Journal of Theoretical and Applied Information Technology
15th November 2024. Vol.102. No. 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PREDICTION AND ASSESSMENT OF SOFTWARE ENGINEERING SKILLSET AMONG COMPUTER SCIENCE STUDENTS USING CONVOLUTIONAL NEURAL NETWORKS THROUGH EXPLAINABLE AI

JASMIN NIZAR [1], R SHARMILA [2], K U JASEENA[3]

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

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

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

E-mail: [1]jasminnizar.nizar@gmail.com, [2]sharmila.ramasamy@kahedu.edu.in, [3]jaseena@mesmarampally.org

ABSTRACT

The dynamic nature of the software industry necessitates a focus on actual skills alongside academic proficiency. Success in software development projects is often attributed to a well-rounded combination of Soft skills, Life skills, and Technical expertise. The development of skills stands out as the most reliable method for helping students to acquire the essential competencies needed to navigate the complexities of the modern world. Practitioners in software engineering often express concern that graduates lack the necessary preparation for a successful career in the field. Consequently, it has become imperative for educational institutions and employers to accurately assess and predict the skill sets of software engineering students. Achieving a balance among these competencies is vital for excelling in the collaborative and ever-evolving realm of software development endeavours. The prediction and assessment of software engineering skillsets among computer science students are essential to ensuring that graduates are well-equipped for the evolving demands of the professional landscape. This proactive approach aligns academic programmes with industry requirements, allowing educators and institutions to gauge the effectiveness of their teaching methodologies and curriculum. This study aims to provide a comprehensive evaluation and prediction of software engineering skills among computer science students, employing a Convolutional Neural Network (CNN) model. The Convolutional Neural Network model is utilised to predict the presence of Soft, Life, and Technical skills in students. The architecture involves a 1D Convolutional Neural Network for feature extraction from these skill categories, followed by a fully connected network for classification. The model incorporates Explainable Artificial Intelligence through interpretability, ensuring accountability. Local and vision-based explainability are explored using techniques such as Shapley Additive Explanations (SHAP), Local Interpretable Model-Agnostic Explanations (LIME), and Layer-wise Relevance Propagation (LRP). The performance of the proposed CNN- XAI model is assessed using metrics such as accuracy, precision, recall, F1 score, and the AUC value. Simulation results show that the suggested CNN-XAI model outperforms other basic models in predictive accuracy.

Keywords: *Convolutional Neural Network, Software Engineering, Explainable Artificial Intelligence, SHAP, LIME, LRP.*

1. INTRODUCTION

In the ever-evolving landscape of technology, the demand for proficient software engineers has reached unprecedented heights. Software engineering, as a swiftly evolving discipline necessitates individuals armed with a diverse array of skills. The significance of cultivating a robust software engineering skillset cannot be emphasized enough in our technology-driven society. Proficient software engineers play a pivotal role, contributing to technological advancement by innovating applications, designing platforms, and ensuring the security and reliability of software systems. Their expertise in dissecting intricate problems, crafting

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

efficient solutions, and producing clean, maintainable code is foundational.

As businesses increasingly lean on digital solutions, software engineering skills emerge as a basis for organizational triumph. Furthermore, the ever-changing technological landscape demands perpetual learning and adaptation, rendering a potent software engineering skillset not just valuable but imperative for professionals. As the demand for software engineer's surges, it becomes essential to delineate the key skills that students must possess for success in this domain, encompassing Soft skills, Life skills, and Technical or Hard skills [1]. Those pursuing software engineering should exhibit a diverse skill set to navigate the multifaceted aspects of the IT industry, encompassing risk management, technology, personnel, and resources. Leveraging advancements in deep learning, it is now conceivable to predict the software engineering skillset of computer science students using various data inputs.

Skills development encompasses the identification of students' skill gaps and addressing them through various training and career opportunities [2]. This multifaceted process extends beyond the mere acquisition of technical expertise, delving into the cultivation of Soft, Life, and Technical skills. Soft skills encompass communication, teamwork, adaptability etc. which are crucial for professional success and effective collaboration. Life skills, including critical thinking and problem-solving etc., contribute to a holistic and adaptable approach to challenges. Simultaneously, Technical skills form the foundation for specialised knowledge and proficiency in specific domains, ensuring students are well-equipped for the intricacies of their chosen fields. By providing a thorough focus on these many skill sets, skill development seeks to equip students for a dynamic and constantly changing work environment.

The software industry, facing a high rate of project failures, reveals a significant gap between industry demands and the readiness of software engineering graduates. Concerns arise about graduates lacking essential skills for the dynamic software industry, suggesting potential shortcomings in current educational approaches. Inadequate knowledge and abilities for software project management tasks, worsened by instructional methods often lacking experiential learning, are cited by the industry as reasons why software engineering graduates may face challenges in the demanding job market. This misalignment between industry expectations and educational outcomes creates a potential barrier between the IT sector and the education system. However, commendable efforts are underway to bridge this gap and establish a more responsive approach to software engineering education. The prediction and assessment of software engineering skillsets, facilitated by advanced technologies like deep learning, play a crucial role in identifying these gaps. Educators can use real-time insights into individual students' strengths and weaknesses to tailor curriculum, teaching methods, and resources, addressing specific skill deficiencies. This proactive strategy aligns educational programmes with industry demands, ensuring students acquire the most relevant and up-to-date skills. Skillset predictions also empower educators to implement interventions and support mechanisms, fostering a more inclusive and effective learning environment.

Identifying and nurturing essential skills among computer science students is a critical undertaking for both educators and industry professionals. The advent of advanced deep learning techniques, particularly Convolutional Neural Networks, offers powerful tools for predicting and assessing software engineering skillsets. Although CNNs are traditionally associated with computer vision tasks, their applicability extends to diverse domains, including software engineering skill assessment. CNNs excel in learning hierarchical representations, making them adept at capturing intricate patterns in complex datasets. In the context of skill prediction, these neural networks can analyse code snippets, project repositories, and other relevant data to extract meaningful features correlated with a student's software engineering proficiency. In addition to CNNs, the integration of explainable artificial Intelligence techniques, such as SHAP, LIME, and LRP, becomes crucial for enhancing model interpretability. While accurate predictions are paramount, understanding the rationale behind the model's decisions is equally vital, especially in educational contexts. XAI methods aim to make the decision-making process of deep learning models more transparent and comprehensible.

By incorporating XAI into the CNN framework, educators gain insights into the specific features contributing to a student's skill assessment. This openness not only strengthens the model's credibility but also facilitates more informed and constructive feedback from educators. Moreover, the significance of employing XAI techniques alongside CNN lies in their ability to reveal the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

inner workings of the model, providing a deeper understanding of the factors influencing skill predictions. The combination of CNNs and XAI techniques, such as SHAP, LIME, and LRP, not only enables accurate software engineering skill predictions but also ensures interpretability, fostering transparency and trust in the educational assessment process. The highlights of the paper are presented below:

(1) The paper introduces a predictive model that incorporates CNN with XAI to predict various skill sets in software engineering, namely, Soft, Life, and Technical skills.

(2) SHAP, LIME, and LRP techniques are employed in this study to enhance model interpretability by identifying pivotal features in skillsets.

(3) Tailored datasets are utilised in the research.

(4) The evaluation of the CNN model with XAI involves a thorough analysis using diverse metrics to gauge its effectiveness.

The paper is structured into several sections for clarity and coherence. Firstly, Section 1 offers a concise introduction to various methods and strategies used for assessing students' skill levels. Following this, Section 2 conducts a thorough review of the relevant literature. Section 3 details the dataset utilised in the study. The proposed methodology is outlined in Section 4, while Section 5 delves into the presentation and analysis of results. The implications of the predictions are discussed in Section 6. Finally, Section 7 encapsulates the research's findings, conclusions, and suggestions for future investigations.

2. LITERATURE SURVEY

The Information Technology (IT) sector stands at the forefront of innovation in the current digital era, creating a continuous demand for professionals with robust software engineering skillsets. Beyond being a role within the IT industry, software engineering serves as the backbone driving the entire sector forward. Recent graduates in software engineering often face challenges as their university-acquired skills may not align with industry requirements [3]. In the digital world, the IT industry is the driving force, and software engineers play a crucial role as unsung heroes. Their skillset proves indispensable in fostering innovation, meeting customer needs, ensuring quality and security, enhancing efficiency, adapting to change, promoting collaboration, and solving complex problems. With ongoing technological advancements, the significance of software engineering skills in the IT industry is set to grow. Therefore, a solid foundation in software engineering is not just beneficial but essential for those aspiring to thrive in this dynamic field [4].

IT companies invest resources, both in terms of time and money, to assist recent graduates in acquiring the necessary skill set [5]. Employers can optimise their resources by hiring practitioners who have undergone appropriate training, leading to cost and time savings. Simultaneously, employees can enhance their employability by recognising the most crucial skills for their field. Conversely, understanding the essential skill set is pivotal for academia when updating curricula. Therefore, identifying weak areas among students and anticipating their skill set is crucial for informed decisions, especially when determining whether to specialise in software engineering [6].

Several studies have been conducted to identify the software engineering skill set required for the IT industry. Akdur [3] suggested that knowing essential skills could enhance job readiness, helping employers minimize training time by quickly integrating well-prepared practitioners. Garousi et al. [4] observed that many software engineering graduates face early-career challenges due to a skills mismatch between academic learning and industry needs. In a related study, Garousi et al. [5] further emphasized that this discrepancy is a key barrier for new graduates. Scaffidi [6] highlighted the importance of both technical and soft skills, particularly qualities like adaptability, innovation, and rapid learning for computer science graduates. Gnatz et al. [7] supported this by describing a course model focused on practical skill transfer.

