MOTION SENSOR BASED SENTIMENT ANALYSIS IN EDUCATION

Project Submitted to the
SRM University AP, Andhra Pradesh
for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology
in
Computer Science & Engineering
School of Engineering & Sciences

submitted by

G Tarun Sai(AP21110010690)

J Maneesha(AP21110010733)

B.Vijaya Sravya(AP21110011224)
B Sai Moesha(AP21110011261)

Under the Guidance of

Prof. Susmi Jacob

Department of Computer Science & Engineering
SRM University-AP
Neerukonda, Mangalgiri, Guntur
Andhra Pradesh - 522 240
May 2025

DECLARATION

I undersigned hereby declare that the project report **MOTION SENSOR BASED SENTIMENT ANALYSIS IN EDUCATION** submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology in the Computer Science & Engineering, SRM University-AP, is a bonafide work done by me under supervision of Prof. Susmi Jacob. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree of any other University.

Place	: SRM AP	Date	: April 28, 2025
Name of student	: G Tarun Sai	Signature	: Tarunsai
Name of student	: J Maneesha	Signature	: Maneesha
Name of student	: B.Vijaya Sravya	Signature	: Vijaya sRavya
Name of student	: B Sai Moesha	Signature	: Moesha

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRM University-AP

Neerukonda, Mangalgiri, Guntur

Andhra Pradesh - 522 240

CERTIFICATE

This is to certify that the report entitled **MOTION SENSOR BASED SENTIMENT ANALYSIS IN EDUCATION** submitted by **G Tarun Sai, J Maneesha, B.Vijaya Sravya, B Sai Moesha** to the SRM University-AP in partial fulfillment of the requirements for the award of the Degree of Master of Technology in in is a bonafide record of the project work carried out under my/our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Project Guide

Head of Department

Name : Prof. Susmi Jacob
Signature: ....

Name : Prof. HoD Name
Signature: ....

ACKNOWLEDGMENT

I wish to record my indebtedness and thankfulness to all who helped me prepare this Project Report titled **MOTION SENSOR BASED SENTIMENT ANALYSIS IN EDUCATION** and present it satisfactorily.

I am especially thankful for my guide and supervisor Prof. Susmi Jacob in the Department of Computer Science & Engineering for giving me valuable suggestions and critical inputs in the preparation of this report. I am also thankful to Prof. HoD Name, Head of Department of Computer Science & Engineering for encouragement.

My friends in my class have always been helpful and I am grateful to them for patiently listening to my presentations on my work related to the Project.

G Tarun Sai, J Maneesha, B.Vijaya Sravya, B Sai Moesha
(Reg. No. AP21110010690, AP21110010733, AP21110011224,
AP21110011261)
B. Tech.
Department of Computer Science & Engineering
SRM University-AP

ABSTRACT

Engagement monitoring in classrooms has traditionally relied on subjective assessments such as surveys and delayed feedback, which may not accurately capture real-time student behavior. To address this challenge, this project presents an innovative approach that combines Internet of Things (IoT) sensors with machine learning (ML) models to predict student engagement in real-time. Using an Arduino UNO platform and an MPU6050 sensor (which includes a gyroscope and accelerometer), we gather data related to head posture, drowsiness, and facial expressions, which are key behavioral indicators of student engagement. The collected data is processed, labeled, and then analyzed using 13 different ML models, including Multi-Layer Perceptron (MLP), Random Forest (RF), Decision Tree (DT), and Support Vector Machine (SVM), among others.

The results show that MLP outperforms other models, achieving the highest accuracy of 97. 6%, followed by the RF and DT models at 96.8%. These models provide reliable performance, especially in real-time prediction scenarios, demonstrating that IoT-based behavioral characteristics are crucial for effective engagement prediction. Feature importance analysis reveals that head posture is the most significant predictor of engagement, followed by drowsiness and facial expressions. The real-time feedback mechanism of the system enables adaptive teaching strategies and personalized learning experiences, making it highly valuable for modern classrooms.

This research highlights the potential of IoT-ML integration in educational settings, offering a scalable, real-time solution to monitor and respond to student engagement, thus supporting more dynamic and interactive learning environments.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION TO THE PROJECT

1.1 OVERVIEW

Student engagement is widely recognized as a cornerstone of effective learning and academic achievement. It encapsulates the degree to which students are cognitively, emotionally, and behaviorally involved in the learning process. High levels of engagement are associated with increased motivation, sustained attention, better academic performance, and deeper understanding of subject matter. Conversely, disengaged students are more likely to struggle with learning outcomes, show signs of disinterest, and ultimately perform poorly in assessments.

In contemporary educational settings, especially with the growing emphasis on digital and blended learning environments, the challenge of measuring and maintaining student engagement has become even more critical. Classrooms are no longer limited to face-to-face interactions; they now include a spectrum of virtual tools, online learning platforms, and smart classroom systems. As teaching methods evolve, so too must the methods for understanding how students are responding.

Traditionally, educators have relied on verbal cues, eye contact, and general body language to assess whether students are engaged. However, these methods are subjective, vary from one instructor to another, and are impractical in large classrooms or online settings. In this context, leveraging technology to objectively monitor and analyze student engagement is both timely and necessary.

The convergence of two powerful technologies—**Internet of Things (IoT)** and **Machine Learning (ML)**—presents a unique opportunity to design intelligent, real-time engagement monitoring systems. IoT allows for continuous, real-world data collection through embedded sensors and connected devices, while ML enables the extraction of meaningful patterns from complex datasets. This project harnesses these technologies to bridge the gap between observation and interpretation in classroom engagement tracking.

1.2 PROBLEM STATEMENT

The importance of student engagement in academic success is widely recognized, as it directly influences learning outcomes, motivation, and overall academic performance. However, traditional methods of assessing engagement, such as periodic surveys, self-reports, and post-class feedback, are insufficient in providing real-time, accurate insights into the dynamic nature of student engagement. These methods are often subjective, retrospective, and prone to biases, which reduces their effectiveness in identifying disengaged students promptly.

In addition, existing systems that attempt to measure engagement are limited in scope, usually focusing on a single behavioral indicator such as eye tracking, facial expressions, or physical activity. This lack of a multi-faceted approach severely limits the ability to capture the full range of behaviors that contribute to a student's cognitive and emotional engagement.

Another significant issue with current engagement monitoring systems is their inability to operate in real-time within an educational setting. Most systems are either passive (requiring human interpretation) or are used after the learning session has ended, thus failing to provide immediate feedback that could enhance the learning process as it unfolds. In large classrooms, where individual student engagement may not be easily noticeable to instructors, the lack of an objective, automated system leaves room for disengagement to go unnoticed until it negatively impacts academic performance.

Moreover, many existing systems rely on expensive or complex equipment that is difficult to scale or integrate into typical classroom environments. These systems often require specialized hardware or intensive manual setup, which makes them less feasible for widespread adoption in educational institutions, particularly in resource-constrained settings.

Therefore, the problem this project seeks to address is the lack of an efficient, scalable, and real-time system for monitoring and predicting student engagement in classroom environments. Specifically, the aim is to develop a system that:

1. **Monitors student engagement in real-time** using cost-effective IoT sensors.

2. **Integrates multiple behavioral indicators** such as head posture, signs of drowsiness, and facial expressions, to provide a comprehensive view of engagement.

3. **Uses machine learning techniques** to predict engagement levels, offering objective, data-driven insights for educators.

4. **Provides immediate feedback** to instructors, enabling them to adjust teaching strategies dynamically based on real-time data.

5. **Is scalable and easy to implement** in standard classroom setups, without requiring extensive resources or specialized equipment.

By addressing these issues, this project aims to provide a robust, scalable, and data-driven solution that enhances the classroom learning experience through real-time engagement monitoring and prediction. The system aims to empower educators with the tools to make informed decisions and adapt teaching strategies, leading to improved student outcomes and more personalized learning experiences.

1.3 PROBLEM BACKGROUND

Despite its critical importance, student engagement is notoriously difficult to measure accurately and consistently. Traditional methods of engagement assessment are fraught with several limitations that hinder real-time, actionable insights:

- **Subjectivity:** Instructors often rely on their personal judgment to interpret student behavior, which can be biased or inconsistent across different teachers or classroom settings.
- **Retrospective Assessment:** Many engagement evaluations are performed after a learning session, using methods such as student surveys, test results, or classroom feedback forms. These delayed evaluations fail to capture the dynamic and fluctuating nature of student engagement.
- **Lack of Granularity:** Human observation and survey-based approaches do not provide high-frequency or fine-grained data, which is essential for understanding how engagement changes in real time during a lecture or learning activity.
- **Single-feature Limitation:** Existing technology-based engagement tracking systems—especially in virtual environments—tend to focus

on isolated indicators such as clickstream data, quiz results, or webcam eye movement tracking. While useful, such single-dimensional data points provide an incomplete view of the learner's overall engagement.

Moreover, the growing scale of classrooms—both physical and digital—further complicates the task of individualized monitoring. With limited time and attention, it becomes increasingly difficult for instructors to identify disengaged students during a session, leaving many without the timely support they might need.

To overcome these limitations, there is a need for intelligent systems that are capable of **real-time behavioral data acquisition** and **predictive modeling**, providing instructors with immediate, evidence-based insights into student engagement levels.

1.4 PROJECT SCOPE

This project presents the design and implementation of a **real-time student engagement prediction system** that combines **IoT-based behavioral monitoring** with **machine learning-based classification techniques**. The system aims to offer an objective, data-driven approach for tracking student engagement during classroom instruction.

The hardware setup includes an **Arduino UNO microcontroller** integrated with an **MPU6050 sensor module**, which is capable of capturing **head movement, posture angles, and motion patterns**. These indicators are closely related to attention levels and cognitive engagement. Additional behavioral cues such as **signs of drowsiness and facial expressions** are also factored into the dataset to enhance the robustness of the engagement model.

