

Machine Learning Tool to Predict Student Categories After Outlier Removal

Anindita Desarkar, Jadavpur University, India*

Ajanta Das, Amity University, Kolkata, India

Chitrita Chaudhuri, Jadavpur University, India

ABSTRACT

Statistical outlier detection techniques use academic performance-oriented results to find the truly brilliant as well as the weakest amongst a colony of students. Machine learning allows further partitions within the remaining student community, based on both merit and personality. The present work proposes a decision tree model for predicting three more appropriate categories. It utilizes text analytic tools to assess student characteristic traits from their textual responses and feedback. The cream of the general pool is chosen to belong to a top class comprising the mentor group, provided they can academically assist the weaker of the lot. But all on the top may not be suited for mentorship role. Textual assessment data delves to reveal character orientations favouring such decisions. The bulk who can manage their own forms the second class. The bottom of the pool benefits with assistance from the mentor group and comprise the third class.

KEYWORDS

Artificial Intelligence(AI), Decision Tree, Instructor Performance, Outlier, Quiz-Based Assessment, Statistical Inter-Quartile Range (IQR) Technique, Student Characteristic Prediction, Text Analytics

1. INTRODUCTION

The future of a society largely depends on the success of its young generation. To achieve this objective, it is essential to make foundations robust to the core – and excellent education and knowledge engineering might well be considered the key-factors in this context. But like many other human endeavours, judiciously choosing the correct procedure and standards is often a tedious, time-consuming and involved process, further hindered by personal opinions and favouritisms. Since AI has already set wings to the power of the machine by providing assistance in all tasks hitherto handled by humans alone, it seems appropriate to create a comprehensive Student Evaluation model, which, given some internal interaction, can take human-like decisions and automatically come up with intermediate suggestions and final categorization.

Such an innovative educational model, allowing constant interactive updates from students, achieves early detection of outlier performance. Removal of the outliers from such a system facilitates the students to be classified into various groups based on their capability. Adequate measures can

DOI: 10.4018/JITR.299380

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

thereafter be arranged for each group; exceptional performers remain a class apart, being the select few who can help the weaker section. Those needing this assistance form a class of their own. There exists an interim class consisting of students who can maintain their progress on their own.

Outlier detection is a technique to identify the presence of unusual patterns within a system, which do not conform to the general expected behavior (Singh & Upadhyaya, 2012). In educational domain, outlier performers refer to the group of students who perform below or above a statistically determined permissible range. Outperformers or positive outliers are those receiving marks above the upper limit. Their innovative responses help to enhance existing knowledge bases, opening up new research domains. On the other hand, a few performers exist who fall below the lowest standard – they are the poor performers or negative outliers who cannot be treated at par with any other groups within the student community. They usually need Special Care.

Test based examination is the most common and widely used technique to assess a student's knowledge level. However, conventional tests do not provide scope for assessing variable degree of understanding and confidence. Improved methodologies are obviously needed to measure these factors by preserving the detailed profile report of each student with graded questions at every quantifiable level of a specific subject. This research work proposes an assessment model to mitigate the above needs.

The general *motivation* behind the currently proposed model arises from the urge to automatically detect knowledge levels - crucial for perfecting a student's learning curve. This may require a detailed profiling report, which includes area-wise expertise in a subject, asking for level-specific adaptive assessment techniques at each stage. Literature surveys on existing learning models reveal that most of these lacks in extracting psychological factors such as levels of patience, confidence and perseverance of the participants. But these traits are essential for perfecting a learner's knowledge base. The proposed model enhances its academic assessment capability by capturing these other character revealing features as well. Thus the proposed system has an edge over existing traditional models in judging both technical and psychological acumen of a student.

The dataset preliminarily comprises of results from academic ability tests for a group of students. In addition, psychological characteristic related information is collected for the group utilizing Human Resource (HR) development techniques and Electronic survey models with textual response from each participant. Text analytic tools are employed to generate numeric scores from the textual responses which help to produce an augmented class value for each individual. The statistical IQR technique is applied on the overall class score to identify and remove outliers from the system. The remaining values are categorized into three types of student classes: those *capable of mentoring* others, those who can manage with *self-mentoring* and the ones who *needs mentoring*. Attributes associated with test result values being generally continuous in nature, need to have appropriate split-points to accommodate the Attribute-Selection method adopted by the Decision Tree Algorithm utilized for class determination. This algorithm closely resembles the Iterative Dichotomiser or ID3 proposed by the eminent Machine Learning exponent J. Ross Quinlan. Since Decision Trees can become large and difficult to interpret, IF-THEN rules are extracted from them to facilitate classification. These rules are then applied to assess the class of a new student. The present model produced satisfactory results on the provided test data set.

Section 2 highlights existing research works in related areas. The outline of the proposed framework is presented in Section 3. Section 4 and Section 5 discusses the experimental setup and decision tree formulation respectively. Section 6 follows with experimental results and inferences drawn accordingly. Section 7 talks about the application areas where the proposed model can be adopted. Some fore-plans of future works along with the conclusion are depicted in Section 8.

2. BACKGROUND

Mentioned below are a few recent researches published in the domain of AI based assessment. Samarakou et al. (Samarakou, Fylladitakis, Prentakis & Athineos, 2014) have proposed an AI based

automatic laboratory course assessment tool which is able to provide personalized feedback considering their individual strength and weakness of their laboratory assessment. Various principles of dynamic assessments are studied and analyzed thoroughly by Cotrus et al (Cotrus & Stanciu, 2014). It's an interactive process between the assessor and the assessed, where an "intelligent quotient" is computed instead of the traditional scoring approach. Mohammad (2015) has also enriched this domain by his research on this. Fani et al. have implemented Dynamic Assessment as a method of enhancing learners' reading comprehension ability (Fani & Rashtchi, 2015). A scaffolding system is presented by Ueno et al. which offers adaptive hints using a probabilistic model, Item Response Theory (IRT) (Ueno & Miyasawa, 2015). Feng et al. have introduced a better tutoring system compared to traditional one (Feng & Heffernan, 2010). The objective is to guess the required assistance level of a student to learn a topic.

Student Performance Prediction System (SPPS) is designed and implemented by Karthikeyan et al (Karthikeyan & Palaniappan, 2017), which is based on enhanced feature selection and ensemble classification algorithms using historical academic data. Another research is also performed by Rashid et al. on the same domain (Rashid & Aziz, 2016) where the objective is to find relationship between student's outcome of a course and their socio-economic backgrounds and earlier accomplishments.

