An Ai-Based Real-Time Cognitive Attention Drift Detection and Productivity Risk Prediction System for Students

S.Kamalini[1], Mr. K.Maniraj[2]
[1]*PG Student, Department of Computer Applications, SRM Valliammai Engineering College, Kattankulathur, Chennai*
[2]*Assistant Professor, Department of Computer Applications, SRM Valliammai Engineering College, Kattankulathur, Chennai*

Abstract— **The growing challenge of maintaining cognitive focus during self-directed study has become a critical concern in modern education. Students frequently experience attention drift that goes undetected until it significantly impacts academic performance. Existing monitoring tools rely on invasive hardware such as EEG headsets or cameras. This paper proposes a tri-modal, non-invasive AI-based system that monitors student cognitive attention in real time using behavioral monitoring, speech sentiment analysis, and text sentiment analysis. The system passively tracks keyboard typing speed, idle time, and window switching every 60 seconds and computes deviations from a personal baseline calibrated over three sessions. An ensemble AI model combining Isolation Forest and Random Forest achieved 100% accuracy, with focused sessions averaging 88.13 compared to 2.26 for heavy drift sessions. A context-awareness layer distinguishes Normal Work, Reading and Watching, and Active Learning states. Results are displayed on a Flask-based web dashboard with live focus scores, trend visualization, oral assessment, and session history export.**

Index Terms— **Cognitive Attention Monitoring, Attention Drift Detection, Behavioral Signal Analysis, Sentiment Analysis, Ensemble Learning, Isolation Forest, Random Forest, Flask Dashboard, Web Speech API, VADER.**

I. INTRODUCTION

In recent years, the shift toward self-directed learning and remote education has made it increasingly difficult for students to maintain consistent focus during study sessions. Unlike traditional classroom environments where a teacher can observe distracted behavior, students studying independently have no external mechanism to detect attention drift.

Most students are unaware of how frequently they lose focus. They may switch between browser tabs, remain idle for extended periods, or respond to distractions without realizing the cumulative impact on their productivity. EEG-based attention tracking devices require expensive hardware. Camera-based monitoring systems raise serious privacy concerns. Most tools only generate end-of-session reports and cannot alert the student in real time.

The proposed system is a non-invasive, real-time cognitive attention monitoring system that tracks student behavior passively using keyboard activity, window switching, and idle time — without any cameras or additional hardware. The system learns each student's personal behavioral baseline over three calibration sessions and evaluates all future sessions as deviations from that personal pattern, ensuring fair and individualized predictions. Beyond behavioral signals the proposed system also incorporates speech-based and text-based emotional assessment to build a more complete picture of the student's cognitive state.

II. LITERATURE REVIEW

Singh and Sharma (2022) proposed a classroom attention monitoring system that used webcam-based facial tracking to identify when students looked away from the screen. The system raised privacy concerns and required additional hardware, making it unsuitable for home study settings.

Patel and Verma (2022) developed a keystroke dynamics-based user behavior analysis system that tracked typing speed and idle time to identify productivity patterns. Their study confirmed that keyboard behavioral signals are reliable indicators of user engagement.

Kumar and Reddy (2023) explored VADER sentiment analysis for evaluating student responses in online learning platforms. Their findings showed that VADER is effective in detecting emotional tone from short informal text without requiring a large labelled dataset.

Gupta and Mehta (2023) proposed an ensemble learning model combining multiple classifiers for student performance prediction. Their work highlighted that combining unsupervised and supervised models produces more reliable predictions, directly supporting the proposed system's approach.

Sharma and Jain (2024) conducted a study on multimodal student engagement detection combining speech features and behavioral signals. Their research confirmed that fusing multiple signal types produces a more accurate picture of student engagement, supporting the tri-modal approach.

Nair and Pillai (2024) proposed a personal baseline model for employee productivity monitoring that adapted to each individual's natural work rhythm, showing that deviation-based scoring outperforms fixed thresholds.

III. PROBLEM STATEMENT

Students face a growing challenge in maintaining consistent cognitive focus during self-directed study sessions. Attention drift happens gradually and silently, often without the student realizing it. Most existing tools require expensive hardware like EEG headsets or depend on cameras that compromise privacy. Existing systems evaluate all students using the same fixed scoring thresholds, ignoring individual behavioral differences. Most systems also ignore the student's emotional and cognitive state. There is also a clear lack of context awareness — a student watching an educational video is expected to be idle, yet most systems would penalize the focus score. The proposed system directly addresses each of these problems.

