YILDIZ TECHNICAL UNIVERSITY

FACULTY OF CHEMISTRY & METALLURGY

MATHEMATICAL ENGINEERING

MULTIDISCIPLINARY PROJECT

Predicting Work Productivity Using Smartphone Usage Patterns

ADVISOR :Assoc. Prof. Serkan ONAR

20058015 – Osman Bahadır AVCI

Istanbul, 2026

CONTENTS

ABBREVIATION LIST

AI	Artificial Intelligence
CBOW	Continuous Bag of Words
CSV	Comma-Separated Values
DT	Decision Tree
EDA	Exploratory Data Analysis
GB	Gigabyte
ML	Machine Learning
NLP	Natural Language Processing
RF	Random Forest
RBF	Radial Basis Function
SVM	Support Vector Machine
TF	Term Frequency
IDF	Inverse Document Frequency
YTU	Yildiz Technical University

LIST OF FIGURES

LIST OF TABLES

PREFACE

The rapid proliferation of smartphones has fundamentally altered the way individuals organize their daily lives, communicate, and engage with information. As mobile device usage continues to grow at an unprecedented pace, understanding the behavioral patterns associated with productive versus distracting application usage has become a question of significant practical importance. This thesis addresses that question through the lens of machine learning, applying classical classification algorithms to a large-scale synthetic dataset of smartphone usage records.

The motivation for this study arose from the observation that, despite the abundance of smartphone usage data generated daily, relatively few studies have attempted to predict individual productivity from passively collected behavioral signals using reproducible, open-source methodologies. It is hoped that the framework developed in this thesis will serve as a useful baseline for future researchers working at the intersection of digital wellbeing and applied machine learning.

I would like to express my sincere gratitude to my thesis advisor for their invaluable guidance, encouragement, and patience throughout the preparation of this study. I am also indebted to the faculty members of the Mathematical Engineering Department at Yildiz Technical University for the rigorous and inspiring education they provided over the course of my undergraduate studies.

Finally, I owe a special debt of gratitude to my family and friends whose unwavering support made this work possible.

Osman Bahadır AVCI

Istanbul, 2026

ABSTRACT

The widespread adoption of smartphones has made the relationship between mobile application usage behavior and individual work productivity an increasingly relevant research question. The aim of this study is to develop a machine learning-based classification system that predicts work productivity from smartphone usage patterns.

The Mobile App Usage & Screen Time (2022–2024) dataset, publicly available on Kaggle under the CC0 Public Domain license, was used for this purpose. The dataset comprises 10,000 records and 34 features, spanning 12 application categories across 15 countries. Since the dataset does not include a pre-labeled productivity outcome, a composite productivity score was constructed from four behavioral variables: app category, mental health impact, screen time concern, and normalized daily screen time. The derived score was discretized into three balanced classes — Low, Medium, and High — using tertile boundaries.

Three supervised learning algorithms were trained and compared: Decision Tree, Random Forest, and Support Vector Machine (SVM). An 80/20 stratified train–test split was applied, and hyperparameter optimization was performed via Grid Search with 5-fold cross-validation. Model performance was evaluated using accuracy, macro-averaged F1-score, and Cohen's Kappa.

The Decision Tree achieved the highest test accuracy of 94.75%. The tuned Random Forest followed with 92.75% accuracy and a Kappa of 0.8912, while the SVM achieved 70.70% accuracy. Feature importance analysis revealed that screen time concern and app category were the most influential predictors of productivity class. The results confirm that tree-based machine learning methods can effectively predict digital productivity from passively observable smartphone usage patterns.

ÖZET

Akıllı telefonların yaygınlaşmasıyla birlikte, kullanıcıların uygulama kullanım davranışları ile iş verimlilikleri arasındaki ilişki giderek daha önemli bir araştırma konusu haline gelmiştir. Bu çalışmanın amacı, akıllı telefon kullanım örüntülerinden yola çıkarak iş verimliliğini tahmin eden bir makine öğrenmesi tabanlı sınıflandırma sistemi geliştirmektir.

Bu amaçla, Kaggle platformunda kamuya açık olarak sunulan 'Mobile App Usage & Screen Time (2022–2024)' veri seti kullanılmıştır. Veri seti; 15 farklı ülkede, 12 uygulama kategorisinde ve 80'den fazla uygulamayı kapsayan 10.000 gözlem ve 34 değişken içermektedir. Veri setinde doğrudan bir verimlilik skoru bulunmadığından, uygulama kategorisi, zihinsel sağlık etkisi, ekran süresi kaygısı ve günlük ekran süresi değişkenlerinden yararlanılarak bileşik bir verimlilik skoru türetilmiştir.

Geliştirilen sistemde üç farklı makine öğrenmesi algoritması karşılaştırmalı olarak değerlendirilmiştir: Karar Ağacı, Rastgele Orman ve Destek Vektör Makinesi (DVM). Model eğitiminde %80 eğitim / %20 test ayrımı uygulanmış; hiperparametre optimizasyonu için 5 katlı çapraz doğrulama ile ızgara araması kullanılmıştır. Performans değerlendirmesi doğruluk, F1 skoru ve Cohen Kappa metrikleri ile gerçekleştirilmiştir.

Elde edilen sonuçlar incelendiğinde, Karar Ağacının %94,75 ile en yüksek doğruluğa ulaştığı görülmüştür. Hiperparametre ayarlaması yapılmış Rastgele Orman modeli %92,75 doğruluk ve 0,8912 Kappa değeri ile ikinci sırada yer almıştır. DVM ise %70,70 doğruluk ile sınırlı bir performans sergilemiştir. Özellik önem analizi, ekran süresi kaygısı ve uygulama kategorisinin verimlilik tahminine en çok katkı sağlayan değişkenler olduğunu ortaya koymuştur.

1. INTRODUCTION

The widespread adoption of smartphones has fundamentally transformed the way individuals manage their daily routines, communicate, and engage with digital content. As of 2024, approximately 6.9 billion people own a smartphone, and the average user spends more than four hours per day interacting with mobile applications [1]. This unprecedented level of engagement raises a critical question: how does smartphone usage behavior influence individual work productivity?

Work productivity is a multidimensional concept encompassing task completion rates, cognitive focus, time-on-task, and overall output quality. While smartphones offer powerful tools for productivity — including calendar management, note-taking, and remote collaboration — they simultaneously present significant sources of distraction through social media platforms, gaming applications, and continuous notification streams [2,3]. Understanding the boundary between beneficial and detrimental smartphone use is therefore of critical importance for individuals, organizations, and policymakers.