Tair et al. (Tair & El-Halees, 2012) have applied various data mining techniques on graduate students' available records to discover rules based on association, classification, clustering and outlier detection. In another research, Ahmed et al. (Ahmed & Elaraby, 2014) have showed how classification technique might be used to predict the final grade of the students analyzing the students' repository. Out of several classification approaches, decision tree (ID3) method is utilised here for forecasting the same. AI based assessment is created and used by Luckin (2017) where it has provided continuous feedback to parents and teachers about the students' learning curve, the support they require and their progress towards achieving the learning goals. An automated system is designed by the researcher Altuhaifa (2016) to analyze students' emotion and activities, which is utilized to predict the nature of the group and decide teaching style accordingly. Various impacts of innovative assessment on the student learning are observed by McDowell (1995). Subramani et al. (Subramani & Iyappan, 2018) have worked to incorporate various technologies in the teaching front to create a rich learning experience for students as well as faculties. Chattopadhyay et al. (Chattopadhyay, Shankar, Gangadhar & Kasinathan, 2018) have showcased how various AI based solutions like Expert Control System (ECS)-based tutoring platform and Agent-based tutoring systems (AbS) which can be implemented in the process of Assessment for Learning (AfL). Murphy (2019) has discussed various ways and applications where AI has been utilized to support teachers and the practice of teaching. The applications include intelligent tutoring systems, automated essay scoring, and early warning systems. Also, the current researchers have worked on how machine-dependent assessment eradicates the chances of *human bias* (Desarkar, Das & Chaudhuri, 2018).

3. OUTLINE/ARCHITECTURAL DIAGRAM OF PROPOSED APPROACH

The proposed system deals with two types of data input: one, designated Quiz Based Assessment (QBA), consists of quiz marks for students, supplemented by dynamic attributes captured interactively such as response time, confidence level, and perseverance of individuals. The second, involving psychological assessment of the students based on their textual feedbacks to HR queries and E-surveys, is referred to as the Trait-Based Assessment (TBA) data. Figure 1 represents how the two data sources are merged to build the complete student database, and Decision Tree Induction is applied next on the available training data to generate the required classifier model. The performance of the model is evaluated by assessing accuracy of class prediction for the test data.

Figure 2 describes how the QBA attributes are collected. Marks obtained is countered with levies deducted for failing to answer correctly at first attempt, and number of questions left unattempted.

Figure 1. Outline of Proposed Approach

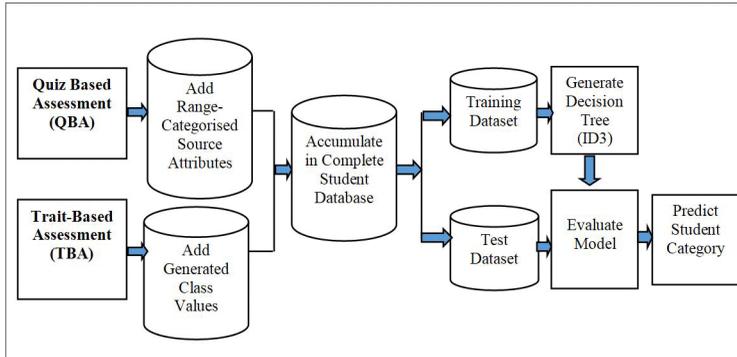
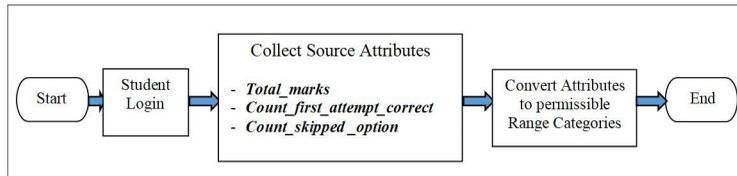


Figure 2. Generating Source Attributes from Quiz Based Assessment (QBA)



Range conversions are applied to the numeric attributes to bring them at par with standard requirement of the system.

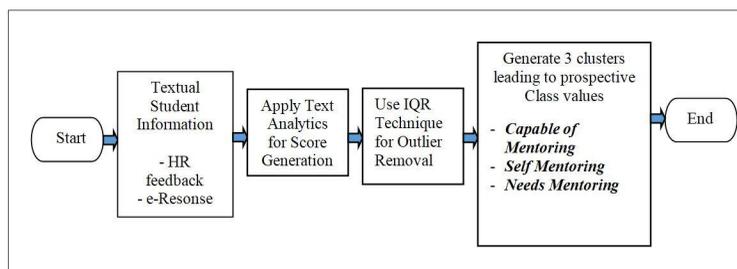
Figure 3 demonstrates how class values are identified from the TBA dataset. A total score for each individual is next produced, by using text analytic tools to generate numeric scores against the textual responses of the participants in this second dataset. The statistical IQR technique is applied on the overall score to identify and remove the outliers from the system. The remaining values are categorized into three classes: *capable of mentoring*, *self-mentoring* and *needs mentoring*.

4. EXPERIMENTAL SETUP

4.1 Dataset Collected Using Quiz-Based Assessment

Performance based assessment is an efficient way to measure knowledge level more effectively – and as such widely used in all domains across the globe. Evaluation can be achieved by testing code-

Figure 3. Populating class Values from Trait-Based Assessment (TBA)



development skills for diverse logical approaches, ability to predict appropriate output for existing code under the given circumstance, or proficiency in enhancing its performance. For the time being our approach caters to the appropriate output evaluation criteria only - students need to find out the correct output for the code provided. The model is platform independent; a suitable programming language under any environment can be chosen for setting the task. As already mentioned, the model is empowered to predict emotional and mental traits of the students.

4.1.1 Question Bank Preparation

A question bank is prepared containing a number of scenarios where each one is based on a specific programming concept. Table 1 describes the structure of a sample question bank.

4.1.2 Dataset Description

A prototype has been designed and developed to establish the approach by implementing the above question bank and providing the test to a select group of 40 new entrants in a session. As shown in Table 1, the assessment model contains 10 questions to be answered within 30 minutes. It is for checking knowledge-level dexterity in basic C Programming. Each question carries with it hints and final solutions, to be provided appropriately based on student interaction. Total marks is set to 100, 10 for each question apiece. Detailed level profiling report is generated for all 40 participating students, out of which 30 are preserved as training dataset. The remaining 10 are utilized as test data.