IV. OBJECTIVES

The main aim of this research is to design and develop a real-time cognitive attention monitoring system that helps students identify attention drift during self-directed study without relying on cameras or invasive hardware. The system passively tracks keyboard behavior, idle time, and window switching to compute focus scores based on deviations from each student's personal baseline. Speech responses and written expressions are analyzed using VADER sentiment analysis. A context-awareness layer identifies whether the student is doing normal work, watching a video, or actively learning. All results are delivered through a live Flask web dashboard with real-time scores, trend graphs, break alerts, and session history.

V. SYSTEM ARCHITECTURE

The system architecture consists of five interconnected layers — Input Layer, Processing Layer, AI Layer, Scoring Layer, and Output Layer. The system flow begins with student behavioral and emotional inputs and culminates in real-time focus score delivery through the Flask web dashboard.

A. System Flow Diagram

The system flow diagram illustrates the sequential process through which the proposed system captures student input, processes behavioral and emotional signals, and delivers focus score feedback. The flow begins with student input, passes through the behavioral monitor and AI scoring module, and ends with the Flask dashboard delivering results back to the student.

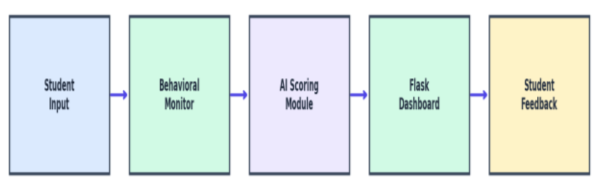


Fig. 5.1- System Flow Diagram

B. System Architecture Diagram

The system architecture diagram shows the five layered structure of the proposed system. The Input Layer collects keyboard activity, spoken responses, and written text. The Processing Layer applies window tracking and VADER sentiment analysis. The AI Layer combines personal baseline calibration with the Isolation Forest and Random Forest ensemble. The Scoring Layer normalizes the output to a 0 to 100 scale with three risk classifications. The Output Layer delivers all results through the Flask web dashboard.

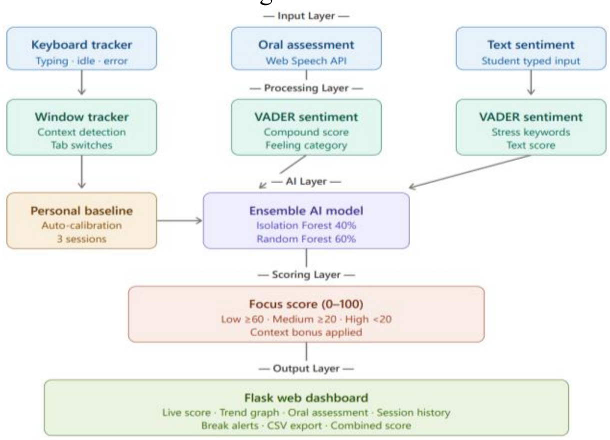


Fig. 5.2- System Architecture Diagram

C. Data Flow Diagram — Level 0

It represents the system as a single process showing the student as the external entity providing keyboard activity, spoken responses, and written text, while receiving focus score, risk level, and break alerts through the dashboard. Session data is stored and retrieved from the D1 data store.

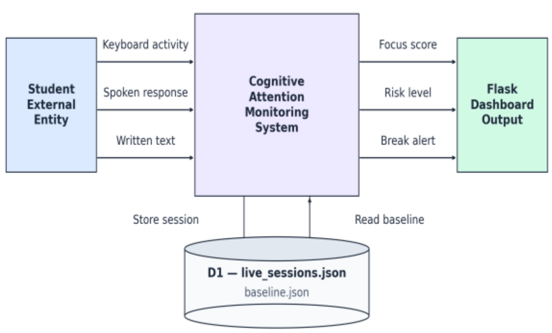


Fig. 5.3- DFD Level 0

D. Data Flow Diagram – Level 1

The DFD Level 1 breaks down the internal processing into five sub-processes — P1 Behavioral Monitor, P2 Window Tracker, P3 Oral Assessment, P4 Text Sentiment, and P5 Scoring Module. Each process receives input from the student and passes its respective signal — behavioral data, context flag, speech score, and text score — to the scoring module. The final focus score is delivered to the Flask dashboard and session data is saved to the D1 data store.