Flambeau et al. [8] used convolutional neural networks (CNNs) in a multi-label classification model for IT job applications, allowing viewers to observe factors influencing the model’s decisions. Sventekova and Lovecek [9] argued that universities should foster soft skills to enhance students’ engagement with the most recent information and participation in research. According to Iriarte and Bayona [10], interpersonal skills, including leadership and conflict resolution, are crucial for the success of software projects. Similarly, Peslak [11] investigated collaboration techniques within educational settings, while Gregory and Hartman [12] analyzed the soft skills

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

necessary for effective project management in information systems.

Zaman et al. [13] explored the complementary roles of political and social skills in software projects, aligning with emerging trends in project management. El-Sabaa [14] compared managerial skills valued by project and functional managers, finding that both groups prioritize distinct skill sets for career success. Jiechieu and Tsopze [15] proposed a CNN-based multi-label classification model aimed at identifying specific skills in job postings.

Research on academic performance prediction has also leveraged deep learning models. For instance, Li X et al. [16] created a model that uses student data to predict academic achievement. Lau et al. [17] combined traditional and neural network analysis for predictive modeling, while Mubarak et al. [18] applied graph convolutional networks to analyze behavioral patterns in academic participation. Khalid and Kumar [19] showed that deep learning models outperform linear regression for assessing students' learning outcomes.

Innovative methodologies continue to emerge, with Li et al. [20] using CNNs for classroom teaching behavior analysis, and Aydoğdu [21] employing an artificial neural network (ANN) to identify predictors of academic success, focusing on variables like attendance and engagement. Liu et al. [22] demonstrated the effectiveness of resume parsing algorithms with CRNN, achieving high accuracy in skill identification. Zhao et al. [23] emphasized skill normalization as a vital factor in labor market analysis, while Kalyani [24] proposed CNNs for predictive accuracy. Studies like Hussain et al. [25] and Neha et al. [26] contrasted a neural network model with expert system algorithms for student categorization, finding the neural approach superior.

Alnasyan et al. [27] proposed a graph convolutional network-based model to detect students' involvement in various behavioural patterns. The paradigm is naturally generalisable to a variety of knowledge graphs. Kartik et al. [28] applied LIME and SHAP for interpretability in academic achievement prediction, highlighting behavioral factors. Ahmed [29] employed clustering algorithms to analyze student academic performance, observing the strong predictive power of SVM. Nizar et al. [30] developed a predictive framework combining Random Forest, principal component analysis, and explainable AI to assess students' technical, soft, and life skills. Zhang and Yang [31] created a deep learning model that used the Nutcracker Optimization Algorithm (NOA) and an enhanced version (I-NOA) to optimize a multi-layer CNN for predicting final grades of students based on online learning behaviors.

Previous research has mostly concentrated on conventional evaluation methodologies, sometimes lacking Explainable AI methods. Because these approaches are difficult to comprehend and cannot clearly reveal the variables causing skill gaps, it is difficult for educators to provide focused treatments. By using interpretable machine learning models more especially, Convolutional Neural Networks (CNN) augmented with explainable AI tools like SHAP, LIME, and LRP. This study sets itself apart and reveals important elements impacting students' abilities. By doing this, it fills the knowledge vacuum created by earlier research by offering a fresh method for comprehending and assessing software engineering's complicated skill sets. Additionally, this study assesses the models' accuracy and reliability, emphasising their capacity to identify skill gaps. These results are then intended to assist teachers in better matching students' abilities to industry norms.

The existing research predominantly focuses on Soft skills and Technical skills within software engineering, leading to a noticeable gap in exploring Life skills. This study acknowledges the crucial role of Life skills in predicting software engineering skill sets. While few articles delve into deep learning models for predicting students' skill sets, the emphasis on explainable AI for identifying and predicting skill improvement areas remains limited. Concentrating specifically on Soft, Life, and Technical skills, this research aims to bridge the gap in current work by highlighting the importance of utilising deep learning techniques in education. By showcasing the potential of these models, the study aims to offer valuable insights into skill assessment and enhancement processes. The research underscores the significance of CNN in accurately predicting whether software engineering students possess Soft, Life, or Technical skills. Considering these factors, it is reasonable to conclude that convolutional neural networks using Explainable XAI emerge as the most suitable approach for achieving the outlined objectives in the proposed study.

3. DATASET

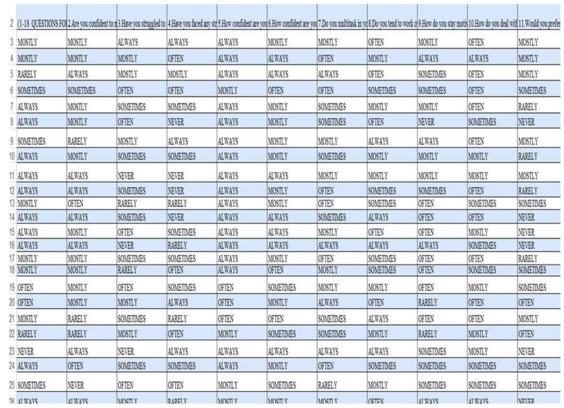
The accuracy of a prediction model is significantly influenced by the quality of the dataset. A high-quality dataset enhances model

Journal of Theoretical and Applied Information Technology
15th November 2024. Vol.102. No. 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

performance and ensures more reliable predictions. In this study, the dataset was created by computer science students from various institutions in Kerala, India. In the upcoming subsection, we will provide an introduction to the dataset and outline the features incorporated within it. In the domain of predictive algorithms, where anticipating outcome is paramount, datasets serve as the foundational bedrock. For this research, surveys emerged as the principal avenue for data collection. A meticulously crafted questionnaire was administered to computer science students across various institutions in Kerala, specifically those pursuing software engineering courses. The survey encompassed Soft, Life, and Technical skills, with each question presenting six possible answers, graded from 0 to 5. This extensive survey engaged over two thousand students, evaluating each skill category through a dedicated assessment quiz comprising a total of fifteen questions. The model effectively gauges a student's competence in Soft, Life, and Technical skills if they score above 60% in each area. The dataset is cultivated from responses collected from two thousand and four students.

The survey questionnaire, comprising three segments and fifty-three questions, investigates the nuanced assessment of students' software engineering skill sets. The initial section focuses on Soft skills, researching aspects such as project management and team communication through five tests and eighteen characteristics. Acknowledging the challenges in supporting Soft skills, especially in managing large projects, the survey navigates the complexities of curricular pedagogy in software engineering. Life skills take centre stage in the second segment, evaluated through five tests and nine features, capturing workplace practices influenced by development strategies and approaches. This includes conflict resolution, leadership, good listening, communication, and presentation skills. The final section concentrates on Technical skills with evaluation through five tests and nine features covering diverse aspects like cost or time management issues, adaptation to new technology, hands-on tool utilisation, time and quality management, and programming abilities. The survey emerges as a valuable tool, providing students with profound insights into their academic strengths and areas for improvement. Figure 1 offers a visual representation, offering a snapshot of the study's dataset.



2	(1-18 QUESTIONS FO	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 yo	8.Do you tend to work o	9.How do you stay moti	10.How do you deal wit	11.Would you prefer
3	MOSTLY	MOSTLY	ALWAYS	ALWAYS	ALWAYS	MOSTLY	MOSTLY	OFTEN	MOSTLY	OFTEN	MOSTLY
4	MOSTLY	MOSTLY	MOSTLY	OFTEN	ALWAYS	ALWAYS	OFTEN	MOSTLY	ALWAYS	ALWAYS	MOSTLY
5	RARELY	ALWAYS	MOSTLY	MOSTLY	ALWAYS	ALWAYS	ALWAYS	OFTEN	SOMETIMES	OFTEN	MOSTLY
6	SOMETIMES	SOMETIMES	OFTEN	OFTEN	MOSTLY	OFTEN	OFTEN	SOMETIMES	SOMETIMES	OFTEN	SOMETIMES
7	ALWAYS	MOSTLY	SOMETIMES	SOMETIMES	ALWAYS	MOSTLY	SOMETIMES	MOSTLY	MOSTLY	OFTEN	RARELY
8	ALWAYS	MOSTLY	OFTEN	NEVER	ALWAYS	MOSTLY	SOMETIMES	OFTEN	NEVER	SOMETIMES	NEVER
9	SOMETIMES	RARELY	MOSTLY	ALWAYS	ALWAYS	MOSTLY	MOSTLY	ALWAYS	ALWAYS	OFTEN	MOSTLY
10	ALWAYS	MOSTLY	SOMETIMES	SOMETIMES	ALWAYS	MOSTLY	SOMETIMES	MOSTLY	MOSTLY	MOSTLY	RARELY
11	ALWAYS	ALWAYS	NEVER	NEVER	ALWAYS	ALWAYS	MOSTLY	MOSTLY	MOSTLY	MOSTLY	MOSTLY
12	ALWAYS	ALWAYS	SOMETIMES	NEVER	ALWAYS	MOSTLY	OFTEN	SOMETIMES	SOMETIMES	OFTEN	RARELY
13	MOSTLY	OFTEN	RARELY	RARELY	ALWAYS	MOSTLY	OFTEN	SOMETIMES	OFTEN	SOMETIMES	SOMETIMES
14	ALWAYS	ALWAYS	SOMETIMES	NEVER	ALWAYS	ALWAYS	SOMETIMES	ALWAYS	OFTEN	OFTEN	NEVER
15	ALWAYS	MOSTLY	OFTEN	SOMETIMES	ALWAYS	ALWAYS	MOSTLY	OFTEN	OFTEN	MOSTLY	NEVER
16	ALWAYS	ALWAYS	NEVER	RARELY	ALWAYS	ALWAYS	ALWAYS	ALWAYS	ALWAYS	SOMETIMES	NEVER
17	MOSTLY	MOSTLY	SOMETIMES	SOMETIMES	ALWAYS	MOSTLY	OFTEN	SOMETIMES	OFTEN	OFTEN	RARELY
18	MOSTLY	MOSTLY	RARELY	OFTEN	ALWAYS	OFTEN	MOSTLY	SOMETIMES	OFTEN	SOMETIMES	SOMETIMES
19	OFTEN	MOSTLY	OFTEN	SOMETIMES	OFTEN	SOMETIMES	MOSTLY	MOSTLY	OFTEN	MOSTLY	SOMETIMES
20	OFTEN	MOSTLY	MOSTLY	ALWAYS	OFTEN	MOSTLY	ALWAYS	OFTEN	RARELY	OFTEN	OFTEN
21	MOSTLY	RARELY	SOMETIMES	RARELY	OFTEN	OFTEN	SOMETIMES	ALWAYS	OFTEN	OFTEN	MOSTLY
22	RARELY	RARELY	MOSTLY	OFTEN	MOSTLY	SOMETIMES	SOMETIMES	MOSTLY	RARELY	MOSTLY	OFTEN
23	NEVER	ALWAYS	NEVER	ALWAYS	ALWAYS	ALWAYS	ALWAYS	ALWAYS	SOMETIMES	MOSTLY	NEVER
24	ALWAYS	OFTEN	SOMETIMES	SOMETIMES	ALWAYS	MOSTLY	OFTEN	SOMETIMES	SOMETIMES	SOMETIMES	SOMETIMES
25	SOMETIMES	NEVER	OFTEN	OFTEN	MOSTLY	SOMETIMES	RARELY	MOSTLY	SOMETIMES	SOMETIMES	SOMETIMES
26	ALWAYS	ALWAYS	MOSTLY	RARELY	MOSTLY	MOSTLY	MOSTLY	OFTEN	ALWAYS	ALWAYS	NEVER

