Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 8s (2025)

Prediction and Assessment of Software Engineering Skill Set among Computer Science Students using Random Forest Algorithm

Jasmin Nizar [1*], R Sharmila[2] and Jaseena K U[3]

[1*]Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India[1]

[2]Professor, Department of Computer Applications, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India[2]

[3]Assistant Professor, Department of Computer Applications, MES College Marampally, Aluva, Kochi, Kerala, India[3]

Article History:

Received: *10-11-2024*

Revised:*24-12-2024*

Accepted:*09-01-2025*

Abstract:

Introduction: Skill assessment of students is an evaluation process that captures the ability of each student, whether they are new to a skill or have mastered it. People working on software development projects need to possess three skills: - Soft skills, Life skills, and Technical skills. It is globally accepted in the IT industry that for computer science students to succeed in their careers to the fullest extent, project management training is required. The skill-based education in software engineering helps to produce go-to-market talent and, in turn, achieve job satisfaction and potential career growth. In the current educational system, it is essential to predict students' skill sets, including Soft, Life, and Technical skills. The skill set of software engineering among computer science students can be accurately predicted using machine learning approaches.

Objectives: To identify skill gaps among students, promote skill-based education within software engineering programs, and enhance workforce readiness by bridging the knowledge gap between academia and the software industry.

Methods: The students' Soft, Life, and Technical skill levels are predicted using the Random Forest (RF) method. To enhance the RF model's effectiveness, this research incorporates Principal Component Analysis (PCA) as an additional step. PCA dissects the input features of the highest quality into their constituent elements. Accumulated Local Effects (ALE), an Explainable AI approach, is used to determine which factors influence skill set predictions. The suggested classification model is assessed using metrics such as accuracy, precision, recall, F1 score, and the AUC value.

Results: The modelling outcomes show that the suggested PCA-based Random Forest model has a higher prediction accuracy of 86.5% for Soft skills, 84.6% for Life skills, and 87% for Technical skills than the standard machine learning models used for comparisons.

Conclusions: Experimental results indicate that the Random Forest model with PCA demonstrates significant generalization capabilities and outperforms baseline approaches in accurately forecasting computer science students' software engineering talents.

Keywords: Skill Assessment, Software Engineering, Principal Component Analysis, Random Forest, Machine Learning, Accumulated Local Effects.

1. Introduction

In the current era, software permeates nearly every aspect of our lives. It is one of the most notable, inspiring, and challenging innovations of modern technology [1]. However, the software industry

continues to experience high rates of project failure. One primary reason is the disconnect between academic training and industry requirements. The business sector frequently asserts that software engineering graduates are ill-prepared to meet the demands of the industry. This is often due to inadequate skills and knowledge, as well as educational methodologies that fail to emphasize experiential learning. Consequently, bridging this gap between academia and the IT industry has become a critical focus.

Skills development involves identifying skill gaps in students and providing training to bridge those gaps, ultimately enabling employment. It plays a pivotal role in career success. While experts emphasize the need to develop students' skills from the beginning of their college journey, they often assume students will acquire these skills on their own. In reality, achieving competency in specific skills, whether technical, soft, or life skills, requires deliberate effort and structured training. Individuals engaged in software development projects need three primary skill sets: soft skills, life skills, and technical skills. While technical and soft skills are distinct, soft skills have been shown to be critical for project success [2]. These include teamwork, leadership, organizational effectiveness, problem-solving, and decision-making [3]. Life skills, such as the ability to collaborate effectively within a team, further support these efforts [4]. Meanwhile, technical skills, rooted in specific domains, involve the application of tools, techniques, and methods to accomplish tasks [2].

A skill-based education system ensures that students acquire not only theoretical knowledge but also essential soft and life skills. For individuals entering the IT industry, analytical thinking, problem-solving abilities, programming proficiency, subject knowledge, interpersonal and cognitive skills, and effective communication are indispensable. Teaching software engineering presents challenges, particularly when students fail to see the practical relevance of their studies. Therefore, fostering technical and transversal abilities within the computer science curriculum is essential to motivate and prepare students. This research focuses on identifying students' skill sets, which contributes to their motivation and career readiness. It evaluates their understanding of project management techniques and their applications. Additionally, it aims to train students to enhance their skills for successful careers.

Machine learning methods offer promising solutions for accurately predicting students' skill sets. These methods deploy impactful algorithms to predict skills, ensuring higher-quality training in software development courses and preparing students for successful careers. By addressing the gap between the IT sector and academia, such approaches can improve software project success rates. Feature extraction plays a crucial role in improving the predictive accuracy of machine learning models. This study employs a PCA-based RF algorithm to forecast the skill sets of software engineering students. Principal Component Analysis (PCA) extracts high-quality features, while the Random Forest (RF) algorithm predicts potential skill sets. The highlights of this paper are as follows:

- PCA-based RF algorithm to predict various skill sets of software engineering students, namely, soft skills, life skills, and technical skills.
- PCA is utilized to extract high-quality and relevant features.
- Random Forest algorithm is employed for skill set prediction.
- A customized dataset is used.

- The proposed PCA-based RF model is evaluated using various metrics.
- The Explainable AI technique, Accumulated Local Effects (ALE), is employed to identify the features contributing to skill set predictions.

The structure of the paper is organized as follows: Section 1 introduces the methods and strategies for predicting students' skill sets. Section 2 discusses the study's motivation. Section 3 reviews related work. Section 4 describes the datasets, while Section 5 explains the proposed methodology. Section 6 presents the findings. Finally, Section 7 concludes the study with key findings and recommendations for future research.

2. Motivation

According to industry perspectives, graduates with software engineering degrees often fall short of industry expectations. This discrepancy surprises academic institutions that emphasize software engineering disciplines. The persistent gap between the software industry and academia necessitates efforts to bridge these differences. Identifying students' weaknesses and predicting their skill sets while they pursue software engineering is crucial. Numerous studies highlight that student graduating from institutions with insufficient emphasis on soft, life, and technical skills—such as communication, teamwork, creativity, and leadership—are at a disadvantage [5, 6, 7, 8]. Companies often invest significant time and money in induction programs to train fresh graduates in these essential skills. This underscores the importance of incorporating skill development into academic curricula. Several researchers have called attention to the relevance of software engineering education to industry requirements [9, 10]. Establishing a strong connection between academic training and industry needs benefits both students and employers [11, 12, 13, 14].

Some studies have explored the skill sets required for success in the IT industry. Pınar Cihan [3] emphasizes addressing the limitations of project-based learning approaches to deliver high-quality instruction in software development courses. Riza Yosia Sunindijo [4] investigates 16 skill elements and their influence on project efficiency in the software sector, focusing on time, cost, and quality. E. Sventekova et al. [5] stress the importance of developing soft skills during the educational journey to foster students' interest in scientific inquiry. Gnatz et al. [9] provide insights into designing courses that effectively impart practical knowledge, based on detailed observations. Carmen Iriarte [15] highlights the critical role of soft skills in determining IT project success. Alan R. Peslak [16] describes strategies for fostering effective group collaboration within course frameworks. Gregory J. Skulmoski [17] examines the essential soft skills required by information systems managers to ensure success at every stage of project implementation.

The skillset prediction helps close the gap between what is needed in the workplace and what is learned in colleges. The methodology enables educators to create focused interventions that address specific skill gaps and enhance overall competency development by accurately identifying students' Technical, Life, and Soft skills. By aligning students' skills with the demands of the IT sector, this data-driven approach ensures students are better prepared to handle real-world challenges. Institutions benefit from more efficient resource allocation, focusing on essential skills required for career growth and

employability. The study's ultimate goal is to advance a more effective, personalized, and industry-relevant educational framework that equips students for success in the workplace.

3. Related Works

Machine learning models produce more accurate results for prediction tasks due to the substantial development of AI approaches. Some works proposed by various researchers for skill set prediction are presented below.