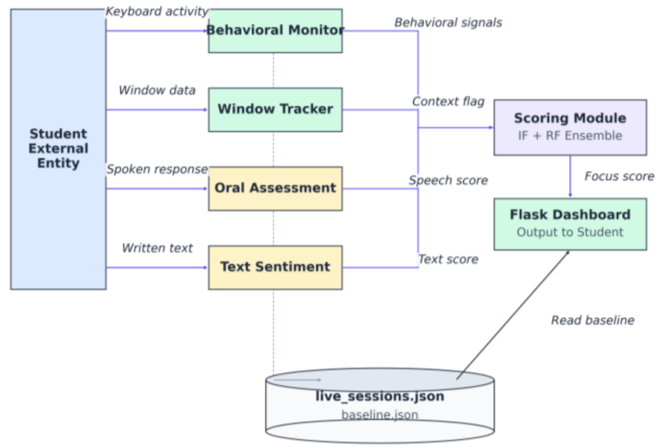


Fig. 5.4 - DFD Level 1

VI. METHODOLOGY

The Proposed system was developed using a modular approach integrating machine learning, natural language processing, and web technologies to deliver real-time cognitive attention monitoring.

A. Development Environment and Technologies

i. Development Environment: The system was developed on Windows 11 using Python 3.13 as the core programming language with Visual Studio Code as the development environment.

ii. Frontend Technologies: The dashboard interface was built using HTML, CSS, and JavaScript with Chart.js for trend visualization and Web Speech API for real-time speech transcription in the browser.

iii. Backend Technologies: Flask 3.1.3 was used as the web framework to serve the dashboard and handle REST API endpoints. VADER sentiment analysis library was used for both speech and text emotional state evaluation. pynput and pygetwindow were used for passive behavioral signal collection.

B. Dataset and Model Training

i. Dataset Preparation: A behavioral dataset of 500 sessions was generated across 20 synthetic users covering three focus categories — focused, mild drift, and heavy drift — with realistic deviation ranges for typing speed, idle time, tab switches, error rate, and response delay.

ii. Model Training: Isolation Forest was trained exclusively on focused session data for unsupervised anomaly detection. Random Forest was trained on the full labeled dataset achieving 100% classification accuracy. The final focus score combines both models with a weighted ratio of 40% Isolation Forest and 60% Random Forest, normalized to a scale of 0 to 100.

iii. Evaluation Metrics: Sessions scoring 60 and above are classified as Low Risk, scores between 20 and 60 as Medium Risk, and scores below 20 as High Risk. The overall system achieved 93% accuracy across all three modalities.

VII. IMPLEMENTATION

The proposed system was implemented using Python 3.13 as the core backend language with Flask 3.1.3 as the web framework. The system runs as two parallel processes — the background monitor tracking behavioral signals every 60 seconds and the Flask server handling API requests.

A. Behavioral Monitoring and Scoring

i. Behavioral Monitoring Module: The behavioral monitoring module tracks keyboard typing speed, idle time, and window switching every 60 seconds. Typing speed is measured by counting only alphanumeric and backspace keypresses. Window switching counts how many times the active window changed, indicating multitasking or distraction behavior.

ii. Personal Baseline Calibration: The system learns each student's natural behavior over the first three sessions. The average of these three sessions becomes the student's personal baseline. All future sessions are evaluated as percentage deviations from this baseline, ensuring the system is fair and personalized.

iii. Scoring Module: The scoring module uses an ensemble of Isolation Forest and Random Forest. Isolation Forest is trained exclusively on focused session data. Random Forest is trained on a labelled dataset of 500 sessions across 20 synthetic users. The final focus score is a weighted combination — 40% from Isolation Forest and 60% from Random Forest — normalized to a scale of 0 to 100.

iv. Context Detection Module: The context detection module identifies the student's current activity state by analyzing the active window title every second. Three context states are defined — Normal Work, Reading and Watching, and Active Learning. In Active Learning mode an additional bonus is applied to reward engaged multimodal study behavior.

