MIT | Academy of Engineering

School of Computer Engineering & Technology

Presentation to NBA Committee

Betterment of Online Platform by Monitoring Attentiveness and Activeness of User

Guide:

Prof. Sunilkumar Bhagat

Students Name (Roll No_Seat No):

Dewansh Singh (037_T214102)

Sakshi Dumbre (026_T214112)

Anurag Taparia (068_T214073)

Vedant Dawange (082_T214058)

Index

Introduction

- Conferencing tools have become one of the most essential tools for almost everyone in this pandemic. With the ”work-from-home” culture amid COVID-19, many online platforms such as Google meet, MS teams, Zoom, Teachmint, etc. have gain lot of popularity .
- Talking specifically on education sector, education rely on online platforms. But virtual classrooms does not matchup engagement of students like a real classroom. As everyone is attending lecture from home , this naturally reduces focus and increases distraction.
- It is observed that lecturers find it hard to monitor and look after every student in such a scenario due to divided webcam feeds, simultaneous presentation and replying on chats. The question arises in their mind that whether the student on the other side of the channel is attentive towards them or not, are they really listening to them?
- And one more problem occurs in extension of above problem is that, it becomes difficult for lecturer to pay attention on each individual. Many of the platforms does not provide analysis of each attendee.
- So, to answer these questions we have decided to perform research on this and come up with a solution.
- The idea of this project is to add feature which could give record/history of attending classes as well as attentiveness, activeness and activities performed in each class which will help the admin/lecturer to keep a check and helps in pupil management.

Problem statement

- Study of the Attentiveness and activeness of a user in online learning platform using Machine Learning.

Objectives to be achieved

1. To provide faculty with analysis so they can keep record/track of the students/users attentiveness, activeness and in-lecture performance using e-learning platforms.
2. We need to analyze the data acquired by the platform and analyze a student's/users attentiveness in the session and it's activities.
3. Developing with new ways to increase the student's attentiveness, interest and student's knowledge towards the faculty is also one of the main objectives of this research.
4. Try to improvise the data collection, so that the faculty can decide the method of teaching through this analysis, and they can choose one method which is most effective, interesting and attention seeker for a specific group of students.

Scope

- **Parameters taken:**

1. **Study different in lecture poll patterns:**

   Identify and analyse different in lecture poll patterns and keep the data storage of marks obtained in particular time.

2. **Storing the time taken**

   Marks obtained in the time taken by the user during Poll questions answering.

3. **Storing the grade points**

   Marks obtained after submitting the poll questions.

4. **Generate detailed attentiveness and activeness report:**

   Analyse the data and generate attentiveness and activeness reports based on the attendee performance. This feature will help both the student and the teacher to evaluate on individual basis.

- **Parameters need to be taken:**

1. **Capture log of hand raise count:**

   Track the total count of hand raise attempts performed by the user and store the log details count in order to analyse the data.

2. **Capture log of window switching count:**

   Track the total count of window switching attempts performed by the user and store the log details count in order to analyse the data.