4.1.3 Rule Set for Marks Distribution

Following are the ruleset based on which marks are assigned to the student at every level. The exact distribution of marks is also reflected in the flowchart depicted in the following section:

IF-THEN Rule Set for Marks Distribution:

- IF (1st Attempt Completely Correct) THEN Provide Marks (10)
- IF (1st Attempt Completely Incorrect) THEN Provide No Marks (0)
- IF (1st Attempt Skipped) THEN Provide No Marks (0)
- IF ((1st Attempt 90% Correct) AND (Skipped 2nd Attempt)) THEN Provide Part Marks (2)

Table 1. Structure of Question Bank

Question No.	Question	Hint	Correct Answer
1	<pre>main() { int i,j, count = 0; for (i = 1; i <= 30 ; i++) { for (j = 1;j <= 40; j++) { count++; } } printf("Count=%d\n", count); }</pre>	Hints: Nested Loop !!! Outer loop executes 30 times & Inner loop executes 40 times !!!	Count=1200
-----	-----	-----	-----
10	<pre>void main() { int a = 5, b = 5, c = 10, result; result = (a == b)&& (c > b); printf("result= %d\n",result); }</pre>	Hints: Check the concept and rules of logical operators !!!	result=1

- IF ((1st Attempt 90% Correct) AND (2nd Attempt Poorer)) THEN Provide Part Marks (2)
- IF ((1st Attempt 90% Correct) AND (2nd Attempt Completely Correct)) THEN Provide Part Marks (8)
- IF ((1st Attempt 90% Correct) AND (2nd Attempt 90% Correct) AND (3rd Attempt Skipped)) THEN Provide Part Marks (2)
- IF ((1st Attempt 90% Correct) AND (2nd Attempt 90% Correct) AND (3rd Attempt Completely Correct)) THEN Provide Part Marks (6)
- IF ((1st Attempt 90% Correct) AND (2nd Attempt 90% Correct) AND (3rd Attempt Poorer)) THEN Provide Part Marks (2)

Note: Explanation of a few key terms

- **Completely Correct:** Answer is 100% correct
- **Skipped:** The student has moved to the next question without answering the existing one
- **Completely Incorrect:** Answer is completely different from correct answer
- **2nd Attempt Poorer:** The correctness of the second attempt is lower than the first attempt
- **1st Attempt 90% Correct:** The first attempt is very near to the right answer

4.1.4 Feature Extraction

Following are the feature details that are extracted from the first dataset:

1. **Total_marks:** Sum total of all marks obtained by a student in the academic quiz.
2. **Count_first_attempt_correct:** Number of questions fully correct at first attempt.
3. **Count_skipped_option:** Total number of times a student skips any question.

The first attribute contributes towards the measurement of academic performance of a student. The second and third attributes determine the level of patience and perseverance of a person, leading to determination of characteristic traits. Total time taken to complete the test is measured, but it only contributes in restricting the number of attempted questions, by setting a prior time limit.

The assessment provides a maximum of three attempts for each question, based on the correctness percentage C of the previous attempt. The flowchart in Figure 4 depicts how features are obtained using C . The “skip” option is available to move on to the next question without attempting the current one. Next question is shown either if the current one is answered completely correctly or incorrectly, or if it is skipped. But, nearly correct attempts involve second and third chances, with partial deduction of marks following rules discussed in Section 4.1.3.

Metrics captured from the results of the participating candidates for an assignment with 10 questions carrying 10 marks each, are reflected in the Sample Feature shown in Table 2, populated from the student database. Citing an example, here Student ID CSE201901 has received total 90 marks. Out of this, 80 ($8 * 10 = 80$) is scored for the 8 Count_first_attempt_correct answers. Remaining 10 marks is collected from 2 questions, one being skipped after first attempt (which earns him 2 marks being quite near to the correct answer), and the other answered correctly in the second attempt earning 8 marks, according to the ruleset presented in Section 4.1.3.

4.1.5 Analyze Dataset for Attribute Splitting

Conversion to categorical values for all continuous valued attributes are generally required for classification using Decision Tree algorithms. Since all attributes are integer numbers in the chosen scenario, these need to be converted into percentage values to form the best possible categories for each of them. The following split categories have been utilised in the current experiment – in each case the initial count is expressed in the percentage form with respect to the total tally of that attribute in the whole set:

Figure 4. Flowchart of Proposed Evaluation Approach

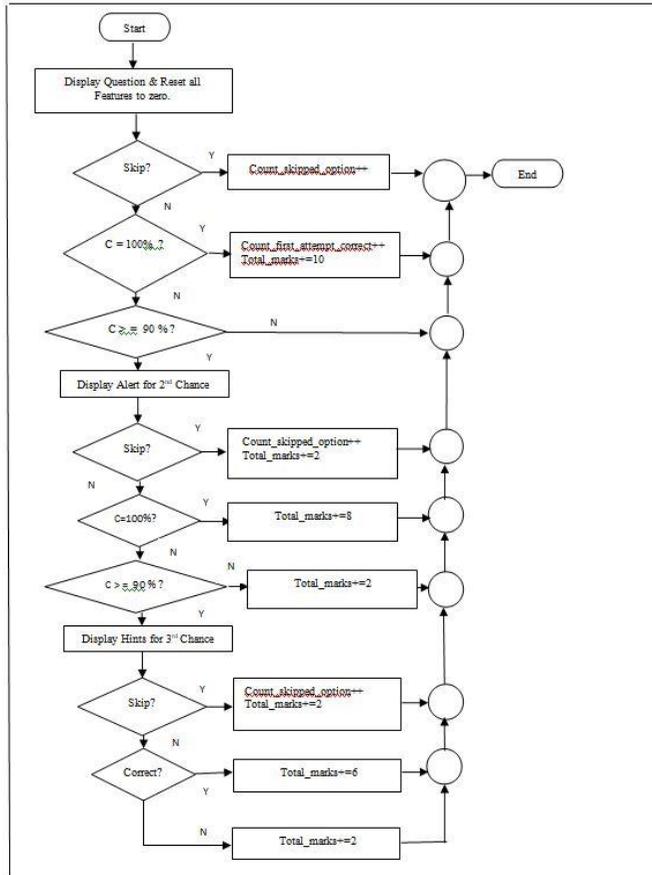


Table 2. Sample Features obtained from Quiz based Assessment

Student ID	Total_marks	Count_first_attempt_correct	Count_skipped_option
CSE201901	90	8	1
CSE201902	60	2	2
CSE201903	32	0	4
.....
CSE201939	44	1	2
CSE201940	58	4	1

- Attribute **Total_marks** Category splits:
 - Category 1 Description : $\geq 80\%$
 - Category 2 Description: $\geq 60\%$ and $< 80\%$
 - Category 3 Description: $\geq 40\%$ and $< 60\%$
 - Category 4 Description: $< 40\%$

- Attribute *Count_first_attempt_correct* Category splits:
 - Category 1 Description: > 70%
 - Category 2 Description: > 40% and <= 70%
 - Category 3 Description: < = 40%
- Attribute *Count_skipped_option* Category splits:
 - Category 1 Description: > 50%
 - Category 2 Description: >= 20% and <=50%
 - Category 3 Description: < 20%

4.1.6 Convert Assessment Result Set With Categorical Values

The dataset projected in Table 2 is converted in the Sample Feature shown in Table 3 based on the category splits discussed in the foregoing section 4.1.5. For example, in the very first record Total_marks 90 becomes >= 80% through *Category 1 Description*.