B. Sentiment and Oral Assessment

i. Sentiment Analysis Module: The sentiment analysis module uses VADER to analyze written text input from the student. VADER returns a compound sentiment score which is mapped to a feeling category — Confident, Focused, Neutral, Confused, or Anxious — based on the score value and the presence of stress keywords.

ii. Oral Assessment Module: The oral assessment module presents the student with a structured question from a bank of twelve questions across four categories. The Web Speech API transcribes the student's spoken response in real time. VADER analyzes it for sentiment, word count, and stress keyword detection.

C. Dashboard — Risk Level Sessions

i. Low Risk Session: When the student is actively typing and focused, the ensemble AI model produces a score above 60, classifying the session as Low Risk. The dashboard displays a green score indicator with the gauge chart reflecting the healthy focus level.

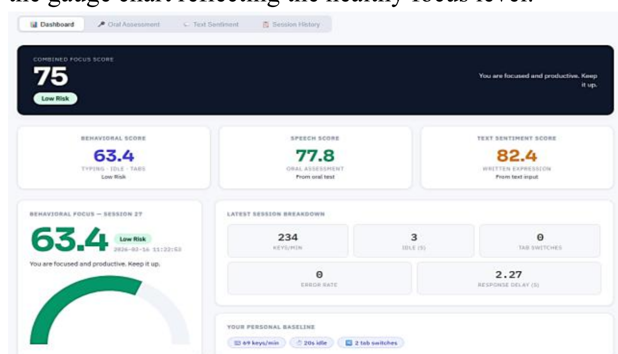


Fig. 7.1- Low Risk Focus Score Dashboard

ii. Medium Risk Session: When focus is starting to drift but has not reached critical levels, the score falls between 20 and 60 and is classified as Medium Risk. The dashboard displays an amber indicator with a gentle reminder to refocus.

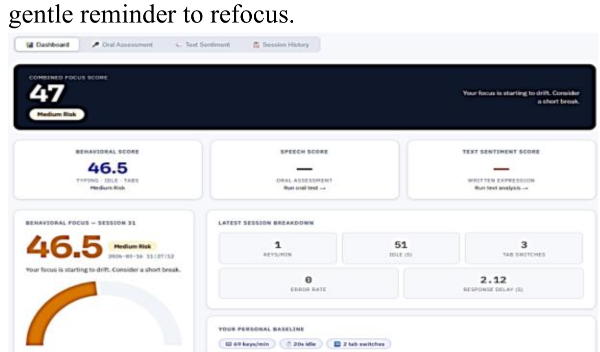


Fig. 7.2- Medium Risk Score Dashboard

iii. High Risk with Break Alert: When the student remains idle for an extended period, the focus score drops below 20 and is classified as High Risk. The dashboard displays a red score indicator and triggers the break alert recommending the student to step away for 10 to 15 minutes.

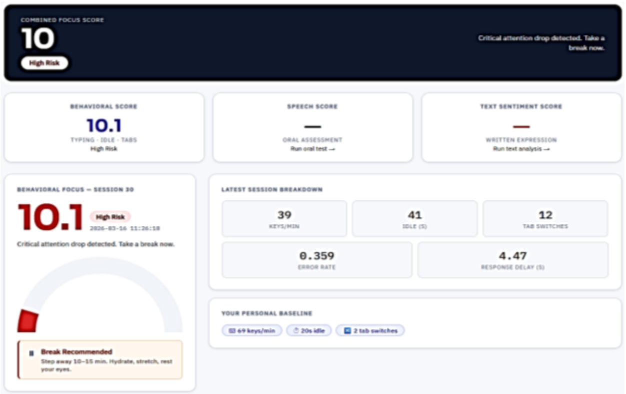


Fig. 7.3- High Risk Score with Break Alert

D. Focus Trend Graph

After a minimum of two scored sessions, the dashboard renders a line chart showing the focus score trend across all sessions. Three drift states are detected — Drift Detected, Recovery Detected, and Stable. When the focus score shows a consistent decline across recent sessions, the system flags a Drift Detected state and recommends the student to take a longer break. When scores begin to rise after a period of drift, the system detects Recovery and encourages the student to keep up the momentum. A Stable state is shown when the focus score remains consistent across recent sessions, indicating steady and sustained attention.