Figure 1: Snapshot of dataset

3.1 Survey Queries

The aim of this study is to predict software engineering students' skill sets, including Soft, Life, and Technical skills. This information shows the importance of integrating computer science education with industrial needs in order to deliver higher-quality software. To accomplish our objective, we posed the following research queries (RQs):

Below are a few of the research queries that were utilised in the survey to gauge the students' Soft skills:

1. Have you struggled to meet deadlines for any submission?
2. Do you have best teamwork experience in your studies?
3. Have you faced any stress in a critical situation of your student project?
4. Do you successfully resolve a conflict in a team work?
5. How do you stay motivated while working on a group project that doesn't interest you?

Below are a few of the research queries that were utilised in the survey to gauge the students' Life skill:

1. Do you coach others if any difficulty faced in group projects?
2. How confident are you in presentation of topic in class?
3. How effective and efficient you are on oral communication?

4. How good being enthusiastic mean to you in team building?
5. How effective are you as a leader in inspiring and creating a core team?

Below are a few of the research queries that were utilised in the survey to gauge the students' Technical skill:

1. How eager are you to study new technology in your studies?
2. Do you think that group project activities have helped you learn new methods, procedures and processes?
3. Do you approach any administrative task in your studies?
4. Rate yourself in programming abilities?
5. How confident are you working on a project by yourself?

3.2 Features

This study's dataset is a comprehensive exploration encompassing three primary dimensions: Soft, Life, and Technical skills. Originating from questionnaire responses, these insights were gathered from computer science students across diverse institutions in Kerala, India, supplementing the dataset with skill-related knowledge assessments. Soft skills assessment delves into decision-making, planning, creativity, teamwork, meeting deadlines, and multitasking through a series of five questions. Life skills are scrutinized via a questionnaire boasting nine features and five questions, uncovering insights into presentation skills, leadership, good listening, communication proficiency, and conflict resolution. Meanwhile, the evaluation of Technical skills, with nine features and five questions, reveals mastery in time management, quality control, adaptation to new technologies, hands-on tool utilization, and programming prowess. The dataset systematically categorizes students, revealing their strengths and areas for improvement across these three skill sets. Tables 1 and 2 serve as a window into the factors integral to predicting Soft, Life, and Technical skills within the domain of software engineering for computer science students.

Table 1. Features utilized to predict Soft skills

Sl No	Features to predict Life Skills	Features to predict Technical Skills
1	Demonstrator	Time Management
2	Good Leader	Quality Management
3	Good Listening	Blend Of Management Plus Technical Subject
4	Oral message communication	Gaining Extra Information
5	Team Building	New Hands-On Tools
6	Area Of Conflict Interest	Group Project Activities
7	Good Presenter	Developing Projects Alone
8	Coach Others	Cost Or Time Management Challenges
9	Interpersonal Communication	Administrative Tasks

Table 2. Features utilized to predict Life and Technical Skills

Sl No	Features 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

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

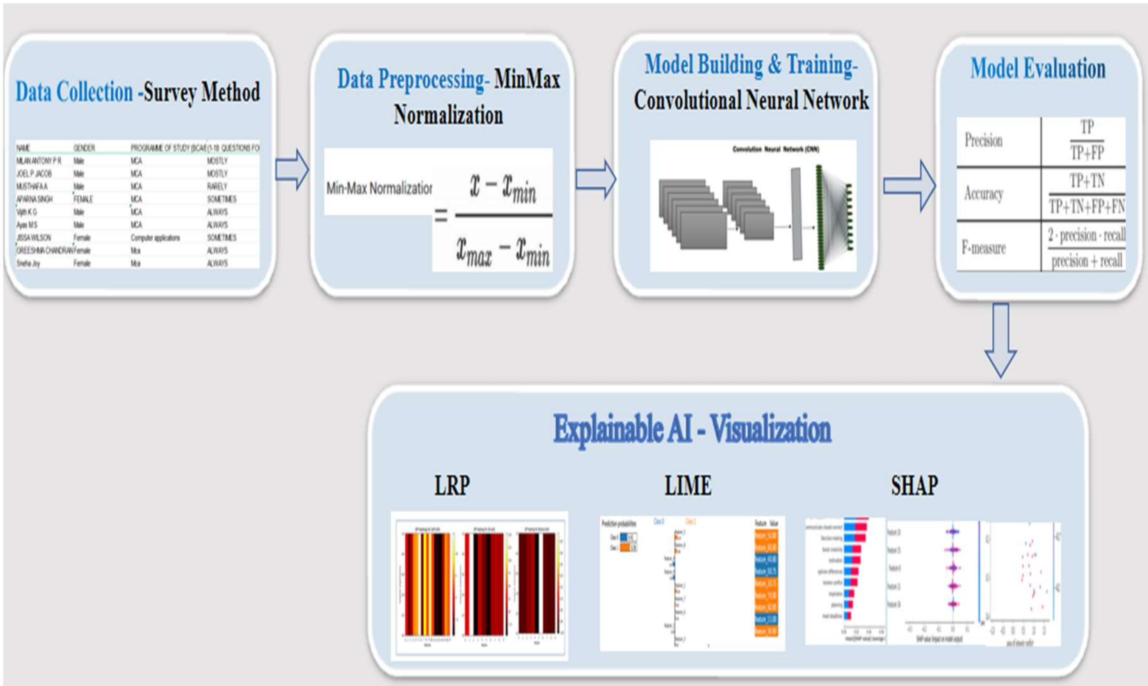


Figure 2: Proposed Framework

4. PROPOSED METHODOLOY

In this study, a Convolutional Neural Network model with XAI is employed to predict the diverse skill sets within software engineering that computer science students might possess. The suggested framework encompasses several crucial phases, starting with data collection and followed by data preprocessing, feature extraction, model selection, model training, and evaluation. The organized structure of this methodology is visually represented in Figure 2, providing a clear roadmap for the implementation and assessment of the predictive model.

4.1 Data Preprocessing

In the pursuit of extracting valuable insights from datasets to enhance predictive capabilities, data preprocessing emerges as a crucial step [32]. This preprocessing stage significantly influences the effectiveness of Convolutional Neural Networks. Efficient pattern detection by these models relies on properly preprocessing the data gathered for testing and training. Convolutional Neural Networks cannot effectively be trained on datasets with missing values, necessitating the reduction of such gaps in the original data to enhance result accuracy. Additionally, certain data properties may be ignored during training to boost efficiency, especially if they don't contribute meaningfully to the model. Properly preprocessed data translates into improved training and more accurate predictions. The study explored various normalisation techniques, including MinMax Normalisation, MaxAbsScaler, and StandardScaler algorithms. After several trial-and-error experiments, the MinMax normalisation method demonstrated the most favourable results.

Normalisation, a crucial pre-processing step involving mapping or scaling, is integral to refining data for analysis [33]. This study adopts MinMax normalisation [34] as the chosen data preprocessing technique, also known as min-max scaling. The Min-Max normalisation method is a straightforward approach that precisely fits data into a predefined range. This method transforms a feature, ensuring all its values lie within the range [0, 1]. While the most commonly used range is [0, 1], other ranges can be selected based on specific requirements. Equation 1 illustrates the fundamental Min-Max

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

normalisation formula, where x' represents the normalised data, min is the lowest value in x, and max is the highest value in x. This technique plays a pivotal role in preparing data for effective utilisation in the subsequent stages of the study.