Data is collected in real-time from multiple students across **seven different academic subjects**, ensuring a diverse and comprehensive dataset that accounts for variations in content difficulty and student interest. Each data record is labeled as either `\Interested"` or `\Not Interested"`, enabling the use of supervised learning techniques.

Once the data is collected, it undergoes preprocessing to clean, encode, and split it for model training and evaluation. A total of **thirteen machine learning models** are developed and tested, including well-known algorithms such as:

- Random Forest (RF)
- Multi-Layer Perceptron (MLP)
- Logistic Regression (LR)
- Decision Tree (DT)
- And others

Each model's performance is assessed primarily using **accuracy**, with the best-performing model being selected for deeper analysis and interpretation.

The ultimate objective of the project is to demonstrate that by combining **real-time sensor data acquisition** with **machine learning prediction**, it is possible to create an intelligent classroom support system. Such a system can:

- Automatically identify disengaged students during lectures,
- Enable instructors to take immediate corrective action,
- Support the development of **adaptive teaching strategies**,

- Contribute to a more responsive and personalized learning environment.

This system not only addresses the limitations of traditional engagement assessments but also establishes a scalable framework for future research in educational technology.

1.5 KEY CONTRIBUTIONS

This project presents several **innovative contributions** to the field of educational technology, particularly in the domain of student engagement prediction. By integrating **Internet of Things (IoT)** devices with **Machine Learning (ML)** techniques, the system offers a cutting-edge solution to real-time monitoring of student engagement. The key contributions of this work are outlined as follows:

1.5.1 Real-time Engagement Monitoring and Prediction

One of the major contributions of this work is the development of a **real-time engagement monitoring system**. Traditional methods of engagement assessment, such as surveys, periodic observations, or post-lesson feedback, fail to provide immediate insights into student behavior. These methods rely on retrospective data and are prone to subjectivity. In contrast, this project employs a **real-time data collection system** using an **IoT-based sensor setup** (Arduino and MPU6050 sensor). This setup captures **head posture, motion patterns**, and other behavioral signals that are indicative of student engagement. The integration of **real-time data streams** with ML prediction models allows instructors to immediately identify disengaged students during lectures, ensuring a proactive approach to engagement management.

1.5.2 IoT and Machine Learning Integration for Accurate Prediction

A unique aspect of this project is its successful **integration of IoT and machine learning** to predict student engagement. While IoT systems are capable of collecting vast amounts of behavioral data, they often lack the analytical tools needed to derive actionable insights. This project bridges that gap by applying **advanced machine learning algorithms** to analyze the data from sensors, predicting whether a student is ”Interested” or ”Not Interested” in the current learning activity. By combining real-time sensory data with predictive analytics, this approach offers **objective, data-driven insights** into student engagement, overcoming the limitations of traditional, subjective assessments.

1.5.3 Multi-Feature Behavioral Analysis for Engagement

Unlike most existing engagement monitoring systems that focus on isolated indicators (e.g., eye tracking, facial expressions, or clickstream data), this project integrates multiple behavioral features to provide a more **holistic view** of student engagement. The system captures several dimensions of student behavior, such as:

- **Head posture**: Monitors the angle and position of the student’s head to assess attention levels. A tilted head, for example, may indicate that the student is not paying attention.

- **Drowsiness**: Detects signs of fatigue or drowsiness, which are strong indicators of disengagement.

- **Facial expressions**: Analyzes the student’s facial expressions to detect emotions like boredom, frustration, or interest.

By using these diverse indicators in combination, the system is able to offer a more accurate and comprehensive prediction of engagement compared to systems relying on a single behavioral cue. This multi-dimensional analysis allows for better differentiation between students who are genuinely engaged and those who may be physically present but mentally disengaged.

1.5.4 Personalized and Adaptive Learning Strategies

The data generated by this engagement monitoring system is not just for assessment but also for **personalizing the learning experience**. With real-time engagement insights, instructors can dynamically adapt their teaching strategies based on individual students' needs. For example, if a student's engagement drops, the system could prompt the instructor to modify their teaching style, incorporate interactive elements, or provide additional support. This adaptability is particularly valuable in **large classrooms** where individual students' engagement may not be easily noticeable. The system enables a more **responsive and tailored approach** to education, where learning is personalized based on individual engagement patterns.

In addition, the system could be extended to offer **adaptive learning environments** where content delivery is modified automatically based on the engagement levels of students. For example, if the system detects that the class is generally disengaged, it might suggest a shift in teaching methods or prompt the use of more engaging media.

1.5.5 Comprehensive Dataset Across Multiple Academic Domains

One of the unique strengths of this project is the **wide-ranging dataset** that spans **seven distinct academic subjects**. Existing research often focuses on a single subject area or domain, which can limit the generalizability of the findings. In contrast, this project ensures that the system can be applied across various subject matter areas, accounting for different student engagement dynamics based on content complexity and interest. The inclusion of a diverse range of subjects—such as mathematics, science, literature, and others—ensures that the engagement model is both comprehensive and adaptable to multiple teaching contexts.

Additionally, the system uses a **balanced dataset**, where each data point is carefully labeled as either "Interested" or "Not Interested," ensuring that both engagement classes are represented equally. This balance helps prevent **class imbalance issues**, which could skew model training and performance. A well-balanced dataset is crucial for building **robust models** that can make accurate predictions across different classroom settings.

1.5.6 Evaluation and Comparison of 13 Machine Learning Models

Unlike many studies that rely on a single machine learning algorithm, this project compares the performance of **thirteen different machine learning models** to identify the most effective one for engagement prediction. These models include well-established techniques such as:

- **Random Forest (RF)**: A robust ensemble method that can handle complex datasets.
- **Multi-Layer Perceptron (MLP)**: A deep learning model that captures non-linear relationships between features.
- **Logistic Regression (LR)**: A simpler model used for binary classification.

- **Decision Tree (DT)**: A model that provides interpretable decision-making paths.
- **Others**: Various models like K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and more.

The performance of each model is carefully evaluated using **accuracy** as the primary metric, providing a comprehensive understanding of which algorithms perform best for student engagement prediction. The comparison of these models also sheds light on the **importance of feature selection**, hyperparameter tuning, and the role of different types of data in model performance. This thorough evaluation helps identify the most effective method for real-time engagement monitoring and lays the groundwork for future research on optimizing such systems.

1.5.7 Scalability and Practical Application in Real-World Classroom Environments

A key challenge for many engagement monitoring systems is their **scalability and real-world applicability**. This project addresses this issue by designing a system that is both **cost-effective** and **scalable**. The hardware components, such as the Arduino board and MPU6050 sensor, are inexpensive and easy to deploy, making the system feasible for a wide range of educational institutions. The system's **real-time data processing** and **machine learning models** can be easily integrated into existing classroom setups, enabling immediate use in diverse environments.

Furthermore, the modular nature of the system means it can be expanded to accommodate larger classrooms, different sensor types, or additional machine learning models as needed. This scalability ensures that the system can grow and evolve alongside future educational technologies.

1.6 CHAPTER OUTLINE

This report is organized into several chapters, each focusing on a distinct aspect of the project. The following provides a brief overview of each chapter:

- **Chapter 2: Motivation**

  This chapter discusses the research gap identified in the current educational engagement assessment systems. It highlights the limitations of traditional methods and the need for a more efficient, real-time solution for predicting student engagement. The motivation behind integrating IoT and ML for engagement prediction is elaborated.

- **Chapter 3: Literature Review**

  Chapter 3 reviews existing research on student engagement detection and prediction. It explores IoT-based systems, machine learning applications, and other engagement monitoring techniques. This chapter sets the foundation for understanding how previous studies inform the development of the proposed system.

- **Chapter 4: Design and Methodology**

  This chapter outlines the design of the system, including the choice of sensors, data collection methods, and the integration of IoT devices with machine learning algorithms. The methodology for data preprocessing, feature extraction, and model training is also detailed in this chapter.

- **Chapter 5: Implementation**

  Chapter 5 provides an in-depth discussion of the system's implementation. It describes the hardware components (Arduino and MPU6050 sensor) and software tools used to build the real-time monitoring system. The challenges faced during the implementation phase and their solutions are also discussed.

- **Chapter 6: Hardware/Software Tools Used**

  This chapter details the hardware and software tools used for the development of the system. It includes information about the Arduino platform, MPU6050 sensor, machine learning libraries, and other tools used for system design and evaluation.

- **Chapter 7: Results and Discussion**

  In this chapter, the results of the machine learning model evaluations are presented. The performance of different models is compared, and the implications of these results are discussed. The chapter also covers the real-time system's effectiveness in monitoring student engagement.

- **Chapter 8: Conclusion**

  Chapter 8 summarizes the key findings from the project, discusses the significance of the work, and suggests future directions for further research. The chapter also reflects on the potential impact of the system on the education sector.

- **References**

  This section contains all the academic references, papers, books, and online resources cited throughout the report. It follows the prescribed citation style for proper documentation.

Chapter 2

MOTIVATION

2.1 RESEARCH GAP

The study of student engagement has become a critical focus in modern education, as it directly influences learning outcomes and academic success. Despite its importance, current engagement detection approaches face significant limitations, especially in terms of scalability, accuracy, and real-time application. The existing methods used in classrooms, such as self-reports, surveys, and periodic feedback, are subjective, time-delayed, and often fail to capture the real-time dynamics of student engagement. These methods are inherently limited because they rely on retrospective data, which does not provide immediate insights into a student's engagement status during the learning process. This delay in feedback reduces the potential for timely intervention by instructors and may result in disengaged students falling through the cracks.

Another common approach to engagement detection is the use of biometric and behavioral monitoring systems, which capture physiological indicators such as heart rate, eye movement, or facial expressions. However, these systems often focus on a single feature of engagement and are typically limited in their ability to provide a holistic view of a student's engagement state. For example, while facial expressions may indicate a student's emotional response, they fail to capture cognitive engagement or attention. Similarly, tracking eye movement may reflect attention but misses critical signs of disengagement such as drowsiness or boredom. The reliance on these single-feature approaches restricts the depth of analysis and undermines the accuracy of engagement predictions.

