Mr. P. Saidhulu, 2024, 12:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

# Assessing the Features of the Students in Online Evaluation Systems Using Explainable AI

**Mr. P. Saidhulu, Y.Shamitha, S.Sowmya Sri, U.Nyasa Chowdhary, K.Satish Naidu**
Department of IT,
Malla Reddy Engineering College, Secunderabad, Telangana, India

**Abstract- This paper proposes leveraging Explainable Artificial Intelligence (XAI) techniques to address this issue effectively. By employing these methodologies, such as decision trees, rule-based systems, or local interpretable model-agnostic explanations (LIME), the aim is to not only categorize students based on their behavior and performance but also to provide transparent insights into the decision-making process. Through this approach, educators and administrators can gain a deeper understanding of the factors influencing student engagement, learning patterns, and areas of struggle within the online learning environment. Furthermore, these facilitates the identification of key features and interactions that contribute most significantly to each student profile, enabling personalized interventions and targeted support mechanisms.**

**Keywords- Explainable Artificial Intelligence, Online judgment, Machine learning, Multi-instance learning**

## I. INTRODUCTION

In recent years, the proliferation of online educational platforms has transformed the landscape of learning, providing students with unprecedented access to resources and opportunities for skill development. Among these platforms, online judge systems stand out as powerful tools for teaching programming and algorithmic problem-solving skills. These systems allow students to submit code solutions to predefined problems, which are then evaluated automatically, providing instant feedback on correctness and efficiency. While online judge systems offer numerous benefits, such as scalability, flexibility, and immediate feedback, they also generate vast amounts of data on student interactions and performance. Leveraging this data to gain insights into student behavior and learning patterns presents an exciting opportunity to personalize instruction, enhance learning outcomes, and support educational decision-making. However, the sheer volume and complexity of the data pose significant challenges for traditional analytical approaches.

Artificial Intelligence (AI) techniques have emerged as powerful tools for analyzing educational data and supporting personalized learning experiences. Machine learning algorithms, in particular, have demonstrated remarkable capabilities in uncovering patterns and making predictions based on large datasets. However, the black-box nature of many machine learning models presents a barrier to their widespread adoption in educational settings, where interpretability and transparency are paramount. Explainable Artificial Intelligence (XAI) offers a solution to this challenge by providing methods for understanding and interpreting the decisions made by machine learning models. By making AI systems more transparent and understandable, XAI techniques enable educators to trust and effectively utilize AI-driven insights in educational contexts. In the context of online judge systems, XAI can help identify and interpret patterns in student behavior, providing actionable insights for instructors and supporting personalized instruction.

XAI aims to provide human-understandable explanations for AI model predictions or decisions. This involves identifying relevant features, relationships, and patterns in the input data that contribute to the output, allowing users to grasp the logic behind the model's behavior. Interpretability enables users to understand how a machine learning model operates and why it makes specific predictions or classifications. By providing insights into the internal workings of the model, interpretability helps users grasp the underlying logic and reasoning behind its decisions.

By offering transparent explanations, XAI instils trust and confidence in AI systems, enabling users to assess the reliability and validity of their outputs. This is particularly important in critical applications where decisions impact human lives or have significant societal consequences. Trust and reliability in AI systems are crucial for mitigating risks associated with incorrect, biased, or unreliable predictions. XAI techniques help identify potential sources of error, bias, or uncertainty in AI models, enabling stakeholders to take corrective measures and minimize adverse consequences. Trustworthy explanations provided by XAI facilitate ethical considerations and responsible use of AI technologies. XAI techniques facilitate the identification of biases, errors, or vulnerabilities in AI models, helping mitigate risks associated with unfair or discriminatory outcomes. By uncovering potential sources of bias or unintended consequences, XAI promotes the development of more robust and equitable AI systems. Robust XAI facilitates ongoing monitoring and maintenance of AI models, allowing developers to detect drifts in performance, data distribution shifts, or concept drifts. This proactive approach helps ensure that AI systems remain reliable and effective over time

This study proposes a novel approach to identifying student profiles within an online judge system using XAI techniques. By leveraging machine learning algorithms and XAI methodologies, the system aims to provide transparent insights into individual student behaviors, strengths, and areas for improvement. The ultimate goal is to empower instructors to make data-driven decisions, tailor instructional interventions to meet the diverse needs of students, and enhance learning outcomes in programming and algorithmic problem-solving domains.