Pınar Cihan [3] highlights how crucial it is to solve the drawbacks of project-based learning strategies in order to provide students in software development courses with high-quality education. In order to assess their impact on improving project efficiency in the software industry, specifically with regard to time, cost, and quality, Riza Yosia Sunindijo [4] looks into 16 ability factors. According to E. Sventekova et al. [5], in order to encourage a stronger interest in engaging in scientific investigation, young people must acquire soft skills throughout their educational journey. Based on in-depth observations, Gnatz et al. [9] offer insights into the design of a course intended to successfully impart practical knowledge to students. The results of the study support Carmen Iriarte's [15] assertion that soft skills are crucial to the success of IT initiatives. Within the course structure, Alan R. Peslak [16] examines and outlines a number of tactics and protocols for encouraging productive group collaboration. The fundamental soft skills that information systems managers need to have in order to succeed at every stage of project implementation are examined by Gregory J. Skulmoski [17].

Zaman et al. [18] suggested a model for prediction where the authors utilized a dataset taken from software companies in Pakistan. This research brings findings that illuminates practical and theoretical contributions for a better understanding of software projects' performance complexities. This study also provides priority for social and political skills. Furthermore, the study also provides a clear idea about the complex performance relationship between software projects. Mezhoudi N et al. [19] suggested that finding the important variables influencing employability and the demands of the modern workforce may be quite beneficial to all parties involved. Students who are aware of their skills and shortcomings may be able to better plan their careers. According to Makhoba et al. [20], based on their skill-set characteristics, students can use this study to determine which degree they are most likely to succeed in. The university administrative staff, bursary sponsors, and scholarship sponsors can use this model to identify students who are academically challenging.

According to González-Marcos et al. [21], the education of project management was designed traditionally. This research work suggests an approach that is student-centered in virtual teams. It investigates the effect on student satisfaction and learning results. The results of this study imply that students in higher education benefit from motivation and psychological states. Current learning theories back up these claims. El-Sabaa [22] analyzed the talents and experiences that project managers and functional managers bring to successful performance and careers, as well as how these attributes vary. The study's conclusion offers an conceptual structure for enhancing the recruitment and effectiveness of project managers. Additionally, project managers' interpersonal abilities have a more significant impact on how they run their projects. Kotsiantis S B [23] proposed that machine learning techniques applied in the educational field constitute an emerging technology. This article uses machine learning approaches to estimate student grades in an effort to close the gap between current

regression techniques and student performance prediction. As a result, a prototype tutor support tool for software has been created. According to Md. S. and Krishnamoorthy [24], this study looks at the predictability and applicability of a characteristic called the context-dependent cognitive abilities scores. This facilitates the evaluation of a student's chances of succeeding and allows for the application of suitable interventions.

Petkovic et al. [25] proposed an analytical method for assessing teamwork in software engineering and predicting learning outcomes. The Random Forest (RF) technique is used to analyze team performance and activity data using a machine learning framework. Two independent accuracy measures were used to validate the study's findings. It is clearly evident that the random forest algorithm can precisely predict how the product team and the software engineering process will function. Lin and You [26] proposed a model to forecast skill sets and create the theoretical framework to research the project members' capacities. This study suggests a theoretical framework that aids team leaders in identifying the skill sets required for project management. Parneet Kaur et al. [27] focus on locating the students who are slow learners and exhibiting them through predictive data mining methods. The student academic records dataset is examined using various classification methods. This helps to predict accuracy and determine which classification algorithm performs best. A knowledge flow model is also demonstrated in this study among the five classifiers. Petkovic et al. [28] used an approach to evaluate and predict how well students will learn to cooperate. Machine learning (ML) is used in the study, namely random forest categorization (RF).

Ramalingam, M. and Ilakkiya, R [29] observed that every year, student performance suffers in both the classroom and during placement, which calls for analysis in order to raise student performance for subsequent batches. The only factors influencing the students' placement performance are their academic and extracurricular accomplishments. As a result, both will be considered. Kolo, D.K. and Adepoju, S.A. [30] suggested that the foundation of a society's efforts to raise the standard of its citizens is its education. The ability to forecast student academic success is necessary to raise the standard of education. The decision tree structure is created and implemented. It was shown that a variety of variables, including the students' financial situation, learning motivation, and gender, had an impact on their performance. E.B. Costa et al. [31] observed that the methods examined in our study can predict which students are most likely to fail; some of these methods are more effective when data pre-processing is used. The support vector machine method performs better than other methods. As stated by M. Kumar et al. [32], the fundamental focus of this article is to provide readers with a comprehensive understanding of the various data mining techniques that have been applied to evaluate student growth and efficiency, Since there are several methods for classifying data, Ahmed A B E D and Elaraby I S [33] suggested using the classification process to forecast students' ultimate scores. In this case, the decision tree (ID3) technique is employed. A graph convolutional network-based approach was presented by Alnasyan et al. [34] to identify students' participation in different behavioural patterns. The paradigm can be used to a wide range of knowledge graphs by default. Zhang and Yang [35] developed a deep learning model using the Nutcracker Optimisation Algorithm to forecast students' final grades.

4. Dataset Description

The dataset comprises of Soft, Lif eand Technical skills which are the key components of this study.

The dataset was prepared from the inputs from the questionnaire distributed among computer science students at various colleges in Kerala, India. This includes skill-related quiz assessments. The Soft skills of each student are evaluated based on 18 features and 5 quiz questions based on decision-making, planning, creativity, teamwork, meeting deadlines, and multitasking. For assessing Life skills, a questionnaire carries nine features and five quiz questions based on presentation skills, leadership, good listening, communication skills, and conflict in an area of interest. To measure Technical skills, there are nine features and five quiz questions consisting of time management, quality management, new technology, new hands-on tools, cost or time management challenges, and so on and so forth. The above-mentioned dataset is used to evaluate, predict, or classify the student's strengths and areas for improvement based on three skillsets. The numerous criteria chosen to predict the Soft skills, Life skills, and Technical skills of software engineering among computer science students are presented in Tables 1, 2, and 3.

Table 1. Attributes used to predict Soft skills

Sl No	Attributes to predict Soft skills
1	Decision-Making
2	Planning
3	Teamwork Experience
4	Confident Management
5	Meeting Deadlines
6	Stress Critical Situations
7	Communicate Closed Connect
8	Boosting Creativity
9	Multitask
10	Working Over Hours
11	Opinion Differences
12	Resolving Conflict
13	Conveying Unpopular Information
14	Working Alone
15	Rearranging Schedules
16	Inspiration
17	Motivation
18	Emotional Intelligence

Table 2. Attributes used to predict Life skills

Sl No	Attributes to predict Life skills
1	Demonstrator
2	Good Leader
3	Good Listening
4	Oral message communication
5	Team Building
6	Area Of Conflict Interest
7	Good Presenter
8	Coach Others
9	Interpersonal Communication

Table 3. Attributes used to predict Technical skills

Sl No	Attributes to predict Technical Skills
1	Time Management
2	Quality Management

3	Blend Of Management Plus Technical Subject
4	Gaining Extra Information
5	New Hands-On Tools
6	Group Project Activities
7	Developing Projects Alone
8	Cost Or Time Management Challenges
9	Administrative Tasks

5. Proposed Methodology

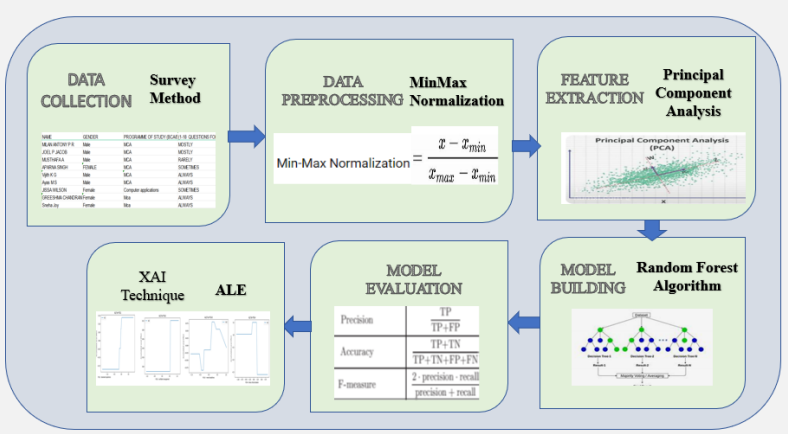

Figure 1: Proposed Methodology

The proposed study combines Random Forest with the PCA algorithm to predict the various skillsets of software engineering among computer science students. Data collection, data preprocessing, feature extraction, model selection, model training, and evaluation are among the methodology's proposed steps. The proposed methodology's layout is depicted in Figure 1.