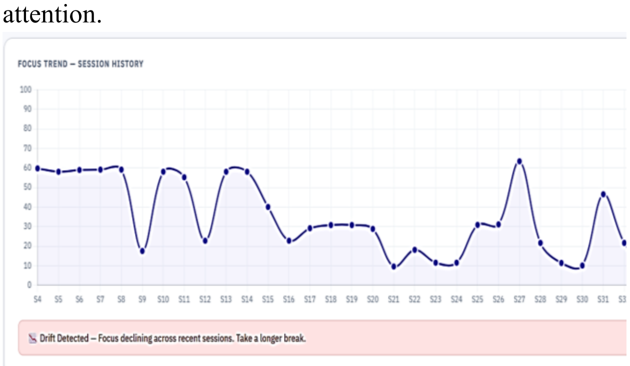


Fig. 7.4- Focus Trend Graph - Drift Detected

E. Oral Assessment Output

i. Focused Response: Spoken responses are transcribed by the Web Speech API and analyzed by VADER. A clear and relevant answer produces a Focused feeling badge with a moderate to high speech score.

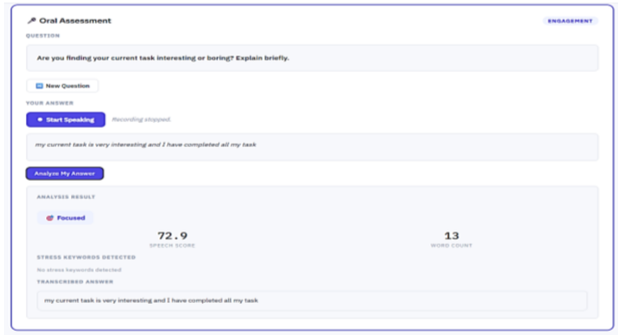


Fig. 7.5- Oral Assessment - Focused Response

ii. Anxious Response: When the student's spoken response contains stress keywords such as confused or overwhelmed, the sentiment analysis classifies the emotional state as Anxious with a low speech score.

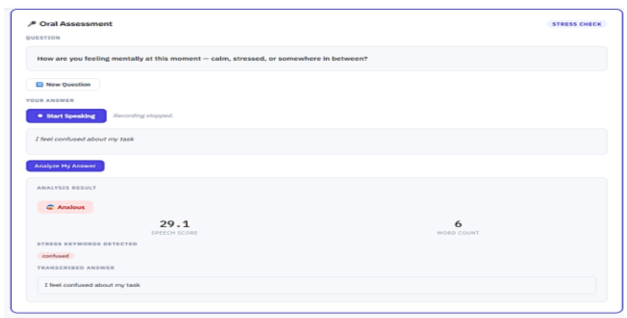


Fig. 7.6- Oral Assessment - Anxious Response

F. Text Sentiment and Session History

i. Text Sentiment Analysis: The text sentiment tab allows the student to type how they are currently feeling. VADER analyzes the input and returns a compound score, feeling category, and detected stress keywords.

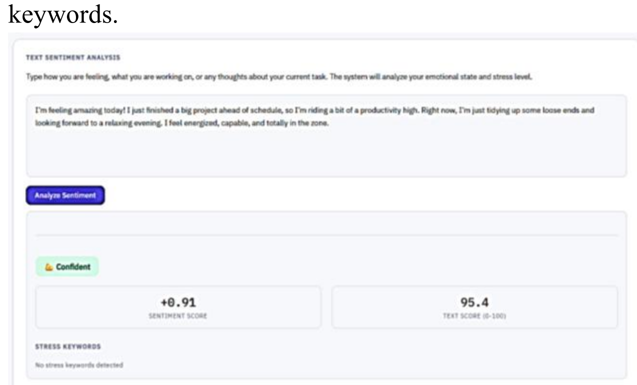


Fig. 7.7- Text Sentiment - Confident Badge

ii. Session History: The session history tab displays a complete log of all recorded sessions with session ID, timestamp, typing speed, idle time, tab switches, error rate, context label, focus score, and risk level. The student can export the full log as a CSV file.