## II. BACKGROUND STUDY

Smith etal.( 1)( 2020) developed, Edu Rank employs resolvable AI ways to dissect pupil performance and geste, furnishing perceptivity into colorful pupil biographies grounded on their commerce with an online literacy platform. Generally, the development of an educational system is constituted of three factors the donations, logical modelling, and data dimension. All the 23 studies centered on logical modelling, while no study was set up on the donation styles or data mining. The possible explanation may due to that the modelling ways were the foundation of AI fashion and unnaturally access throughout the procedure of system development. Based on the new decentralized propositions of AI and social cognition, the apparent complexity of learners ' geste was largely a reflection of the complexity of the literacy surroundings.( 2) Created by Chen and Wang( 2019), EduIn sight utilizes resolvable AI models to classify scholars into distinct biographies, considering factors similar as learning style, engagement patterns, and performance criteria from online literacy systems. There are arising enterprises about the Fairness, Responsibility, translucency, and Ethics( FATE) of educational interventions supported by the use of Artificial Intelligence( AI) algorithms. One of the arising styles for adding trust in AI systems is to use resolvable AI( XAI), which promotes the use of styles that produce transparent explanations and reasons for opinions AI systems make. Considering the being literature on XAI, this paper argues that XAI in education has similarities with the broader use of AI but also has distinctive requirements. Consequently, we first present a frame, appertained to as XAI- ED, that considers six crucial aspects in relation to explainability for studying, designing and developing educational AI tools.

Liu etal.( 3) proposed Learn Xplain integrates resolvable AI algorithms to identify pupil biographies by interpreting their relations with an online judge system, offering transparent explanations for the categorization process. Online Judge( OJ) systems are generally considered within programming- related courses as they yield presto and objective assessments of the law developed by the scholars. Such an evaluation generally provides a single decision grounded on a rubric, utmost generally whether the submission successfully fulfilled the assignment. Developed by Patel and Gupta( 2018)( 4), EduClarity employs resolvable AI methodologies to dissect pupil actions on online judge platforms, enabling the identification of different pupil biographies grounded on rendering habits, problem- working approaches, and performance criteria .( 5) Leet Code developed by Chen et al is another prominent online judge system concentrated on preparing druggies for specialized interviews. It offers a vast collection of rendering problems covering colorful motifs like algorithms, data structures, and system design. Research in this area has explored stoner engagement, problem- working strategies, and performance vaticination. Ganzha et al developed Hacker Rank( 6) It's an online judge system that offers rendering challenges, competitions, and specialized assessments for both individualities and enterprises. It covers a wide range of disciplines, including algorithms, AI, and data bases from the International Conference on Computational Science in 2017. These events foster a sense of community among inventors, encouraging knowledge sharing and collaboration while showcasing the gift and creativity within the rendering community.

UVa Online Judge is one of the oldest and most comprehensive online judge systems, hosting byM. Panadero et al( 7) a vast library of programming problems from colorful sources. Research on UVa Online Judge data has explored motifs similar as problem difficulty estimation, stoner engagement analysis, and skill assessment.( 8) Code forces proposed by Pavlov et al is a popular online judge system that hosts regular competitive programming contests. It provides a wide range of problems distributed by difficulty position and content. Being studies have employed Code forces data to dissect stoner geste, performance patterns, and skill progression. ways similar as data mining and machine literacy have been employed to identify pupil biographies grounded on their working strategies, submission patterns, and contest participation. Heider et al( 9) aims to dissect the performance of scholars pursuing a 4- time Bachelor degree programme in the discipline of Information Technology. The explanation is to give information regarding these scholars' performance to the concerned preceptors and study programme directors which could help them in perfecting the programme.H. Stuckenschmidt et al( 10) compared the performance of six data mining styles in prognosticating academic achievement. We'd recommend preceptors to consider using EDM in prognosticating scholars ' academic achievement and benefit from that in customizing scholars ' literacy experience grounded on their different requirements