5.1 Data Collection

Any prediction model relies heavily on data. For this study, surveys are the main method of data gathering. A survey was carried out among computer science graduates of different institutions in Kerala to identify the gaps in skills using a well-drafted questionnaire. The survey was done among students who studied software engineering as a course in their curriculum. The questionnaire consists of several questions covering various parameters related to Soft skills, Life skills, and Technical skills. Each question has six options with a score of 5, 4, 3, 2, 1, and 0, respectively. A quiz assessment of each category of skill is conducted in the survey. The dataset consists of more than a thousand samples. A score of five represents the most-rated skill, and zero represents the least-rated skill. There are five quiz questions, each from three categories of skills, for a total of fifteen. If the score of a student is more than 60% against each category of skills, then the model can correctly predict whether they possess that competence: - Soft skills, Life skills, and Technical skills. One thousand and twenty-six students were required to complete the survey.

The Survey questionnaires evaluate the students' skill sets in software engineering. Questionnaires consist of fifty-three questions and have three parts. The first part is focused on students' Soft skills, the second on their Life skills and the third part observed students' Technical skills. Eighteen features and five Softskill quizzes were used to evaluate the Soft skills of each student. Soft skills in software

engineering, such as overseeing large projects and working collaboratively with team members, receive comparatively less emphasis and support in curriculum pedagogy. Nine features and fiveLifeskill quizzes were used to evaluate the Life skills. The development of Life skills has been influenced by advancements in the workplace, such as development strategies and techniques. They include presentation skills, leadership, good listening, communication skills, and conflict in an area of interest. Nine features and five Technical skill quizzes were used to evaluate the Technical skills. The features comprise time management, quality management, new technology, new hands-on tools, cost or time management challenges, programming abilities, and so on. This survey method will help the students to understand some of their academic skills, and gives a general idea of their skillset gap. Figure 2 represents a screenshot of the dataset of the study.

Some of the sample questions used in the survey to assess the Soft skills of the students are presented below:

- Do you feel comfortable leading a team of over twenty people?
- How confident are you to communicate within your closed connect / group?
- How confident are you to spice up in group discussions to boost creativity?
- Is there anyone you would rather delegate decision-making authority to?
- Do you go about rearranging your schedule if something unplanned occurs?

Some of the sample questions used in the survey for measuring the Life skills of the students are given below:

- How good are you in resolving a conflict in area of interest in group work?
- Are you a good listener in the communication of group work?
- Are you self-efficient in giving presentations / facing larger groups?
- How good are you at interpersonal communication?
- How good a leader you are in motivating and building a core team?

Some of the questions used to measure Technical skills are provided below:

- Would you like to prefer a blend of management + Technical subject than pure management or Technical course?
- Are you open to learning new hands-on tools and techniques?
- How confident are you in developing a project alone?
- Did you think cost/compute/time management challenges are more crucial than Technical challenges in a group project?
- Have you possessed quality management in your projects?

(1-18 QUESTIONS FOR	2.Are you confident to n	3.Have you struggled to	4.Have you faced any str	5.How confident are you	6.How confident are you	7.Do you multitask in ye	8.Do you tend to work o	9.How do you stay motiv	10.How do you deal with	11.Would you pre
SOMETIMES	SOMETIMES	SOMETIMES	SOMETIMES	SOMETIMES	SOMETIMES	NEVER	NEVER	SOMETIMES	OFTEN	SOMETIMES
RARELY	NEVER	SOMETIMES	OFTEN	MOSTLY	RARELY	MOSTLY	ALWAYS	OFTEN	OFTEN	MOSTLY
ALWAYS	MOSTLY	MOSTLY	OFTEN	ALWAYS	OFTEN	RARELY	MOSTLY	SOMETIMES	SOMETIMES	MOSTLY
MOSTLY	MOSTLY	OFTEN	SOMETIMES	MOSTLY	OFTEN	SOMETIMES	SOMETIMES	SOMETIMES	OFTEN	SOMETIMES
OFTEN	NEVER	MOSTLY	SOMETIMES	ALWAYS	MOSTLY	NEVER	MOSTLY	RARELY	MOSTLY	SOMETIMES
SOMETIMES	SOMETIMES	SOMETIMES	ALWAYS	ALWAYS	SOMETIMES	RARELY	NEVER	SOMETIMES	MOSTLY	ALWAYS
MOSTLY	OFTEN	SOMETIMES	SOMETIMES	MOSTLY	OFTEN	OFTEN	SOMETIMES	RARELY	OFTEN	RARELY
MOSTLY	ALWAYS	MOSTLY	SOMETIMES	ALWAYS	SOMETIMES	RARELY	SOMETIMES	MOSTLY	OFTEN	NEVER
OFTEN	SOMETIMES	SOMETIMES	SOMETIMES	RARELY	SOMETIMES	RARELY	MOSTLY	SOMETIMES	RARELY	SOMETIMES
MOSTLY	ALWAYS	MOSTLY	MOSTLY	OFTEN	OFTEN	OFTEN	ALWAYS	OFTEN	SOMETIMES	RARELY
MOSTLY	MOSTLY	OFTEN	OFTEN	ALWAYS	OFTEN	RARELY	RARELY	SOMETIMES	MOSTLY	SOMETIMES
MOSTLY	MOSTLY	SOMETIMES	SOMETIMES	OFTEN	SOMETIMES	ALWAYS	ALWAYS	MOSTLY	MOSTLY	OFTEN
MOSTLY	NEVER	MOSTLY	MOSTLY	SOMETIMES	SOMETIMES	ALWAYS	OFTEN	RARELY	SOMETIMES	SOMETIMES
ALWAYS	OFTEN	OFTEN	SOMETIMES	MOSTLY	SOMETIMES	RARELY	MOSTLY	SOMETIMES	SOMETIMES	SOMETIMES
SOMETIMES	RARELY	ALWAYS	MOSTLY	OFTEN	SOMETIMES	MOSTLY	ALWAYS	OFTEN	OFTEN	OFTEN

Figure 2: Screenshot of dataset

5.2 Data Preprocessing

Techniques for preprocessing data prepare the dataset for model creation. Before building models, the data collected for training and testing should be suitably preprocessed to help the models quickly pick up on trends. Data with missing values cannot be analyzed to train a machine learning model. 90% of our time is spent preparing the data because of this. So missing values are inputted using the average. Using standard scaler normalization, it worked out, but we achieved better results with minmax normalization.

Minmax normalization is one of the many data preprocessing methods used in this study. In machine learning, it is extensively utilized to standardize numerical characteristics to a certain range. This normalization method transforms each feature in the dataset into a new range between a specified minimum and maximum value, typically between 0 and 1. The resulting normalized values will lie between [0, 1], where the dataset's minimum and maximum values will be changed to 0 and 1, respectively. Although other ranges can be chosen, if necessary, [0, 1] is the most common choice due to its ease of interpretation and compatibility with many machine learning algorithms. Eq. (1) displays the equation for min-max normalization, where min is the value in x that is the smallest, max is the value in x that is the largest, and x' is the normalized data [36].

$$x' = \frac{x - min}{\max - min} \qquad (1)$$

5.3 Feature Extraction

Sometimes adding more features can make the model less accurate. Therefore, feature extraction can be used to simplify the model and prevent data over-fitting. By extracting data with a lot of features, feature extraction aims to produce altered feature domain. This paper utilizes the PCA technique for the purpose of feature extraction. In a data set with more dimensions, The direction of the most significant variance is identified via PCA, and it is then projected onto a new sub-space with a smaller number of dimensions. Equation (2) illustrates how the mean-subtracted data matrix is used to calculate the covariance matrix for the available data. When D is utilised as the diagonal matrix for the Eigenvalues of C, the covariance matrix C is diagonalized by the Eigenvector matrix V that is produced in Equation (3). The Eigenvector that corresponds to the highest Eigenvalue represents the primary component of the data. The appropriate Eigenvectors supply the required components because of their importance. After that, the model receives the attributes extracted and generates a prediction [37].