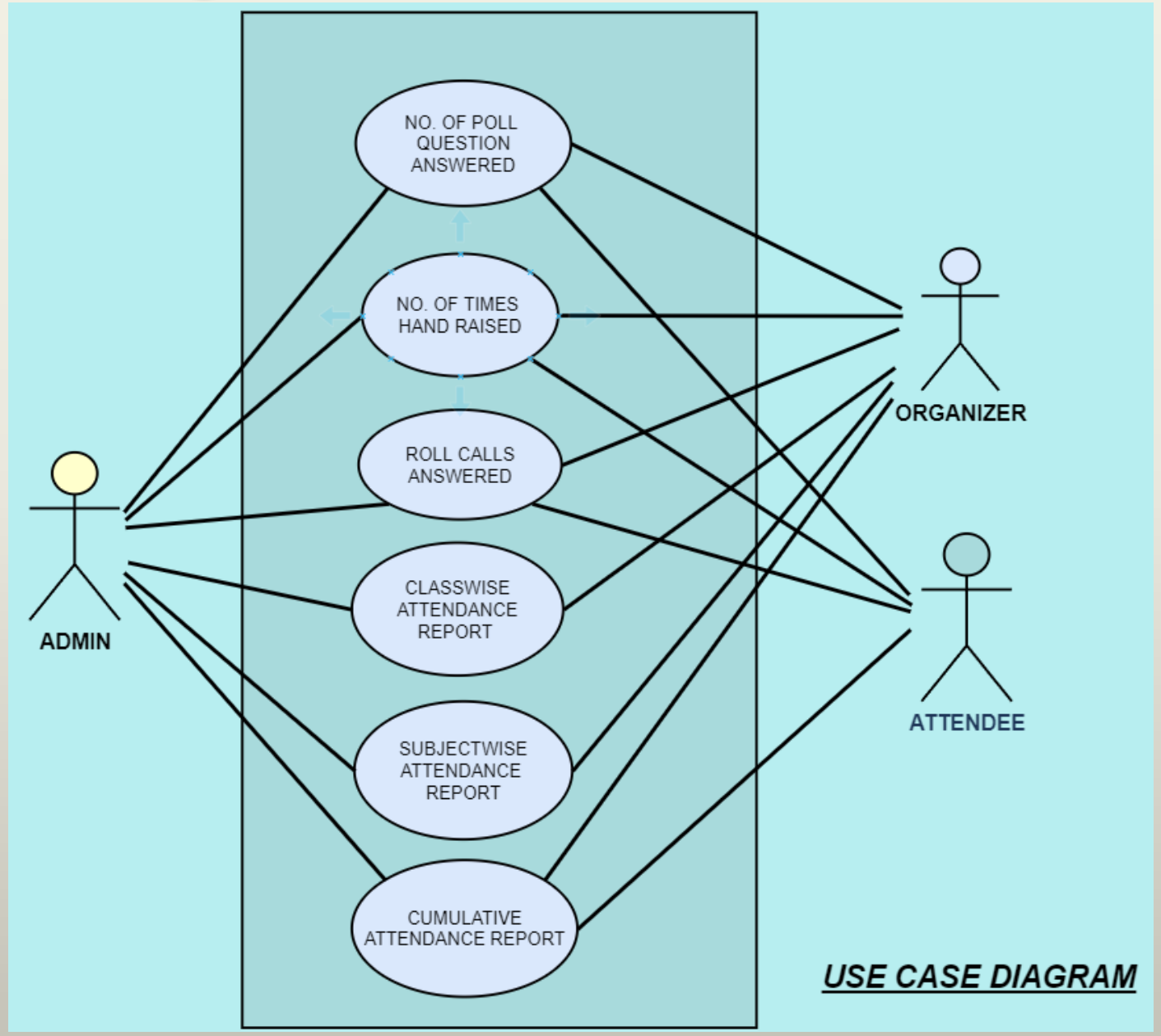
3. **Capture log of unmute attempts:**

   Track the total count of unmute attempts performed by the user and store the log details count in order to analyse the data.