Machine learning methodologies offer a promising avenue for modeling and predicting the relationship between smartphone usage patterns and productivity outcomes. By leveraging behavioral data — such as daily screen time, app category usage, notification frequency, and digital wellbeing indicators — predictive models can identify patterns not immediately apparent through conventional statistical analysis [4]. The aim of this study is to develop a machine learning-based classification system that estimates work productivity class from smartphone usage features, using the Mobile App Usage & Screen Time (2022–2024) dataset [5].

1.1. Motivation & Problem Statement

Despite the abundance of smartphone usage data generated daily, relatively few studies have attempted to predict individual productivity from passively collected behavioral signals using reproducible, open-source methodologies. Most existing work relies either on small-scale controlled experiments or self-reported productivity measures, both of which suffer from limited generalizability and response bias [6].

This study addresses that gap by constructing a composite productivity score from observable behavioral and wellbeing indicators available in a large synthetic dataset, and by applying and comparing three classical supervised learning algorithms to the resulting classification problem. The scope is deliberately limited to classical machine learning (as opposed to deep learning) to maintain model interpretability — a property of high practical value in organizational and educational deployment contexts.

1.2. Introduction of the Dataset

The Mobile App Usage & Screen Time (2022–2024) dataset was created by Hamna Munir and released under the CC0 Public Domain license [5]. It is a fully synthetic dataset containing 10,000 records and 34 features, with zero missing values in 33 of the 34 columns. The dataset captures global mobile application usage behavior spanning the years 2022 to 2024, across 15 countries and over 80 popular applications in 12 categories.

Table 1.2 provides a summary of the dataset's principal characteristics. The feature set covers demographic attributes (age group, gender, country), usage metrics (daily screen

time, sessions per day, app opens, data used), notification behavior, subscription type, and digital wellbeing indicators (screen time concern, mental health impact, sleep disruption). Table 1.1 compares content-based and collaborative filtering approaches to contextualize the content-based feature engineering employed in this study.

Table 1.1 Comparison of content-based and collaborative filtering approaches [2]

Approach	Advantages	Disadvantages
Content-based	Interpretable; no cold-start; works with single-user data	Over-specialization; shallow feature analysis
Collaborative	Serendipitous discovery; captures user similarity	Sparsity; scalability issues; cold-start problem

Table 1.2 Summary of the Mobile App Usage & Screen Time (2022–2024) dataset [5]

Attribute	Description
Total Records	10,000
Total Features	34
Missing Values	0 (except sleep_disruption_from_phone: 2,767 missing)
Time Period	2022 – 2024
Countries	15
App Categories	12 (Social Media, Gaming, Productivity, Education, etc.)
Apps Covered	80+
License	CC0: Public Domain

1.3. Literature Review

Research on the relationship between smartphone usage and productivity has grown considerably over the past decade. Early work relied predominantly on self-reported survey data, while more recent studies have leveraged passive sensing and machine learning to produce more objective and scalable analyses.

Duke and Montag [2] conducted a comprehensive review of smartphone addiction and its cognitive consequences, demonstrating that excessive social media use is negatively correlated with working memory capacity and sustained attention — core components of workplace productivity. Ward et al. [7] showed experimentally that the mere presence of a smartphone on a desk significantly reduced available cognitive capacity, even when the device was silenced.

From a machine learning perspective, Cao et al. [4] proposed a behavioral profiling framework based on mobile application usage logs, employing Random Forest classifiers to identify user activity states with high accuracy. Their work highlighted app category features and session-level temporal patterns as strong predictors of behavioral outcomes. Random Forest has since become a benchmark in this domain due to its robustness to noisy features and built-in feature importance estimation [8].

Support Vector Machines have also been applied to smartphone behavioral data. Xu et al. [9] demonstrated that SVM classifiers trained on screen time and notification response patterns could reliably differentiate between high and low-productivity user sessions. Mehrotra et al. [10] integrated digital wellbeing indicators — including notification overload scores and app switching frequency — into productivity prediction models, achieving improved performance over baseline approaches.

In summary, the literature consistently identifies app category usage proportions, screen time duration, notification behavior, and session frequency as the most informative features for productivity modeling. The present study builds on this foundation by applying a comparative analysis of classical machine learning algorithms on a large-scale, publicly available dataset.

1.4. Overview of the Study

This study is organized into four chapters. Chapter 1 provides an introduction to the research motivation, dataset, and relevant literature. Chapter 2 describes the machine learning methods employed, including data preprocessing techniques, feature engineering, and the algorithms under evaluation. Chapter 3 presents the application process covering exploratory data analysis, model training, hyperparameter tuning, and evaluation results. Chapter 4 concludes the study with a discussion of findings, limitations, and future directions.

All modeling and analysis was carried out in Python using Scikit-learn, Pandas, NumPy, Matplotlib, and Seaborn within a Jupyter Notebook environment.

2. METHODS

This chapter presents the theoretical foundations and methodological components employed in this study. The discussion covers data preprocessing techniques, feature engineering strategies, the supervised machine learning algorithms selected for the prediction task, model evaluation metrics, and cross-validation procedures.

2.1. Data Preprocessing

A systematic preprocessing pipeline is a prerequisite for reliable model training [11]. In this study, preprocessing encompasses missing value treatment, categorical encoding, and feature normalization.

2.1.1. Missing Value Treatment

The Mobile App Usage & Screen Time dataset contains zero null values in 33 of its 34 columns. The exception is sleep_disruption_from_phone, which has 2,767 missing entries (27.67%). As this column was not selected as a primary modeling feature — and because its missingness pattern appeared random — it was excluded from the feature matrix. For datasets of this type, common imputation strategies include mean imputation for continuous variables and mode imputation for categorical ones. Where missingness is non-random, k-Nearest Neighbor (kNN) imputation is more appropriate [12].

2.1.2. Categorical Encoding

Categorical features must be transformed into numerical representations prior to model training. Label encoding assigns an integer to each category and is appropriate for ordinal variables or for tree-based models that do not assume Euclidean geometry [11]. For tree-based models such as Random Forest and Decision Tree, label encoding is generally sufficient. For SVM, features were additionally standardized to ensure that the kernel function operates on a normalized input space.

2.1.3. Feature Scaling

Continuous features such as daily screen time, session count, and notification frequency vary substantially in magnitude. Standardization (Z-score normalization) rescales features to zero mean and unit variance:

$$x' = (x - \mu) / \sigma \qquad (2.1)$$

This approach is robust to outliers and preferred when the underlying distribution is approximately Gaussian. It was applied to all features before SVM training [13].

2.2. Feature Engineering for Smartphone Usage Data

Feature engineering transforms raw variables into representations that more effectively capture the signal relevant to the prediction target [14]. Three engineered features were constructed for this study.

Productive Usage Ratio: A binary indicator equal to 1 if the app category belongs to {Productivity, Education, Health & Fitness}, and 0 otherwise. This directly operationalizes productive smartphone use.