$$\boldsymbol{C} = \frac{\mathbf{1}}{\mathbf{n-1}}\boldsymbol{B} * \boldsymbol{B} \qquad (2)$$

$$\boldsymbol{V^{-1}CV} = \mathbf{D} \qquad (3)$$

5.4 Model Selection and Training

The proposed model was developed using the Random Forest with PCA technique. The functions offered by the Python Scikit-learn (Sklearn) machine learning toolkit were used to implement the random forest. A well-liked supervised classification and regression method is Random Forest. A collection of unpruned classification or regression trees produced from a randomly selected sample of training data is known as the Random Forest [38]. Attributes are chosen at randomly during the induction phase, and the group's forecasts are combined to create the final forecast. Figure 3[39] illustrates the schematic of the Random Forest algorithm. Following is a description of the training part of the algorithm:

Trees hj(X,Θj) are used as the base learners in a Random Forest. The trained tree is denoted as hˆ j(x,θ j ,D) with training data D = {(x1, y1),...,(xN, yN)}, where xi = (xi,1,..., xi,p) T represents the p predictors, and yi is the response, given a particular realization θ j of Θj. The following is a description of the Random Forest classification algorithm [40].

Let's designate D = {(x1, y1),...,(xN, yN)} as the training dataset, where xi = (xi,1,..., xi,p) T. For each index from 1 through J:

1. Generate a bootstrap sample Dj from D with a size of N.

2. Employ binary recursive partitioning to construct a tree using the bootstrap sample Dj as the training data.

a. Aggregate all observations into a single node to initiate.

b. For each undivided node, execute the following steps recursively until the

stopping criterion is met:

i. Randomly select m variables at random from the p available variables.

ii. Among the m variables, find the best binary divide from step 1 for each variable.

iii. Divide the node into two descendant nodes using the partition that was established in step ii.

The formula to forecast the categorization at a new point x is as follows:

$$\hat{\mathbf{f}}(\mathbf{x}) = \mathbf{argmax}_{\mathbf{y}} \sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{J}} \mathbf{I}(\hat{\mathbf{h}}\mathbf{j}(\mathbf{x}) = \mathbf{y}) \qquad (4)$$

In this instance, hˆ j(x) is the jth tree's forecast of the answer variable at x.

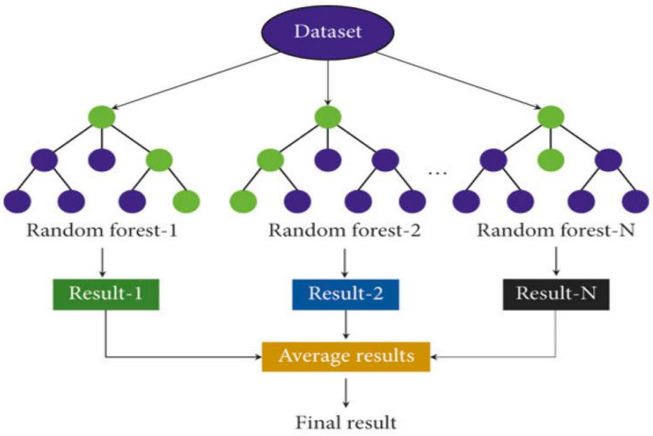

Figure 3: Schematic of Random Forest Algorithm

In this study, a randomized search is employed for hyperparameter tuning. Randomized search is one of the most popular algorithms for finding the best parameters. The number of estimators, minimum leaf sample, maximum feature, maximum depth, and bootstrap were chosen as the random forest algorithm's training parameters. When working with huge datasets, it is impossible to avoid the fact that the model's overall training time will increase as it is trained and cross-validated with different parameters [41]. The parameters with the best scores are shown in Table 4.

The number of estimators determines the number of individual decision trees that will be employed in the random forest ensemble. The minimum leaf sample determines how many samples must be present at each decision tree's leaf node. In a decision tree, the maximum feature indicates the greatest number of characteristics that are taken into account for determining the optimum divide at each node. Each decision tree's maximum depth is determined by its maximum depth. Additionally, each decision tree is built using a random forest using a method known as bootstrapped sampling, which generates numerous datasets by randomly sampling the original data and replacing it.

Table 4. Optimum values for the parameters

Sl No	Name of Parameter	Value
1	Number of estimators	351
2	Minimum samples of split	5
3	Minimum samples of leaf	2
4	Maximum depth	5
5	Maximum feature	'sqrt'
6	Bootstrap	False

5.4.1 Accumulated Local Effects (ALE)

Interpretable machine learning uses a technique called Accumulated Local Effects (ALE) [42] to examine how individual features affect a model's predictions. ALE is computationally efficient and unbiased in the presence of correlated features because, in contrast to partial dependence plots, it isolates the impacts of each feature within narrow intervals, accounting for feature interactions. ALE plots, which are averaged over the feature's distribution, show how the predictions change when a feature changes. By offering a localised perspective on feature impact, this method guarantees that the

analysis is still reliable even in cases when features are not independent. When it comes to comprehending intricate, nonlinear models like random forests or neural networks, ALE is particularly helpful.

5.5 Models Used for Comparison

The models used for comparison with the proposed study are Logistics Regression (LogR), Decision Trees (DT), Random Forest (RF) and Support Vector Machine (SVM). And these models are evaluated using the metrics accuracy, precision, recall, F1-score, and AUC.

5.5.1 Logistic Regression

A supervised learning technique that forecasts the result of a dependent variable that is classified is called logistic regression [43]. This implies that an independent or categorical value must be the outcome. It delivers probabilistic values in the 0 - 1 range rather than an exact number between 0 and 1. It can be True or False, Yes or No, 0 or 1, etc.

5.5.2 Decision Tree

A visual depiction of a structure that resembles a flowchart is called a decision tree [29]. The work's requirements are represented by the nodes, decision-making criteria are shown by the branches, and the task's outcome is the final node. The initial element in the series represents the tree's root node. The contents of each property are used by the roots node to determine how to split the tree into pieces. This iterative structure further splits the tree into multiple branches.

5.5.3 Support Vector Machine

For classification task, Support Vector Machines (SVM) [29] identify the appropriate hyperplane for dividing data into discrete classes. Method that helps to categorise many types of data. It works especially well when the data cannot be separated linearly or when you need to develop a strong decision boundary that adapts effectively to new, untested data.

5.6 Model Evaluation

The performance of the suggested framework is evaluated using the metrics—accuracy, precision, recall, F1-score, and AUC value—that have been found to be the most frequently used metrics for classification tasks. All these measurements are good indicators. So, it stands to reason that the models will function more effectively with a higher value. The precision and recall of a perfect classifier are both equal to one. Accuracy is calculated using the formula in Equation (5) and represents the ratio of accurately anticipated occurrences to all observed data. The proportion of observations among all retrieved observations is referred to as precision. It is mathematically represented in Equation (6). Recall is the fraction of actual positive instances that were appropriately identified by the model. It is mathematically represented in Equation (7). The harmonic mean of precision, recall, and F1-score can be computed using Equation (8) [44].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

$$Precision = \frac{Number\ of\ retrieved\ and\ relevant\ observations}{Total\ retrieved\ observations} = \frac{TP}{TP+FP} \quad (6)$$

$$Recall = \frac{Number\ of\ retrieved\ and\ relevant\ observations}{Total\ relevant\ observations} = \frac{TP}{TP+FN} \quad (7)$$

$$F1\ Score = 2 * \frac{(recall * precision)}{(recall + precision)} \quad (8)$$

True positive (TP) represents the count of relevant observations accurately identified by the model, as described in equations (5) to (8). The number of irrelevant observations that the model mistakenly labels as relevant is indicated by the term "false positive" (FP). The amount of relevant data that the model mistakenly classified as irrelevant is known as a false negative, or FN. The area will be between 0 and 1 always, considering that both TPR and FPR have a range of 0 to 1. Better model performance is indicated by a higher AUC value.

