.

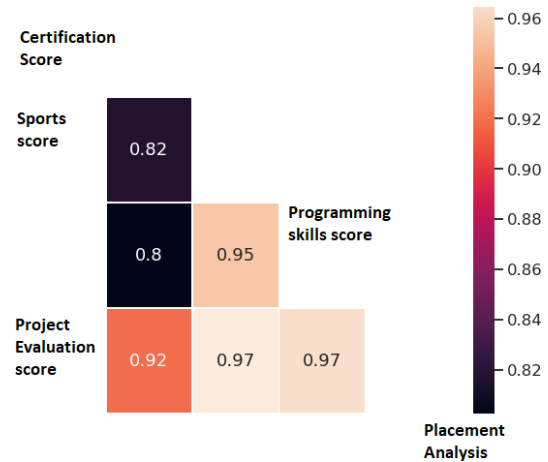


Fig. 6. Correlation Analysis for Student Evaluation

Following fig. 6 and Fig. 7 is showing the relationship of student attempted for certification exam and cleared certification exam for placement

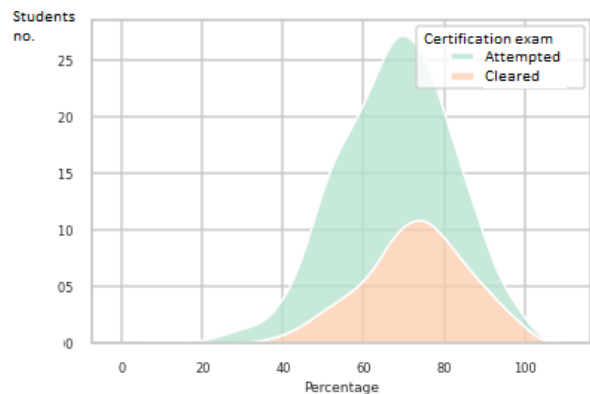


Fig. 7. Correlation Analysis for Student Evaluation

Fig. 7 gives the evaluation of success ratio of certification and placed student. It gives more than 87% Skilled candidate to any IT Company.

VI. CONCLUSION

In conclusion, our research aims to overcome the limitations inherent in the traditional academic evaluation system for IT students. The current practice of enforcing a rigid 60% academic benchmark creates obstacles for highly talented individuals, impeding their access to job opportunities and neglecting the recognition of their practical skills. Our proposed Ensemble Student Evaluation algorithm seeks to revolutionize the evaluation process by incorporating a diverse set of parameters, including certifications, projects, programming skills, and sports activities. This innovative approach strives to offer a more holistic and equitable assessment, enabling a nuanced

understanding of students' capabilities beyond academic scores.

By harnessing APIs and the Ensemble Student Evaluation algorithms, our system ensures a fair and detailed evaluation, better preparing students for the dynamic IT industry. The inclusion of individual progress reports empowers students to actively participate in their development, fostering a sense of ownership and motivation for continuous improvement. In essence, our research aims to contribute to a more inclusive and dynamic job market that recognizes and values the multifaceted nature of individual potential in the field of Information Technology.

The proposed method for evaluating success ratio in certifications and placing students has achieved an accuracy rate of more than 87% demonstrating its effectiveness in identifying skilled candidates for IT companies.

VII. REFERENCES

- [1] Data collection and analysis of github repositories and user. Fraggiskos Chatziasimidis Gnomon Informatics SA 21 Antonis Tritsis Str. 57001 Thessaloniki, Greece +30-2310804150 f.chatziasimidis@gmail.com
- [2] XML-based Fully Automatic Assessment System for IT Skills Aicha Chorana, Abdallah Lakhdari, Hadda Cherroun, and Slimane Oulad Naoui Computer science and mathematics department, Faculty of Science
- [3] Identifying Student Project Impact Using the Buxton and Hanney Payback Framework. Marion Slack, PhD, Terri L. Warholak, PhD a University of Arizona College of Pharmacy, Tucson, Arizona.
- [4] Code-quality evaluation scheme for assessment of student contributions to programming projects. 'His-Min Chen & Bao-An Nguyen' Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan.
- [5] Design and Evaluation of Automated Scoring Programming Assignments Yuki Akahane, Hiroki Kitaya, and Ushio Inoue Graduate School of Engineering Tokyo Denki University Tokyo, Japan.
- [6] Aicha Chorana, Abdallah Lakhdari, Hadda Cherroun, and Slimane Oulad Naoui. Xml-based fully automatic assessment system for it skills. In 2014 International Conference on Web and Open Access to Learning (ICWOAL), pages 1–6. IEEE, 2014.
- [7] RepoQuester: A Tool Towards Evaluating GitHub Projects. Kowndinya Boyalakuntla¹, Meiyappan Nagappan², Sridhar Chimalakonda¹, Nuthan Munaiah³ Indian Institute of Technology Tirupati, India.
- [8] Small Text Extraction from Documents and Chart Images Rominkumar Busa, Shahira K C, Graduate student member, IEEE, and Lijiya A, Member, IEEE.
- [9] Automated Assessment Tools for grading of programming Assignments: A review. Sidhidatri Nayak Reshu Agarwal Amity Institute of Information Technology Amity University, Indiasidhidatri.nayak@gmail.com.
- [10] Software Certification for Industry – verification and validation issues in expert Systems. Anca I. Vermesan Det Norske Veritas AS Veritasveien 1, 1322 Hovik, Norway Email: Anca.Vermesan@dnv.com.