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

4.2 Model Selection and Training

This section unveils the formulation of the suggested Convolutional Neural Network model, specifically utilizing a 1D Convolutional Neural Network architecture tailored for handling sequential data. The versatility and efficiency of Convolutional Neural Networks in managing sequential data are well established, with typical architectures featuring multiple convolution layers followed by pooling layers. Throughout the network's progression, the dimensions of the input data decrease and the depth increases as more feature maps are incorporated. At its pinnacle, a standard feed-forward neural network with fully connected layers is integrated, ultimately leading to the final layer responsible for producing predictions. These networks excel at capturing spatial and temporal dependencies through the application of relevant filters. The architecture of convolutional neural networks revolves around three key layers: the convolution layer, pooling layer, and fully connected layer. The convolution layer, situated at the initial stage, extracts essential features from the input using filters, generating a feature map. The introduction of nonlinearity through the RELU layer and subsequent spatial dimension down sampling in the pooling layer reduce parameters while preserving vital information. The flattened output is then connected to a fully connected layer, responsible for computing class scores essential for input classification [35].

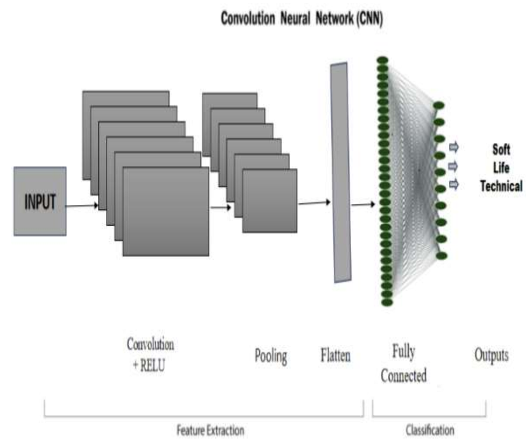


Figure 3: Anatomy of 1D Convolutional Neural Network

The anatomy of a Convolutional Neural Network adopted for classification is shown in figure 3 includes six functional modules: Input layer, Convolution Layer, ReLU Activation, Pooling Layer, Dense Layer, and Output layer.

1. Input Layer: Use a series of values to represent the input data.

2. Convolution Layer: To find patterns or features in an input sequence, convolutional layers with filters glide over it.

3. ReLU Activation: Non-linearity is introduced using a Rectified Linear Unit (ReLU) activation function following each convolutional process.

4. Pooling Layer: By sampling the spatial dimensions from below, pooling layers shorten the sequence while preserving crucial information.

5. Flatten Layer: A vector is created by flattening the output of the convolutional and pooling layers.

6. Dense Layer: The final classification judgements are made by integrating the information learnt by the convolutional layers with one or more fully connected (dense) layers.

7. Output: The class with the highest probability is selected as the final prediction after the network provides probabilities for each class.

Table 3: The architecture of 1D Convolutional Neural Networks

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Layer (Type)	Output shape	Param #
conv1d (Conv1D)	(None, 8, 2048)	6144
max_pooling1d	(None, 4, 2048)	0
flatten (Flatten)	(None, 8192)	0
dense (Dense)	(None, 512)	419486
dense_1 (Dense)	(None, 256)	131328
dense_2 (Dense)	(None, 128)	32896
dense_3 (Dense)	(None, 64)	8256
dense_4 (Dense)	(None, 1)	65
		Total params: 4373505 (16.68 MB) Trainable params: 4373505 (16.68 MB) Non-trainable params: 0 (0.00 Byte)

In a Convolutional Neural Network, the transformation of feature map values is a critical step, facilitated by the utilisation of activation functions. Among the diverse range of activation functions available, the Rectified Linear Unit (RELU) is employed in this architecture. The specific configuration of the convolutional neural network employed in this study is detailed in Table 3. The first convolution layer extracted 6144 features. In a 1D CNN, the max pooling layer plays a pivotal role in reducing the dimensionality of the input sequence while preserving essential characteristics. Subsequently, the input features undergo flattening in the flattening layer, and they are then fed into the dense layer, also referred to as the fully connected layer. The output layer is commonly used for binary classification problems, as it consists of a single neuron. This architecture had 4,373,505 extracted parameters in total, all of which were trainable, and the model contained no non-trainable parameters. This well-designed CNN architecture is ready to gather and analyse pertinent data efficiently for further classifications and predictions.

4.3 Explainable Artificial Intelligence (XAI)

Explainable Artificial Intelligence (XAI) [36] is a rapidly emerging field focused on developing models and systems that produce comprehensible and interpretable outcomes. Traditional machine learning models, particularly deep neural networks, often function as "black boxes," making it difficult for users to understand the reasoning behind their decisions. XAI aims to address this opacity by designing algorithms that offer clear explanations alongside accurate predictions or classifications. By enhancing the interpretability of AI models, XAI promotes more accessible and understandable interactions between humans and intelligent systems, fostering trust, accountability, and ethical use of AI across various domains.

4.3.1 Shapely additive explanations (SHAP)

Shapley Additive Explanations [SHAP] [37] is a potent and popular Explainable Artificial Intelligence approach. Inspired by cooperative game theory and the concept of Shapley values, it offers a comprehensive understanding of how each feature contributes to the output of a machine learning model. By displaying how each element influences the model's predictions, SHAP values are used to provide a unique significance value to each feature.

This enables users to better understand how sophisticated models, such as ensemble techniques or deep neural networks, make decisions. By decomposing predictions into intelligible parts, SHAP values promote transparency and help in the identification of significant elements, hence facilitating model interpretation. Being a flexible and user-friendly tool, SHAP has grown to be an essential component of XAI.

4.3.2 LIME

Within Explainable Artificial Intelligence, a popular approach is Local Interpretable Model-Agnostic Explanations (LIME) [37]. It provides interpretable, human-readable explanations that are

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

locally accurate, so resolving the interpretability problem of intricate machine learning models. In order to estimate the model's decision boundary near a particular instance, LIME operates by varying the input data and tracking the resulting changes in the model's predictions. Through the process of fitting an easily comprehensible "local" model to these perturbed samples, LIME offers insights into the behavior of the model in the near vicinity of a particular input. With this method, consumers may understand model predictions individually. Because of its adaptability and model-neutrality, LIME is a useful tool for boosting AI system transparency and confidence.

4.3.3 LRP

Explainable Artificial Intelligence uses a technique called Layer-wise Relevance Propagation (LRP) [37] to shed light on how deep neural networks make decisions. Each neuron in the network is given a relevance score by LRP, which measures each neuron's contribution to the final prediction. Following that, these scores are transmitted backward across the network's layers, emphasizing the significance of various input characteristics. By assigning importance to certain characteristics and connections, LRP seeks to increase transparency regarding the inner workings of neural networks. By doing this, LRP enhances interpretability and user confidence in complicated deep learning models by assisting users in understanding which aspects of the input data were critical in shaping the model's output.

4.4 Models Used For Comparisons

This study conducts a systematic comparison between the proposed CNN with XAI model and well-established algorithms, including Artificial Neural Networks (ANN), LightGBM (Light Gradient Boosting Machine), Adaptive Boosting (AdaBoost), and Extreme Gradient Boosting (XGBoost). The primary objective of this comparative analysis is to glean valuable insights into the heightened prediction capabilities afforded by CNN with the Explainable Artificial Intelligence (XAI) methodology. Through this exploration, the study endeavours to shed light on the superior predictive performance and interpretability achieved by the CNN model in comparison to other prominent algorithms.

4.4.1 Artificial neural networks (ANN)

The architecture of the human brain serves as an inspiration for Artificial Neural Networks [38], which are the foundations used to build deep learning models. An Artificial Neural Network is a system of linked nodes arranged in layers that can learn intricate patterns and representations from input. These networks, which have an input layer, hidden layers, and an output layer and adjust their weights during training to optimise for certain tasks, may be used for a variety of tasks such as image recognition, natural language processing, and regression issues since they can modify their weights during training to optimise for certain tasks. However, interpretability issues are frequently caused by their intricate and deep structure.

4.4.2 LightGBM (Light gradient boosting machine)

LightGBM [39] is a fast and efficient gradient boosting system that works especially well with big datasets. Using a gradient-boosted decision tree construction technique, it utilises a tree-based learning approach to repeatedly enhance model performance. Compared to conventional depth-wise methods, LightGBM's leaf-wise growth strategy minimises loss more efficiently, leading to quicker training periods and improved prediction accuracy. Its efficiency in handling high-dimensional data has led to its appeal in machine learning contests, where it is frequently utilised for tasks like regression and classification.

4.4.3 Extreme gradient boosting (XGBoost)

The robust and scalable gradient boosting library Extreme Gradient Boosting, or XGBoost [39], is well-known for its efficacy and efficiency in predictive modelling. Similar to other boosting algorithms, XGBoost constructs a group of weak learners—typically decision trees—and then combines them to produce a strong model. Compared to conventional gradient boosting techniques, XGBoost is quicker and less prone to overfitting since it incorporates regularisation terms, pruning procedures, and parallel computing abilities. It has grown to be a well-liked option in machine learning contests and the standard method for a number of tasks, such as ranking, regression, and classification issues. XGBoost is widely used in business and academia because of its outstanding efficiency, speed, and adaptability.

4.4.4 Adaptive boosting (AdaBoost)

Adaptive Boosting, or AdaBoost [39], is an approach for ensemble learning that combines weak learners—usually decision trees—to create a strong model, therefore improving performance. AdaBoost gives misclassified instances weights during training so that weaker learners can concentrate on the hardest examples. The weak learners' contributions are added together and weighted according to their accuracy to create the final model. AdaBoost is renowned for being easy to use, efficient, and versatile in its ability to generalise to a wide range of datasets. Frequently employed in binary classification issues, it finds utility in face identification and further computer vision assignments.