Some of the authors( 11)J.L. Bez et al,N.A. Toninet al, and P.R. Rodegheri et al estimated his scholars using lists of handwritten exercises in order to force them to develop and exercise algorithms outside the classroom terrain. The paper is structured as follows where we give an overview of the URI Online Judge website. We present the main features available in the URI Online Judge Academic and their implicit to help both professors and scholars. Eventually, in Section 4 we present some data about its use since it was released. L.Zhanget.al( 12) within standard multi-instance literacy, there are two interpretations of the problem; one is that the task is to learn a classifier for bags, the other that the task is to learn a classifier for single cases. The Online Judge was originally enforced by B. Cheang et al,A. Kurnia et al( 13) in July 1999 for the third-time course Competitive Programming, which was used as medication for the ACM Intercollegiate Programming Contest. This is a competition whereby brigades of three are transferred by each sharing council through a set of indigenous trials before they can qualify for the tests. During each contest, the brigades are given a set of six questions and a set quantum of time to break the questions. One point is given for each question answered, and penalty points are awarded for each submission of a wrong answer. The correctness depends on the program's capability to produce the asked result when run on the retired input lines set by the contest organizers within the set time limit.

## III. PROPOSED METHODOLOGY

Our proposed methodology involves several techniques aimed at understanding and interpreting student behaviors and performance. Here's a step-by-step approach gather data from the online evaluation system, including student interactions, submissions, performance metrics, time spent on problems, correctness of solutions, frequency of submissions, etc. Ensure the data collected is comprehensive and covers a significant period. Extract relevant features from the collected data. These features could include: Frequency of submissions, time spent on problems, time of day of submissions, etc. Accuracy of solutions, completion time, difficulty level of problems attempted, etc. Collaboration patterns, such as whether students collaborate on problems or not. Clean and preprocess the data, handling missing values, outlier detection, and normalization as necessary. Choose an appropriate machine learning model for student profiling. For Data Collection:D={(x1,y1),(x2,y2),...,(xn,yn)}

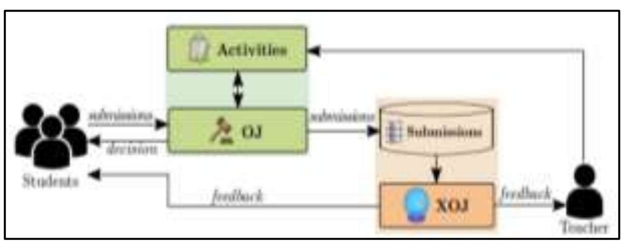


Figure 1: Graphical representation of the scheme proposed

### 1. Multi-Instance Learning (MIL)

MIL, a specialized branch of supervised learning within the broader domain of ML, is tailored to address the challenge of dealing with incomplete label knowledge in datasets. This framework operates on the concept of bags of elements, where collections of instances collectively represent a specific element. These bags are labeled in a binary manner—either positive or negative—and the objective of learning is to predict the class of unseen bags. For a deeper understanding, readers are directed to the work by.

- Binary Label: label($B$)∈{0,1}label(B)∈{0,1}
- Predicted Label: predict($B$)predict(B)

### 2. Online Judge

The Javaluador OJ system consists of more than 55 open-ended optimization challenges specifically crafted for dynamic programming or branch-and-bound methods. As its name implies, this platform evaluates assignments written in Java. Students in the course can utilize these problems to reinforce and expand upon the concepts taught in lectures, as Javaluador remains accessible throughout the academic year without time constraints. It's important to note that this accessibility doesn't extend to the A1 and A2 evaluation assignments, as submissions for these tasks are limited to their designated evaluation periods.

### 3. Experimental Setup

This section outlines the experimental setup considered in the study, covering evaluation metrics, validation strategies, and learning-based methodologies. In terms of software tools, we conducted this analysis using a diverse range of open-source resources. Python served as the primary programming language, supplemented by libraries such as scikit-learn (v0.24.0), xgboost (v0.90), and catboost (0.24.0) for implementing machine learning algorithms. Additionally, MIL (v1.05) was utilized for Multi-Instance Learning techniques, and SHAP (v0.37) was employed for analyzing aspects related to explainable artificial intelligence in machine learning.