The area beneath the ROC curve is known as the "Area Under the Curve" (AUC). The AUC value serves as a valuable metric for assessing the performance of classification models, indicating the model's ability to distinguish between positive and negative classes across different thresholds. From the Receiver Operator Characteristic (ROC) curves, the AUC value may be calculated. The fluctuation in false positive and true positive rates is depicted by the ROC curve. Typically, AUC values fall between 0.5 and 1, with 1 indicating that the generated model is the best possible classifier.

5.7 Parameter settings of the model

The effectiveness of the suggested model, Random Forest with PCA, is compared with benchmark models, Logistics Regression (LogR), Decision Trees (DT), Random Forest (RF), and Support Vector Machine (SVM). Support vector machine algorithms perform well with both high- and low-dimensional datasets because of the kernel function. The various parameters of the decision tree are carefully adjusted in order to have a robust decision tree with good performance. A random forest is a meta-learner that uses averaging to fit numerous decision trees to various dataset sub-samples, improving prediction accuracy and lowering overfitting [40]. Table 5 displays the parameter settings for the machine learning models utilized for comparison.

Table 5. The parameters used by the models used for comparison.

Models	Parameters	Number or Type
SVM	Kernel function	Radial basis function (RBF)
DT	Maximum depth of base estimator	10
	Minimum samples of leaf of base estimator	5
	Learning rate	0.1
	Number of estimators	300
	Maximum depth of base estimator	10
RF	Number of estimators	3
	Minimum samples of split	5
	Minimum samples of leaf	2
	Maximum feature	Sqrt
	Bootstrap	FALSE

6 Results and Discussions

In this study, Random Forest with PCA is recommended for predicting students' skill sets, with PCA being used for feature extraction. Assessment of performance involves metrics such as accuracy, precision, recall, F1-score, and AUC. The effectiveness of the proposed RF-PCA model is compared with Decision Trees (DT), Logistic Regression (LogR), Support Vector Machine (SVM), and Random Forest (RF) algorithms.

6.1 Assessment of Soft Skills

Outcomes of the Soft Skills assessment are shown in Table 6. The results indicate that while Random Forest, Logistic Regression, Decision Tree, and Support Vector Machine all perform reasonably well, Random Forest with PCA achieves superior results with an accuracy of 86.56%. The Random Forest with PCA model's recall, F1-score, and precision all exceed 0.82, demonstrating the model's excellent generalizability and its ability to predict Soft Skills with the highest degree of accuracy.

6.2 Assessment of Life Skills

Table 7 presents the results related to the prediction of Life Skills. The findings demonstrate that Random Forest with PCA outperforms Support Vector Machine (SVM), Logistic Regression, Decision Tree, and Random Forest, achieving an accuracy of 84.62%. Additionally, recall, F1-score, and precision for the Random Forest with PCA model are all above 0.82, underscoring the model's robustness and its capability to accurately predict Life Skills.

6.3 Assessment of Technical Skills

In Table 8, the outcomes of the Technical Skills prediction are presented. The results indicate that Random Forest with PCA outperforms all other models, including Random Forest, Logistic Regression, Decision Tree, and Support Vector Machine, achieving an accuracy of 87.05%. The Random Forest with PCA model's recall, F1-score, and precision all exceed 0.84, highlighting the model's exceptional generalizability and its capacity to precisely predict Technical Skill levels.

Table 6. Analysis of Accuracy, Precision, Recall, F1-score, and AUC values across different models for predicting Soft skills

ModelEmployed	Accuracy	Precision	Recall	F1-Score	AUC value
LogR	0.7014	0.7166	0.7678	0.7413	0.6929
DT	0.8059	0.7920	0.8839	0.8354	0.7958
SVM	0.8009	0.7812	0.8928	0.8333	0.7958
RF	0.8507	0.8202	0.9375	0.8750	0.8395
RF-PCA	**0.8656**	**0.8244**	**0.9642**	**0.8888**	**0.8529**

Table 7. Analysis of Accuracy, Precision, Recall, F1-score, and AUC values across different models predicting Life skills

Model Employed	Accuracy	Precision	Recall	F1-Score	AUC value
LogR	0.6069	0.5700	0.6129	0.5906	0.6073
DT	0.8109	0.8125	0.7959	0.8041	0.8105
SVM	0.6716	0.6310	0.6989	0.6632	0.8243
RF	0.8208	0.7786	0.9134	0.8407	0.8175
RF-PCA	**0.8462**	**0.8252**	**0.8673**	**0.8457**	**0.8462**

Table 8. Analysis of Accuracy, Precision, Recall, F1-score, and AUC values across different models predicting Technical skills

Model Employed	Accuracy	Precision	Recall	F1-Score	AUC value
LogR	0.7064	0.7289	0.7222	0.7255	0.7051
DT	0.8308	0.7747	0.9052	0.8349	0.8347
SVM	0.8258	0.7542	0.9368	0.8356	0.8347
RF	0.8606	0.8474	0.9259	0.8849	0.8661
RF-PCA	**0.8705**	**0.8442**	**0.9537**	**0.8956**	**0.8747**

Figure 4 demonstrates the ROC curves for five different algorithms used in predicting the Soft, Life and Technical skills of computer science students: Logistic Regression, Decision Tree, Support Vector Machine, Random Forest, and Random Forest-PCA.

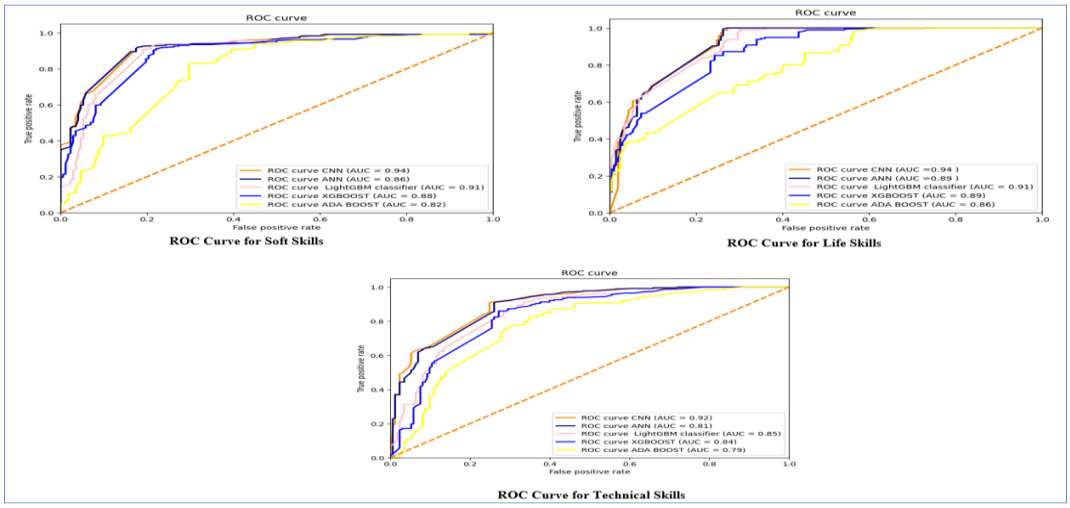

Figure 4: ROC curves for Soft, Life and Technical skills

6.4 Confusion Matrix of Proposed Model

The confusion matrix is an excellent tool for assessing classifier performance. After using neural networks on a training set of data, the results are displayed as confusion matrices, which are two-dimensional matrices. The confusion matrix's operation is depicted in Figure 5 [20]. The confusion matrix for the Soft skill of the RF-PCA is shown in Figure 6, and with an accuracy of 86.56%, it is the most accurate classifier. The confusion matrix for Life skill of the RF-PCA is shown in Figure 7, and with an accuracy of 84.62%, It is the classifier with greatest results. The RF-PCA's Technical skill confusion matrix is shown in Figure 8, and with an accuracy of 87.05%, it is the best classifier available.