4.2 Determining Class Value From Trait-Based Assessment

In the previous section, dataset is organized for all the source attributes except the target or class variable. A value of the decision variable should be assigned against each row to form a complete data-set which will be used to build up the decision tree. Detailed feedback against each student is received from HR department as well as from an internally conducted e-Survey. These two are combined to generate the values of the decision variable as explained in the following subsections.

4.2.1 Collection Procedure for Trait-Based Dataset

The detailed characteristics of the students are mostly collected from HR ratings on various positive and negative qualities of the individuals, expressed through phrases such as ‘*Outstanding knowledge level and very passionate learner, Super Confident, Great Patience*’ or ‘*Average knowledge level, Diffident, Impatient*’. The objective is to measure the quantitative presence and levels of three main qualities in a student: knowledge in a specific subject, confidence and patience.

Table 4 presents an e-Survey report to disclose the participants’ willingness and capability to mentor others. For example, the knowledge level excellency of the first student (Student ID = CSE201901), is discerned from inputs like “Outstanding knowledge level and very passionate learner”. His confidence and patience levels are also ascribed high positive scores, induced by superlative comments like “Super Confident” and “Great Patience”. The last column records his mentoring enthusiasm. However, the next student (Student ID = CSE201902), although having very good knowledge level, fares poorly in confidence and patience levels, on top of his mentorship denial.

Table 3. Sample Feature Table with Categorical Values from Quiz based Assessment

Student ID	Total_marks	Count_first_attempt_correct	Count_skipped_option
CSE201901	>= 80%	> 70% and <= 100%	< 20%
CSE201902	>= 60% & < 80%	< = 40%	20 - 50%
CSE201903	< 40%	< = 40%	20 - 50%
.....
CSE201939	>= 40% & < 60%	< = 40%	20 - 50%
CSE201940	>= 40% & < 60%	< = 40%	< 20%

Table 4. Sample Features obtained from Trait-based Assessment

Student ID	Knowledge Level Comments from HR	Confidence Level Comments from HR	Patience Level Comments from HR	Mentorship Capability (Response from e-Survey)
CSE201901	Outstanding knowledge level and very passionate learner	Super Confident	Great Patience	Want to be Mentor, Loves Mentoring
CSE201902	Very Good knowledge level	Diffident	Impatient	Denied to be mentor
CSE201903	Poor knowledge level	Diffident	Impatient	Denied to be mentor

CSE201939	Average knowledge level	Diffident	Impatient	Denied to be mentor
CSE201940	Average knowledge level	Confident	Patient	Denied to be mentor

4.2.2 Text Analytics for Score Generation

Inputs received from the preceding Table 4 data are used to generate the class value as follows: a numerical score is obtained from the textual description of each participant – both from HR ratings, and their self-assessment e-survey reports. Text Analytic techniques parse the responses and assign quantitative values for a person (Alessia, Ferri, Grifoni & Guzzo, 2015). The sentiment analysis tool used here is AFINN, which uses a list of words rated for valence in the range between -5 to +5 (Aung & Myo, 2018; Naldi, 2019). It has an internal “Score” method returning the sum of word valence scores for a text string. For instance, AFINN.Score(“Outstanding knowledge level”) returns +5, AFINN.Score(“very passionate learner”) returns +2, while AFINN.Score(“Outstanding knowledge level and very passionate learner”) returns +7. *Total Score* for the samples in Table 4 are calculated in Table 5 by summing up the scores received from parameters like *Knowledge Level Comments*, *Confidence Level Comments* etc.

Table 5. Sample Student Character Dataset with Sentiment Score

Student ID	Knowledge Level Comment	Knowledge Level Cumulative Score	Confidence Comment	Confidence Score	Patience Comment	Patience Score	Mentorship Capability	Mentorship Capability Score	Total Score
CSE201901	Outstanding knowledge level and very passionate learner	7	Super Confident	5	Great Patience	3	Want to be Mentor, Loves Mentoring	4	19
CSE201902	Very Good knowledge level	3	Diffident	-2	Impatient	-2	Denied to be mentor	-2	-3
CSE201903	Poor knowledge level	-2	Diffident	-2	Impatient	-2	Denied to be mentor	-2	-8

CSE201939	Average knowledge level	0	Diffident	-2	Impatient	-2	Denied to be mentor	-2	-6
CSE201940	Average knowledge level	0	Confident	2	Patient	0	Denied to be mentor	-2	0

4.2.3 Outlier Removal Through IQR Technique

Anomalies or outliers need to be checked and extracted in all domains, and the present scenario is no exception (Singh & Upadhyaya, 2012). As in most cases, both positive and negative outliers can co-exist here. The positive bucket helps to identify people who are presumably most suitable for research related activities. The negative bucket, on the other hand, contains those who would benefit from special attention. The well-known Interquartile Range (IQR) technique is deployed here for detecting the anomalies (Krishnaiah, Narsimha & Chandra, 2014). In any data series, the IQR is expressed as $(Q3 - Q1)$, where $Q1$ and $Q3$ are the 1st and 3rd quartile values representing 25th and 75th percentile respectively. Two specific examples from the chosen sample are discussed below.

The student bearing ID CSE201922, with a total score of -18, is identified as negative outlier since his score falls below the lowest permissible range evaluated to -13.25 according to the statistical expression $(Q1 - 1.5 * IQR)$. Another student bearing ID CSE201901, is selected as a positive outlier because his total score evaluates to 19 and the upper statistical limit below positive outliers is set to 12.75 on the basis of the expression $(Q3 + 1.5 * IQR)$. The final training dataset is prepared after discarding these outliers from the system, and three class values are created based on the total score obtained within these.