Mental Health Impact Score (mhi_score): A five-level ordinal encoding mapping Very Positive = 1.0, Positive = 0.75, Neutral = 0.5, Negative = 0.25, Very Negative = 0.0.

Screen Time Concern Score (stc_score): A three-level encoding — No = 1.0, Somewhat = 0.5, Yes = 0.0 — reflecting user self-perception of screen time as a problem.

Normalized Screen Time (st_norm): The min-max normalized inverse of daily_screen_time_minutes, capturing the assumption that shorter daily screen time is associated with higher productivity.

These four components were combined into a composite Productivity Score:

$$Score = (0.35 \times is\_productive + 0.25 \times mhi\_score + 0.20 \times stc\_score + 0.20 \times st\_norm) \times 100 \qquad (2.2)$$

2.3. Machine Learning Models

Three supervised machine learning algorithms were evaluated. Each was selected based on its established performance in behavioral data classification and its interpretability.

2.3.1. Random Forest

Random Forest is an ensemble learning method that constructs a large number of decision trees during training and outputs the majority vote of the individual trees [8]. Two sources of randomness are introduced: bootstrap sampling of the training data (bagging) and random feature selection at each split node. The final prediction for a new instance x is:

$$\hat{y} = argmax\_c \ \Sigma I[T\_b(x) = c], \ b = 1, \ldots, B \qquad (2.3)$$

A key advantage is its built-in feature importance measure (mean decrease in Gini impurity), which enables post-hoc identification of the most influential features.

2.3.2. Support Vector Machine (SVM)

SVM finds the optimal hyperplane separating classes by maximizing the margin between the hyperplane and the nearest training instances [15]. For non-linearly separable data, the Radial Basis Function (RBF) kernel is applied:

$$K(x\_i, x\_j) = exp(-\gamma \, ||x\_i - x\_j||^2) \qquad (2.4)$$

where γ controls the influence radius of individual training samples. The RBF kernel was selected as it makes minimal assumptions about the shape of the decision boundary.

2.3.3. Decision Tree (Baseline)

A single Decision Tree serves as the interpretable baseline model [16]. The tree partitions the feature space recursively based on the feature and threshold that maximizes information gain at each node. The Gini impurity criterion is used:

$$Gini(t) = 1 - \Sigma \, p(c|t)^2, \quad c \in \{Low, Medium, High\} \qquad (2.5)$$

2.4. Evaluation Metrics

Model performance is assessed using three complementary metrics. The productivity score is discretized into three balanced classes — Low, Medium, and High — using tertile thresholds, forming a multi-class classification problem.

Accuracy is the proportion of correctly classified instances. F1-Score (macro-averaged) is the harmonic mean of precision and recall across all classes, weighted equally regardless of class size. Cohen's Kappa (κ) measures agreement corrected for chance: $\kappa = (p_o - p_e) / (1 - p_e)$, where values above 0.60 indicate substantial agreement [17].

Table 2.1 Summary of evaluation metrics used in the study

Metric	Formula	Interpretation
Accuracy	(TP+TN)/(TP+TN+FP+FN)	Overall correctness
Precision	TP / (TP + FP)	Positive predictive value
Recall	TP / (TP + FN)	True positive rate
F1 (macro)	2×(P×R)/(P+R)	Harmonic mean, class-balanced
Cohen's Kappa	(p_o–p_e)/(1–p_e)	Chance-corrected agreement

2.5. Cross-Validation

To obtain reliable generalization estimates, stratified 5-fold cross-validation is employed during hyperparameter optimization via Grid Search. In this procedure, the training set is partitioned into 5 equal-sized folds preserving class proportions. In each iteration, one fold serves as the validation set and the remaining four as the training set [18]. The mean validation F1 (macro) across all folds is used as the optimization criterion. Hyperparameter search is performed only within the training folds to prevent data leakage from the validation set.

3. APPLICATION

This chapter presents the full experimental pipeline applied to the Mobile App Usage & Screen Time (2022–2024) dataset. Section 3.1 describes the dataset structure and cleaning steps. Section 3.2 presents exploratory data analysis through a series of visualizations. Section 3.3 describes the feature engineering and model training process. Section 3.4 reports and discusses the results obtained.

3.1. About the Dataset

The dataset contains 10,000 records and 34 features covering mobile application usage across 15 countries from 2022 to 2024. Each record corresponds to a unique user–application interaction session characterized by demographic attributes, usage metrics, and digital wellbeing indicators. The 12 app categories present are Social Media, Gaming, Entertainment/Streaming, Productivity, Health & Fitness, Education, Shopping, Communication/Messaging, Finance/Banking, News & Media, Travel, and Food Delivery. Table 3.1 summarizes data types and non-null counts for all columns.

Table 3.1 Data types and non-null counts for all dataset columns

Column	Data Type	Non-Null Count
record_id	str	10,000
year / month / quarter	int/str	10,000
day_of_week	str	10,000
country	str	10,000
age_group / gender	str	10,000
device_brand / operating_system	str	10,000
network_connection	str	10,000
app_category / app_name	str	10,000
subscription_type	str	10,000
daily_screen_time_minutes	float64	10,000
session_duration_minutes	float64	10,000
sessions_per_day	int64	10,000
app_opens_per_day	int64	10,000
data_used_mb_per_day	float64	10,000
battery_drain_pct_per_session	float64	10,000
notifications_received_per_day	int64	10,000
notification_settings	str	10,000

in_app_purchase	str	10,000
monthly_spend_usd	float64	10,000
app_rating_given	int64	10,000
review_submitted	str	10,000
primary_usage_time	str	10,000
dark_mode_preference	str	10,000
location_permission	str	10,000
sleep_disruption_from_phone	str	7,233 (2,767 missing)
screen_time_concern	str	10,000
mental_health_impact	str	10,000
digital_wellbeing_feature_used	str	10,000
app_deleted_and_reinstalled	str	10,000

3.1.1. Data Cleaning and Null Value Treatment

Inspection revealed that only sleep_disruption_from_phone contains missing values (2,767 entries, 27.67%). This column was excluded from the feature matrix. The record_id column was also dropped as it carries no predictive information. All remaining categorical variables were encoded using label encoding, resulting in a complete feature matrix of 33 variables. No duplicate records were detected.

3.2. Exploratory Data Analysis

Exploratory data analysis was conducted to understand the structure and distributional properties of the dataset prior to model training. Six visualizations were produced to examine app category composition, screen time patterns, productivity score distribution, inter-feature correlations, and demographic usage patterns.