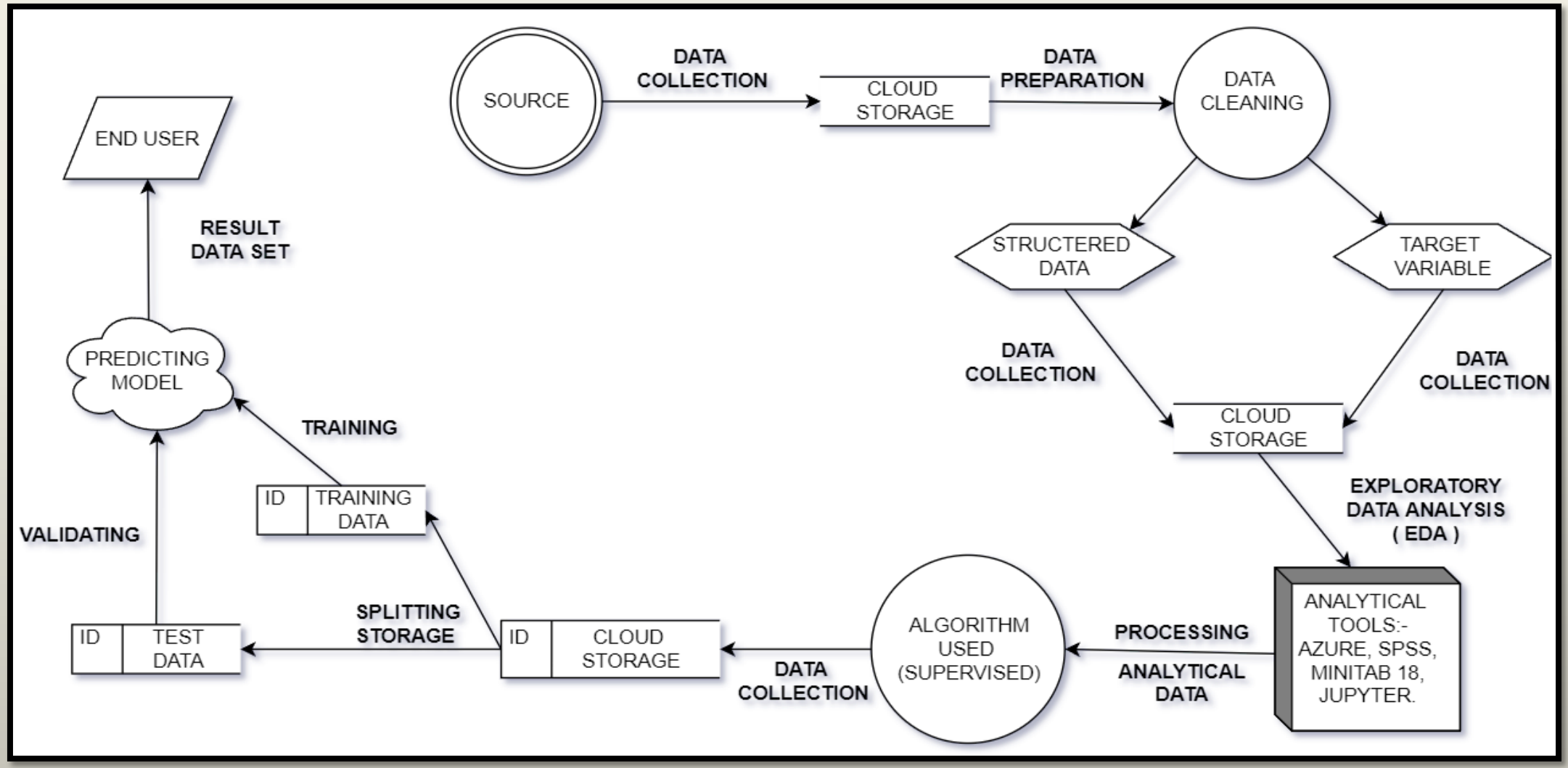
4. **Define feedback models:**

   Use various and appropriate Machine Learning models and algorithms to interpret the data generated and evaluate the responses.