4.5 Model Evaluation

The evaluation of the proposed model hinges on pivotal metrics such as accuracy, precision, recall, and F1-score, universally recognized as essential benchmarks for assessing classification problems. Higher values of these measures indicate greater efficacy, making them reliable indicators of the model's success. A perfect classifier would ideally have precision and recall equal to one, highlighting their crucial function in assessing the correctness and comprehensiveness of the model.

Accuracy, denoting the ratio of correctly predicted events to all observed data, is computed using the formula in Equation (2). Precision, another crucial metric, gauges the proportion of meaningful observations among all retrieved observations and is expressed mathematically in Equation (3). Conversely, recall evaluates the fraction of actual positive samples correctly identified by the model, with its mathematical representation provided in Equation (4). Equation (5) [41] is frequently used to compute the harmonic mean of precision, recall, and F1-score in order to provide a comprehensive evaluation. This combination of criteria provides a thorough assessment of the model's overall performance and a sophisticated understanding of its prediction powers and ability to balance precision and recall.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

$$Precision = \frac{\text{Number of retrieved and relevant observations}}{\text{Total retrieved observations}} = \frac{TP}{TP+FP} \tag{3}$$

$$Recall = \frac{\text{Number of retrieved and relevant observations}}{\text{Total relevant observations}} = \frac{TP}{TP+FN} \tag{4}$$

$$F1\ Score = 2 * \frac{(recall * precision)}{(recall + precision)} \tag{5}$$

True Positive (TP) represents the instances where the model correctly identifies relevant observations, as outlined in equations (5) to (8). Conversely, "false positive" (FP) refers to the model incorrectly labelling irrelevant observations as relevant, while "False Negative" (FN) occurs when the model mistakenly classifies relevant data as irrelevant. The Receiver Operator Characteristic (ROC) curve, particularly the "Area Under the Curve" (AUC), stands as a pivotal metric for evaluating classification models. Ranging between 0 and 1, both True Positive Rate (TPR) and False Positive Rate (FPR) contribute to the AUC value, with a higher AUC indicating superior model performance. This metric, derived from the ROC curve, quantifies a model's discrimination power by illustrating the balance between false positives and true positives at various thresholds. An AUC value falling between 0.5 and 1 characterizes a good classification model, with a higher value closer to 1 denoting superior classification performance. Therefore, AUC serves as a critical indicator in assessing the overall effectiveness and discriminatory power of a classification model.

4.6 Parameter settings of the model

In this study, hyperparameter tuning played a crucial role in optimizing the algorithm's performance by determining the most effective configuration. Key parameters, such as the number of hidden layers, processing nodes in each layer, maximum iteration of epochs, and activation functions in the dense and output layers, were carefully selected for training the model. When dealing with extensive datasets, the training and cross-validation processes inherently demand a longer duration, as acknowledged in previous research [40]. The comparison of models involved a detailed examination of the parameters used, as outlined in Table 4. This comprehensive approach aimed to identify the best combination of hyperparameters for achieving optimal model performance and accuracy.

(3)

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4: The parameters used by the models for comparison

Classification Models	Parameters	Number or Type
CNN	1D Convolution layer	(kernel_size=3, activation='relu')
	Pool layer	(pool_size=2)
	No of dense layers	6
	Number of neurons in Dense layers	42048, 1024, 512, 256,128,64
	Activation function of the dense layers	Relu
	Activation function of the output layer	Linear
	Maximum iteration of epochs	200
ANN	Number of layers	5
	Number of neurons in the hidden layer	1048, 1024, 512, 256,128
	Activation function of the hidden layers	Relu
	Activation function of the output layer	Linear
	Maximum iteration of epochs	200
LightGBM, AdaBoost,XG Boost	Number of estimators	351
	Minimum samples of split	5
	Minimum samples of leaf	2
	Maximum depth	5
	Maximum feature	'sqrt'
	Bootstrap	FALSE

5. RESULTS AND DISCUSSIONS

The study recommends the adoption of Convolutional Neural Networks in tandem with explainable Artificial Intelligence to predict students' skill sets, utilizing XAI for heightened interpretability. The assessment of model performance involves metrics such as accuracy, precision, recall, f1-score, and Area Under the Curve (AUC). In order to assess the efficacy of the proposed CNN with XAI model, a thorough comparative analysis is carried out against alternative algorithms, namely Artificial Neural Network, LightGBM, Adaptive Boosting, and XGBoost. This comprehensive comparison is designed to offer valuable insights into the relative strengths and performance characteristics of the CNN model integrated with SHAP, LIME, and LRP for interpreting and predicting students' skill sets.

5.1 Assessment of Soft, Life, and Technical skills

The assessments for Soft, Life, and Technical skills are detailed in Tables 5, 6, and 7, respectively. Comparative analysis shows that Artificial Neural Networks, LightGBM, AdaBoost, and XGBoost yield similar results. However, it is the Convolutional Neural Network that truly shines, boasting remarkable accuracy of 88.27% for Soft skills, 89.77% for Life skills, and 88.02% for Technical skills. These metrics collectively underscore CNN's exceptional generalizability and its ability to predicting Soft, Life, and Technical skills with remarkable accuracy. The findings highlight the CNN approach as a robust model for assessing and predicting skills across various domains, offering valuable insights into individuals' proficiencies in these areas.

Table 5. Comparison of Accuracy, Precision, Recall, F1-score and AUC values of different models for predicting Soft skills

Model Employed	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC value**
ANN	0.8852	0.8043	0.9946	0.8894	0.8926
LightGBM	0.8304	0.7679	0.9333	0.8425	0.9107
AdaBoost	0.7605	0.7064	0.8279	0.7623	0.8611
XGBoost	0.8227	0.7510	0.9408	0.8353	0.8949
CNN	**0.8977**	**0.8445**	**0.9804**	**0.9074**	**0.9461**

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 6. Comparison of Accuracy, Precision, Recall, F1-score and AUC values of different models for predicting Life skills

Model Employed	Accuracy	Precision	Recall	F1 Score	AUC Value
ANN	0.8703	0.8520	0.9342	0.8912	0.8601
LightGBM	0.8653	0.8228	0.9587	0.8855	0.9113
AdaBoost	0.7780	0.7768	0.8552	0.8141	0.8244
XGBoost	0.8553	0.8320	0.9342	0.8801	0.8863
CNN	**0.8827**	**0.8764**	**0.9322**	**0.9034**	**0.9413**

Table 7. Comparison of Accuracy, Precision, Recall, F1-score and AUC values of different models for predicting Technical skills

Model Employed	Accurac	Precision	Recall	F1-Score	AUC value
ANN	0.8104	0.7469	0.9289	0.8280	0.8125
LightGBM	0.8004	0.7447	0.9035	0.8165	0.8532
AdaBoost	0.7506	0.6932	0.8832	0.7767	0.7956
XGBoost	0.7730	0.7245	0.8680	0.7898	0.8447
CNN	**0.8802**	**0.8502**	**0.9444**	**0.8947**	**0.9279**

Figure 4 illustrates the ROC curves for five different algorithms used in predicting the soft skillset of computer science students: Artificial Neural Networks, LightGBM, AdaBoost, XGBoost, and CNN.

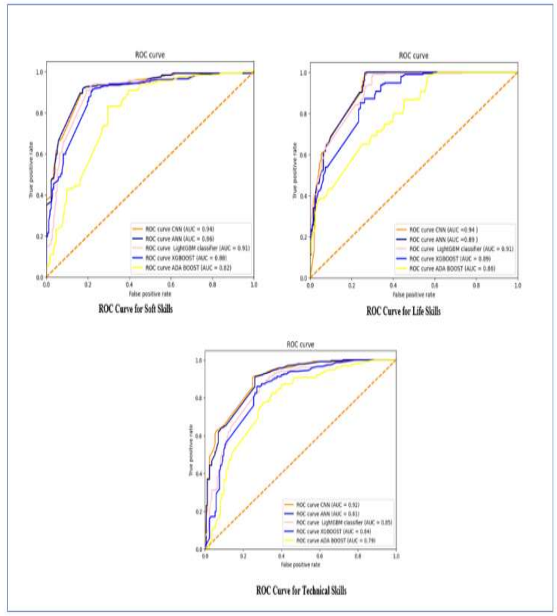


Figure 4: ROC curve for Soft, Life and Technical skills

5.2 Comparison of the proposed model with other models

The CNN model's effectiveness is extensively assessed through a comprehensive evaluation utilizing a wide array of statistical metrics, offering a thorough comprehension of its overall performance. This evaluation encompasses accuracy, precision, recall, F1-score, and AUC, providing a comprehensive analysis of the CNN model's predictive prowess. Figure 5 provides a visual depiction of the proposed model's comparative performance in predicting Soft, Life, and Technical skills, highlighting its superior capabilities. These visual representations unmistakably showcase the CNN model's dominance over alternative models, solidifying its position as a robust and reliable choice for predicting diverse skill sets.