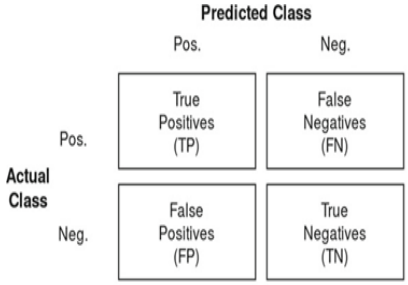

Figure 5: Confusion matrix breakdown

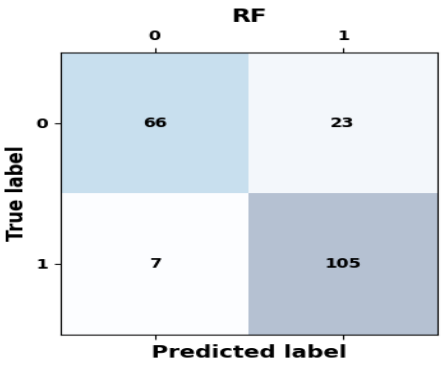

Figure 6: Confusion matrix for Soft skills

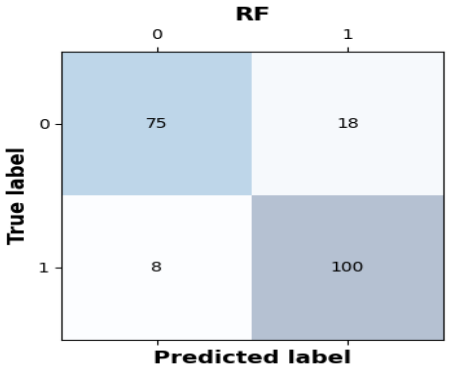

Figure 7: Confusion matrix for Life skills

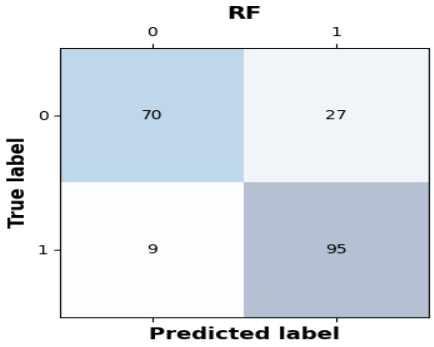

Figure 8: Confusion matrix for Technical skills

6.5 Comparison of the proposed model with other models

The model's effectiveness is thoroughly assessed using various evaluation metrics, providing insights into its overall performance. This study employs diverse statistical measures, including accuracy, precision, recall, F1-score, and AUC, for the comprehensive evaluation of the RF-PCA model. To gauge its performance, the RF-PCA model is compared against alternative machine learning models such as Random Forest (RF), Logistic Regression (LogR), Decision Tree (DT), and Support Vector Machine (SVM). This comparative analysis across accuracy, precision, recall, F1-score, and AUC highlights the RF-PCA model's proficiency in predicting skill sets. Figures 9,10 and 11 visually depict the comparison of the proposed model with others in predicting Soft skills, Life skills, and Technical skills, showcasing its superior performance across these metrics. The results from the figures unmistakably indicate the RF-PCA model's outperformance compared to alternative models. The figures clearly show that the suggested RF-PCA model outperforms other models. The findings from the figures unmistakably indicate the RF-PCA model's outperformance compared to alternative models.

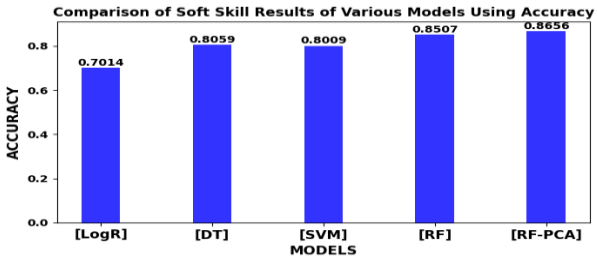

Figure 9: Accuracy values for prediction of Soft skills

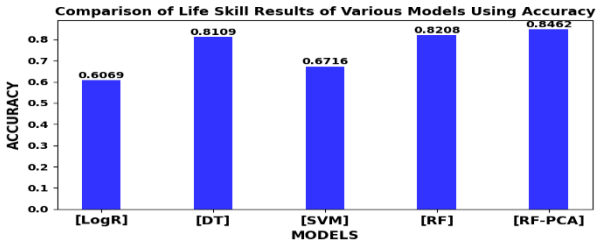

Figure 10: Accuracy values for the prediction of Life skills

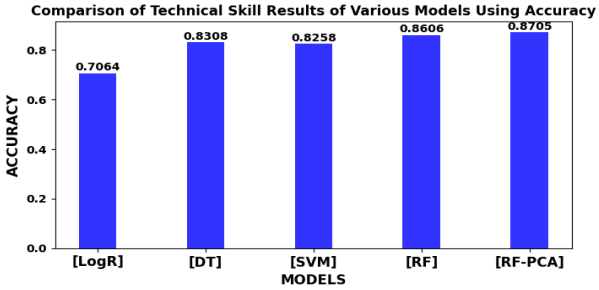

Figure 11: Accuracy values for prediction of Technical skills

6.6 Interpretability of the proposed model using ALE

Accumulated Local Effects (ALE) explains how the predictions of the Random Forest model are influenced by the PCA components, which are linear combinations of the original features. ALE isolates the effects of each PCA component by averaging the changes in the model's predictions when the value of each component is varied within predetermined intervals, while keeping other components constant. This approach ensures that the impact of each component is evaluated independently. The findings reveal the relationship between PCA components and model predictions, offering insights into how different aspects of data variance contribute to the skill set predictions.