Initial Design



Project Flow Diagram

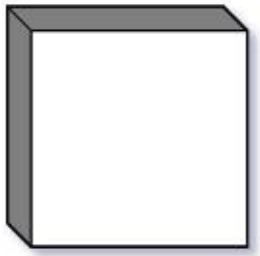


DESCRIPTION OF COMPONENTS USED IN PROJECT FLOW DIAGRAM

START END
OF THE DIAGRAM

COMPONENT TO
STORE DATA

COMPONENT TO
STORE DATA
WITH ID

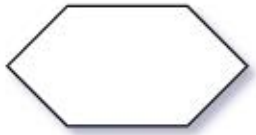


EXTERNAL ENTITY BOX FOR
INVOLVEMENT OF EXTERNAL
AGENT

COMPONENT
FOR OBJECT

COMPONENT FOR
RESULT STATE

COMPONENT FOR
PROCESSING STATE



CHECK BOX TO CHECK
THE CONDITION OF
INVOLVEMENT

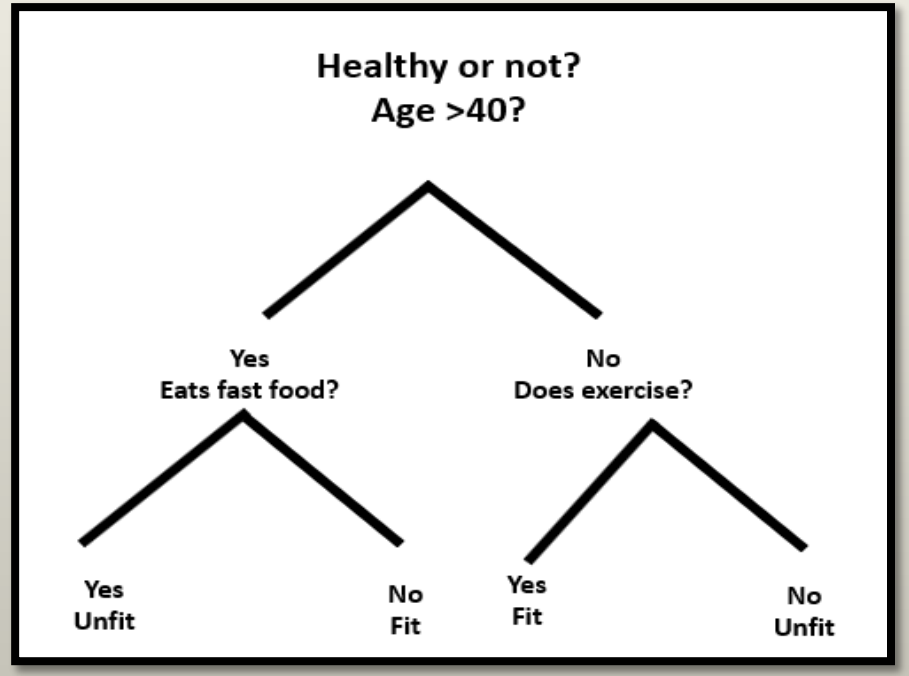
DIRECTIONAL
CONNECTOR

Algorithm

- **Supervised Algorithm:**

1. **Decision Tree**

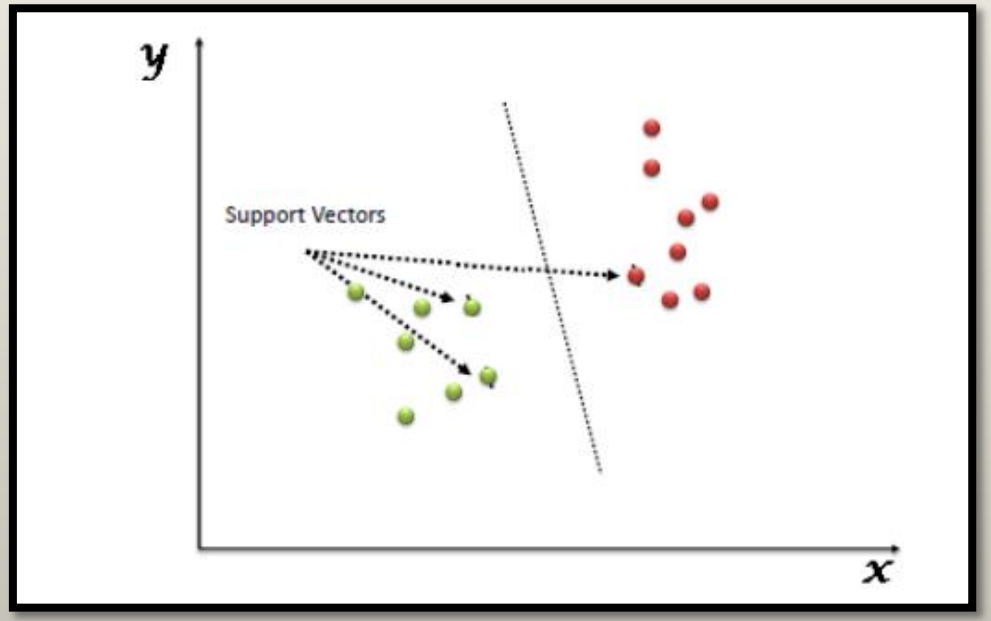
Introduction Decision Trees are **a type of Supervised Machine Learning** where the data is continuously split according to a certain parameter. Where we come out with the possibility of predicting the outcome. Basically the root data get split into leaf nodes and these leaf nodes are separated through two possibility either **“yes”** or **“no”**.