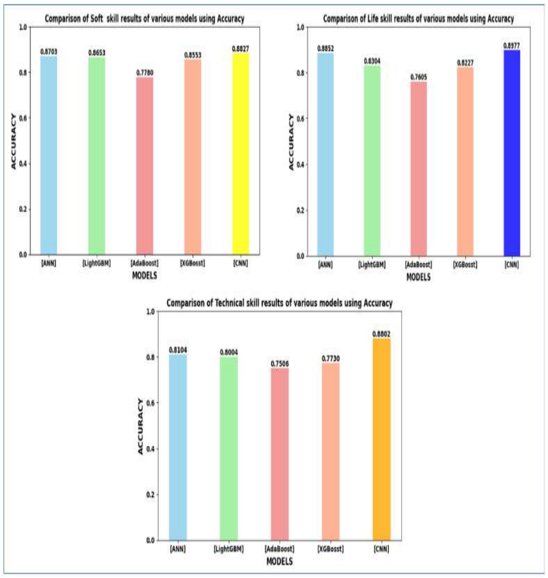
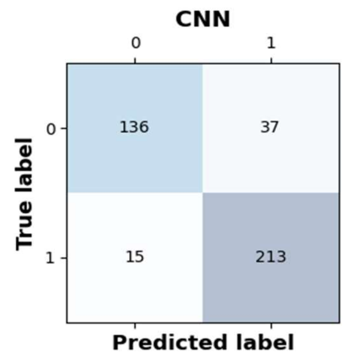


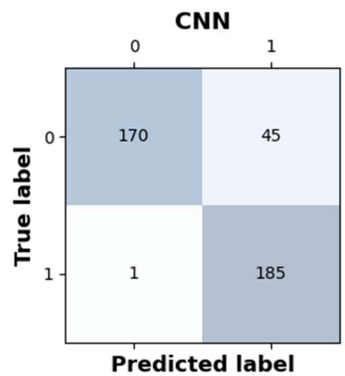
Figure 5: Accuracy values for prediction of Soft, Life, Technical skills

5.3 Confusion Matrix of Proposed Model

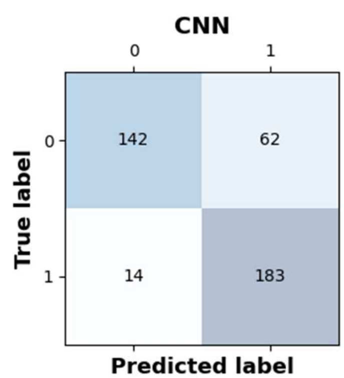
A confusion matrix [42] is employed in this study for accessing the effectiveness of classification model's. In order to shed light on the accuracy and mistakes of the model, it presents a summary of the counts of true positive, true negative, false positive, and false negative predictions. Reaffirming its usefulness as a strong predictive model, the CNN model with XAI exhibits amazing adaptability in obtaining high accuracy across a variety of skill domains. The CNN classifier's predictions for Soft, Life, and technical skills are visually represented by the confusion matrix in Figure 6. Out of 401 predictions, 349 instances for Soft skills were correctly predicted by the classifier, whereas 52 erroneous predictions were made. There were 355 accurate predictions and 46 inaccurate ones for life skills. The classifier produced 325 right predictions and 76 false ones for its technical skills. This dissection provides an in-depth assessment of the classifier's performance in many skill areas.



Soft skills



Life skills



Technical skills

Figure 6: Confusion matrix for Soft, Life and Technical skills

5.4 Explainability of the proposed model using SHAP

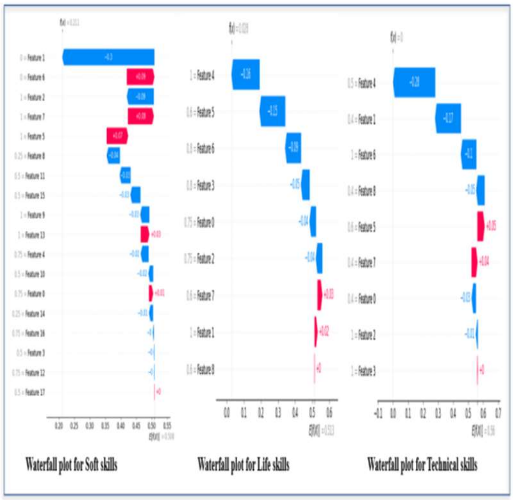


Figure 7: Waterfall plot for Soft, Life, and Technical skills

The waterfall plot [42] illustrates the contribution of each feature to the output of the model for the first sample in the test dataset. The plot will show how each feature's value contributes to the final prediction made by the model for that particular sample. In the waterfall plot generated by SHAP, the colours represent the direction and magnitude of the feature contributions to the model output. Blue bars represent positive contributions to the model output. This means that an increase in the feature value leads to an increase in the model's output or prediction. Red bars represent negative contributions to the model output. This means that an increase in the feature value leads to a decrease in the model's output or prediction. The longer the blue bar, the more positive impact the feature has on the prediction, whereas the longer the red bar, the more negative impact the feature has on the prediction.

The numeric values displayed on top of each bar indicate the magnitude of the contribution of each feature to the model output. These values represent the absolute SHAP values, which quantify the impact of each feature on the model prediction, regardless of the direction (positive or negative). By examining the lengths and colours of the bars in the waterfall plot, you can gain insights into how each feature affects the model's predictions and understand the relative importance of different features in the prediction process. Figure 7 depicts a waterfall plot showcasing the contributions of various features to Soft, Life, and Technical skills. In Life skills, features 4 and 5 exhibit positive contributions, while in Technical skills, features 4 and 1 significantly contribute to the model's output.

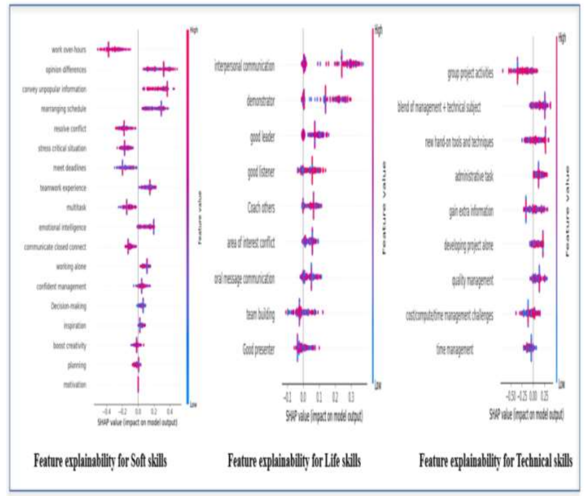


Figure 8: Summary plots for Soft, Life and Technical skills

The SHAP [43] generates a summary plot visualising the global feature importance across all instances in the test data. The colour of each bar indicates the direction of the feature's effect, with red indicating a positive impact and blue indicating a negative impact. This plot allows for quick identification of which features have the most significant influence on model predictions and their directionality. Additionally, the ordering of features along the y-axis provides insights into their relative importance compared to one another. Overall, the summary plot offers a comprehensive overview of the model's features, aiding in the interpretation and understanding of its predictive behaviour. The fusion of Convolutional Neural Network architecture with the interpretability offered by the SHAP algorithm enhances the decision-making process when predicting the skill levels of students within the field of software engineering. Figure 8 illustrates the summary plot for Soft, Life, and Technical skills.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

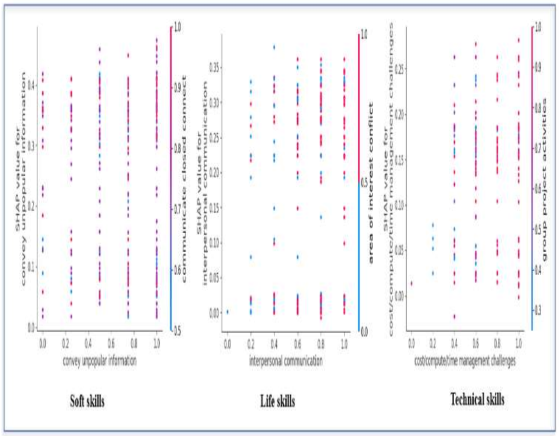


Figure 9: Dependence plots for Soft, Life and Technical skills

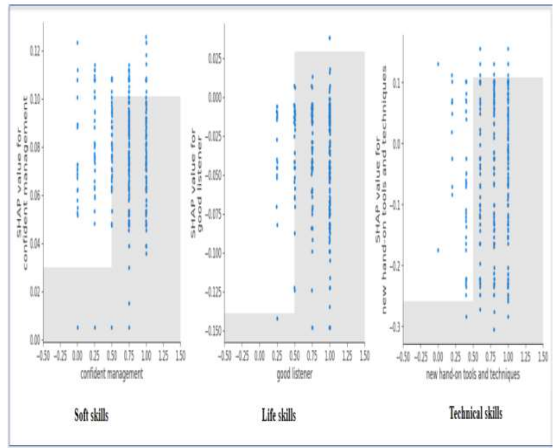


Figure 10: Scatter plots for Soft, Life and Technical skills

Figure 9 displays dependence plots for Soft, Life, and Technical skills. Dependence plots [42] are specifically designed to elucidate the influence of individual features on model predictions, providing valuable insights into the behavior of the model. To further investigate Soft skills, a dependency plot is created for the particular characteristic of "teamwork experience." This graphical depiction facilitates understanding of the relationship between this attribute and the model's predictions. By utilizing SHAP values for analysis, the dependency plot clarifies how differences in "teamwork experience" impact predictions in the soft skill domain. Transitioning to Life skills, the focus is placed on a distinguishing feature, "interpersonal communication." Within the life skill spectrum, the resulting graphic offers a dynamic depiction of the complex interaction between "interpersonal communication" and model predictions. This focused examination significantly enhances the interpretability of the model's behavior regarding the selected characteristic. In Technical skills, the characteristic of "administrative task" takes center stage. Plotting the data provides a comprehensive understanding of how differences in this variable affect the model's predictions within the technical skill domain. This methodological approach yields insightful information for well-informed analysis and decision assistance, fostering a thorough grasp of the model's decision-making process.