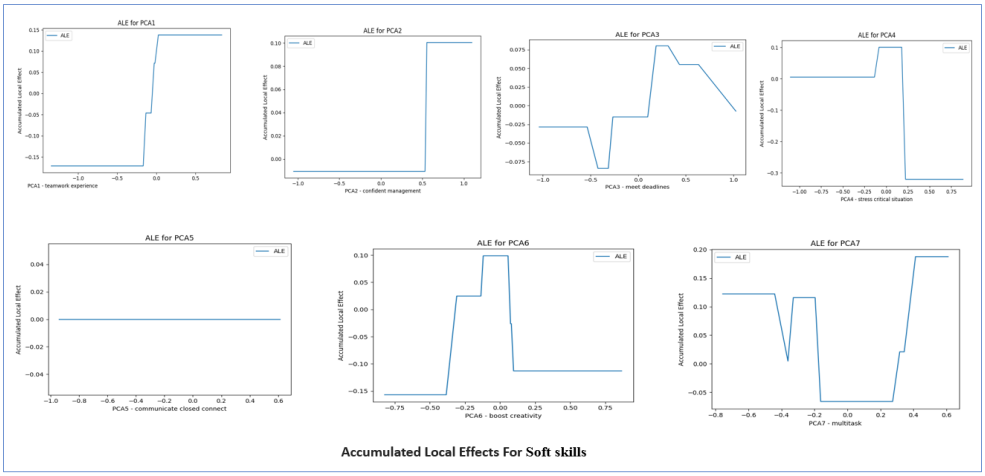

Figure 12: ALE plot for Soft skills

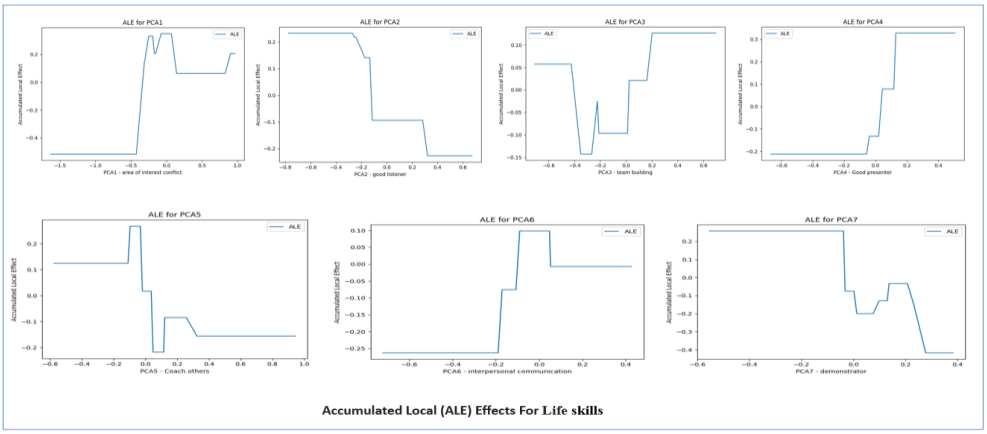

Figure 13: ALE plot for Life skills

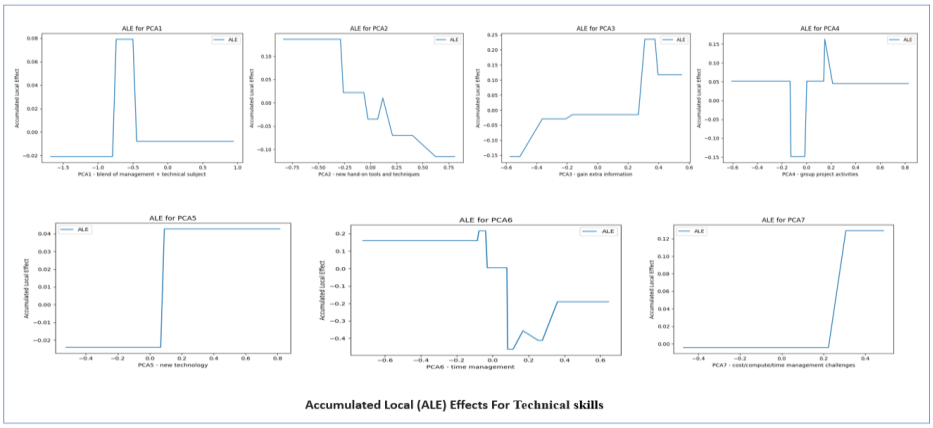

Figure 14: ALE plot for Technical skills

Plotting the ALE for each PCA component for Soft, Life, and Technical skills is illustrated in Figures 12, 13, and 14. PCA components with clear and consistent patterns, such as PCA1 and PCA4, exhibit trends that are straightforward to interpret and likely play a significant role in accurately predicting Soft skills. The model's predictions are consistently and significantly influenced by components like PCA2 and PCA6, emphasizing their importance in correctly predicting Life skills. PCA3 and PCA7 display notable trends and variations, indicating their strong influence in accurately forecasting Technical skills.

ALE curves with positive slopes indicate that increasing the value of the component enhances the likelihood of predicting a specific class, whereas curves with negative slopes suggest the opposite effect. When combined with PCA loadings, this provides a comprehensive interpretation of feature importance and model behavior by pinpointing the original features that contribute most to influential components. The plot of ALE for all PCA components in Soft, Life, and Technical skills is shown in Figure 15. Each original feature's contribution to each PCA component is visually represented in the PCA loadings heatmap where color intensity signifies the magnitude and direction of the contribution. These heatmaps facilitate the interpretation of how specific features affect the model's predictions through PCA components. When paired with ALE analysis, the heatmap traces these components back

to the original features. While ALE calculates the impact of changes in each PCA component on predictions, the heatmap identifies which original features are most influential. Together, they illustrate the importance of the transformed components to the model and reveal how the original features indirectly influence predictions by contributing to key PCA components.

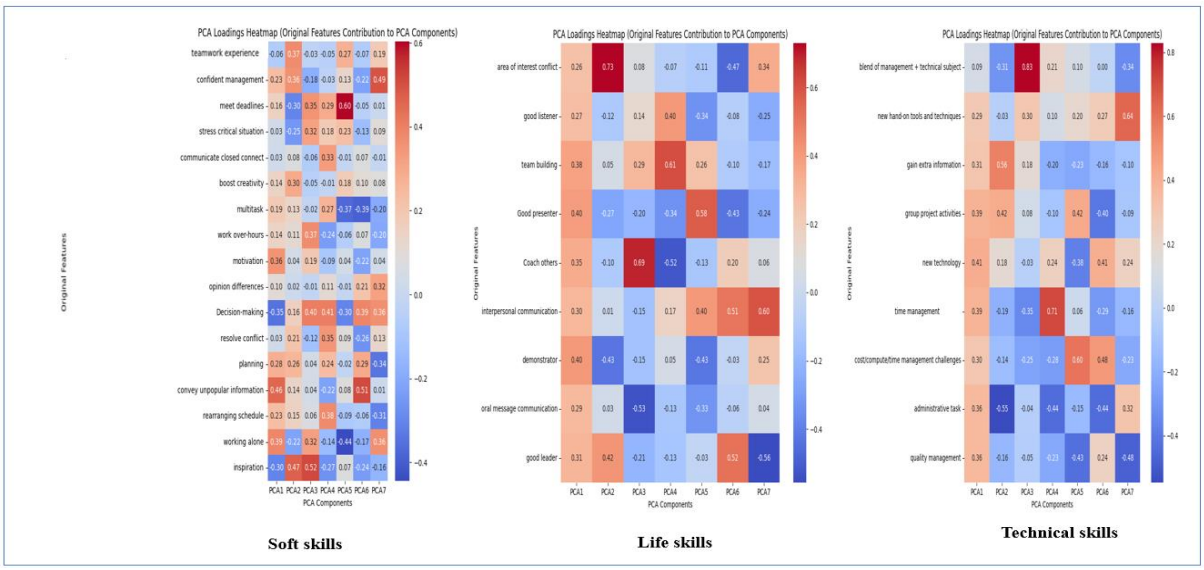

Figure 15: Plot ALE for all PCA Components in Soft, Life, and Technical skills

7.Conclusion And Future Works

This proposed study predicts students' skill sets, including Soft, Life, and Technical skills, using a Random Forest model combined with the PCA approach and ALE. The performance of the proposed model is compared with machine learning techniques, including logistic regression, decision trees, and support vector machines. Experimental results indicate that the Random Forest model with PCA demonstrates significant generalization capabilities and outperforms baseline approaches in accurately forecasting computer science students' software engineering talents. The use of ALE in RF-PCA provides a clear interpretation of how PCA components, derived from original features, influence the predictions of a Random Forest model. ALE enhances the explainability of complex models by isolating the impact of each PCA component and linking them to the original features using PCA loadings. This combination retains the advantages of PCA's dimensionality reduction while enabling a deeper understanding of feature relevance and helping identify key factors influencing model decisions. The study aims to identify skill gaps among students, promote skill-based education in software engineering programs, and better prepare students for the workforce while bridging the knowledge gap between academia and the software industry.

Through skillset prediction and assessment, this study seeks to identify students with high competence levels to improve the teaching-learning process. This approach greatly benefits educators and students by providing insights into both technical and cognitive abilities. It fosters a culture of deeper understanding, empowering students to critically evaluate their work and take greater ownership of

their education. This method offers a more accurate and comprehensive analysis than traditional evaluation techniques, enabling businesses and educational institutions to select and develop proficient software engineers with enhanced expertise. Educators also gain a better understanding of students' personalities and strengths, facilitating more personalized instruction. Although the study's objectives of promoting skill-based learning and aligning education with market demands are achieved, there are some limitations. For example, the model primarily relies on the combination of PCA and Random Forest, which, while effective, may not capture intricate correlations as well as more advanced methods like deep learning. Additionally, focusing on a limited set of skills (Soft, Life, and Technical skills) may overlook other critical abilities such as creativity or adaptability. Furthermore, this study did not explore the model's adaptability to disciplines outside software engineering or its scalability across diverse student demographics.

Future advancements could include incorporating deep learning-based feature extraction and prediction algorithms to improve the precision and robustness of skillset predictions. Expanding the scope to include a broader range of skills and testing the model on larger datasets would also enhance its generalizability. Despite these limitations, the study represents a significant step in leveraging skill-focused assessments to bridge the knowledge gap between academia and industry.