Class Value Description:

- **Needs Mentoring:** Total Score below 0 [but above the lower outlier limit -13.25]
- **Self- Mentoring:** Total Score ≥ 0 and < 5
- **Capable of Mentoring:** Total Score ≥ 5 [but below the upper outlier limit 12.75]

4.2.4 Assigning Class Values

The three identified class values are assigned against the student dataset based on the total score obtained. Table 6 presents a glimpse of the dataset.

The class values are ascertained and attached to the available dataset and may be presented in the form depicted in Table 6.

4.3 Combine to Form Student Complete Dataset

The combined student dataset is formed by adding class values to the original source attributes. A few such samples are presented in Table 7.

5. DECISION TREE FORMULATION USING TRAINING DATASET

Decision trees (DTs) have several advantages compared to other classification techniques. Being logically comprehensible, interpretability is more. They can handle both categorical and quantitative

Table 6. Sample Dataset with Total Sentiment Score and corresponding Class value

Student ID	Total Sentiment Score	Expected Class
CSE201901	19	Positive Outlier
CSE201902	-3	Needs Mentoring
CSE201903	-8	Needs Mentoring
.....
CSE201939	-6	Needs Mentoring
CSE201940	0	Self Mentoring

Table 7. Combined Sample Dataset

Student ID	Total _marks	Count _first_attempt_ correct	Count_skipped _option	Decision/Class
CSE201901	>= 80%	> 70% and <= 100%	< 20%	Positive outlier
CSE201902	>= 60% & < 80%	< = 40%	20 - 50%	Needs Mentoring
CSE201903	< 40%	< = 40%	20 - 50%	Needs Mentoring
.....
CSE201939	>= 40% & < 60%	< = 40%	20 - 50%	Needs Mentoring
CSE201940	>= 40% & < 60%	< = 40%	< 20%	Self Mentoring

values. They can classify and solve regression problems too. Automatic variable screening is performed implicitly - a major advantage. Moreover, DTs need less effort for data preparation.

While fitting a DT on a training data set, it is most important to select the splitting attribute at each non-leaf node. Several techniques exist for this: the one used here calculates *Entropy* and *Information Gain* for all attributes available at a node – as implemented in the ID3 algorithm (Hall & Lande, 1998; Pach & Abonyi, 2008; Sachdeva, Hanmandlu & Kumar, 2012; Wang & Lee, 2006). The mathematical effort is based on Information theory and being statistics driven is simple. The technique avoids being overtly sensitive to outliers, as splits occur according to sample proportions rather than absolute values. Furthermore, DTs can easily be mined to extract rules, and the major objective here is to utilize these rules for final predictions (Das & Desarkar, 2018).

5.1 Overview of ID3 Algorithm

1. Entropy of every attribute should be calculated using the data set.
2. The set should be splitted into subsets by using the attribute for which the resulting entropy (after splitting) is minimum or information gain is maximum.
3. A decision tree node should be built containing that attribute.
4. Recurse on subsets using the remaining attributes.

5.2 Information Gain Calculations

Let $Info(D)$: expected information needed to classify tuples in training dataset D and let p_i : probability that a tuple in D belongs to class C_i (i : 1 to m):

A : splitting attribute with n distinct values $\{a_1, a_2, \dots, a_n\}$

D_j : subsets of D for corresponding splits $\{D_1, D_2, \dots, D_n\}$

$Info_A(D)$: extra information needed to partition D based on A

Then,

$$1) p_i = |C_i, D| / |D|$$

$$2) Info(D) = - \sum p_i \log_2(p_i), \text{ for } i = 1 \text{ to } m$$

$$3) Info_A(D) = \sum (|D_j|/|D|) \times Info(D_j), \text{ for } j = 1 \text{ to } n$$

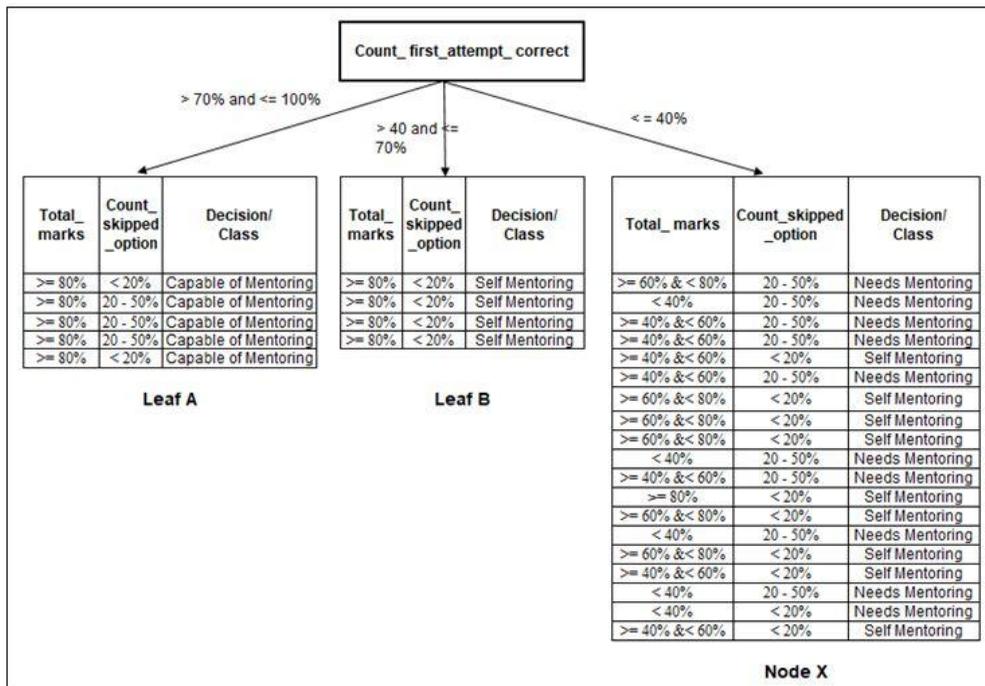
$$4) Gain(A) = Info(D) - Info_A(D)$$

Note:- A log function to the base 2 is used as the information is encoded in bits.

5.3 Application Development Tools

The adaptive assessment model and classification based technique have been developed in Core java. AFINN is used as text analytics tool for converting textual responses to numerical scores.

Figure 5. First level of Decision Tree



6. EXPERIMENTAL RESULTS AND PERFORMANCE ANALYSIS

This section contains various snapshots of the result-set while performing the experiment. It includes the decision tree built on the training dataset, the corresponding ruleset and the final prediction accuracy while applying the rules on the test dataset.

