International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 05 Issue: 03 | March – 2026 DOI:10.55041/ISJEM05958
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Human-Independent AI-Based Productivity Enforcer

S. Rishi Vardhan[1], K. Chaithanya[2], G. Kishore[3], M. Vandana[4]

Supervisor**: Dr.M.V. Krishna Mohan**, Assistant Professor, Dept. of CSE, VIET

[1]*Department of CSE (AIML), Visakha Institute of Engineering and Technology, Andhra Pradesh, India*
[2]*Department of CSE (AIML), Visakha Institute of Engineering and Technology, Andhra Pradesh, India*
[3]*Department of CSE (AIML), Visakha Institute of Engineering and Technology, Andhra Pradesh, India*
[4]*Department of CSE (AIML), Visakha Institute of Engineering and Technology, Andhra Pradesh, India*

Abstract - This paper presents a Human-Independent AI-Based Productivity Enforcer designed to improve user productivity productivity by automatically restricting access to distracting applications and activating focus mode when necessary. Unlike traditional productivity tools that rely on user discipline, the through intelligent monitoring and automated control mechanisms. The system analyses user application usage patterns in real time and identifies non-productive behavior using machine learning techniques such as K-Means clustering. Based on this analysis, applications are classified into productive and non-productive categories. A threshold-based decision mechanism is applied to enforce proposed system ensures consistent enforcement through automation. The system is implemented using an Android-based interface integrated with backend services and locally executed AI models to ensure privacy and efficiency. Experimental results demonstrate a significant reduction in non-productive screen time and improved user focus. The proposed approach provides a scalable and adaptive solution for enhancing productivity among students and professionals.

Key Words: Artificial Intelligence, Productivity Enforcement, App Usage Monitoring, K-Means Clustering, Focus Mode, Behavior Analysis.

1. INTRODUCTION

In today's digital era, the widespread use of smartphones and internet-based applications has significantly increased user exposure to distractions such as social media, entertainment platforms, and continuous notifications. These distractions negatively impact concentration, time management, and overall productivity, especially among students and professionals. Existing productivity tools mainly focus on tracking user activity or providing reminders; however, they rely heavily on user self-discipline, making it difficult for users to sustain consistent productivity over extended periods of time..

To address these limitations, this paper proposes a Human-Independent AI-Based Productivity Enforcer that actively monitors and controls user behavior. The system utilizes real-time application usage tracking combined with machine learning techniques to identify non-productive patterns. By applying algorithms such as K-Means clustering, the system classifies applications based on their impact on productivity and dynamically adapts to user behavior over time.

A key contribution of this work is the implementation of an automated enforcement mechanism that restricts access to distracting applications and activates focus mode when predefined thresholds are exceeded. This reduces dependency on manual intervention and ensures consistent productivity. Additionally, the system incorporates locally executed AI models to maintain user privacy while delivering efficient performance.

The proposed solution provides a scalable, adaptive, and intelligent approach to productivity management, making it suitable for real-world applications.

2. LITERATURE REVIEW

Recent studies have highlighted the growing impact of digital distractions on user productivity, particularly due to excessive usage of smartphones and internet-based applications. Tools such as RescueTime and Forest have been developed to monitor application usage and provide insights into user behavior. While these tools help users understand their time utilization, they primarily depend on self-discipline and manual intervention, making it difficult for users to sustain consistent productivity over extended periods of time in work or studies.

Machine learning techniques have been increasingly used to analyse user behavior and identify productivity patterns. Research shows that parameters such as screen time, application usage frequency, and session duration can be used to classify user activities into productive and non-productive categories. Algorithms such as clustering and behavior prediction models have demonstrated effectiveness in identifying usage trends and improving productivity recommendations for the users.

In addition, advancements in mobile computing and context-aware systems have enabled real-time activity monitoring using smartphones. These systems utilize application usage statistics and sensor data to understand user behavior patterns. However, most existing solutions focus only on monitoring and suggesting improvements rather than actively enforcing productivity.

Another major limitation of current systems is their dependence on cloud-based processing, which raises concerns related to data privacy and security. User data is often transmitted and processed externally, increasing the risk of data exposure.

The proposed system addresses these limitations by integrating real-time monitoring, AI-based classification, and automated enforcement into a unified framework. Unlike traditional approaches, it not only analyses user behavior but also actively restricts non-productive activities through app blocking and focus mode. Furthermore, the use of locally executed AI models

ensures better privacy, faster processing, and improved system efficiency.