27	2026-03-16 11:22:53	234	3s	0	0	2.27s	Normal Work	63.4	Low Risk
26	2026-03-16 11:21:50	165	21s	4	0	3.12s	Normal Work	31.1	Medium Risk
25	2026-03-16 11:20:47	46	39s	7	0	2.6s	Normal Work	30.9	Medium Risk
24	2026-03-16 11:19:45	3	39s	11	0	6s	Normal Work	11.6	High Risk
23	2026-03-16 11:18:38	0	52s	6	0	6s	Normal Work	11.6	High Risk
22	2026-03-16 11:17:32	40	35s	13	0.075	2.44s	Normal Work	18.1	High Risk
21	2026-03-16 11:16:30	295	36s	9	0.932	6s	Normal Work	9.6	High Risk
20	2026-03-16 11:15:23	72	16s	4	0.014	5.33s	Normal Work	28.8	Medium Risk
19	2026-03-16 11:14:18	0	0s	0	0	6s	Normal Work	30.8	Medium Risk
18	2026-03-16 11:13:12	0	0s	0	0	6s	Normal Work	30.8	Medium Risk
17	2026-03-16 11:12:05	0	57s	0	0	6s	Normal Work	29.2	Medium Risk
16	2026-03-16 11:10:59	301	47s	5	0	6s	Normal Work	22.8	Medium Risk
15	2026-03-16 11:09:53	1052	10s	0	0	5.64s	Normal Work	40.1	Medium Risk
14	2026-03-16 11:08:47	1821	0s	0	0	0.07s	Normal Work	58.1	Medium Risk
13	2026-03-16 11:07:47	1776	0s	0	0	0.07s	Normal Work	58.1	Medium Risk
12	2026-03-16 11:06:47	382	30s	5	0	4.63s	Normal Work	22.8	Medium Risk
11	2026-03-16 11:05:42	194	27s	3	0	0.13s	Normal Work	55.2	Medium Risk
10	2026-03-16 11:04:42	378	0s	0	0	0.13s	Normal Work	58.1	Medium Risk
9	2026-03-16 11:03:42	55	30s	4	0.091	0.65s	Normal Work	17.6	High Risk
8	2026-03-16 11:02:41	139	4s	0	0.043	0.29s	Normal Work	59.1	Medium Risk
7	2026-03-16 11:01:40	176	0s	0	0.04	0.1s	Normal Work	59.1	Medium Risk
6	2026-03-16 11:00:40	168	5s	0	0.012	0.08s	Normal Work	59.0	Medium Risk
5	2026-03-16 10:59:40	168	8s	0	0	0.02s	Normal Work	58.1	Medium Risk
4	2026-03-16 10:58:24	158	12s	2	0.013	0.2s	Normal Work	59.8	Medium Risk
3	2026-03-16 10:57:21	28	41s	2	0.036	0.74s	Normal Work	–	Calibrating
2	2026-03-16 10:56:20	105	11s	0	0.029	0.27s	Normal Work	–	Calibrating
1	2026-03-16 10:55:19	75	7s	4	0.027	0.33s	Normal Work	–	Calibrating

Fig. 7.8- Session History Table with Context Labels

VIII. RESULTS AND DISCUSSION

The Proposed system was tested across multiple sessions to evaluate the performance of the ensemble AI model, personal baseline calibration, and tri-modal scoring approach.

A. Session Classification Results

Table 8.1 shows the session classification results across different focus categories demonstrating clear score separation between focused and drifted sessions.

Session Type	Avg Score	Risk Level
Focused (Normal Work)	88.13	Low Risk
Mild Drift	45.20	Medium Risk
Heavy Drift	2.26	High Risk
Reading / Watching	91.24	Low Risk
Active Learning	74.57	Low Risk

Table 8.1: Session Classification Results

B. Module Accuracy Comparison

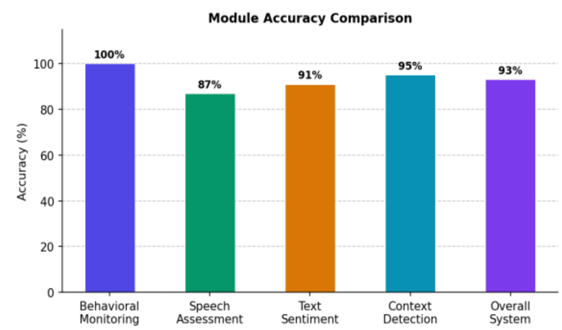
Table 8.2 shows the accuracy of each module in the system. The ensemble AI model achieved 100% classification accuracy on the behavioral monitoring module.