6.1 Decision Tree Using ID3

By applying ID3 algorithm, and information gain calculations as discussed in the preceding section for each attribute, the root node is selected by choosing the attribute with largest information gain. The various categories of that attribute are selected as different branches of the node. Here, Figure 5 shows the first level of the decision tree where **Count_first_attempt_correct** is chosen as root node having the highest information gain.

The first and second nodes appearing in the left consists of pure class values (*Capable of Mentoring* and *Self Mentoring* respectively). Hence, they form pure leaf nodes. The right most node, on the other hand, remains impure as it contains a heterogeneous mixture of class values. So, this node requires further branching. Hence, information gain is again calculated for the rest of the attributes to decide the next available branching attribute. The process continues until either all branches end in a pure leaf, or all attributes are utilized or there are no more training tuple to be considered. Figure 6 provides the next level of the decision tree where 'Count_skipped_option' and 'Total_marks' are identified as the branching criteria respectively.

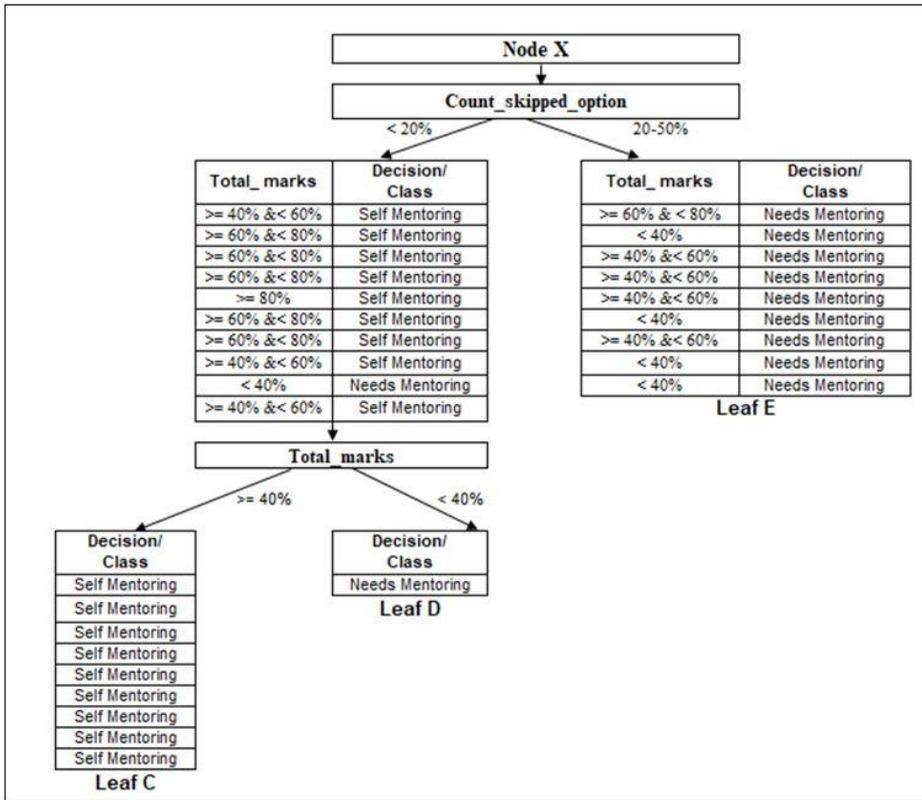
6.2 Knowledge Mining Through Rule-Sets

Following are the rule-set details derived from the above decision tree:

Rule 1 [Leaf A]: **IF (Count_first_attempt_correct >70%) THEN Capable of Mentoring**

Rule 2 [Leaf B]: **IF (Count_first_attempt_correct >40% and <= 70%)**

Figure 6. Second level of Decision Tree



THEN Self Mentoring

Rule 3 [Leaf C]: **IF (Count_first_attempt_correct <= 40% and Count_skipped_option < 20% and Total_marks >= 40%) THEN Self Mentoring**

Rule 4 [Leaf D]: **IF (Count_first_attempt_correct <= 40% and Count_skipped_option < 20% and Total_marks < 40%) THEN Needs Mentoring**

Rule 5 [Leaf E]: **IF (Count_first_attempt_correct <= 40% and Count_skipped_option >= 20%) THEN Needs Mentoring**

6.3 Accuracy Prediction and Error Analysis

Table 8 presents the test data set, complete with its original class values, as well as the calculated predicted class values obtained from the above rule-set. Accuracy is found to be 90% for the dataset used - as nine records out of ten are predicted accurately. The single instance of mismatch in the whole dataset is highlighted in the table below.

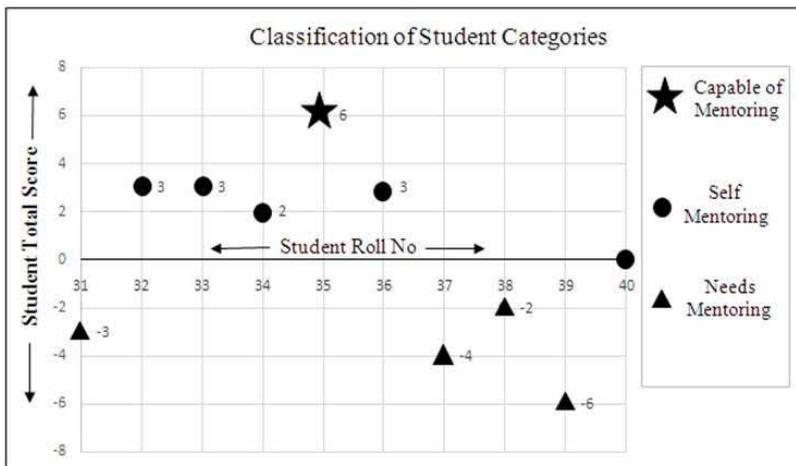
6.4 Inference

The distribution of the three different classes for the test dataset is depicted in Figure 7, with the individual *Total Scores* mentioned for each student beside the datapoints. These points are plotted on a vertical scale of *Student Total Score* ranging between -8 and +8, with the *Student Roll Nos* spread over the horizontal axis in the middle.