Scatter plots [43], as generated by SHAP offer a visual depiction of the relationship between feature values and their corresponding SHAP values within a specific feature. Each data point on the plot represents a sample from the dataset, with its position on the x-axis indicating the feature value and, on the y-axis, representing the SHAP value associated with that feature. By employing the colour blue, scatter plots can denote a specific subset or category within the data, aiding in interpretation. They provide valuable insights into how changes in feature values impact SHAP values, thereby illuminating the influence of each feature on the model's output.

Figure 10 illustrates scatter plots for Soft, Life, and Technical skills. In Soft skills, the scatter plot for confident management reveals a discernible trend, indicating that higher levels of confidence correspond to more effective management outcomes. Conversely, in Life skills, the scatter plots focus on "being a good listener." The clustered points within these plots suggest instances where active listening positively influences personal connections and enhances conflict resolution capabilities. Within the realm of Technical skills, the scatter plots pertain to "new hands-on tools and techniques," offering further insights into how proficiency in these areas impacts model predictions or outcomes. Analysing the scatter plot helps in understanding the importance and contribution of individual features to the model's predictions.

Journal of Theoretical and Applied Information Technology
15th November 2024. Vol.102. No. 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5.5 Explainability of the proposed model using LIME

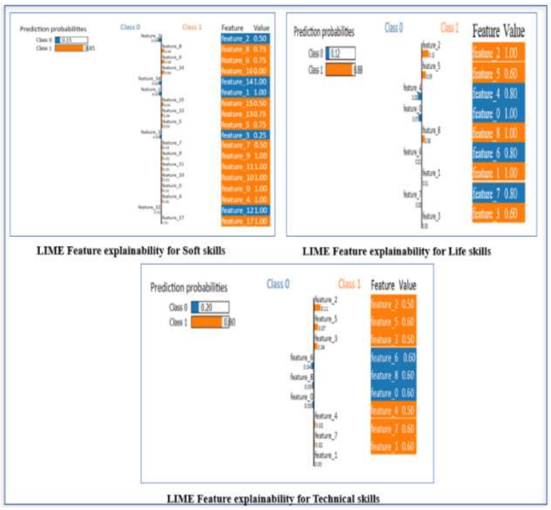


Figure 11: Visualization of LIME for Soft, Life and Technical skills

The Figure 11 visualization from LIME illustrates the key features and their impact on the prediction outcome for Soft, Life, and Technical skills. In a binary classification scenario, two probabilities are presented: one for Class 1 and one for Class 0. The features are colour-coded, with orange indicating support for Class 1 and blue for Class 0. Longer horizontal bars denote features with greater importance in the prediction process. The Soft skills explanation highlights the top eighteen features identified by the LIME model, where features 8, 6, and 16 are the top contributors to Class 1. Similarly, the Life skills explanation focuses on the top nine features, with features 2, 5, and 8 being the primary contributors to Class 1. In the Technical skills explanation, features 2, 5, and 3 emerge as the top contributors to Class 1. These explanations shed light on how these features are represented in the dataset and influence the prediction outcome. The aim is to offer a clear understanding of the LIME model's output, emphasizing the importance of each component in the prediction process [44].

5.6 Explainability of the proposed model using LRP

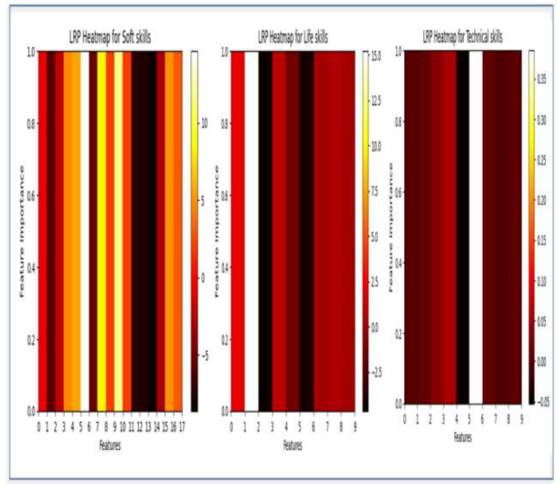


Figure 12: LRP Heatmaps for Soft, Life and Technical skills

LRP heatmaps [45] are visual representations that help determine how important or relevant input characteristics are to the model's predictions. Each cell in the heatmap corresponds to a feature, and the colour intensity of each cell represents the relevance score. A hotter colour indicates higher relevance, while cooler colours indicate lower relevance. Features or regions of the heatmap that are coloured red indicate that they have a strong positive influence on the model's prediction. These features are crucial for understanding why the model made a particular decision. Light red, yellow, or orange indicates moderate relevance. White represents neutral or near-zero relevance. Black represents zero relevance or absence of importance. The black regions of the heatmap suggest that the corresponding features have no relevance to the model's prediction. Thus, regions of the heatmap with hotter colours suggest features that have a stronger influence on the model's prediction for the first test sample. The X-axis of the heatmap represents the feature index. The Y-axis of the heatmap represents the feature's importance. This axis provides a qualitative indication of the importance of each feature in influencing the model's decision for the Soft, Life and Technical skills. The heatmap provides a visual representation of the relevance scores obtained from the LRP analysis, allowing you to identify important features and better understand how they contribute to the model's predictions. The different colour intensities help in quickly identifying which features are most relevant to the model's decision-making process.

Figure 12 visualises a heatmap to visualise the output of layer-wise relevance propagation (LRP) analysis for the Soft, Life and Technical skills. In Soft skills, features 1 and 6 demonstrate the highest relevance, while for Life skills, feature 4 emerges as the most significant. Regarding Technical skills, features 0, 1, 6, 7, and 8 are identified as the most influential factors influencing the model's decision-making process.

The integration of explainable AI methods such as SHAP, LIME, or LRP provides transparency by elucidating how the model makes predictions, fostering trust and understanding among educators and learners. By analysing feature importance, researchers can identify critical skills assessed by the model, aiding in curriculum design and personalised learning strategies. Furthermore, explainability enables error analysis, allowing educators to address model biases and improve assessment accuracy. This transparency supports tailored feedback and promotes ethical considerations by detecting and mitigating biases, ensuring fairness in skill assessments and educational outcomes. Ultimately, explainability enhances the interpretability of AI systems in educational contexts, facilitating more informed decision-making and personalised skill development approaches.

6. IMPACT OF SKILLSET PREDICTION

The impact of skillset prediction, particularly through the integration of CNN with XAI, is profound in guiding customised intervention strategies for students lacking in Soft, Life, or Technical skills. Upon identification of areas of weakness, it becomes imperative to develop tailored training plans and allocate resources specifically designed to address these deficiencies. These interventions should be accompanied by individualised skill development plans, meticulously crafted to cater to the unique needs of each student. Moreover, the implementation of ongoing monitoring and feedback mechanisms is crucial to ensuring the effectiveness of skill enhancement initiatives. Continuous tracking of students' progress allows for timely adjustments and interventions, facilitating a dynamic learning environment where improvements can be swiftly addressed and supported.

By adopting such a proactive approach, educational institutions can foster comprehensive skill development among computer science graduates. Not only does this strategy empower students to overcome their limitations, but it also equips them with the necessary competencies to thrive in a swiftly changing professional environment. Ultimately, the integration of skillset prediction and personalised intervention strategies contributes significantly to the holistic growth and success of students in their academic and professional pursuits.

The use of explainable AI techniques indeed strengthens the model by revealing the factors that influence predictions, improving interpretability, and allowing educators to understand the model's decision-making logic. However, some essential assessment needs remain unaddressed. Despite the model's promising results, further research is necessary to evaluate its applicability across diverse student demographics and curricula, which would help ensure its robustness and fairness. Key unresolved issues for future exploration include validating the effectiveness of explainability techniques (such as SHAP, LIME, and LRP) in real-world educational settings, mitigating potential biases, and refining the model to assess specific subskills within broader domains. Addressing these challenges could significantly enhance the value and reliability of this prediction model in educational contexts.

7. CONCLUSION

This study integrates Convolutional Neural Networks with explainable AI methodologies such as SHAP, LIME, and LRP to create a robust model for projecting students' skill sets across the Soft, Life, and Technical domains. By leveraging these advanced techniques, our model outperforms traditional methods like Artificial Neural Networks, LightGBM, AdaBoost, and XGBoost, showcasing significant potential for generalisation and accuracy in skill assessment. The incorporation of explainable AI techniques such as SHAP, LIME, and LRP not only enhances the predictive power of our model but also provides insights into the underlying factors driving skill assessments. This transparency facilitates targeted interventions and curriculum enhancements, empowering educational institutions to better prepare computer science students for the dynamic demands of the workforce. Moreover, our approach not only identifies skill-rich students more effectively but also enriches the teaching-learning process within educational institutions. By providing students with insights

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

into their own skill development, it fosters a sense of self-awareness and encourages them to become more independent learners. Additionally, educators benefit from a deeper understanding of their students' personalities and strengths, enabling more personalised instruction and support.

Though the model's predictive performance is encouraging, there is additional opportunity for enhancement. The model's adaptability to different datasets may be limited by its reliance on CNNs; future research might investigate hybrid models or other architectures to get around this. Furthermore, despite the interpretability offered by SHAP, LIME, and LRP, each approach has computational cost and scalability limits that may present difficulties in real-time applications. Future research addressing these elements may improve the model's applicability and increase its adaptability to a range of educational settings.