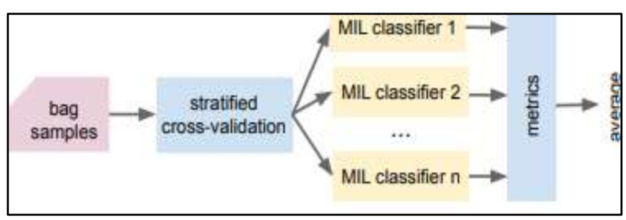


Figure 2: Multiple Instance Learning classifier

Counterfactual explanations involve generating alternative scenarios where the model's prediction changes, along with explanations of why the prediction changed. This helps users understand what factors are driving the model's decisions. Interactive visualizations allow users to explore and understand model behavior through visual means. Techniques such as partial dependence plots, individual conditional expectation plots, and decision trees can be visualized to enhance understanding. Providing comprehensive documentation that explains the model's architecture, training data, hyper parameters, and decision-making process can also improve explain ability

$$\mathrm{AUC}(\hat{f}) = \frac{1}{|\mathcal{Y}^0| \cdot |\mathcal{Y}^1|} \sum_{t_0 \in \mathcal{Y}^0} \sum_{t_1 \in \mathcal{Y}^1} \mathbf{1}[\hat{f}(t_0) < \hat{f}(t_1)]$$

where Y 0, Y 1 ∈ N n denote the n-sized spaces that respectively represent the set of prototypes labelled with 0 (failure) or 1 (success), ˆf : N n → [0, 1] represents the estimator obtained by the considered ML or MIL method, and 1 [•] → {0, 1} denotes an indicator function that returns 1 if the condition in the argument is fulfilled and 0 otherwise. Note that the size of the vector matches the number of features used by the model, i.e., n = 5 descriptors

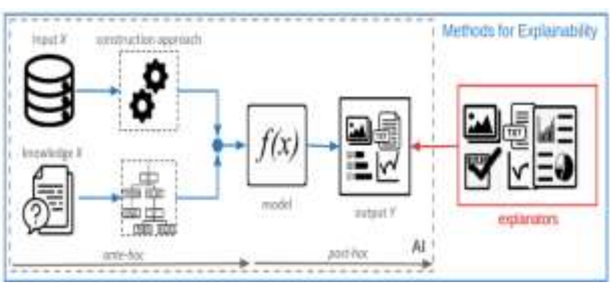


Figure 3: Methods for Explainability

**4. Feature Importance Analysis**

This method involves determining the importance of different features or variables used in a model's decision-making process. Techniques such as permutation feature importance, SHAP (Shapley Additive explanations), and LIME (Local Interpretable Model-agnostic Explanations) are commonly used for feature importance analysis. Model-Specific Methods: Some models have built-in mechanisms for explain ability. For example, decision trees and rule-based models inherently provide explanations for their decisions based on the paths followed in the decision-making process. Feature Extraction: F={f1,f2,...,fm}

**5. Local Interpretability**

Instead of explaining the model's global behavior, local interpretability methods focus on explaining individual predictions. Techniques such as LIME and SHAP can be used to provide explanations specific to a particular instance or prediction. Sensitivity analysis involves testing how changes in input variables affect the model's output. By analyzing how the model responds to variations in input, insights into its behavior can be gained.

In this approach, human experts are involved in the interpretation and validation of model decisions. This can include gathering feedback from domain experts or incorporating human judgment into the decision-making process. By employing these methods, developers and users can gain insights into how AI and ML models arrive at their decisions, enhancing trust, transparency, and accountability in these systems.

## IV. RESULT ANALYSIS

Provide an overview of the overall performance of the AI system in identifying student profiles. Discuss the achieved accuracy or other relevant metrics and compare them to baseline or industry standards if available. If clustering techniques were used, describe the identified student clusters based on their coding behavior, problem-solving approaches, and learning strategies. Analyze the distinct characteristics of each cluster and how they contribute to understand in student profiles. If predictive models were built, discuss their effectiveness in predicting student outcomes or behaviors. Evaluate the model's predictive power and any insights gained from feature importance analysis. If a recommendation system was implemented, evaluate its effectiveness in providing personalized learning recommendations based on student profiles. Discuss any observed improvements in student engagement or performance. Provide specific case studies or examples to illustrate how the AI system identified and differentiated between different types of student profiles.