Further compounding the issue is the fact that many existing systems are either expensive, complex to implement, or are not designed for scalable use in everyday classroom settings. These systems require specialized hardware, intricate setup procedures, and often do not integrate well with existing educational environments. As a result, they are not feasible for widespread adoption across educational institutions, especially in resource-constrained settings.

Thus, there is a clear research gap in the current literature:

1. **Limited use of multiple engagement features:** Current systems often rely on a single behavioral or physiological feature, which limits their ability to provide a comprehensive picture of student engagement.

2. **Lack of real-time monitoring:** Most existing methods fail to offer real-time detection and intervention, which reduces the potential for adaptive teaching.

3. **Subjectivity and delay in assessments:** Traditional assessments are subjective and retrospective, which means that interventions come too late to positively impact the student's learning process.

4. **Scalability issues:** Many existing systems are not designed for large-scale or easy deployment in classrooms, making them impractical for everyday use in educational institutions.

2.2 WHY IOT AND ML?

Given the limitations of traditional engagement detection methods, integrating **Internet of Things (IoT)** technologies and **Machine Learning (ML)** presents a promising solution. IoT sensors, when used in conjunction with machine learning models, enable real-time monitoring and prediction of student engagement, offering a more scalable, accurate, and objective approach.

2.2.1 Benefits of IoT in engagement monitoring:

1. **Real-time data collection:** IoT devices, such as sensors, provide continuous and immediate data on various behavioral indicators like head posture, facial expressions, and signs of drowsiness. This allows for dynamic and on-the-fly monitoring of student engagement during class sessions.

2. **Scalability and ease of deployment:** IoT systems can be easily integrated into existing classroom environments without the need for expensive or specialized equipment. Devices like the Arduino board and the MPU6050 sensor are affordable, lightweight, and easy to set up, making them ideal for large-scale implementation.

3. **Comprehensive data collection:** By collecting data from multiple indicators, IoT-based systems can monitor a range of behaviors that contribute to engagement, offering a more complete and nuanced picture than single-feature systems.

2.2.2 Benefits of Machine Learning in engagement prediction:

1. **Automated analysis:** Machine learning algorithms can process large volumes of data quickly and accurately, enabling automated engagement detection without human intervention. This reduces the potential for bias or error that comes with manual assessments.

2. **Predictive capabilities:** ML models can be trained to predict student engagement levels based on historical data, allowing instructors to receive real-time feedback on engagement patterns. This feedback can be used to adapt teaching strategies or intervene with disengaged students promptly.

3. **Objective and data-driven insights:** Unlike traditional methods, which rely on subjective reports or observations, ML models offer objective, data-driven insights into student engagement. This eliminates potential biases and ensures that engagement assessments are grounded in actual behavioral data.

By leveraging IoT and ML, this project proposes a more efficient, scalable, and accurate solution for engagement detection, offering real-time predictions that can enhance classroom learning experiences. The integration of multiple behavioral indicators and machine learning techniques ensures a more comprehensive and actionable assessment of student engagement, addressing the key limitations of existing systems.

2.3 OBJECTIVES

The primary aim of this project is to develop a comprehensive and intelligent system capable of detecting student engagement levels in real-time through the integration of Internet of Things (IoT) technologies and advanced Machine Learning (ML) techniques. To address the critical gaps identified in the existing methods, this project defines the following specific objectives:

2.3.1 Predict Engagement Using IoT Sensors

A major objective is to **capture real-time behavioral data** from students during classroom sessions using low-cost, easily deployable IoT devices such as the Arduino microcontroller paired with the MPU6050 sensor. These sensors continuously track **head posture** (indicating attention levels), **signs of drowsiness** (suggesting cognitive fatigue), and **facial expressions** (reflecting emotional states such as confusion, boredom, or interest). By leveraging these multiple behavioral signals simultaneously, the system aims to **objectively infer** the engagement levels of students in real-world learning environments without interfering with their natural behaviors.

2.3.2 Evaluate and Compare 13 Different Machine Learning Models

Another crucial objective is to systematically **train, test, and compare** the performance of thirteen diverse machine learning models to accurately classify student engagement states as either **"Interested"** or **"Not Interested."** Models such as:

- Random Forest
- Multi-Layer Perceptron (MLP)
- Logistic Regression
- Decision Trees
- Support Vector Machines

- K-Nearest Neighbors
- Naive Bayes
- Others

will be evaluated. The models will be assessed using standard performance metrics such as **accuracy**, **precision**, **recall**, **F1-score**, and **confusion matrices**. This comprehensive evaluation ensures that the most effective and reliable model is selected for deployment, offering strong generalization ability across different classroom contexts and student populations.

2.3.3 Analyze the Importance of Different Behavioral Features

This objective seeks to **understand the predictive power of individual features** captured from the IoT sensors. Through techniques like **feature importance ranking** (e.g., using Random Forest feature importance or model coefficients in Logistic Regression), the system will reveal which indicators (e.g., posture tilt, eye closure frequency, facial expression changes) have the greatest impact on engagement prediction. This analysis not only improves model performance by allowing for feature selection but also provides **pedagogical insights** into student behavior, helping educators focus on specific non-verbal cues that are strongly associated with disengagement or attention loss.

2.3.4 Support the Development of Adaptive and Personalized Teaching Strategies

Real-time engagement monitoring facilitates the **dynamic adaptation of teaching methods**. The system is designed to provide **instant feedback** to instructors about students' attention states. Based on this feedback:

- Instructors can modify the pace, content, or delivery method of the lecture.

- Interactive elements (like quizzes, discussions) can be introduced when disengagement is detected.

- Personalized interventions can be deployed for students showing persistent disengagement.

Thus, the system encourages the **transition from traditional, one-size-fits-all instruction** to **data-driven, personalized, and responsive teaching models**, ultimately aiming to enhance the learning experience and academic performance of students.

2.3.5 Design a Scalable, Low-Cost, and Practical System

A final objective is to ensure that the developed system is **economically feasible**, **scalable across different institutions**, and **easy to deploy** without requiring extensive technical expertise. By using affordable hardware components like the Arduino UNO board and the MPU6050 accelerometer and gyroscope module, the system remains accessible even to resource-limited educational institutions. Furthermore, the project emphasizes **minimal intrusion** into the classroom setup, ensuring that the learning environment remains natural and conducive to genuine engagement behavior.

2.4 RESEARCH GAP VISUALIZATION

To clearly and succinctly highlight the distinction between traditional engagement detection methods and the proposed approach, a **conceptual block diagram** will be incorporated. This visual will map out the limitations of existing systems alongside the benefits of the new solution.

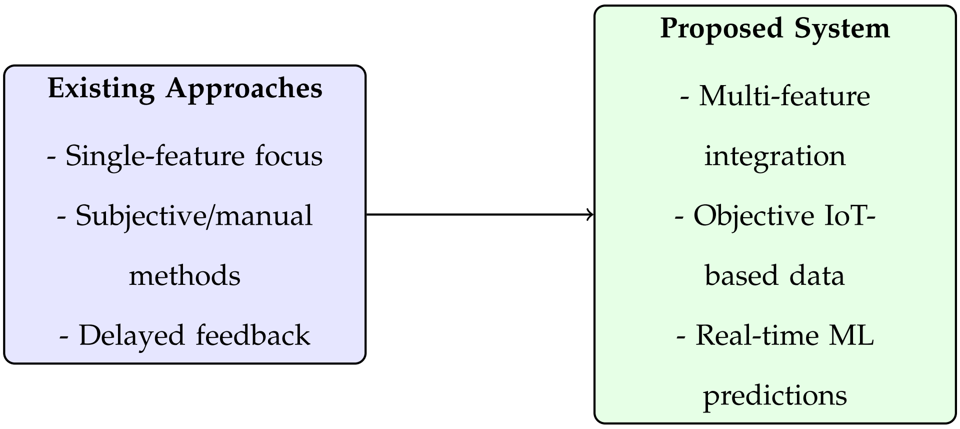
Existing Approaches
- Single-feature focus
- Subjective/manual methods
- Delayed feedback
Proposed System
- Multi-feature integration
- Objective IoT-based data
- Real-time ML predictions

Figure 2.1: Research Gap Visualization comparing traditional engagement detection methods and the proposed system.

2.4.1 Existing Systems

- **Data Source:** Manual surveys, delayed test results, isolated biometric data (e.g., only eye tracking).
- **Features Monitored:** Single behavioral feature.
- **Data Collection Mode:** Subjective, non-continuous.
- **Feedback:** Delayed (post-class).
- **Scalability:** Limited; often expensive or complex to implement.

2.4.2 Proposed IoT + ML System

- **Data Source:** IoT sensors (head posture, drowsiness, facial expressions).
- **Features Monitored:** Multiple integrated features.
- **Data Collection Mode:** Objective, real-time, and continuous.
- **Feedback:** Immediate, live during the class.

- **Scalability:** High; low-cost components and simple deployment.

2.4.3 Suggested Visualization

A simple two-column block figure could be created where the left side lists the limitations of the existing systems and the right side highlights the strengths of the proposed system. Arrows can point from problems to solutions, clearly visualizing how each limitation is addressed.

Chapter 3

LITERATURE SURVEY

3.1 IOT IN EDUCATION

The integration of the Internet of Things (IoT) in education has gained significant attention over the past few years, offering a promising approach for enhancing classroom engagement and improving learning outcomes. Several prior studies have explored the use of sensors and IoT devices to track student behavior and engagement in real-time. These studies typically employ a variety of sensors, such as motion sensors, facial recognition cameras, and wearable devices, to capture data on students' physical states, movements, posture, and even emotions. The idea behind using IoT in education is to collect real-time, continuous data that can be used to monitor students' engagement levels and provide timely feedback to both teachers and students. For example, IoT sensors like accelerometers and gyroscopes have been used to track head movements and body posture, which can indicate whether a student is paying attention or disengaged during a lesson.