Table 8. Combined Student Test Dataset

Student ID	Total_marks	Count_first_attempt_correct	Count_skipped_option	Original Class	Predicted Class
CSE201931	>= 60% & < 80%	< = 40%	20 - 50%	Needs Mentoring	Needs Mentoring
CSE201932	>= 40% & < 60%	< = 40%	< 20%	Self Mentoring	Self Mentoring
CSE201933	>= 60% & < 80%	> 40 and <= 70%	< 20%	Self Mentoring	Self Mentoring
CSE201934	>= 60% & < 80%	< = 40%	< 20%	Self Mentoring	Self Mentoring
CSE201935	>= 60% & < 80%	< = 40%	< 20%	Capable of Mentoring	Self Mentoring
CSE201936	>= 60% & < 80%	> 40 and <= 70%	< 20%	Self Mentoring	Self Mentoring
CSE201937	< 40%	< = 40%	20 - 50%	Needs Mentoring	Needs Mentoring
CSE201938	>= 40% & < 60%	< = 40%	20 - 50%	Needs Mentoring	Needs Mentoring
CSE201939	>= 40% & < 60%	< = 40%	20 - 50%	Needs Mentoring	Needs Mentoring
CSE201940	>= 40% & < 60%	< = 40%	< 20%	Self Mentoring	Self Mentoring

Figure 7. Classification of Student Categories



A student case base can be constructed based on the summarized Table 8. Considering a *threshold* percentage for each course to evaluate it within a dynamic scenario, the following ratings can also help to predict the quality of the course offered and the performance of the instructor.

Assuming x to be the number of students ‘Capable of Mentoring’, y to be the number of students selected for ‘Self Mentoring’, and z to be the group that ‘Needs Mentoring’, the following three cases can be considered:

$$\text{Case 1: } \left(\frac{x}{x + y + z} \right) \geq \text{Threshold}$$

Indicates an ideal situation where majority group belongs to class ‘Capable of Mentoring’:

$$\text{Case 2: } \left(\frac{y}{x + y + z} \right) \geq \text{Threshold}$$

Represents a rather general level with most participants being capable of ‘Self Mentoring’:

$$\text{Case 3: } \left(\frac{z}{x + y + z} \right) \geq \text{Threshold}$$

Suggests course revamp, since most students fall into the ‘Needs Mentoring’ group.

In the present instance, x evaluates to 10%, y to 50% and z to 40% of the total population, as is evident from our result tables and graphs. By choosing an appropriate threshold value, the present course can be easily categorized according to the above case representations.

Another factor to be apprehended is that a healthy $x:z$ ratio needs to be maintained to strike a balance between mentors and weak students. Several analyses can be further performed on this output, which can facilitate the decision making process. Following are a few such perspectives which can be explored easily:

- Analysis of instructor performance.
- Identification of outstanding courses.
- Identification of Outlier performers.
- Creation of summarized student profiles for the organization.
- Arranging psychological counseling for students lacking in patience and perseverance.

7. APPLICATION DOMAIN

Some areas where the proposed assessment technique can be implemented to enhance student overall performance, as suggested by the works of Taneja et al. (Taneja, Safapour & Kermanshachi, 2018), are discussed below:

- **Pre-Final Assessment:** Conduct mock tests to assess the negative outliers (poor performers) in advance.
- **Progress Assessment:** Can be used to identify the knowledge gap of the students after completing a specific topic. In case a large number of students are found to commit numerous errors, the system can be made to generate appropriate alerts for the instructor indicating probable knowledge gaps existing among the students.
- **Learners’ Self Assessment:** Automatic feedbacks generated by the system are of immense importance even in the absence of the Instructor.
- **Analyzing Instructor Performance:** Appropriate measures can be implemented by analyzing the instructor performance from the various ratios calculated above. Effective measures include change in teaching style, incorporation of modern equipment while teaching (Kagama & Irungu, 2018).

- **Identification of Outstanding Courses:** Courses, belonging to the Case 1 category (as discussed above in the Inference section), can be universally recognized and accepted as Model Practices elsewhere in the Academic World.
- **Career Selection through Student Character Analysis:** Classification of students on the basis of their character traits can be utilized by the placement section to provide suitable job-profiles at the end of a course, if required.
- **Arrange Psychological Counselling:** Psychological Counselling can be arranged in a few critical situations as and when predicted by the system.

8. CONCLUSION AND FUTURE SCOPE

The current model utilizes various AI driven approaches, such as decision tree induction and character-trait classification using text analytic tools to build an innovative and interactive assessment system in the education domain. The terminal objective is not only to judge students' capability, but also to provide continuous support in attaining their goals. Based on the experimental outcome, it can be claimed that the system succeeds in identifying five types of performers amongst the students - the truly extraordinary ones marked as *positive outliers*, the *capable-of-mentoring* class, the *self-mentoring* class, the *needs-mentoring* class, and the *negative outliers* comprising the ones needing special attention.

Some of the possible outcomes of the model that can be explored further include judging the best instructor and identifying extraordinary courses, as hinted in the Inference section. A future scope in this domain may also involve the implementation of the proposed technique in a large scale with provisions for self-assessment tools to remote students too. The measure of performance improvement of the students involved would be the best metric for the evaluation of the proposed technique. Claims of the tool being a purely generic one is supported by the fact that, with minor modifications, the system can obviously accommodate topics other than the one explored here.

There remain a number of unexplored grounds within the research space – such as conducting the experiment with large datasets using scalable and incremental versions of decision tree induction with improved tree pruning mechanisms and comparing the results with other machine learning techniques. These could not be accommodated in the present work due to environmental constraints, but must remain as lucrative essentials in furthering this endeavor.

FUNDING AGENCY

The publisher has waived the Open Access Processing fee for this article.