Refrences

[1] Mahanti R, Mahanti PK. Software engineering education from indian perspective. In18th Conference on Software Engineering Education & Training (CSEET'05) 2005 Apr 18 (pp. 111-117). IEEE.
[2] Belzer K. Project management: still more art than science. InPM Forum Featured Papers 2001 (pp. 1-6).
[3] Cihan P, Kalıpsız O. Evaluation of Students' Skills in Software Project. TEM Journal. 2014 Feb 1;3(1).
[4] Sunindijo RY. Project manager skills for improving project performance. International journal of business performance management. 2015 Jan 1;16(1):67-83.
[5] Sventekova E, Lovecek T. Project-based teaching, practice in the academic environment. InLatest advances in educational technologies: proceedings of the 11th WSEAS international conference on education and educational technology (EDU 12): Singapore City, Singapore 2012 May 11.
[6] Cihan P, Kalıpsiz O. Assessing the human factors in software development courses students project. International Conferance on Education and Educational Technologies. 2013.
[7] Begel A, Simon B. Struggles of new college graduates in their first software development job. InProceedings of the 39th SIGCSE technical symposium on Computer science education 2008 Mar 12 (pp. 226-230).
[8] Begel A, Simon B. Novice software developers, all over again. InProceedings of the fourth international workshop on computing education research 2008 Sep 6 (pp. 3-14).
[9] Gnatz M, Kof L, Prilmeier F, Seifert T. A practical approach of teaching software engineering. InProceedings 16th Conference on Software Engineering Education and Training, 2003.(CSEE&T 2003). 2003 Mar 20 (pp. 120-128). IEEE.
[10] Yeh RT. Educating future software engineers. IEEE Transactions on education. 2002 Feb;45(1):2-3.
[11] Garcia I, Pacheco C, Coronel N. Learn from practice: defining an alternative model for software engineering education in Mexican universities for reducing the breach between industry and academia. InProceedings of the International Conference on Applied Computer Science 2010 Sep 15 (pp. 120-124).
[12] Karunasekera S, Bedse K. Preparing software engineering graduates for an industry career. In20th Conference on Software Engineering Education & Training (CSEET'07) 2007 Jul 3 (pp. 97-106). IEEE.
[13] Subrahmanyam GV. A dynamic framework for software engineering education curriculum to reduce the gap between the software organizations and software educational institutions. In2009 22nd Conference on Software Engineering Education and Training 2009 Feb 17 (pp. 248-254). IEEE.

[14] Mahmood Z. A framework for software engineering education: a group projects approach. International Journal of Education and Information Technologies–Powered GoogleDoc Journals. 2007 Mar;1(3):153-6.
[15] Iriarte C, Bayona Orè S. Soft skills for it project success: A systematic literature review. InTrends and Applications in Software Engineering: Proceedings of the 6th International Conference on Software Process Improvement (CIMPS 2017) 6 2018 (pp. 147-158). Springer International Publishing.
[16] Peslak AR. Teaching software engineering through collaborative methods. Issues in Information Systems. 2004;5(1):247-53.
[17] Skulmoski GJ, Hartman FT. Information systems project manager soft competencies: A project-phase investigation. Project Management Journal. 2010 Mar;41(1):61-80.
[18] Zaman U, Jabbar Z, Nawaz S, Abbas M. Understanding the soft side of software projects: An empirical study on the interactive effects of social skills and political skills on complexity–performance relationship. International Journal of Project Management. 2019 Apr 1;37(3):444-60.
[19] Mezhoudi N, Alghamdi R, Aljunaid R, Krichna G, Düştegör D. Employability prediction: a survey of current approaches, research challenges and applications. Journal of Ambient Intelligence and Humanized Computing. 2023 Mar;14(3):1489-505.
[20] Makhoba L, Jadhav A, Sixhaxa K, Ajoodha R. Evaluation of Student Skill-Sets as Predictors of Success at Higher Education Institutions. InProceedings of International Conference on Communication and Computational Technologies: ICCCT 2022 2022 Sep 27 (pp. 585-600). Singapore: Springer Nature Singapore.
[21] González-Marcos A, Alba-Elías F, Navaridas-Nalda F, Ordieres-Meré J. Student evaluation of a virtual experience for project management learning: An empirical study for learning improvement. Computers & Education. 2016 Nov 1;102:172-87.
[22] El-Sabaa S. The skills and career path of an effective project manager. International journal of project management. 2001 Jan 1;19(1):1-7.
[23] Kotsiantis SB. Use of machine learning techniques for educational proposes: a decision support system for forecasting students' grades. Artificial Intelligence Review. 2012 Apr;37:331-44.
[24] Md S, Krishnamoorthy S. Student performance prediction, risk analysis, and feedback based on context-bound cognitive skill scores. Education and Information Technologies. 2022 Apr;27(3):3981-4005.
[25] Petkovic D, Sosnick-Pérez M, Huang S, Todtenhoefer R, Okada K, Arora S, Sreenivasen R, Flores L, Dubey S. Setap: Software engineering teamwork assessment and prediction using machine learning. In2014 IEEE frontiers in education conference (FIE) proceedings 2014 Oct 22 (pp. 1-8). IEEE.
[26] Lin HY, You J. Teamwork-Performance Prediction by Using Soft Skills and Technological Savvy Skills. Journal of University Teaching and Learning Practice. 2021;18(8):9.
[27] Kaur P, Singh M, Josan GS. Classification and prediction based data mining algorithms to predict slow learners in education sector. Procedia Computer Science. 2015 Jan 1;57:500-8.
[28] Petkovic D, Okada K, Sosnick M, Iyer A, Zhu S, Todtenhoefer R, Huang S. Work in progress: a machine learning approach for assessment and prediction of teamwork effectiveness in software engineering education. In2012 frontiers in education conference proceedings 2012 Oct 3 (pp. 1-3). IEEE.
[29] Ramalingam M, Ilakkiya R. Data mining algorithms (KNN & DT) based predictive analysis on selected candidates in academic performance. In2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence) 2021 Jan 28 (pp. 332-337). IEEE.
[30] Kolo DK, Adepoju SA. A decision tree approach for predicting students academic performance.
[31] Costa EB, Fonseca B, Santana MA, de Araújo FF, Rego J. Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses. Computers in human behavior. 2017 Aug 1;73:247-56.
[32] Kumar M, Singh AJ, Handa D. Literature survey on student's performance prediction in education using data mining techniques. International Journal of Education and Management Engineering. 2017 Nov 1;7(6):40-9.
[33] Ahmed AB, Elaraby IS. Data mining: A prediction for student's performance using classification method. World Journal of Computer Application and Technology. 2014 Feb;2(2):43-7.

[34] Alnasyan B, Basheri M, Alassafi M. The Power of Deep Learning Techniques for Predicting Student Performance in Virtual Learning Environments: A Systematic Literature Review. Computers and Education: Artificial Intelligence. 2024 May 3:100231.
[35] Zhang X, Yang L. A Convolutional Neural Network-Based Predictive Model for Assessing the Learning Effectiveness of Online Courses Among College Students. International Journal of Advanced Computer Science & Applications. 2024 Sep 1;15(9).
[36] KU J, Kovoor BC. A Wavelet-based hybrid multi-step Wind Speed Forecasting model using LSTM and SVR. Wind Engineering. 2021 Oct;45(5):1123-44.
[37] Gupta I, Sharma V, Kaur S, Singh AK. PCA-RF: an efficient Parkinson's disease prediction model based on random forest classification. arXiv preprint arXiv:2203.11287. 2022 Mar 21.
[38] Cutler A, Cutler DR, Stevens JR. Random forests. Ensemble machine learning: Methods and applications. 2012:157-75.
[39] Fu H, Qi K. Evaluation model of teachers' teaching ability based on improved random forest with grey relation projection. Scientific Programming. 2022;2022(1):5793459.
[40] Nair A, Parameter tuning with grid search: A hands-on introduction. *Analytics India Magazine.* 2019.
[41] Fraj MB, In Depth: Parameter Tuning for Random Forest. 2019.
[42] Okoli C. Statistical inference using machine learning and classical techniques based on accumulated local effects (ALE). arXiv preprint arXiv:2310.09877. 2023 Oct 15.
[43] Boateng EY, Abaye DA. A review of the logistic regression model with emphasis on medical research. Journal of data analysis and information processing. 2019 Oct 12;7(04):190.
[44] Zhu L, Qiu D, Ergu D, Ying C, Liu K. A study on predicting loan default based on the random forest algorithm. Procedia Computer Science. 2019 Jan 1; 162:503-13