3. METHODOLOGY

The proposed system is designed to monitor user activity, analyse behavioural patterns, and enforce productivity using artificial intelligence and automated control mechanisms. The system operates through a combination of real-time data collection, machine learning-based classification, and rule-based enforcement using android controls to block the specific apps.

3.1 System Overview

The system consists of four major modules: App Usage Monitor, AI Classification Module, App Blocking Mechanism, and Focus Mode Controller. The App Usage Monitor continuously collects user activity data such as screen time, application usage duration, frequency of access, and session patterns. This data is then processed and analysed to identify user behavior trends
.

3.2 Data Collection

Data Collection and Feature Extraction User activity data is collected in real time using system-level APIs. The collected data includes parameters such as total usage time, number of app launches, session duration, and time-based usage patterns. These features are extracted and structured for further analysis by the AI model.

3.3 AI Classification

AI Classification The system utilizes the K-Means clustering algorithm to classify applications based on their usage behavior. Applications are grouped into categories such as productive, moderately used, and non-productive. The clustering process is based on features such as usage duration and frequency.

The objective of clustering is to minimize intra-cluster variance and maximize inter-cluster separation, enabling accurate classification of application usage patterns.

3.4 Algorithm

Productivity Enforcement Algorithm A threshold-based decision mechanism is implemented to enforce productivity. When the usage of a non-productive application exceeds a predefined threshold, the system automatically triggers restriction mechanisms.

Algorithm:

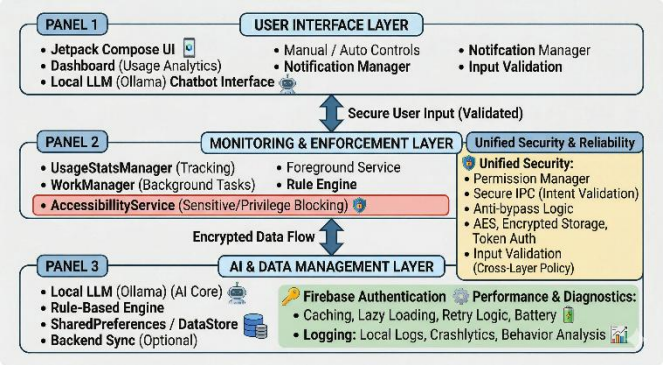
Step 1	Monitor user app usage in real time
Step 2	Extract usage features (time, frequency, duration)
Step 3	Apply K-Means clustering to classify apps
Step 4	Identify non-productive applications
Step 5	Compare usage with threshold limits
Step 6	If threshold exceeded → block application
Step 7	Activate focus mode and restrict notifications
Step 8	Update system behavior dynamically

Pseudo Code:

```
if usage_time(app) > threshold AND app == non_productive:
block(app)
activate_focus_mode()
else:
allow(app)
```

3.5 System Architecture

System Architecture The system follows a layered architecture consisting of an Android-based frontend, backend API services, database management, and AI processing modules. The frontend handles user interaction and displays usage analytics, while the backend processes data and manages communication between components. Firebase and MySQL are used for data storage, and AI models are executed locally using Python-based frameworks.



3.6 Working Mechanism

Working Mechanism, the system operates in a continuous cycle of monitoring, analysis, and enforcement. User activity is tracked in real time, analysed using AI techniques, and appropriate actions are taken automatically. The system adapts to user behavior over time, improving accuracy and personalization.

Overall, the methodology ensures an intelligent, adaptive, and automated approach to productivity management, reducing dependency on user self-discipline.

4. RESULTS AND DISCUSSION

The proposed system was evaluated to analyse its effectiveness in monitoring user activity and enforcing productivity. The App Usage Monitor accurately tracked screen time, application usage duration, and user interaction patterns in real time.

The App Blocking mechanism successfully restricted access to non-productive applications when predefined thresholds were exceeded. The system responded within a short duration, ensuring real-time enforcement

The Focus Mode feature significantly reduced user distractions by limiting notifications and restricting access to selected applications. This resulted in improved task engagement and reduced app-switching behavior.

The AI-based classification using K-Means clustering demonstrated high accuracy in identifying usage patterns and categorizing applications into productive and non-productive groups. The system dynamically adapted to user behavior, improving decision-making over time.

Overall, the results indicate a noticeable reduction in non-productive screen time and an improvement in user productivity. The integration of AI with automated enforcement mechanisms

proved to be effective in maintaining consistent focus and disciplined application usage

An experimental study was conducted on a group of users to evaluate the system performance. Fig. 1 shows the comparison of average daily screen time before and after implementing the system. A significant reduction in screen time can be observed across all users.