2. Support Vector Machine (SVM)

"Support Vector Machine" (SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges. However, it is mostly used in classification problems.

In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is a number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well.



Techniques

- For Decision Tree:

a) Calculate the impurities of data using Entropy.

b) Calculation of Gini Index will also lead to find impurity.

- For Support Vector Machine:

a) Calculate the accuracy with default hyper parameters.

b) Calculate the accuracy with different types of “kernel” and “C” value.

for example: “Linear” Kernel and C=1000.

Understanding Algorithm Mathematically

- Entropy is nothing but the **measure of impurity.**
- The Mathematical formula for Entropy is as follows -

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

- Where ‘Pi’ is simply the frequent probability of an element/class ‘i’ in our data. For simplicity’s sake let’s say we only have two classes , a positive class and a negative class. Therefore ‘i’ here could be either + or (-).

- So if we had a total of 100 data points in our dataset with 30 belonging to the positive class and 70 belonging to the negative class then ‘P+’ would be 3/10 and ‘P-’ would be 7/10. Pretty straightforward.

- The **Gini Index or Gini Impurity** is calculated by subtracting the sum of the squared probabilities of each class from one.
- It favors mostly the larger partitions and are very simple to implement. In simple terms, it calculates the probability of a certain randomly selected feature that was classified incorrectly.
- The Gini Index varies **between 0 and 1**, where 0 represents purity of the classification and 1 denotes random distribution of elements among various classes. A **Gini Index of 0.5** shows that there is equal distribution of elements across some classes.
- Mathematically, The Gini Index is represented by:-

$$G = \sum_{i=1}^{C} p(i) * (1 - p(i))$$

- Where, C is the total number of classes and p(*i*) is the probability of picking the data point with the class *i.*
- The **Gini Index works on categorical variables** and gives the results in terms of **"success"** or **"failure"** and hence performs only binary split. It isn't computationally intensive as its counterpart – Information Gain.
- From the Gini Index, the value of another parameter named Gini Gain is calculated whose value is maximized with each iteration by the Decision Tree to get the perfect CART.

Model

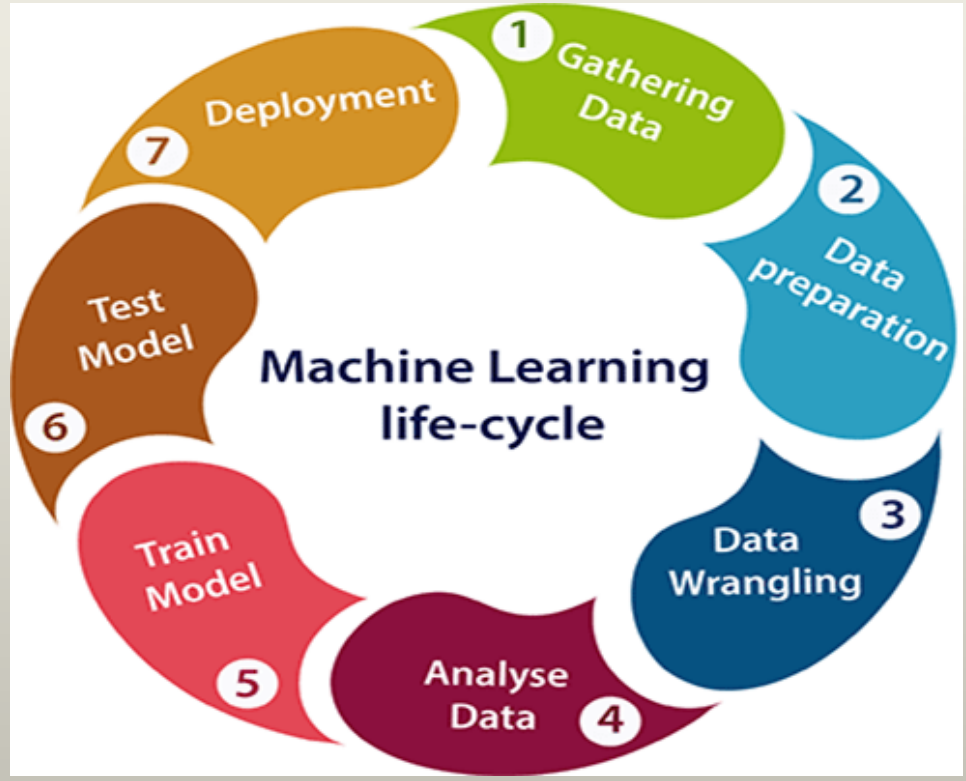
- ***Life Cycle of Machine Learning:***

In the complete life cycle process, to solve a problem, we create a machine learning system called "model", and this model is created by providing "training". But to train a model, we need data, hence, life cycle starts by collecting data.