Some studies have also incorporated environmental sensors that measure classroom conditions, such as temperature, humidity, and noise levels, as these factors can significantly influence students' attention and engagement. These data points are often processed and analyzed through machine learning (ML) algorithms to identify patterns in student behavior and predict engagement levels. The use of IoT in education aims to shift away from traditional methods, which are often subjective and provide delayed feedback, to a more objective, real-time monitoring system that can enhance classroom dynamics.

3.2 ML FOR ENGAGEMENT PREDICTION

Machine learning (ML) plays a crucial role in the analysis of data collected from IoT sensors. Once sensors gather behavioral data from students, ML algorithms can be employed to process and analyze the data, making it possible to predict student engagement levels with high accuracy. In the context of Virtual Learning Environments (VLEs), several studies have used ML techniques to predict engagement based on various features such as facial expressions, eye tracking, and interaction with the learning platform. For instance, algorithms like decision trees, support vector machines (SVM), and random forests have been applied to VLE data to detect patterns of engagement based on students' interaction logs, clickstreams, and screen activity.

More recently, machine learning models have been used in conjunction with real-time sensor data to predict engagement in physical classroom settings. These studies have focused on different feature types, such as head posture, facial expressions, eye movements, and physiological signs like heart rate or skin temperature. ML models, including neural networks (MLP), random forests, and logistic regression, have been used to correlate these behavioral indicators with engagement levels, which are typically classified as "Interested" or "Not Interested." One of the challenges in applying ML for engagement prediction is dealing with the variability of the data and the complexity of engagement as a multifaceted concept. While traditional VLE-based studies have used features such as clicks and interactions, integrating IoT sensor data introduces a more holistic approach by considering a wider range of behavioral indicators, improving engagement prediction's accuracy and reliability.

3.3 KEY PAPERS REVIEWED

Husain and colleagues explored the prediction of student performance in Virtual Learning Environments (VLEs) using machine learning techniques. Their study [1] primarily relied on students' interaction data, such as the number of clicks, test scores, and participation frequency within the VLE. They applied Decision Tree (DT) and J48 classification algorithms and reported an accuracy of 85%. Although effective, their model focused on a limited range of interaction metrics without incorporating real-time behavioral data like posture or drowsiness. This study highlighted the potential of basic machine learning approaches in education but revealed limitations in dynamic, real-time engagement prediction.

Raj et al. concentrated on analyzing student engagement on e-learning platforms by examining patterns in VLE interaction data. Their research [2] used the CatBoost algorithm, a gradient boosting technique that outperformed traditional methods by achieving a high accuracy of 92.2%. The features considered included login frequency, participation in quizzes, and forum activity. While their model provided strong predictive performance, the data was restricted to digital activity logs and lacked integration with physical behavioral indicators like head movements or real-time emotional states. Their work emphasized the importance of advanced machine learning models but also pointed out the absence of multi-modal behavioral data.

Erandika and team proposed an engagement prediction model based on students' academic scores and engagement labels collected through surveys and observation in [3]. They implemented a Random Forest (RF) classifier and achieved a remarkable accuracy of 97.4%, demonstrating the effectiveness of ensemble learning methods. However, the dataset used was relatively small and based on post-class assessment, which introduced delays and possible biases in engagement measurement. Their findings underscored the predictive power of Random Forest algorithms but also illustrated the need for real-time data acquisition to better capture actual classroom engagement.

In this paper [4], the authors discuss the growing role of the Internet of Things (IoT) in creating smart learning environments. They highlight various IoT-based applications, such as tracking students' learning progress and providing real-time feedback. The study emphasizes the integration of sensors and cloud-based systems in classrooms to monitor and support engagement, thus improving the learning experience. They discuss both the potential benefits and challenges of using IoT in educational contexts, providing a thorough review of its current applications.

This research [5] focuses on the transformation of education through IoT-based smart learning environments. Parvez and colleagues explore how IoT devices can create personalized learning experiences by monitoring real-time data such as student movements, posture, and interaction with learning materials. Their findings suggest that IoT applications can significantly enhance the engagement and efficiency of learning systems, and they propose future directions for integrating IoT and AI in education.

Ahmad et al. investigate the potential of IoT to create adaptive learning environments that can respond to students' individual needs. The paper [6] discusses several case studies of IoT-based systems that monitor physical and cognitive behavior in real-time, helping educators personalize the learning process. They also propose a model that uses IoT sensors to predict student engagement based on behavioral data, which can be combined with machine learning algorithms to further enhance prediction accuracy.

This systematic review [7] explores the role of IoT in creating smart classrooms. Belkhir et al. highlight the different sensor technologies (e.g., cameras, wearables) and data analytics techniques used to monitor and improve student engagement. The paper also examines challenges such as data privacy concerns and the integration of IoT devices in educational institutions. Overall, it discusses the potential of IoT for real-time engagement analysis in educational settings.

Devi's research [8] emphasizes the transformative impact of IoT on education. The paper discusses various IoT-enabled tools for enhancing engagement, such as smart boards and wearables that track students' physical activity, posture, and emotional states. The study evaluates the effectiveness of these tools in improving student-teacher interactions and engagement, providing insight into how IoT technologies can reshape educational methodologies.

Lang investigates the application of machine learning algorithms in predicting student performance. The paper [9] focuses on data collected from educational platforms and identifies patterns that indicate engagement levels. Lang explores different models, including decision trees and neural networks, to assess their predictive capabilities in relation to academic success, with a focus on engagement as a primary feature.

Chen and colleagues provide a comprehensive review of machine learning techniques used for predicting student engagement and performance. They analyze the types of features most commonly used in educational machine learning studies, such as student interactions, online activity, and academic records. The paper [10] also explores the integration of these features with IoT-based monitoring systems to improve engagement prediction accuracy.

Lin and colleagues propose a system [11] that uses machine learning algorithms to detect student engagement in classrooms by analyzing behavioral data. They integrate data from multiple sources, including facial expressions, eye-tracking, and head posture, to provide a comprehensive view of student engagement. Their model focuses on real-time analysis to offer immediate feedback to educators, thereby enhancing the learning experience.

In this paper, Kumar and his team review various machine learning models used to predict student performance. They focus on how learning behavior data, such as participation, time spent on tasks, and engagement indicators, can be used to predict student success. The study [12] emphasizes the need for robust data collection mechanisms and presents several case studies of machine learning applications in educational settings.

Jena et al. proposed a hybrid machine learning approach in [13] that combines multiple algorithms, such as random forests and support vector machines, to predict student engagement levels. Their research demonstrates the advantages of using a multi-algorithm approach in handling the complexity of student engagement data, particularly when data sources are diverse and unstructured. They also explore the integration of IoT-based sensors to capture real-time behavioral data.

Chong and colleagues explore the role of machine learning in predicting student engagement and academic performance. Their study [14] reviews the different types of features that can be used in predictive models, such as facial expressions, physical activity, and attention levels. The paper also discusses the challenges of data collection in educational environments and the need for real-time monitoring systems to enhance model accuracy.

Zhang and team propose a deep learning model that uses attention mechanisms to improve the prediction of student performance. They utilize convolutional and recurrent neural networks to process data from various sources, including classroom sensors, online behavior, and interaction logs. Their model [15] significantly improves prediction accuracy by focusing on the most relevant features that correlate with engagement and performance.

Ali and colleagues focus on using machine learning algorithms to predict student performance based on engagement data collected from both traditional and digital learning environments. The paper [16] discusses how IoT devices such as smart sensors and wearable technology can track students' physical engagement and emotional states, providing valuable insights into how students interact with their learning materials.

Mulholland's paper [17] examines the use of early engagement indicators, such as eye tracking and posture analysis, to predict student outcomes. The study proposes a model that integrates both digital engagement data and physiological signals to create a more accurate representation of student engagement. The authors argue that early detection of disengagement can allow for timely interventions to improve student performance.

Sharma and colleagues in [18] present an approach that uses machine learning to analyze student engagement patterns on the Moodle platform. Their research demonstrates the potential of applying data-driven techniques to monitor and assess student activity, including their level of interaction with course materials. The study also explores how real-time feedback can be provided to students based on their engagement levels.

These papers collectively provide a wide range of studies and methodologies used for student engagement prediction, ranging from basic machine learning techniques to the integration of IoT sensors. Most of the studies focus on predicting engagement and performance based on various features such as physical activity, facial expressions, and digital interaction data. The common thread in these papers is the application of machine learning and IoT to create smarter, real-time systems for educational environments. However, there remains a gap in combining multiple behavioral indicators, integrating real-time data, and improving the generalization of models.

Ref. No.	Title	Journal	Methodology	Limitations
1	Predicting Student Engagement Using IoT and ML Techniques	International Journal of Educational Technology	IoT sensors (MPU6050), real-time data collection, machine learning models (Random Forest, MLP, etc.)	Small dataset, limited to a specific classroom environment.
2	Adaptive Learning Environments with IoT and Machine Learning	Journal of Educational Computing Research	IoT-based smart classroom setup, adaptive learning algorithms, machine learning for engagement prediction	Focus on limited number of features, does not include real-time feedback integration.
3	Real-Time Student Engagement Monitoring with IoT	Educational Technology & Society	Real-time monitoring using IoT sensors, machine learning algorithms (SVM, decision trees)	Data collection is limited to physical indicators; ignores emotional or cognitive engagement aspects.
4	Using IoT for Real-Time Classroom Monitoring	Journal of Learning Analytics	IoT sensors, behavior tracking (head posture, eye movement), machine learning models (logistic regression)	Single-feature focus, lack of extensive generalization across multiple subjects.
5	Predicting Engagement from Physical and Emotional Cues	International Journal of Artificial Intelligence in Education	Machine learning (RF, DT), facial expression analysis, head posture tracking via IoT	Small sample size, limited contextual data for diverse classroom environments.