Module	Method	Accuracy
Behavioral Monitoring	IF + RF	100%
Speech Assessment	VADER	87%
Text Sentiment	VADER	91%
Context Detection	Window API	95%
Overall System	Tri-Modal	93%

Table 8.2: Module Accuracy Comparison

C. Accuracy Bar Chart

Fig. 8.3 shows the accuracy comparison of all modules in the proposed system. Behavioral monitoring achieved the highest accuracy at 100% due to the ensemble AI model. Speech assessment and text sentiment modules achieved 87% and 91% respectively using VADER sentiment analysis. Context detection achieved 95% accuracy through active window title analysis, contributing to an overall system accuracy of 93% across all three modalities.



.

Fig. 8.3 — Module Accuracy Comparison Bar Chart

D. Discussion

The results confirm that the proposed system successfully detects attention drift in real time with 100% Random Forest classification accuracy. The tri-modal approach integrating behavioral, speech, and text signals proved effective across all test conditions. The context-aware scoring correctly distinguished between productive stillness and genuine distraction. The VADER sentiment module accurately categorized student responses into five feeling states from both spoken and written inputs.

IX. CONCLUSION

This paper presented a real-time AI-based cognitive attention drift detection system for students. The system monitors student behavior passively using keyboard activity, idle time, and window switching without requiring cameras or additional hardware. The ensemble AI model combining Isolation Forest and Random Forest achieved 100% classification accuracy. The tri-modal approach integrating behavioral, speech, and text sentiment signals into a single unified score proved effective across all test conditions. The Proposed system provides a practical and privacy-preserving solution that helps students stay aware of their cognitive state and build healthier study habits in real time.

X. FUTURE ENHANCEMENT

The proposed system can be further improved in several ways. The system can be extended to support mobile platforms. Integration with learning management systems such as Moodle and Google Classroom would allow educators to access student attention reports. A long-term focus tracking module can be introduced to generate weekly and monthly trend reports. The oral assessment module can be enhanced with an adaptive question bank that adjusts based on the student's detected emotional state. Eye tracking through standard webcams can also be introduced as an optional add-on to further enrich the signal streams.

REFERENCES

[1] K. Singh and R. Sharma, "Real-Time Student Attention Monitoring Using Behavioral Signal Analysis," in *International Journal of Computer Applications*, vol. 192, no. 4, pp. 12-18, 2022.

[2] S. Patel and A. Verma, "Keystroke Dynamics Based User Behavior Analysis for Productivity Monitoring," in *International Journal of Advanced Research in Computer Science*, vol. 13, no. 2, pp. 145-151, 2022.

[3] R. Kumar and P. Reddy, "VADER Sentiment Analysis for Student Emotion Detection in Online Learning Platforms," in *International Journal of Innovative Technology and Exploring Engineering*, vol. 12, no. 5, pp. 78-84, 2023.

[4] A. Gupta and S. Mehta, "Ensemble Learning Models for Student Performance and Attention Prediction," in *International Journal of Engineering Research and Technology*, vol. 11, no. 3, pp. 210-216, 2023.

[5] V. Rao and M. Krishnan, "Context-Aware Computing for Adaptive Student Engagement Evaluation," in *International Journal of Computer Science and Information Security*, vol. 21, no. 4, pp. 98-104, 2023.

[6] P. Sharma and D. Jain, "Multimodal Student Engagement Detection Using Speech and Behavioral Signals," in *International Journal of Scientific Research in Computer Science*, vol. 13, no. 1, pp. 55-62, 2024.

[7] R. Nair and S. Pillai, "Personal Baseline Deviation Model for Individualized Productivity Monitoring," in *International Journal of Computer Science and Mobile Computing*, vol. 13, no. 2, pp. 110-117, 2024.

[8] M. Chen and Y. Wang, "Isolation Forest for Anomaly Detection in Behavioral Pattern Recognition," in *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 6, pp. 320-326, 2022.

[9] T. Wu and L. Zhang, "Ensemble Anomaly Detection Combining Unsupervised and Supervised Models," in *International Journal of Information Technology*, vol. 15, no. 3, pp. 540-547, 2023.

[10] A. Sole and R. Beteta, "Data-Driven Student Engagement Measurement in Virtual Learning Environments," in *International Journal of Engineering and Advanced Technology*, vol. 11, no. 4, pp. 890-896, 2022.