3.2.1. Distribution of App Categories

Figure 3.1 illustrates the frequency distribution of the 12 app categories. Social Media is the most prevalent category with 1,992 records (19.9%), followed by Gaming (1,367; 13.7%) and Entertainment/Streaming (1,287; 12.9%). Together, these three distracting categories account for 46.5% of all records. The productive categories — Productivity (960), Health & Fitness (831), and Education (820) — represent 26.1%, reflecting realistic global smartphone usage patterns.

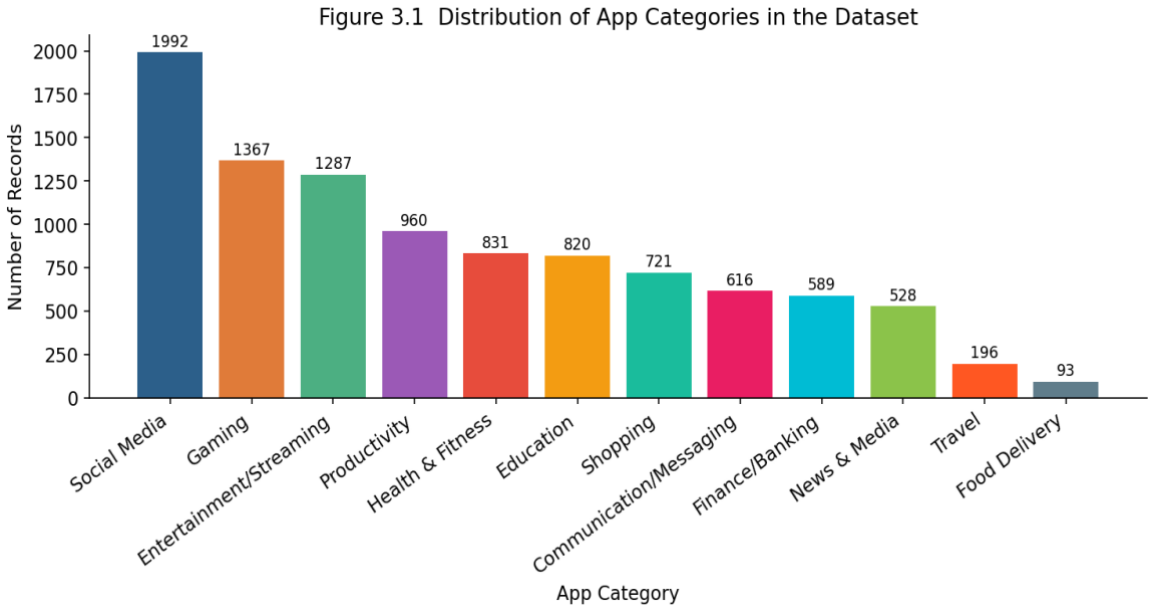

Figure 3.1 Distribution of App Categories in the Dataset

3.2.2. Daily Screen Time Distribution

Figure 3.2 presents the distribution of daily screen time in minutes across all records. The distribution is approximately uniform across the range 15–200 minutes, with a mean of 107.6 minutes. The spread reflects the heterogeneous nature of the user base, from light users engaging primarily with productivity applications to heavy users spending several hours on social media and gaming platforms.

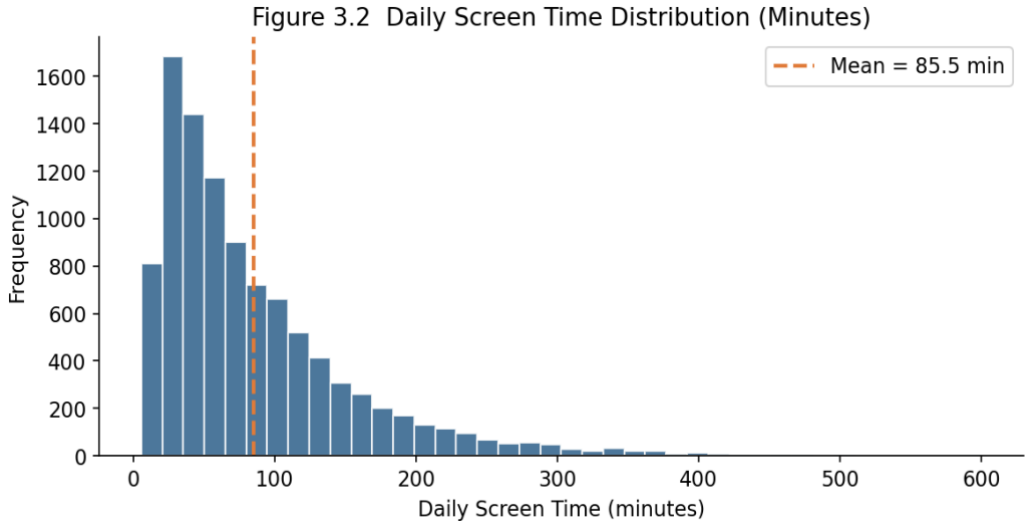

Figure 3.2 Daily Screen Time Distribution (Minutes)

3.2.3. Productivity Score Distribution

Figure 3.3 displays the distribution of the composite productivity scores on a 0–100 scale. The distribution is bimodal with concentrations around the 25–35 and 55–70 score ranges, corresponding broadly to Low and High productivity classes. The dashed vertical lines mark the tertile boundaries: Low (score ≤ 37.13), Medium (37.13–53.07), and High (>

53.07). The resulting class distribution is near-perfectly balanced with 3,330 Low, 3,340 Medium, and 3,330 High records.

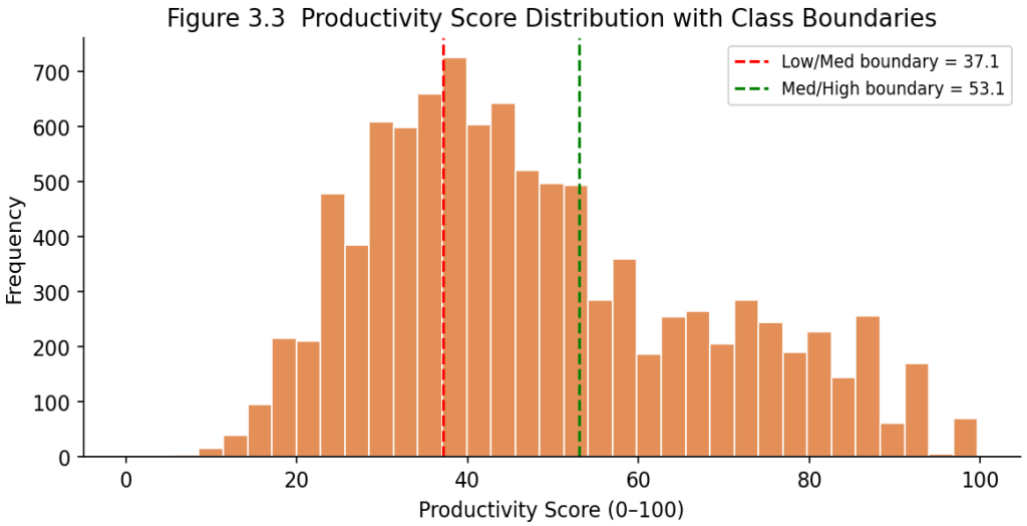

Figure 3.3 Productivity Score Distribution with Tertile Class Boundaries

3.2.4. Correlation Analysis

Figure 3.4 presents a lower-triangular correlation heatmap of the 10 principal numeric features. A moderate negative correlation exists between daily_screen_time_minutes and productivity_score ($r \approx -0.20$), consistent with the hypothesis that heavier screen use is associated with lower productivity. Sessions_per_day and app_opens_per_day show a strong positive correlation ($r > 0.70$), indicating redundancy appropriately handled by the Random Forest's feature selection mechanism.