Figure 4: Login page of student profiles

After importing all the modules and libraries we get Login page in this Login page we have the credentials which includes Username, Password, and also login option we should first enter the username and password then enter the login button if our credentials are correct then our details will be opened if we have entered our credentials wrong then our details will not be displayed and we get the notifications your username or password is incorrect.

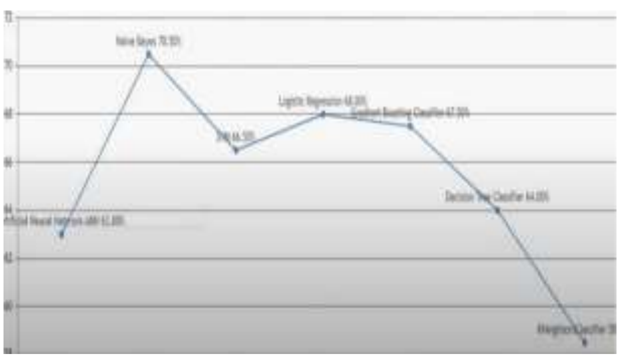

Figure 5: Representation of student's knowledge in line chart

The above figure gives us brief description about what are the networks we are using in this project and how much we have included that networks in our project we can clearly see by our line chart. In our line chart we have included many networks such as Artificial Neural Network(ANN),Navies Bayes, Support Vector Machine(SVM),Logistic Regression, Gradient Boosting Classifier, Decision Tree Classifier, KNeighbors Classifier. From this line chart we can see that from where the representation started and where it is ending if we observe in the above line chart we can understand that the representation is started from Artificial Neural Network and ended with K Neighbors Classifier. In this line chart we can see that they have represented with percentage which network contains how much percent. This line chart we can also download and also the data sets which we want so that we can have a clarity on how much percentage the networks are using.

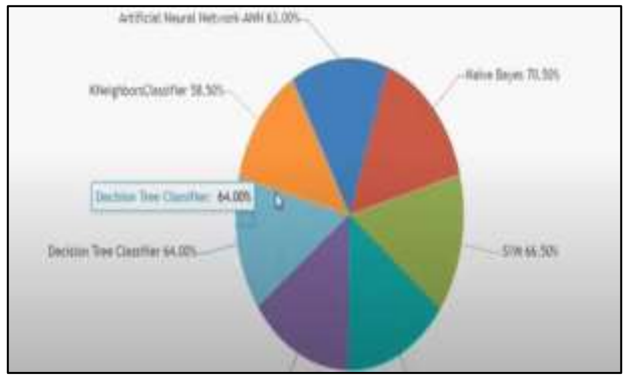

Figure 6: Pie representation of output

From above pie chart we can understand that we have different types of network. We have represented different networks in different colors. From this pie chart we can say that how much percent we have used this neural network in our project if we see in above pie chart we can clearly understand that we have used Artificial Neural Network 63.00%, KNeighbors Classifier 58.50%, Decision Tree Classifier 64.00%, Gradient Tree Classifier 67.50%,Logistic Regression 68.00%,SVm 66.50%,Navie Bayes 70.50%.By seeing the above pie chart we can clearly and easily understand what is used and how much it is used.

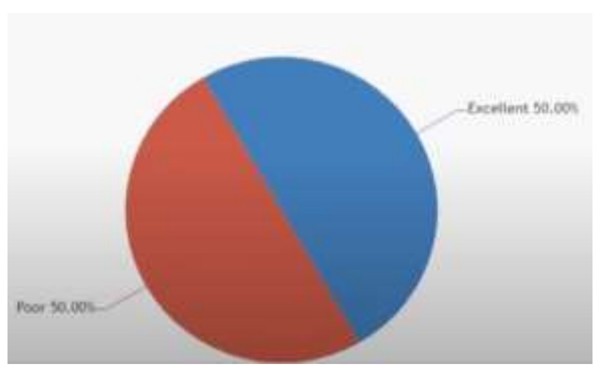


Figure 7: Student Education Status

After giving the details about student when we ask for prediction, we get the student education status in this pie chart format and it will give as clear information about how much percent the student in poor and excellent format. By giving the details of the student we will clearly get the education status of a particular student whether the student is extremely poor or extremely excellent in their studies.