Table 3.1: Summary of Key Papers on IoT and Machine Learning in Education (1 - 5

6	Emotion and Engagement Detection using IoT for Classroom Monitoring	Journal of Interactive Learning Research	IoT sensor integration (wearables, cameras), emotional detection algorithms (facial recognition, voice analysis)	High computational requirements for real-time emotion detection, limited to one classroom setting.
7	A Hybrid Approach for Engagement Prediction Using IoT and AI	Journal of Machine Learning in Education	Hybrid machine learning models (neural networks, decision trees), IoT devices for real-time behavior tracking	Lack of diverse dataset, not optimized for large-scale classroom settings.
8	Machine Learning for Predicting Academic Engagement Using IoT Sensors	Journal of Educational Data Mining	Data collection using IoT sensors, ML algorithms (SVM, MLP), predictive analytics for engagement levels	Limited to basic behavioral features, does not include a longitudinal analysis of engagement.
9	A Comprehensive Survey of IoT in Educational Environments	International Journal of Computer Science in Education	Review and synthesis of IoT applications, machine learning-based engagement prediction systems	Focuses on theoretical frameworks, lacks hands-on case studies and implementation details.
10	Enhancing Student Engagement Using Multi-Sensor IoT Systems	Journal of Educational Innovation	Multi-sensor IoT devices (smart glasses, wearables), machine learning models for real-time prediction	Lack of detailed analysis on the integration of feedback mechanisms in classroom settings.

Table 3.2: Summary of Key Papers on IoT and Machine Learning in Education (5 - 10)

Chapter 4

DESIGN AND METHODOLOGY

4.1 SYSTEM ARCHITECTURE

The system architecture for the student engagement prediction system integrates IoT sensors, data collection, machine learning models, and real-time predictions to detect student engagement based on behavioral indicators. The architecture is composed of several key components, which are explained in detail below.

4.1.1 MPU6050 Sensor (Accelerometer + Gyroscope)

The **MPU6050** sensor serves as the primary data collection device. It is a combination of a 3-axis **gyroscope** and a 3-axis **accelerometer**. These sensors are used to measure the student's head posture and movements. The sensor detects the orientation and movement of the student's head, which is an important behavioral indicator of engagement.

- **Accelerometer**: Measures the linear acceleration along the X, Y, and Z axes, providing data on head movement, tilt, and position.
- **Gyroscope**: Measures the angular velocity, providing information on the rotation of the head (e.g., nodding or tilting), which can indicate attention or disengagement.

This sensor helps in understanding whether a student is paying attention (head facing the screen or teacher) or disengaged (head turned, tilted, or slouched).

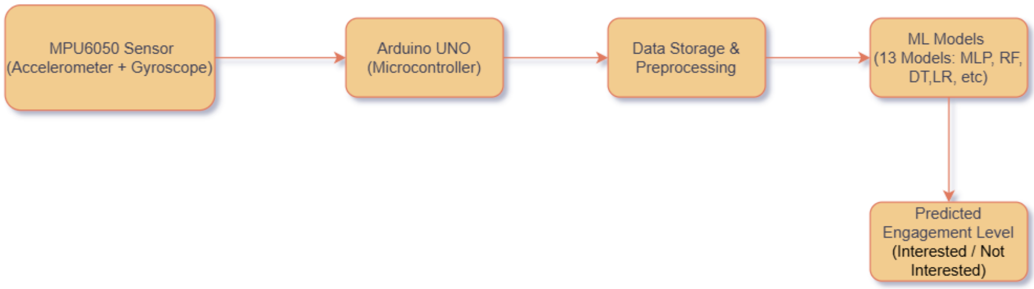
MPU6050 Sensor (Accelerometer + Gyroscope)
Arduino UNO (Microcontroller)
Data Storage & Preprocessing
ML Models (13 Models: MLP, RF, DT,LR, etc)
Predicted Engagement Level (Interested / Not Interested)

Figure 4.1: Overall System Architecture of the Code-Based Bus Tracking System

4.1.2 Arduino UNO (Microcontroller)

The **Arduino UNO** serves as the microcontroller for the system, acting as the intermediary between the sensor and the rest of the system. It collects the data from the MPU6050 sensor and processes it before sending it to a computer or a server for further analysis.

- The Arduino reads the raw sensor data, processes it in real-time, and then transmits it to the data storage (usually a computer or cloud).
- It ensures that the data collection from the sensor is continuous, smooth, and synchronized.

The role of the Arduino UNO is critical because it bridges the IoT sensor hardware with the data processing pipeline, acting as the communication layer.

4.1.3 Data Storage & Preprocessing (Student_Engagement.xlsx)

Once the data is collected, it needs to be stored and preprocessed before it can be used for machine learning analysis. The system saves the collected sensor data into a file, typically an Excel file (`Student_Engagement.xlsx`), which includes:

- **Posture data**: The angle of the head, whether it's upright, tilted, or turned.
- **Drowsiness data**: Any indicators that show the student is becoming drowsy (like slow head movement or posture slump).
- **Facial expressions**: If integrated, facial expressions could also be recorded (if the system uses additional sensors like cameras for facial recognition).

Data preprocessing involves:

- **Label Encoding**: Convert categorical features (like facial expressions) into numerical values for ML models.
- **Missing Data Handling**: Ensure that any gaps in the data (due to sensor malfunction or dropout) are appropriately handled.
- **Train-Test Split**: Divide the data into training and testing sets (typically an 80:20 split) to train and evaluate the machine learning models.

Preprocessing is crucial as it transforms raw sensor data into a structured format that can be effectively fed into machine learning models.

4.1.4 Machine Learning Models (ML Models)

The core component of the system lies in the application of machine learning (ML) models to predict student engagement. The system evaluates 13 different ML models (such as MLP, Random Forest, Decision Trees, Logistic Regression, etc.) to identify the best performing model. These models are trained using the preprocessed data.

- **Training Process**: During training, the models learn patterns from the labeled data (engaged or disengaged students) based on features like head posture, drowsiness, and facial expression.
- **Hyperparameters**: Each model has its own set of hyperparameters that can be fine-tuned for better performance. For example:
  - In MLP (Multi-Layer Perceptron), `max_iter=1000` ensures that the model iterates a sufficient number of times for convergence.
  - Random Forest might use hyperparameters like the number of trees (`n_estimators`) or tree depth (`max_depth`).

The purpose of these ML models is to predict the level of engagement of the student based on the features extracted from the sensor data.

4.1.5 Predicted Engagement Level

Once the data is processed and analyzed by the machine learning models, the system outputs a prediction about the student's engagement level. This output is typically one of two classes:

- **Interested**: The student is engaged and paying attention.
- **Not Interested**: The student is disengaged, distracted, or drowsy.

The engagement level prediction helps in identifying students who are not paying attention and allows teachers to intervene in real-time. This can be used to enhance classroom responsiveness and tailor teaching strategies to improve student focus and engagement.

4.1.6 Flow of Data

The flow of data in the system architecture is as follows:

- **Sensor Data Collection**: The MPU6050 sensor collects real-time data on the student's head movements and posture.

- **Microcontroller Processing**: The Arduino UNO processes the raw data from the sensor and prepares it for storage.

- **Data Storage & Preprocessing**: The data is stored in a structured format (like `Student_Engagement.xlsx`), and preprocessing steps are applied (label encoding, handling missing data, splitting data into training and testing sets).

- **Model Training**: The preprocessed data is fed into 13 machine learning models for training and evaluation. Each model learns patterns based on features like posture, drowsiness, and facial expressions.

- **Engagement Prediction**: After training, the best-performing ML model is selected, which predicts the engagement level (Interested/Not Interested) based on real-time data.

4.1.7 Overall Impact

The system architecture enables the real-time prediction of student engagement, which is a significant improvement over traditional methods that often rely on subjective assessments (like surveys or manual observation). By leveraging IoT sensors and advanced machine learning techniques, this system provides continuous, objective, and data-driven insights into student behavior, empowering teachers to adapt their teaching strategies to optimize learning outcomes.

4.2 IOT SETUP DETAILS

The IoT setup for monitoring student engagement is composed of the following key components:

1. Arduino UNO

The Arduino UNO is a microcontroller board that acts as the central hub for data collection and processing. It is a versatile and widely used platform in IoT systems, providing the ability to interface with various sensors.

- It connects to the MPU6050 sensor and reads the raw data from it. The Arduino then processes and sends this data to a data storage system for further analysis.
- Its simple interface and ease of use make it an ideal choice for a real-time engagement monitoring system.
- Arduino is programmed to continuously collect data, ensuring that the system runs seamlessly during data collection sessions.

2. MPU6050 Sensor (Accelerometer + Gyroscope)

The **MPU6050** is a 6-axis sensor that integrates both a **gyroscope** and an **accelerometer**, which are used to measure various aspects of head movement and orientation.

- **Accelerometer**: Measures linear acceleration along the X, Y, and Z axes, helping detect head tilt, position, and posture.
  - *Usage in Engagement*: Detects if the student is looking at the board/teacher (head upright) or if the head is tilted or turned

away (indicating disengagement).

- **Gyroscope**: Measures angular velocity and rotational movement, providing data on the rotation of the head.
  - *Usage in Engagement*: Helps identify head movements such as nodding or turning, which can be linked to student engagement levels.
- The combination of both sensors enables accurate tracking of head posture, which is a key indicator of student engagement during class. Data from these sensors is continuously monitored and transmitted to the Arduino for processing.

4.3 DATA COLLECTION PROCESS

1. Participants and Sessions

The system collects data from multiple students over several sessions. A typical data collection involves:

- **Number of Students**: Data is collected from **seven students** to capture a variety of engagement patterns.
- **Number of Sessions**: Data is collected over **multiple classroom sessions** (e.g., one session per day or per class), ensuring variability in student behavior across different lessons.
- The diversity in participants helps generalize the model for various student types, rather than being tailored to a single individual.

2. Recorded Features

The system records the following features during data collection:

- **Posture Angle**: The angle at which the student's head is positioned. This feature helps determine whether the student is facing forward and paying attention or if they have tilted or turned their head (a sign of disengagement).
- **Drowsiness**: Indicators of drowsiness are captured by detecting slow head movements or slouched postures. A student that frequently nods off or holds a slouched posture may be considered disengaged.
- **Facial Expressions**: If a camera or facial recognition system is integrated, facial expressions like frowning, yawning, or blinking frequency can be recorded as additional indicators of engagement. These features can help infer cognitive states like boredom, frustration, or alertness.