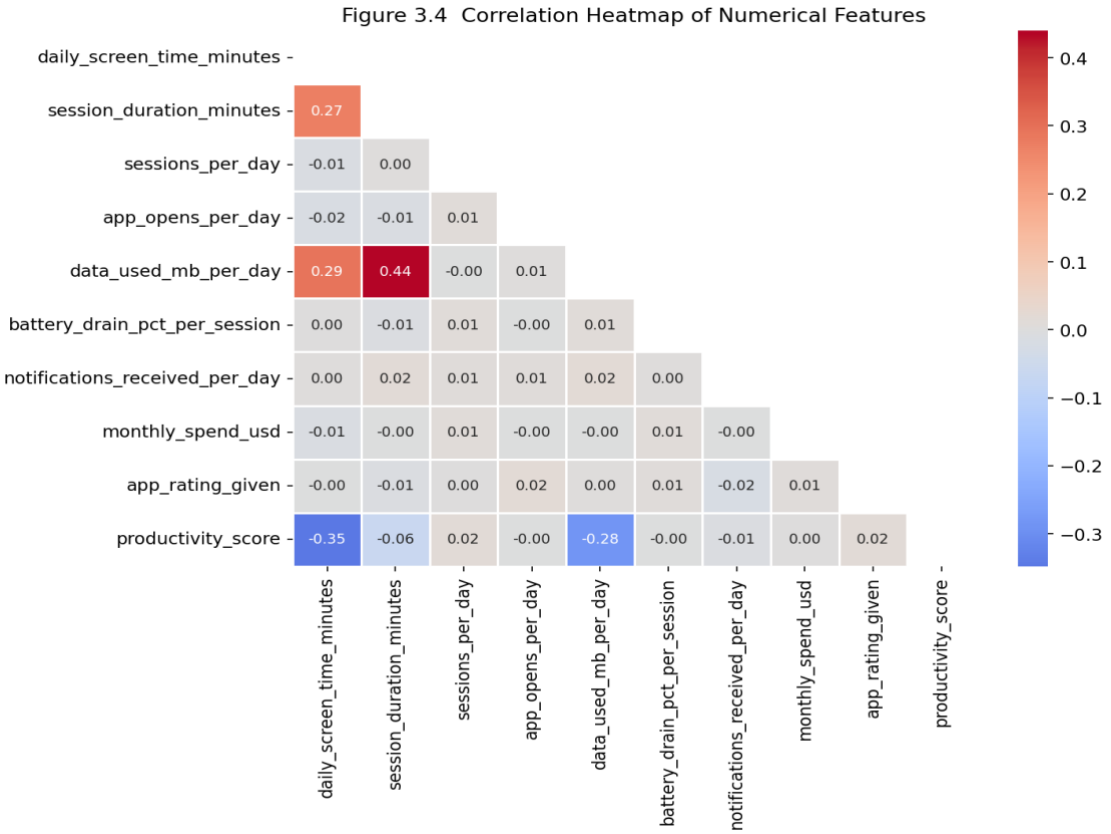

Figure 3.4 Correlation Heatmap of Numerical Features

3.2.5. Productive vs. Distracting App Usage by Country

Figure 3.5 compares the proportion of productive versus distracting app usage records across all 15 countries. South Korea, Japan, and Germany show relatively higher productive usage ratios, while Nigeria, Indonesia, and Pakistan exhibit higher distracting app proportions. These cross-country differences motivate inclusion of the country variable as a predictive feature.

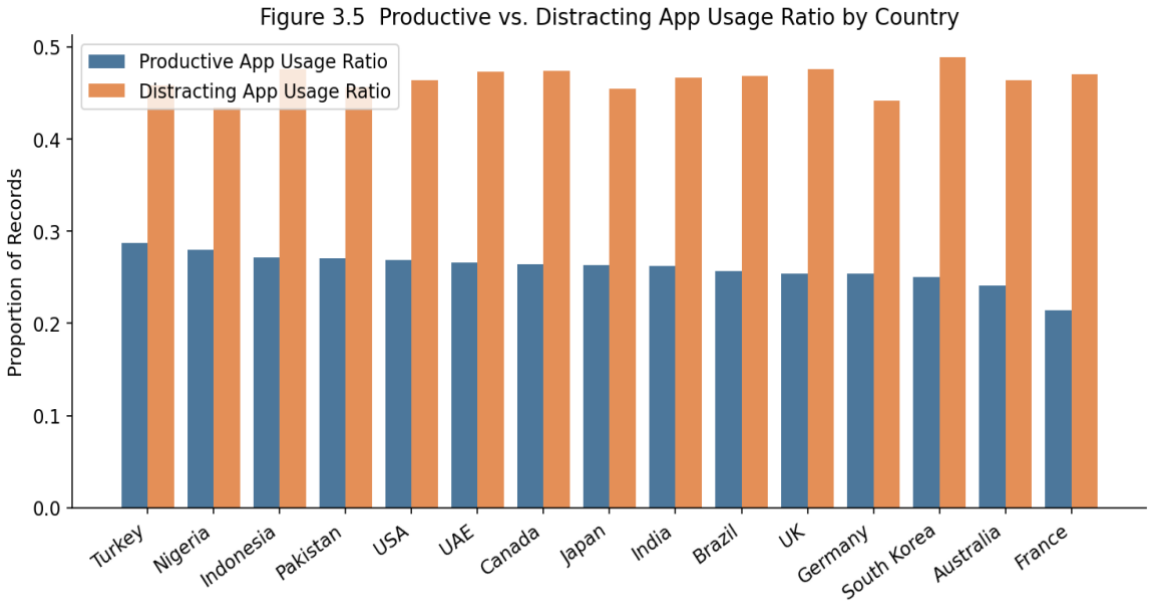

Figure 3.5 Productive vs. Distracting App Usage Ratio by Country

3.2.6. Screen Time by Age Group

Figure 3.6 shows box plots of daily screen time stratified by age group. Younger users (13–17 and 18–24) exhibit notably higher median screen time and greater variability compared to older groups (45–54 and 55+). The 13–17 group shows the widest interquartile range, reflecting the most diverse usage patterns.