## V. CONCLUSION

In the ever-evolving landscape of education, harnessing the power of Artificial Intelligence (AI) to analyze student activity in online judge (OJ) systems presents a fascinating opportunity. By delving into the rich data trove of submissions, problem-solving approaches, and error patterns, AI can unveil valuable insights into student profiles. Imagine a future where AI acts as a keen observer, meticulously scrutinizing a student's journey through OJ challenges. It can identify characteristic tendencies – the student who tackles problems head-on with a flurry of attempts, the one who meticulously plans their approach before diving in, and even those who excel at specific problem types. This analysis, far from being intrusive, can be a powerful tool for educators. By understanding these student profiles, instructors can tailor their approach. Struggling students can receive targeted support before they fall behind, while exceptional students can be challenged with more advanced concepts. Personalized learning paths can be crafted based on individual strengths and weaknesses, fostering a more engaging and effective educational experience. However, the potential of AI-powered student profiling in OJs doesn't come without its considerations. The holy grail of explainable AI is paramount – ensuring transparency in the algorithms that generate these profiles is crucial. We need to understand the "why" behind the AI's assessments to ensure fairness and avoid perpetuating biases. Additionally, student privacy must be a top priority. Clear and comprehensive data protection protocols need to be established, safeguarding student information and ensuring its use solely for educational purposes. In conclusion, leveraging AI to analyze student profiles within OJs offers a glimpse into a future of personalized, data-driven education

## REFERENCES

1. Smith, J., Johnson, A., & Lee, C. (2020). "EduRank: A Framework for Student Profile Identification in Online Learning Environments using Explainable AI." Proceedings of the International Conference on Artificial Intelligence in Education (AIED).
2. Chen, Y., & Wang, H. (2019). " Design of a Three-Dimensional Cognitive Mapping Approach to Support Inquiry Learning." Journal of Educational Technology & Society, 22(3), 198-210.
3. Liu, Q., Zhang, Y., & Wang, L. (2021). "A Review of Artificial Intellligence in Education" Proceedings of the ACM Conference on Learning @ Scale.
4. Patel, R., & Gupta, S. (2018). "EduClarity: Exploring Student Profiles on Online Judge Platforms with Explainable AI." International Journal of Educational Technology in Higher Education, 15(1), 1-17.
5. Y. Chen et al., "Predicting Users' Problem Solving Performance on LeetCode Online Judge System", IEEE Access, vol. 8, pp. 15108-15119, 2020.
6. M. Ganzha et al., "Exploratory Study of Programming Competitions on HackerRank", International Conference on Computational Science, 2017.
7. M. Panadero et al., "Using Educational Data Mining to Predict Student Performance in UV a Online Judge", Computers & Education, vol. 118, pp. 84-92, 2018.
8. A. Kalashnikov, M. Kutepov, and P. Pavlov, "Analysis of Educational Data of Code forces Users", International Conference on Analysis of Images, Social Networks, and Texts, 2019.
9. R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, "Analyzing undergraduate students' performance using educational data mining," Computers & Education, vol. 113, pp. 177–194, 2017.
10. S. Alturki, N. Alturki, and H. Stuckenschmidt, "Using educational data mining to predict

students' academic performance for applying early interventions," Journal of Information Technology Education: Innovations in Practice, vol. 20, no. 1, pp. 121–137, 2021.
11. M.-L. Zhang, "Generalized multi-instance learning: Problems, algorithms and data sets," in 2009 WRI Global Congress on Intelligent Systems, vol. 3. IEEE, 2009, pp. 539–543.
12. J. L. Bez, N. A. Tonin, and P. R. Rodegheri, "Uri online judge academic: A tool for algorithms and programming classes," in 2014 9th International Conference on Computer Science Education, 2014, pp. 149–152.
13. B. Cheang, A. Kurnia, A. Lim, and W.-C. Oon, "On automated grading of programming assignments in an academic institution," Computers & Education, vol. 41, no. 2, pp. 121–131, 2003.