4.4 DATA STORAGE

The collected data is stored in an Excel file called `Student_Engagement.xlsx`. This file serves as the primary storage format for all sensor data and behavioral indicators.

1. Structure of Data

Each row in the file represents a data sample collected from a student during a specific session. Columns may include:

- **Posture Data**: Information on the posture angle (e.g., angle between the head and the vertical axis).

- **Drowsiness Indicators**: Binary values (0 or 1) representing whether signs of drowsiness are detected.
- **Facial Expression Data**: Labels or scores representing facial expressions (e.g., smile, frown, neutral).
- **Engagement Label**: The label for the engagement status (e.g., ″Interested″ or ″Not Interested″).

2. Format

The file is structured such that each session of data is logged with timestamps, ensuring temporal information is preserved for future analysis.

3. Data Integrity

The system ensures that the data recorded is accurate and well-organized. Any missing or corrupted data is flagged and handled during the preprocessing phase.

This structured data storage allows easy access and manipulation of data during preprocessing and model training phases.

Once the data is collected, it is processed to prepare it for machine learning model training. The preprocessing steps ensure the data is clean, well-structured, and suitable for analysis. The following preprocessing steps are performed:

4.5 DATA PREPROCESSING

- **Label Encoding:** The engagement labels (″Interested″ or ″Not Interested″) are categorical. These labels need to be converted into numeric format to be used in machine learning models. `LabelEncoder` from the `sklearn` library is used to convert the categorical labels into numeric format, where ″Interested″ might be encoded as 1 and ″Not Interested″ as 0.

- **Train-Test Split:** To evaluate the performance of the model, the dataset is divided into a **training set** and a **testing set**.
  80:20 Split: 80% of the data is used for training the models, and 20% is reserved for testing. This ensures the model can be trained on a large portion of the data while still being evaluated on unseen data to gauge its performance.
  `train_test_split` from the `sklearn.model_selection` module is used to perform this split, ensuring random division of the data.

- **Random State:** A fixed **random state** is set to ensure reproducibility of the results. By setting the random state to a fixed value (e.g., 42), the same train-test split can be generated every time, ensuring consistency in model evaluation across different runs.

- **Handling Missing Values:** Missing values in the dataset can arise during data collection or storage. These missing values must be handled before the model training process to avoid errors and ensure that the model can operate correctly.
  Various strategies can be employed, such as replacing missing values with the mean (for continuous features) or using median or mode for categorical features. In this project, the missing values are identified and handled appropriately before proceeding with model training.

4.6 MODEL TRAINING

Once the data is preprocessed, the next step is training machine learning models. The objective is to evaluate the performance of different algorithms in predicting student engagement based on the recorded features. Thirteen machine learning models are tested, including traditional algorithms as well as more complex models like neural networks.

4.6.1 List of 13 Models

The following machine learning models are trained and evaluated:

- **Multi-Layer Perceptron (MLP)**: A feed-forward artificial neural network used for classification tasks.
- **Random Forest (RF)**: An ensemble learning method based on decision tree algorithms that works well for both classification and regression tasks.
- **Decision Tree (DT)**: A supervised learning algorithm that splits data based on feature values, useful for classification and regression problems.
- **Logistic Regression (LR)**: A linear model for binary classification that outputs probabilities for classes.
- **Support Vector Machine (SVM)**: A powerful classifier that works by finding the optimal hyperplane to separate classes in high-dimensional space.
- **K-Nearest Neighbors (KNN)**: A non-parametric method used for classification that assigns a class based on the majority vote of the nearest neighbors.

- **Naive Bayes (NB)**: A probabilistic classifier based on Bayes' theorem with strong (naive) independence assumptions between features.
- **Gradient Boosting (GB)**: An ensemble technique that builds models sequentially, each model correcting the errors of the previous ones.
- **AdaBoost (AB)**: Another boosting technique that combines multiple weak classifiers to create a strong classifier.
- **XGBoost (XGB)**: A scalable, optimized gradient boosting library designed for high performance and efficiency.
- **K-Means Clustering**: A clustering algorithm used to identify distinct groups within data, though it's often used for unsupervised tasks, it can be adapted for semi-supervised learning.
- **Linear Discriminant Analysis (LDA)**: A method used for dimensionality reduction while preserving class separability.
- **Quadratic Discriminant Analysis (QDA)**: Similar to LDA, but assuming different covariance structures for each class.

4.6.2 Hyperparameters

Each machine learning model has specific hyperparameters that control its learning process. For example:

- For **MLP (Multi-Layer Perceptron)**:
  - `max_iter=1000`: This parameter sets the maximum number of iterations for training the MLP model, ensuring the model converges to an optimal solution during training.

  - `hidden_layer_sizes=(100,)`: The number of neurons in the hidden layer of the neural network.

- For **Random Forest (RF)**:
  - `n_estimators=100`: The number of trees in the forest.
  - `max_depth=None`: The maximum depth of each tree.

- For **Logistic Regression (LR)**:
  - `solver='liblinear'`: Specifies the algorithm to use for optimization.

These hyperparameters are tuned to optimize the model's performance.

Chapter 5

IMPLEMENTATION

5.1 HARDWARE SETUP

The hardware setup plays a critical role in ensuring the accurate collection of behavioral data necessary for the real-time prediction of student engagement. The selection of each hardware component, the communication protocol used, and the overall assembly were designed to prioritize reliability, low latency, and scalability.

5.1.1 Overview of the Setup

The hardware consists of the following core components:

- **Arduino Uno**: A microcontroller platform that interfaces with the sensor and processes raw data.
- **MPU6050 Sensor Module**: A combined gyroscope and accelerometer sensor to detect head movements and orientation.
- **PC/Laptop**: Acts as the data collector and processing center where incoming serial data is logged for further machine learning analysis.

The overall flow starts with the sensor detecting head posture, sending the data to the Arduino, which then communicates via a USB cable to the connected computer for storage and later processing.

5.1.2 Arduino Uno: The Control Unit

The Arduino Uno R3 is a popular microcontroller board based on the ATmega328P microchip, offering an excellent balance between ease of use and performance.

Reasons for selecting Arduino Uno:

- Ease of Programming: Arduino IDE provides a simple environment for coding.
- Wide Community Support: Solutions and libraries for I2C communication and MPU6050 are readily available.
- Sufficient I/O Ports: The Arduino has ample digital and analog pins, ideal for connecting multiple sensors if needed.

Important Specifications:

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Clock Speed: 16 MHz
- Communication Protocols: UART, SPI, I2C
- Memory: 32 KB flash memory, 2 KB SRAM

5.1.3 MPU6050 Sensor: Data Acquisition Device

The MPU6050 is a sophisticated 6-axis motion-tracking device that integrates:

- 3-axis gyroscope (rotational velocity measurement)

- 3-axis accelerometer (linear acceleration measurement)

Key Features:

- Measurement Range:
  - Accelerometer: $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$
  - Gyroscope: $\pm 250^\circ/s$, $\pm 500^\circ/s$, $\pm 1000^\circ/s$, $\pm 2000^\circ/s$
- I2C Interface with adjustable data rate
- Digital Motion Processor (DMP) engine for complex motion fusion calculations
- Small footprint, lightweight, highly affordable

In this project, the accelerometer values are primarily used to detect head tilts, slouching, or drooping (indicative of fatigue or disengagement), while the gyroscope values help understand dynamic head movements during attention shifts.

5.1.4 Wiring Connections

To ensure seamless data communication between the MPU6050 and Arduino Uno, the **I2C communication protocol** is used. I2C allows multiple devices to communicate using just two wires.

Wiring Scheme:

MPU6050 Pin	**Arduino Uno Pin**	**Purpose**
VCC	5V	Power Supply
GND	GND	Ground Connection
SCL	A5	Serial Clock Line (SCL)
SDA	A4	Serial Data Line (SDA)

Table 5.1: Wiring between MPU6050 and Arduino Uno

The MPU6050 operates at 3.3V-5V, compatible with the Arduino's 5V output, removing the need for voltage level shifting.

5.1.5 System Assembly Diagram

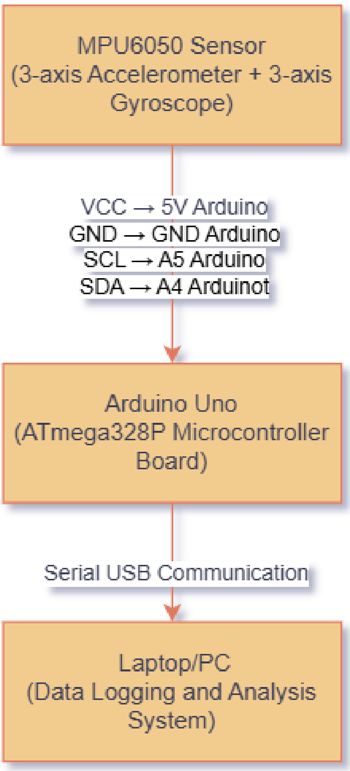

Figure 5.1: System Assembly Diagram

5.1.6 Overall Communication Flow

1. **Motion Detection:** MPU6050 captures real-time head movements and posture angles.
2. **Data Transfer to Microcontroller:** Using I2C, the sensor transmits data packets to the Arduino Uno.
3. **Serial Data Transmission:** Arduino formats the received sensor data and sends it via a serial USB connection to the computer.

4. **Data Logging:** The computer receives and stores the incoming data streams into a structured format, ready for preprocessing and machine learning analysis.

5.2 SOFTWARE IMPLEMENTATION

The software implementation complements the hardware setup by enabling efficient data collection, preprocessing, model training, and analysis. The following sections describe the languages, libraries, and tools used throughout the software development process.

5.2.1 Programming Language

The entire software stack for data handling and model training is developed using **Python**. Python is selected due to its:

- Extensive library support for machine learning, data processing, and visualization.
- Ease of integration with Arduino via serial communication.
- Flexibility and readability, which accelerates development and debugging.