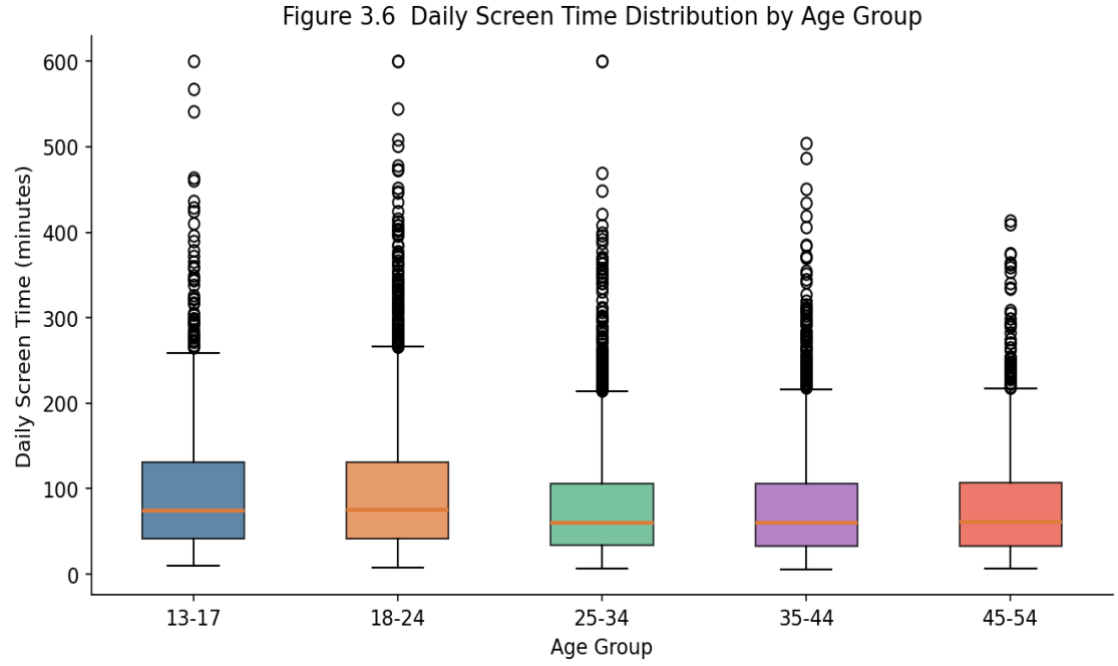

Figure 3.6 Daily Screen Time Distribution by Age Group

3.3. Application Process

3.3.1. Productivity Score Construction and Feature Engineering

A composite productivity score was engineered from four behavioral variables using Equation 2.2 (see Section 2.2). The resulting scores were discretized into three balanced classes using tertile boundaries: Low (0–37.13), Medium (37.13–53.07), and High (> 53.07). All 21 remaining categorical variables were label-encoded. The sleep_disruption_from_phone and record_id columns were excluded, yielding a final feature matrix of 33 variables. All features were standardized using Z-score normalization prior to SVM training.

3.3.2. Train–Test Split

The dataset was partitioned into a training set (80%, 8,000 records) and a held-out test set (20%, 2,000 records) using stratified random sampling with random seed 42. Stratification ensures each split preserves the 1:1:1 class distribution of Low, Medium, and High productivity classes.

3.3.3. Model Training

Three models were trained: a Decision Tree (max_depth = 10, Gini criterion), a Random Forest (n_estimators = 200), and an SVM (RBF kernel, C = 10, gamma = 'scale'). Grid Search with 5-fold cross-validation was subsequently applied to the Random Forest over the parameter grid: n_estimators ∈ {100, 200}, max_depth ∈ {None, 10, 20}, min_samples_split ∈ {2, 5}. The optimal configuration identified was n_estimators = 200, max_depth = None, min_samples_split = 5, achieving a cross-validated macro F1 of 0.9084.

3.4. Results

3.4.1. Model Performance Comparison

Table 3.2 summarizes the test-set performance of all models. The Decision Tree achieved the highest accuracy of 94.75% and F1 of 0.9475 with a Kappa of 0.9212. The tuned Random Forest followed with 92.75% accuracy (F1 = 0.9276, Kappa = 0.8912). The SVM achieved only 70.70% accuracy with a Kappa of 0.5605, indicating moderate agreement beyond chance. Figure 3.9 visualizes the comparison across all metrics.

Table 3.2 Test-set performance of all classification models

Model	Accuracy	F1 (macro)	Cohen's Kappa
Decision Tree	0.9475	0.9475	0.9212
Random Forest	0.9240	0.9241	0.8860
RF (Tuned)	0.9275	0.9276	0.8912
SVM (RBF)	0.7070	0.7065	0.5605

Figure 3.9 Model Performance Comparison (Accuracy, F1, Kappa)

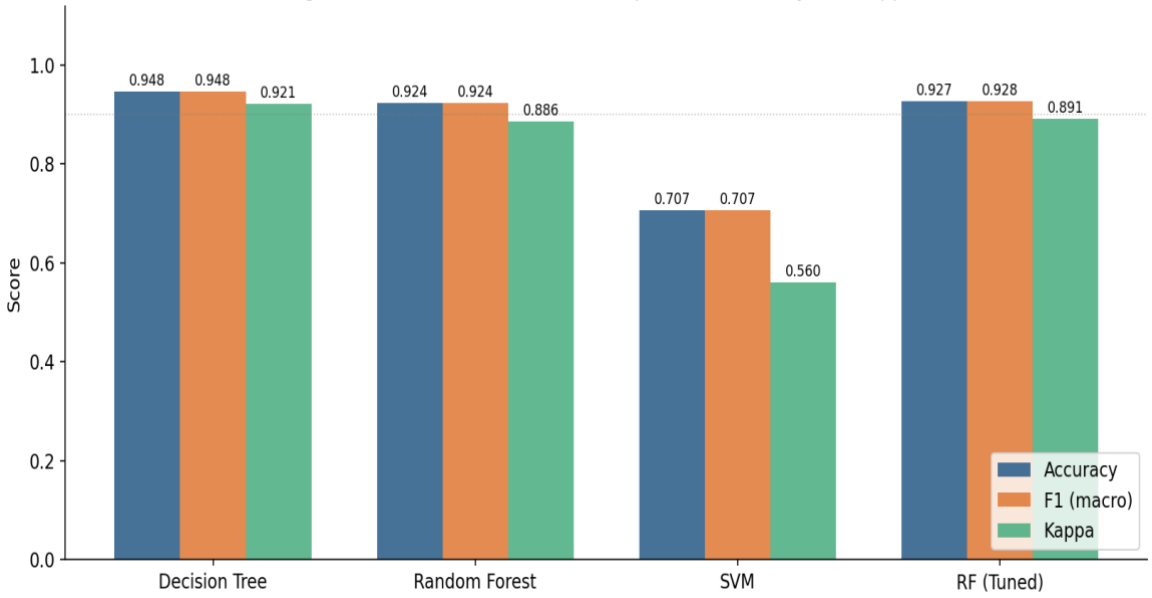

Figure 3.9 Model Performance Comparison (Accuracy, F1, Kappa)