Looking ahead, future iterations of this research will explore the integration of deep learning-based feature extraction and prediction models through hybrid learning approaches. This ongoing evolution promises even greater accuracy and efficiency in estimating students' skill sets, further bridging the gap between educational institutions and the software sector. This study significantly contributes to the advancement of skill-based education and facilitates collaboration between academia and industry. By harnessing the power of Convolutional Neural Networks and explainable AI methodologies, we empower both students and educators to navigate the complexities of skill development in the digital age.

REFERENCES

[1] Cihan, P. and Kalipsiz, O., 2014. "Evaluation of students' skills in software project". TEM Journal, 3(1), p.42.

[2] González-Morales, D., De Antonio, L.M.M. and García, J.L.R., 2011, April. "Teaching Soft skills in software engineering". In 2011 ieee global engineering education conference (educon) (pp. 630-637). IEEE.

[3] Akdur, D., 2022. "Analysis of software engineering skills gap in the industry". ACM Transactions on Computing Education, 23(1), pp.1-28.

[4] Garousi, V., Giray, G., Tüzün, E., Catal, C. and Felderer, M., 2019. "Aligning software engineering education with industrial needs: A meta-analysis". Journal of Systems and Software, 156, pp.65-83.

[5] Garousi, V., Giray, G., Tuzun, E., Catal, C. and Felderer, M., 2019. "Closing the gap between software engineering education and industrial needs". IEEE software, 37(2), pp.68-77.

[6] Scaffidi, C., 2018. "Employers' needs for computer science, information technology and software engineering skills among new graduates". International Journal of Computer Science, Engineering and Information Technology, 8(1), pp.1-12.

[7] Gnatz, M., Kof, L., Prilmeier, F. and Seifert, T., 2003, March. "A practical approach of teaching software engineering. In Proceedings 16th Conference on Software Engineering Education and Training". (CSEE&T 2003). (pp. 120-128). IEEE.

[8] Flambeau, J.K.F. and Norbert, T., 2021. "Skills prediction based on multi-label resume classification using CNN with model predictions explanation. Neural Computing & Applications". 33(10), pp.5069-5087.

[9] Sventekova, E. and Lovecek, T., 2012, May. "Project-based teaching, practice in the academic environment". In Latest advances in educational technologies: proceedings of the 11th WSEAS international conference on education and educational technology (EDU 12): Singapore City, Singapore.

[10] Iriarte, C. and Bayona Orè, S., 2018. "Soft skills for it project success: A systematic literature review. In Trends and Applications in Software Engineering": Proceedings of the 6th International Conference on Software Process Improvement (CIMPS 2017) 6 (pp. 147-158). Springer International Publishing.

[11] Peslak, A., 2004. "Teaching software engineering through collaborative methods". Issues in Information Systems, 1(1), pp.247-253.

[12] Skulmoski, G.J. and Hartman, F.T., 2010. "Information systems project manager soft competencies: A project-phase investigation". Project Management Journal, 41(1), pp.61-80.

[13] Zaman, U., Jabbar, Z., Nawaz, S. and Abbas, M., 2019. "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, 37(3), pp.444-460.

[14] El-Sabaa, S., 2001. "The skills and career path of an effective project manager". International journal of project management, 19(1), pp.1-7.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[15] Jiechieu, K.F.F. and Tsopze, N., 2021. "Skills prediction based on multi-label resume classification using CNN with model predictions explanation". Neural Computing and Applications, 33, pp.5069-5087.
[16] Li, X., Zhang, Y., Cheng, H., Li, M. and Yin, B., 2022. "Student achievement prediction using deep neural network from multi-source campus data". Complex & Intelligent Systems, 8(6), pp.5143-5156.
[17] Lau, E.T., Sun, L. and Yang, Q., 2019. "Modelling, prediction and classification of student academic performance using artificial neural networks". SN Applied Sciences, 1, pp.1-10.
[18] Mubarak, A.A., Cao, H., Hezam, I.M. and Hao, F., 2022. "Modeling students' performance using graph convolutional networks". Complex & Intelligent Systems, 8(3), pp.2183-2201.
[19] Khalid, A. and Kumar, A., 2023, March. "Convolutional Neural Networks for Students Academic Performance Analysis and Alerting System". In 2023 2nd International Conference for Innovation in Technology (INOCON) (pp. 1-7). IEEE.
[20] Li, G., Liu, F., Wang, Y., Guo, Y., Xiao, L. and Zhu, L., 2021. "A convolutional neural network (CNN) based approach for the recognition and evaluation of classroom teaching behavior". Scientific Programming, 2021, pp.1-8.
[21] Aydoğdu, Ş., 2020. "Predicting student final performance using artificial neural networks in online learning environments". Education and Information Technologies, 25(3), pp.1913-1927.
[22] Liu, J., Shen, Y., Zhang, Y. and Krishnamoorthy, S., 2021, May. "Resume parsing based on multi-label classification using neural network models". In Proceedings of the 6th International Conference on Big Data and Computing (pp. 177-185).
[23] Zhao, M., Javed, F., Jacob, F. and McNair, M., 2015, January. "SKILL: A system for skill identification and normalization". In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 29, No. 2, pp. 4012-4017).
[24] Kalyani, B.S., Harisha, D., RamyaKrishna, V. and Manne, S., 2020. "Evaluation of student's performance using neural networks". In Intelligent Computing, Information and Control Systems: ICICCS 2019 (pp. 499-505). Springer International Publishing.
[25] Hussain, S., Muhsin, Z., Salal, Y., Theodorou, P., Kurtoğlu, F. and Hazarika, G., 2019. "Prediction model on student performance based on internal assessment using deep learning". International Journal of Emerging Technologies in Learning, 14(8).
[26] Neha, K., Kumar, R., Jahangeer Sidiq, S. and Zaman, M., 2023, February. "Deep Neural Networks Predicting Student Performance". In Proceedings of International Conference on Data Science and Applications: ICDSA 2022, Volume 2 (pp. 71-79). Singapore: Springer Nature Singapore.
[27] Alnasyan, B., Basheri, M. and Alassafi, M., 2024. "The Power of Deep Learning Techniques for Predicting Student Performance in Virtual Learning Environments: A Systematic Literature Review". *Computers and Education: Artificial Intelligence*, p.100231.
[28] Kartik, N., Mahalakshmi, R. and Venkatesh, K.A. (2023) "XAI-Based Student Performance Prediction: Peeling back the layers of LSTM and random Forest's black boxes". SN ComputerScience,4(5).https://doi.org/10.1007/s42979-023-02070-y.
[29] Ahmed, E. (2024) "Student performance prediction using machine learning algorithms," Applied computational intelligence and soft computing, 2024(1), pp. 1–15. Available at: https://doi.org/10.1155/2024/4067721.
[30] Nizar J, R. Sharmila, K.U. Jaseena (2024) "A Random Forest Model for Prediction of Software Engineering Skill Set among Computer Science Students through Explainable AI", International Journal of Intelligent Systems and Applications in Engineering, 12(21s), pp. 2633–2650.
[31] Zhang, X. and Yang, L., 2024. "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, 15(9).
[32] Chen, S. and He, H., 2018, November. "Stock prediction using convolutional neural network". In IOP Conference series: materials science and engineering (Vol. 435, p. 012026). IOP Publishing.
[33] Al Shalabi, L., Shaaban, Z. and Kasasbeh, B., 2006. "Data mining: A preprocessing engine". Journal of Computer Science, 2(9), pp.735-739.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[34] Patro, S.G.O.P.A.L. and Sahu, K.K., 2015. "Normalization: A preprocessing stage". arXiv preprint arXiv:1503.06462.
[35] Ni, L., Li, Y., Wang, X., Zhang, J., Yu, J. and Qi, C., 2019. "Forecasting of forex time series data based on deep learning". Procedia computer science, 147, pp.647-652.
[36] Holzinger, A., Saranti, A., Molnar, C., Biecek, P. and Samek, W., 2022. "Explainable AI methods-a brief overview". In International workshop on extending explainable AI beyond deep models and classifiers (pp. 13-38). Springer, Cham.
[37] Saranya, A. and Subhashini, R., 2023. "A systematic review of Explainable Artificial Intelligence models and applications: Recent developments and future trends". *Decision analytics journal*, *7*, p.100230.
[38] Kurani, A., Doshi, P., Vakharia, A. and Shah, M., 2023. "A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting. Annals of Data Science", 10(1), pp.183-208.
[39] Sagala, N.T. and Ibrahim, M.A., 2022, July. "A Comparative Study of Different Boosting Algorithms for Predicting Olympic Medal". In 2022 IEEE 8th International Conference on Computing, Engineering and Design (ICCED) (pp. 1-4). IEEE.
[40] Kumar, P., Batra, S. and Raman, B., 2021. "Deep neural network hyper-parameter tuning through twofold genetic approach". Soft Computing, 25, pp.8747-8771.
[41] Khan, M.S.I., Rahman, A., Debnath, T., Karim, M.R., Nasir, M.K., Band, S.S., Mosavi, A. and Dehzangi, I., 2022. "Accurate brain tumor detection using deep convolutional neural network". Computational and Structural Biotechnology Journal, 20, pp.4733-4745.
[42] shap., 2018. Census income classification with scikit-learn.
[43] DataCamp.,2023. "Explainable AI: Understanding and trusting machine learning models".
[44] Sastrawan, I.K., Bayupati, I.P.A. and Arsa, D.M.S., 2022. "Detection of fake news using deep learning CNN–RNN based methods". ICT Express, 8(3), pp.396-408.
[45] Javed, A.R., Khan, H.U., Alomari, M.K.B., Sarwar, M.U., Asim, M., Almadhor, A.S. and Khan, M.Z., 2023. "Toward explainable AI-empowered cognitive health assessment". Frontiers in Public Health, 11, p.1024195.