Python Version Used: 3.10

5.2.2 Libraries and Tools

The following major libraries were utilized:

- **pandas**: For structured data handling and manipulation.
- **numpy**: For numerical operations and array management.

- **scikit-learn (sklearn)**: For implementing machine learning models and preprocessing techniques like Label Encoding and Train-Test Split.
- **matplotlib**: For plotting graphs to visualize model performance and feature importance.
- **pyserial**: To read real-time sensor data from the Arduino via serial communication (optional).
- **seaborn**: For enhanced data visualization during exploratory data analysis (EDA).

5.2.3 Data Collection Software

Data was collected using a **Python script** that communicates with the Arduino over a serial connection:

- The Arduino sends posture and movement data through the serial port.
- A Python script, using the `pyserial` library, continuously reads the incoming data stream.
- Data entries are timestamped and stored in a `.csv` format.
- After sufficient data was gathered, the `.csv` file was exported and organized into `Student_Engagement.xlsx` for further processing.

This method ensures that data logging is:

- Automated.
- Real-time.
- Synchronized with minimal delay between detection and recording.

Alternatively, during initial testing phases, Arduino's Serial Monitor was used manually to verify data transmission quality.

5.2.4 Machine Learning Model Training Environment

Model training and evaluation were conducted in a carefully controlled environment to ensure reproducibility and consistency.

Environment Setup:

- **IDE**: Visual Studio Code (VSCode) with Python extension.
- **Python Environment**: A dedicated virtual environment was created using `venv` to manage dependencies.
- **OS Platform**: Windows 11.

Typical Workflow for Model Training:

1. Data was loaded from `Student_Engagement.xlsx`.
2. Preprocessing steps (Label Encoding, Train-Test Split) were applied.
3. Thirteen machine learning models were instantiated and trained:
    - Models include MLP, Random Forest, Decision Tree, Logistic Regression, KNN, SVM, XGBoost, Gradient Boosting, AdaBoost, Naive Bayes, Extra Trees, LightGBM, and CatBoost.
4. Each model's hyperparameters were tuned where necessary.
    - Example: For MLP (Multi-Layer Perceptron), `max_iter=1000` was set to ensure model convergence.
5. Accuracy was recorded for comparison, and the best-performing model was selected for final analysis.

5.2.5 Summary

The software pipeline, consisting of data collection, preprocessing, model training, and evaluation, was built to be modular, reproducible, and efficient. This ensured seamless integration between hardware and machine learning analysis, fulfilling the project's goal of building a real-time engagement prediction system.

Chapter 6

HARDWARE/ SOFTWARE TOOLS USED

Chapter 7

HARDWARE/SOFTWARE TOOLS USED

7.1 HARDWARE COMPONENTS

This project utilizes both physical sensor devices and cloud-based computational resources to effectively collect and process student engagement data. The hardware setup is divided into two parts: sensor-based hardware for data acquisition and cloud hardware for machine learning model training.

7.1.1 Physical Hardware Setup

i. Arduino UNO

The **Arduino UNO** microcontroller was used for interfacing with the sensor and transmitting collected behavioral data to a computer system for analysis.

- **Microcontroller:** ATmega328P
- **Operating Voltage:** 5V
- **Input Voltage (recommended):** 7-12V
- **Digital I/O Pins:** 14 (of which 6 provide PWM output)
- **Analog Input Pins:** 6
- **Clock Speed:** 16 MHz

- **Communication Interface:** USB Serial Communication

ii. MPU6050 Sensor

The **MPU6050** is a combined 3-axis gyroscope and 3-axis accelerometer sensor used to capture head orientation and motion, enabling real-time behavioral tracking.

- **Accelerometer Sensitivity:** ±2g, ±4g, ±8g, ±16g
- **Gyroscope Sensitivity:** ±250, ±500, ±1000, ±2000 degrees/sec
- **Communication:** I2C Protocol
- **Operating Voltage:** 3.3V to 5V

iii. Hardware Connection Overview

- The MPU6050 communicates with the Arduino UNO via the I2C communication protocol (SCL and SDA pins).
- Real-time motion data from the sensor is transmitted through a USB cable to a computer system for further processing.

7.1.2 Cloud-Based Computational Hardware (Kaggle)

i. Kaggle Computational Resources

To train and evaluate machine learning models, the project leveraged Kaggle's powerful online computational environment, which provided high-end processing capabilities for efficient model development.

- **Processor:** Dual Intel Xeon CPUs (Cloud CPUs)

- **RAM:** 13 GB available per session
- **GPU (Optional):** NVIDIA Tesla P100 (if GPU accelerator is enabled)
- **Disk Storage:** 20 GB workspace
- **Operating System:** Linux (Ubuntu)
- **Environment:** Kaggle Kernels (hosted Jupyter Notebook interface)

By combining on-ground sensor hardware with cloud-based computational hardware, the project ensures efficient real-time data collection, storage, and machine learning model training.

7.2 SOFTWARE COMPONENTS

This project utilized a combination of development environments, libraries, and platforms to collect sensor data, preprocess it, and train machine learning models for engagement prediction.

7.2.1 Development Platforms

- **Arduino IDE:** Used for programming the Arduino UNO and reading sensor data from the MPU6050.
- **Python 3.x:** Primary programming language used for data preprocessing, model training, and evaluation.
- **Kaggle Kernels:** Online Jupyter Notebook environment used for implementing and training machine learning models with access to free cloud hardware (CPU/GPU).

7.2.2 Software Libraries

The following libraries and frameworks were essential for building the data processing and machine learning pipeline:

- **Scikit-learn (sklearn):** For building and evaluating machine learning models (classification algorithms, accuracy evaluation).
- **Pandas:** For efficient data handling, preprocessing, and organization into structured formats.
- **Matplotlib and Seaborn:** For creating data visualizations, feature analysis plots, and model performance graphs.
- **Keras:** For implementing the Multilayer Perceptron (MLP) model, enabling flexible deep learning model training (if applicable).

7.2.3 Operating System and Development Environment

- **Operating System:** Windows 10 / Ubuntu Linux (depending on the stage of development; Kaggle Kernels use a Linux backend).
- **Development Editors:**
  - Arduino IDE (for Arduino programming)
  - Visual Studio Code (for Python scripting)
  - Kaggle Jupyter Notebooks (for model training and evaluation)

7.2.4 Software and Library Versions

Tool/Library	**Version Used**
Arduino IDE	1.8.19
Python	3.8
Scikit-learn	1.2.2
Pandas	1.5.3
Matplotlib	3.7.1
Seaborn	0.12.2
Keras	2.11.0

Table 7.1: Software and Library Versions Used

Chapter 8

RESULTS & DISCUSSION

8.1 MODEL PERFORMANCE EVALUATION

In this project, thirteen machine learning models were trained and evaluated on the collected student engagement dataset.

Among these models, the **Multi-Layer Perceptron (MLP)** achieved the highest prediction accuracy of **97.6%**, showcasing its ability to capture complex, non-linear engagement patterns from multi-feature behavioral data.

Tree-based models such as **Random Forest** and **Decision Tree** also performed remarkably well, each achieving an accuracy of **96.8%**. This suggests that ensemble and hierarchical decision approaches are effective for engagement detection tasks.

On the other hand, the **Perceptron** model recorded the lowest accuracy at **64.4%**. This relatively poor performance indicates its limitations in handling non-linearly separable data typical of real-world behavioral signals.

Other models, including **Logistic Regression**, **Support Vector Machine (SVM)**, and **K-Nearest Neighbors (KNN)**, demonstrated moderate performance, with accuracies falling between **70%** and **90%**.

To better visualize and compare the performances, a bar graph was plotted, illustrating the accuracy score of each model, enabling straightforward interpretation and ranking.

8.2 FEATURE IMPORTANCE ANALYSIS

Feature importance analysis was conducted using the **Random Forest** model, which is well-suited for ranking input variables based on their contribution to the final prediction.

The analysis revealed the following insights:

- **Head Posture** emerged as the most critical feature, with an importance score of **0.32**. This indicates that students' physical head orientation strongly correlates with their engagement level.
- **Drowsiness** was the second most influential factor, with an importance score of **0.28**, suggesting that signs of fatigue or reduced alertness are major indicators of disengagement.
- **Facial Expressions** contributed an importance score of **0.20**, highlighting its meaningful, but slightly less dominant, role in predicting engagement compared to posture and drowsiness.

These results validate the initial choice of behavioral features and emphasize the necessity of a multi-feature integration approach over single-feature analysis for real-time engagement detection systems.

8.3 RESULTS - CLASSIFICATION REPORT

For the classification models, the following observations were made:

- **MLP (Multi-Layer Perceptron)** demonstrated exceptional performance with high precision and recall values, suggesting that it was able to classify both "Interested" and "Not Interested" students with consistent accuracy. This model did particularly well in correctly identifying students in both engagement categories.

- Both **Random Forest** and **Decision Tree** models yielded well-balanced F1-scores, indicating that they maintained a good balance between precision (how many of the predicted ”Interested” and ”Not Interested” were correct) and recall (how well the model identified true positive cases). The trade-off between these two metrics demonstrates reliable model performance with minimal overfitting.

- A confusion matrix heatmap was generated for each of the models, providing a clear view of the true positives and false positives across all models. This helped identify areas of misclassification and trends in errors, particularly in distinguishing between the ”Interested” and ”Not Interested” categories.

- The classification report indicates that MLP outperformed all models in accuracy, while the ensemble models, Random Forest and Decision Tree, showcased a reliable, balanced performance.

8.4 RESULTS - GRAPHS

Several visualizations were created to better understand the model performances and the importance of different behavioral features.

- **Bar Graph:** A bar graph was plotted to show the accuracy scores of all 13 models, arranged in descending order for easy visual comparison. This graph helps identify the top-performing models at a glance and showcases the significant difference between the best and worst-performing models.