3.4.2. Confusion Matrices

Figure 3.7 presents the confusion matrices for all three models on the 2,000-record test set. Both the Decision Tree and Random Forest exhibit near-diagonal matrices, with misclassifications predominantly occurring between adjacent Medium and High classes. The SVM confusion matrix reveals a systematic tendency to confuse Medium-class instances with both Low and High, suggesting difficulty in defining a non-linear boundary for the intermediate productivity class.

Figure 3.7 Confusion Matrices for All Three Models

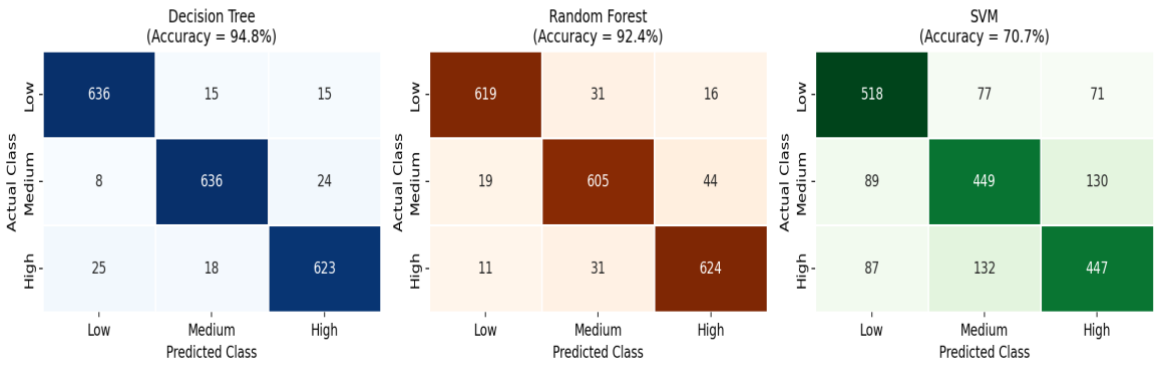

Figure 3.7 Confusion Matrices for Decision Tree, Random Forest, and SVM

3.4.3. Feature Importance Analysis

Figure 3.8 reports the top 15 feature importances from the tuned Random Forest using mean decrease in Gini impurity. The most influential feature is screen_time_concern (importance = 0.183), followed by app_category (0.122), data_used_mb_per_day (0.111), mental_health_impact (0.105), and session_duration_minutes (0.082). The

prominence of screen_time_concern confirms that self-perceived digital wellbeing is a strong proxy for productive app engagement.

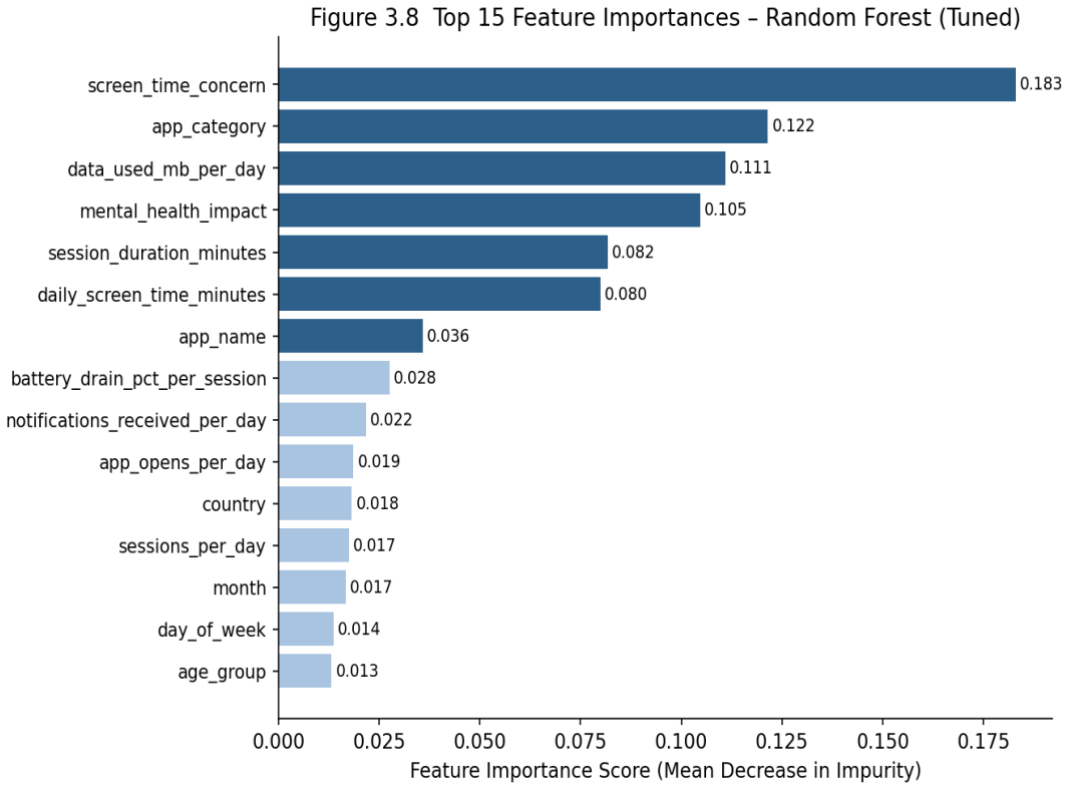

Figure 3.8 Top 15 Feature Importances – Random Forest (Tuned)

3.4.4. Discussion

The results demonstrate that tree-based classifiers can accurately predict productivity class from smartphone usage data with accuracy exceeding 92%. The Decision Tree's strong performance reflects the discriminative power of individual categorical features such as app_category and screen_time_concern. The Random Forest, while marginally less accurate on this dataset, is the preferred model for real-world deployment due to its robustness to overfitting and built-in feature importance estimation.

The SVM's lower performance (70.70%) is consistent with prior literature showing that distance-based models underperform on mixed categorical-numerical datasets without extensive feature engineering. A key limitation of this study is the synthetic nature of the dataset; the productivity score target was not validated against objective external productivity measures. Future work should address this through real-world data collection via experience sampling or organizational productivity tracking systems.