REFERENCES

- Ahmed, A. B. E. D., & Elaraby, I. S. (2014). Data Mining: A prediction for Student's Performance Using Classification Method. *World Journal of Computer Application and Technology*, 2(2), 43–47. doi:10.13189/wjcat.2014.020203
- Alessia, D., Ferri, F., Grifoni, P., & Guzzo, T. (2015). Approaches, tools and applications for sentiment analysis implementation. *International Journal of Computers and Applications*, 125(3).
- Altuhaifa, F. (2016). Students' Weakness Detective in Traditional Class. *International Journal of Advanced Research in Artificial Intelligence*, 5(6), 16–20. doi:10.14569/IJARAI.2016.050604
- Aung, K. Z., & Myo, N. N (2018). Lexicon Based Sentiment Analysis of Open-Ended Students' Feedback. *International Journal of Engineering and Advanced Technology*, 8(2), 46-51.
- Chattopadhyay, S., Shankar, S., Gangadhar, R. B., & Kasinathan, K. (2018). Applications of Artificial Intelligence in Assessment for Learning in Schools. In *Handbook of Research on Digital Content, Mobile Learning, and Technology Integration Models in Teacher Education* (pp. 185-206). IGI Global. doi:10.4018/978-1-5225-2953-8.ch010
- Cotrus, A., & Stanciu, C. (2014). A study on dynamic assessment techniques, as a method of obtaining a high level of learning potential, untapped by conventional assessment. *Procedia: Social and Behavioral Sciences*, 116, 2616–2619. doi:10.1016/j.sbspro.2014.01.622
- Desarkar, A., Das, A., & Chaudhuri, C. (2019). *Automated Evaluation of Outlier Performers in Programming Sessions Using Case-Based Reasoning Techniques*. In *Computational Intelligence in Data Mining*. Springer.
- Fani, T., & Rashtchi, M. (2015). Dynamic assessment of reading comprehension ability: Group or individualized. *Education Journal*, 4(6), 325–331. doi:10.11648/j.edu.20150406.11
- Feng, M., & Heffernan, N. (2010, June). Can we get better assessment from a tutoring system compared to traditional paper testing? can we have our cake (better assessment) and eat it too (student learning during the test)? In *International Conference on Intelligent Tutoring Systems* (pp. 309-311). Springer. doi:10.1007/978-3-642-13437-1_54
- Hall, L. O., & Lande, P. (1998). Generation of Fuzzy Rules from Decision Trees. *Journal of Advanced Computational Intelligence*, 2(4), 128–133. doi:10.20965/jaciii.1998.p0128
- Karthikeyan, K., & Kavipriya, P. (2017). On Improving Student Performance Prediction in Education Systems using Enhanced Data Mining Techniques. *International Journal of Advanced Research in Computer Science and Software Engineering*, 7(5), 935–941. doi:10.23956/ijarcsse/SV7I5/0348
- Krishnaiah, V., Narsimha, G., & Chandra, S. (2014). Survey of Classification Techniques in Data Mining. *International Journal on Computer Science and Engineering*, 1, 18–20.
- Luckin, R. (2017). Towards artificial intelligence-based assessment systems. *Nature Human Behaviour*, 1, 28. doi:10.1038/s41562-016-0028
- McDowell, L. (1995). The impact of innovative assessment on student learning. *Innovations in Education & Training International*, 32(4), 302–313. doi:10.1080/1355800950320402
- Mohammad, S. K. (2015). *Dynamic Assessment: From Theory to Practice*. LAP Lambert Academic Publishing.
- Murphy, R. F. (2019). *Artificial Intelligence Applications to Support K–12 Teachers and Teaching: A Review of Promising Applications, Challenges, and Risks*. RAND Corporation.
- Naldi, M. (2019). *A review of sentiment computation methods with R packages*. arXiv preprint arXiv:1901.08319.
- Pach, F. P., & Abonyi, J. (2008). Association Rule and Decision Tree based Methods for Fuzzy Rule Base Generation. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 13, 45–50.
- Rashid, T. A., & Aziz, N. K. (2016). Student Academic Performance Using Artificial Intelligence. *Zanco Journal of Pure and Applied Sciences*, 28(2), 56–69.

- Sachdeva, K., Hanmandlu, M., & Kumar, A. (2012, March). Real Life Applications of Fuzzy Decision Tree. *International Journal of Computers and Applications*, 42(10), 24–28. doi:10.5120/5729-7800
- Samarakou, M., Fylladitakis, E. D., Prentakis, P., & Athineos, S. (2014). *Implementation of Artificial Intelligence Assessment in Engineering Laboratory Education*. International Association for Development of the Information Society.
- Singh, K., & Upadhyaya, S. (2012). Outlier Detection: Applications and Techniques. *International Journal of Computer Science Issues*, 9(1), 307.
- Subramani, P. C. (2018). Innovative methods of Teaching and Learning. *Journal of Applied and Advanced Research*, 3, 20. doi:10.21839/jaar.2018.v3iS1.161
- Tair, M. M. A., & El-Halees, A. M. (2012). Mining educational data to improve students' performance: A case study. *International Journal of Information*, 2(2), 140–146.
- Taneja, M. P., Safapour, M. E., & Kermanshachi, S. (2018). Innovative Higher Education Teaching and Learning Techniques: Implementation Trends and Assessment Approaches. *2018 American Society for Engineering Education (ASEE) Annual Conference & Exposition*, 24-27 doi:10.18260/1-2--30669
- Ueno, M., & Miyasawa, Y. (2015, June). *Probability based scaffolding system with fading*. In *International Conference on Artificial Intelligence in Education* (pp. 492-503). Springer.
- Wang, T. C., & Lee, H. (2006). Constructing a Fuzzy Decision Tree by Integrating Fuzzy Sets and Entropy. *WSEAS Transactions on Information Science and Applications*.
- Kagama, J., & Irungu, C. (2018). An Analysis of Teacher Performance Appraisals and their Influence on Teacher Performance in Secondary Schools in Kenya. *International Journal of Education*, 11(1), 93–98. doi:10.17509/ije.v11i1.11148
- Das, A., & Desarkar, A. (2018). Decision tree-based analytics for reducing air pollution. *Journal of Information & Knowledge Management*, 17(02), 1850015. doi:10.1142/S0219649218500156

Anindita Desarkar was awarded Master's of Technology in Computer Science in 2017 and is currently pursuing PhD under the supervision of Dr. Chitrita Chaudhuri and Dr. Ajanta Das from Jadavpur University in data mining domain. Her research area includes Evolutionary Algorithms, Data Mining and Data Analytics. She has already publications in International Journals, Conferences and Book Chapters. Anindita is having more than fifteen years of Industry experience in Data Warehouse domain in Tier 1 Company.

Ajanta Das is Professor in the Institute of Information Technology, Amity University. She has been awarded PhD in Computer Science & Engineering from Jadavpur University in 2009. She is having more than 24 years of work experience including 7 years of industry experiences. She started her career with Industry and is proud to work for reputed, famous, pioneer companies like, Tata Steel, Jamshedpur, India and Lexis Nexis Inc., Boston, USA. She is experienced with development and release of new real-life projects in Industry. Her major interest of teaching includes database, software engineering and distributed computing. Her field of Specialization is Distributed Computing. Her focused research area includes Cloud Computing, Internet of Things, Data Analytics, Wireless Sensor Network and Machine Learning.

Chitrita Chaudhuri is Associate Professor in the Department of Computer Science of Jadavpur University. She received Bachelor and Master's degree in Electronics & Tele Communication Engineering from Jadavpur University in 1980 and 1982 consecutively. Prior to joining Jadavpur University she has also worked with Birla Institute of Technology, Mesra, Kolkata Extension Centre from 1998 to 2001. Dr. Chaudhuri has been awarded PhD in Computer Science from Jadavpur University in 2016. Her fields of specialization include data mining and machine learning.