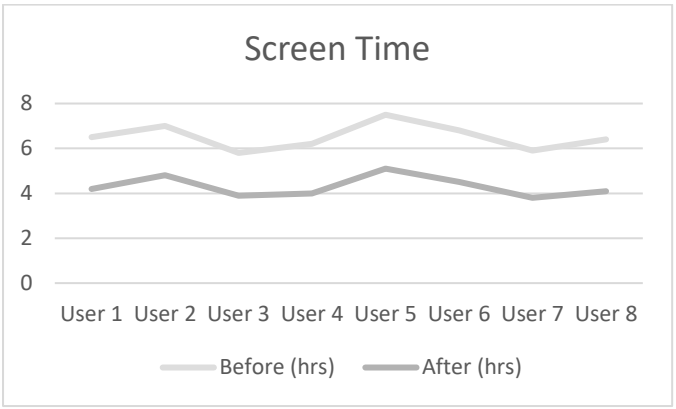


Fig. 1: Average daily screen time before and after AI-Based Productivity Enforcement

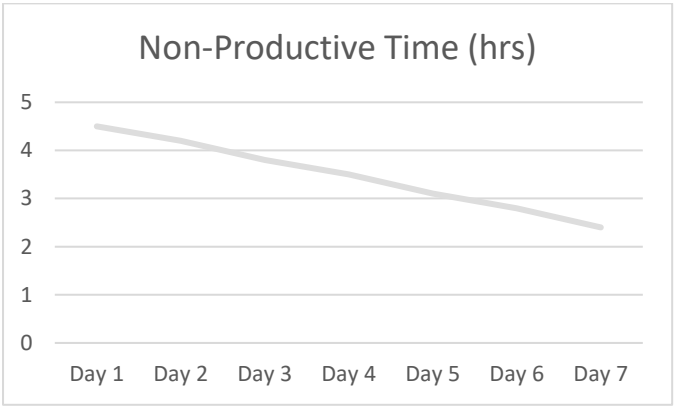


Fig. 2: illustrates the reduction in non-productive usage over a 7-day period. A steady decline is observed, indicating improved user behavior due to continuous monitoring and automated enforcement.

5. CONCLUSIONS

This paper presented a Human-Independent AI-Based Productivity Enforcer that utilizes artificial intelligence to monitor and control user behavior. The system successfully integrates real-time tracking, machine learning-based classification, and automated enforcement mechanisms to improve productivity.

The implementation of app blocking and focus mode significantly reduced distractions and improved user engagement. The use of K-Means clustering enabled accurate classification of application usage patterns, allowing the system to make intelligent decisions.

Unlike traditional productivity tools, the proposed system reduces dependency on user discipline by enforcing productivity automatically. The system also ensures user privacy through local AI execution and efficient data handling. Future work can focus on enhancing AI models, improving cross-platform compatibility, and incorporating more advanced behavioural analysis techniques.

REFERENCES

1. Maldonado, M., Chang, C.-C.K., Gravano, L., Paepcke, A.: The Stanford Digital Library Metadata Style. Int. J. Digit. Libr. 1 (1997) 108–121.

2. Bruce, K.B., Carcelle, L., Pierce, B.C.: Linking Object Encodings. In: Abadi, M., Ito, T. (eds.): Notional Aspects of CPU Software. Lecture Notes in Computer Science, Vol. 1281. Springer-Verlag, Berlin Heidelberg New York (1997) 415–438.

3. van Leeuwen, J. (ed.): Computer Skill Today. Recent Trends and Developments. Lecture Proceedings in Supercomputer Science, Vol. 1000. Springer-Verlag, Berlin Heidelberg New York (1995).

4. Michalewicz, Z.: Genetic Systems + Data Structures = Evolution Programs. 3rd ed. Springer-Verlag, Berlin Heidelberg New York (1996).

5. Pothuraju, V.V. Satyanarayana, et al.: AI-Powered Recommender Systems for E-Commerce. In: IEEE Proceedings of the International Conference on Recent Innovations in Science, Engineering and Technology (ICRISET-2025) (2025).

6. Pothuraju, V.V. Satyanarayana, et al.: Next-Generation National Voting Framework Using Aadhaar-Integrated Decentralized Blockchain and Analytics. Global Journal of Research in Engineering & Computer Sciences.

7. Pothuraju, V.V. Satyanarayana, et al.: Smart Contract-Based Decentralized Voting System for Transparent Elections on Ethereum Blockchain. International Scientific Journal of Engineering and Management.