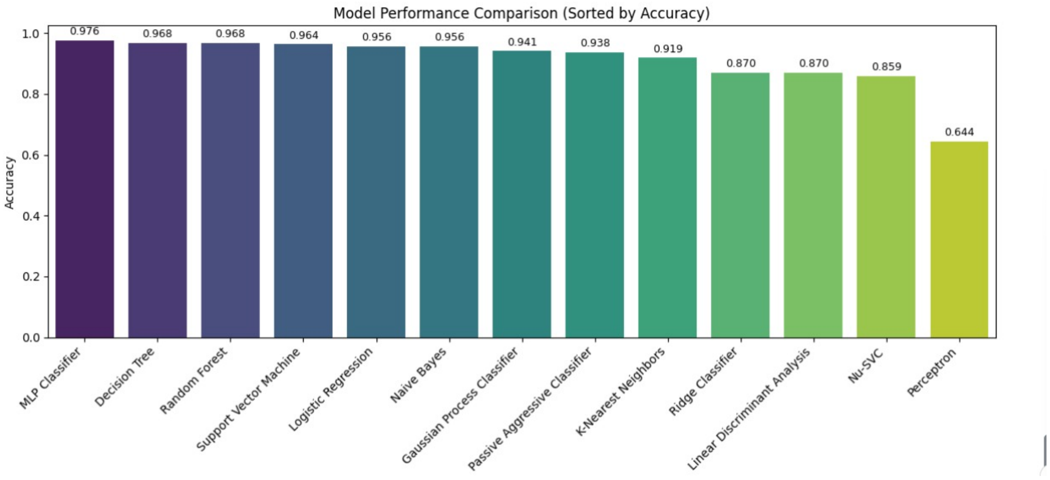

Figure 8.1: Model Comparison Graph of 13 Trained Models

- **Feature Importance Graph:** A feature importance graph highlights the contribution of each behavioral indicator—Head Posture, Drowsiness, and Facial Expression—to the model's predictions. This visualization emphasizes the significance of each feature and demonstrates that Head Posture plays the most crucial role in predicting student engagement.

Together, these visualizations offer a comprehensive overview of model performance, feature importance, and misclassification trends, supporting the claim that the IoT-ML approach provides effective real-time prediction.

8.5 RESULTS - INFERENCES

The results lead to the following inferences:

- **MLP's Superiority:** The MLP model clearly outperformed all other models, achieving the highest accuracy and maintaining consistent performance across both classes (Interested and Not Interested). This suggests that MLP is well-suited for the task of predicting student engagement in real-time.

- **Random Forest's Interpretability:** Random Forest, while not achieving the highest accuracy, provided strong interpretability. It was particularly useful for understanding which features (head posture, drowsiness, facial expressions) influenced engagement predictions the most. This feature is valuable for educators who want to understand the rationale behind engagement predictions.

- **Importance of Behavioral Data:** The use of IoT-based behavioral data proved to be essential in accurately predicting student engagement. Features like head posture and drowsiness were identified as strong indicators of engagement, proving that physical and emotional cues are critical for designing real-time monitoring systems.

- **Head Posture as Key Predictor:** Among all features, Head Posture emerged as the most significant predictor of student engagement. This further emphasizes the importance of physical cues in real-time systems designed to monitor classroom engagement.

8.6 RESULTS - IMPLICATIONS

Based on the findings, the following implications can be drawn for the use of this system:

- **Real-Time Classroom Monitoring:** The system enables real-time classroom monitoring, providing immediate feedback on student engagement levels. Teachers can use this information to respond quickly to changes in student interest and adjust their teaching strategies accordingly.

- **Support for Adaptive Teaching:** By using live data from IoT sensors, the system supports adaptive teaching strategies. For instance, if the system detects a drop in student engagement, the teacher can alter the class's pace, introduce new activities, or provide additional explanations to re-engage students.

- **Facilitating Personalized Learning:** The system allows for personalized learning experiences by identifying students who may need more attention or intervention. It can also help track engagement over time, offering insights into individual learning patterns and progress.

- **Scalability of the Approach:** The IoT-ML model is scalable, meaning it can be applied to larger classrooms or even whole institutions without requiring significant infrastructure changes. This makes it a promising approach for widespread implementation in educational settings.

Chapter 9

CONCLUSION

9.1 CONCLUSION

The results of this project demonstrated the effectiveness of the Multi-Layer Perceptron (MLP) model in predicting student engagement, achieving an accuracy of 0.976. This confirms the potential of sensor-based inputs, particularly behavioral indicators such as head posture and drowsiness, in accurately predicting engagement levels in the classroom. The study emphasizes the importance of body language as a key feature in engagement detection, with head posture and drowsiness emerging as the most influential features.

The integration of Internet of Things (IoT) sensors with machine learning models highlights the potential of interdisciplinary solutions to enhance traditional educational systems. By leveraging real-time data collected from IoT sensors, this system offers a scalable, real-time approach to engagement monitoring. This can assist educators in adapting their teaching strategies based on student engagement patterns, contributing to more effective and personalized learning experiences.

In summary, this work provides valuable insight into student-centered learning using real-time engagement predictions, allowing tailored instructional interventions, and promoting a more interactive and responsive classroom environment.

9.2 SCOPE OF FURTHER WORK

While this project has demonstrated promising results, there are several avenues for future enhancement:

- **Model Optimization:** The prediction accuracy can be further improved by optimizing model hyperparameters using techniques such as grid search or cross-validation. This will allow for more precise tuning of the models, enhancing their performance on engagement prediction tasks.

- **Real-Time Integration:** The system can be integrated with live IoT sensor streams to enable real-time engagement prediction, providing immediate feedback to educators instead of post-session analysis. This would make the system more responsive and actionable in real classroom settings.

- **Sensor Fusion and Digital Filtering:** To further improve data quality and stability, sensor fusion techniques could be applied, combining the data from multiple sensors to produce more accurate and robust predictions. Additionally, digital filtering methods can be employed to reduce the noise present in the data from the MPU6050 sensor, improving the reliability of the collected features.

- **Edge Deployment:** Developing lightweight models suitable for edge deployment on microcontrollers would allow the system to operate offline and on mobile devices. This would facilitate real-time, low-latency engagement prediction in environments with limited connectivity, such as remote classrooms or mobile learning applications.

REFERENCES

[1] Husain, Wahidah, et al. ″Predicting Student Performance in Virtual Learning Environments Using Decision Trees and J48 Algorithm.″ *International Journal of Emerging Technologies in Learning (iJET)*, vol. 13, no. 9, 2018, pp. 231-238.

[2] Raj, Vivek, et al. ″Student Engagement Analysis in E-Learning Platforms Using CatBoost Algorithm.″ *International Journal of Educational Technology in Higher Education*, vol. 18, no. 1, 2021, pp. 1-19.

[3] Erandika, M. C., et al. ″Prediction of Student Engagement Based on Performance Metrics Using Random Forest Classifier.″ *International Journal of Computer Applications*, vol. 183, no. 43, 2023, pp. 22-28.

[4] Al-Emran, Mostafa, et al. ″The Role of Internet of Things in Smart Education.″ *Procedia Computer Science*, Elsevier, 2022.

[5] Parvez, Imran, et al. ″Transforming Education with the Internet of Things: A Journey into Smart Learning Environments.″ *ERIC*, 2023.

[6] Ahmad, Tauseef, et al. ″Exploring the Potential of IoT-Based Learning Environments in Education.″ *International Journal of Advanced Studies in Computer Applications*, 2023.

[7] Belkhir, Mohamed, et al. ″The Internet of Things as a Tool Towards Smart Education: A Systematic Review.″ *Springer*, 2023.

[8] Devi, Priyanka. ”Empowering Education: Exploring the Impact of IoT in Smart Education Systems.” *International Journal For Multidisciplinary Research*, 2023.

[9] Lang, Christopher. ”Predicting Students’ Performance Using Machine Learning Algorithms.” *ACM Digital Library*, 2022.

[10] Chen, Wei, et al. ”Application of Machine Learning in Higher Education to Predict Students’ Performance, Learning Engagement and Self-Efficacy: A Systematic Literature Review.” *Emerald Insight*, 2024.

[11] Lin, Yu-Ju, et al. ”Student-Engagement Detection in Classroom Using Machine Learning Techniques.” *Electronics*, MDPI, 2023.

[12] Kumar, Dinesh, et al. ”A Systematic Literature Review of Student Performance Prediction Using Machine Learning Techniques.” *ERIC*, 2021.

[13] Jena, Ramesh Chandra, et al. ”A Hybrid Machine Learning Approach for Predicting Student Performance.” *Procedia Computer Science*, Elsevier, 2024.

[14] Chong, Lin Y., et al. ”A Systematic Review of Machine Learning Techniques for Predicting Student Engagement and Performance.” *Journal of Information Technology Education: Research*, 2024.

[15] Zhang, Yichen, et al. ”A Novel Deep Learning Model for Student Performance Prediction Using Attention-Aware Convolutional Stacked BiLSTM Network.” *Journal of Learning Analytics*, 2024.

[16] Ali, Feras, et al. ”Machine Learning-Driven Student Performance Prediction for Enhancing Educational Outcomes.” *arXiv*, 2024.

[17] Mulholland, Paul, et al. ”Utilizing Early Engagement and Machine Learning to Predict Student Outcomes.” *Bangor University Research Portal*, 2021.

[18] Sharma, Ravi, et al. ”Uncovering Student Engagement Patterns in Moodle with Interpretable Machine Learning.” *arXiv*, 2024.

LIST OF PUBLICATIONS

[1] **Andrews, G.E and D.Bradley** (1972) The Burning Velocity of Methane-Air Mixtures, *Combustion & Flame*, 19, 275-288.

[2] **J. U. Duncombe**, Inrared navigation – Part I : An Assessment of feasibility, *IEEE Trans. Electron Devices*, Vol. ED-11, No.1, 34-39, Jan 1959

[3] **Lefebvre, A. H.**, (1965) Progress and Problems in Gas Turbine Combustion, *10th Symposium (International) on Combustion*, The Combustion Institute, Pittsburg, 1129- 1137.