Then we analyse the data and do necessary activities required to clean the missing values and pre-process.

Then according to data we split the data in training and testing data.

Then we train our model through train dataset and test our model while calculating the accuracy.



Transition and Implementation

- Almost 100% work is done..

Step-1: Gathering Data 

Step-2: Data Preparation 

Step-3: Data Wrangling 

Step-4: Analyse Data 

Step-5: Train Model 

Step-6: Test Model 

Methodology

- **The first** will be for user, after attending the lectures/meeting our platform will provide us with a certain data and by using the provided data we will analyze a user's attentiveness in the session and its activities.
- Here we are going to consider marks obtained in **Poll questions taken in live sessions, log count of hand raises, log count of window switching, log count user unmute themselves.**
- **Second** will be for person who is conducting some meeting, we will provide the faculty with the analysis so they can keep record/track of the students/users attentiveness, activeness and in-lecture performance using e-learning platforms.
- We will also be improvising the data collection, so that the faculty can decide the method of teaching through this analysis, and they can choose one method which is most effective, interesting and attention seeker for a specific group of students

Conclusion

- So from the project we can understand the attentiveness of student .It does not only consist about whether student have joined lecture but also considers various parameters which will help lecturer to understand how much a particular student was attentive in a lecture.

- Research done on the platform MS Teams we have come to the conclusion that the parameter which are available are not sufficient for telling whether the student is active and attentive or not. Timestamp, no of replies to polls and questions in a particular lecture are some of the parameters that are known and helpful to know activeness of a student. The other parameter which we have concluded are log of the tab changed, hand raise and mute & unmute of a particular student.

- With help of this project lecturer can know which students are attentive as well as active in the lecture. It will help lecturer to keep track on inattentive students.

Selected References

1. Nandish Amish Shah, Sidhdharth Sivasubramanian, Meenakshi K, Aditya Agarwal, ASSESSMENT OF STUDENT ATTENTIVENESS TO E-LEARNING BY MONITORING BEHAVIOURAL ELEMENTS,2021 International Conference on Computer Communication and Informatics (ICCCI -2021), Jan. 27 – 29, 2021, Coimbatore, INDIA.
2. Divya Dinesh, Kamal Bijlani, Athi Narayanan S*, Student Analytics for Productive Teaching/Learning,* 2016 International Conference on Information Science (ICIS).
3. Wei Weng, Feng Xie, Shiying Lu, Lingyan Niu, Shunying Wang, Xiaohan Jiang*, Analysis on Current Situation and Effect of Online Teaching Effect of Universities Based on Mobile Internet,* 2020 International Conference on Computers, Information Processing and Advanced Education (CIPAE).
4. Anish Patil ,Ankur Singh, Neha Chauhan,*Attentiveness Monitoring and user Record Maintenance in Virtual Classrooms,* Proceedings of the Fifth International Conference on Intelligent Computing and Control Systems (ICICCS 2021) IEEE Xplore Part Number: CFP21K74-ART; ISBN: 978-0-7381-1327-2
5. Krishan Kumar, Salsabil Nusair, Bibhya Sharma, Krishna Raghuwaiya, Gavin Jahir Khan,*An exploration on effectiveness of anonymous peer assessment strategy in online formative assessments,* December 18,2020 at 11:23:31 UTC from IEEE Xplore
6. Robert Ross, Carolyn Bell,*Turning the classroom into an escape room with decoder hardware to increase student engagement,* ©2019 IEEE
7. Oliver Au, Raymond So and Lap-Kei Lee,*Attentiveness and Self-Studying Are Keys to Academic Performance,* 2016 International Symposium on Educational Technology,

Thank You!!