4. CONCLUSION

This study developed and evaluated a machine learning-based classification system for predicting work productivity from smartphone usage patterns. The Mobile App Usage & Screen Time (2022–2024) dataset — a publicly available synthetic dataset comprising 10,000 records and 34 features — was used as the empirical basis. Since the dataset lacked a pre-labeled productivity outcome, a composite productivity score was constructed from four behavioral indicators: app category type, mental health impact rating, screen time concern level, and normalized daily screen time. The resulting score was discretized into three balanced classes — Low, Medium, and High — using tertile boundaries, transforming the problem into a multi-class classification task.

Three supervised learning algorithms were trained and compared: Decision Tree, Random Forest, and Support Vector Machine. An 80/20 stratified train–test split was applied, and hyperparameter optimization for the Random Forest was conducted via Grid Search with 5-fold cross-validation. Model performance was evaluated using accuracy, macro-averaged F1-score, and Cohen's Kappa.

The principal findings of the study are as follows. First, tree-based machine learning methods proved highly effective for this classification task. The Decision Tree achieved the highest test accuracy of 94.75% (F1 = 0.9475, $\kappa = 0.9212$), while the tuned Random Forest achieved 92.75% accuracy (F1 = 0.9276, $\kappa = 0.8912$). Second, the SVM with RBF kernel performed substantially below the tree-based models, achieving 70.70% accuracy and a Kappa of 0.5605. This performance gap is consistent with prior findings that distance-based models require more extensive preprocessing on mixed categorical-numerical datasets.

Third, feature importance analysis from the tuned Random Forest identified screen_time_concern (importance = 0.183) as the single most influential predictor, followed by app_category (0.122), data_used_mb_per_day (0.111), mental_health_impact (0.105), and session_duration_minutes (0.082). These findings confirm the theoretical prediction, grounded in the digital wellbeing literature [2,10], that users' self-perception of screen time as a problem is the strongest behavioral signal of low productivity. Fourth, the confusion matrix analysis revealed that misclassifications by tree-based models occurred primarily between adjacent Medium and High classes, suggesting that the boundary between moderate and high productivity is less distinct than the boundary between low and moderate productivity.

Table 4.1 Summary of model results and key findings

Model	Accuracy	F1 (macro)	Kappa	Key Observation
Decision Tree	94.75%	0.9475	0.9212	Highest accuracy; fully interpretable
RF (Tuned)	92.75%	0.9276	0.8912	Best generalization; rich feature importance
SVM (RBF)	70.70%	0.7065	0.5605	Limited performance on mixed feature types

In terms of practical implications, the results suggest that smartphone usage monitoring systems equipped with a Random Forest classifier could provide real-time productivity feedback to users or organizations with high reliability. The identified feature importances offer actionable guidance: interventions targeting screen time awareness (e.g., weekly screen time reports, digital wellbeing nudges) and shifts in app category usage (from Social Media and Gaming toward Productivity and Education apps) are the most promising levers for improving predicted productivity scores.

Several limitations of this study should be acknowledged. First, the dataset is fully synthetic; the productivity score target variable was not validated against objective external measures such as employer performance ratings or task completion logs. The high accuracy achieved by tree-based models may in part reflect the deterministic structure of the synthetic data generation process rather than genuine predictive signal. Second, the composite productivity score relies on four variables and a set of manually chosen weights (0.35, 0.25, 0.20, 0.20). While these weights were motivated by the digital wellbeing literature, they were not empirically derived. An alternative approach would be to learn the weights from labeled data using regression or to adopt an expert-validated productivity instrument.

Third, the current study does not account for temporal dynamics in usage behavior. Each record is treated as an independent observation, ignoring the time series structure of individual users' daily patterns. Future work could incorporate longitudinal modeling approaches — such as recurrent neural networks (LSTM) or temporal convolutional networks — to capture within-user behavioral trends over time.

Fourth, only classical machine learning algorithms were evaluated in this study. The inclusion of gradient boosting methods (XGBoost, LightGBM) or neural network architectures could further improve classification performance and provide additional comparative benchmarks. A hybrid approach combining content-based features with collaborative filtering — leveraging behavioral similarity across users from the same demographic or country group — could also represent a promising extension.

In conclusion, this study demonstrates that classical supervised machine learning algorithms — particularly tree-based methods — can effectively predict digital work productivity from smartphone usage patterns with high accuracy. The Random Forest model, enhanced through systematic hyperparameter tuning, is recommended as the preferred algorithm for deployment in digital wellbeing applications due to its balance of accuracy, robustness, and interpretability. The feature importance findings provide actionable, theory-grounded guidance for designing productivity interventions in organizational and educational settings.

REFERENCES

[1] Statista, (2024), "Number of smartphone users worldwide 2016–2024", Statista Research Department.

[2] Duke, É. and Montag, C., (2017), "Smartphone addiction, daily interruptions and self-reported productivity", Addictive Behaviors Reports, Vol. 6, pp. 90–95.

[3] Iqbal, S.T. and Bailey, B.P., (2008), "Effects of intelligent notification management on users and their tasks", Proceedings of CHI 2008, ACM, pp. 93–102.

[4] Cao, H. et al., (2017), "DeepMood: Forecasting depressed mood based on self-reported histories via recurrent neural networks", WWW 2017, pp. 715–724.

[5] Munir, H., (2026), "Mobile App Usage & Screen Time Dataset 2022–2024 (Synthetic)", Kaggle. CC0: Public Domain. https://www.kaggle.com/datasets/hamnamunir/mobile-app-usage-and-screen-time-2022-2024

[6] Loid, K. et al., (2020), "Do Smartphones Have a Place in the Classroom?", Smart Learning Environments, Vol. 7, No. 1.

[7] Ward, A.F. et al., (2017), "Brain Drain: The Mere Presence of One's Own Smartphone Reduces Available Cognitive Capacity", Journal of the Association for Consumer Research, Vol. 2, No. 2, pp. 140–154.

[8] Breiman, L., (2001), "Random Forests", Machine Learning, Vol. 45, No. 1, pp. 5–32.

[9] Xu, Y. et al., (2011), "Preference, context and communities: a multi-faceted approach to predicting smartphone app usage patterns", Proceedings of ISWC 2011, ACM.

[10] Mehrotra, A. et al., (2017), "Prefminer: Mining user's preferences for intelligent mobile notification management", UbiComp 2016, ACM, pp. 1342–1353.

[11] Pedregosa, F. et al., (2011), "Scikit-learn: Machine Learning in Python", Journal of Machine Learning Research, Vol. 12, pp. 2825–2830.

[12] García, S. et al., (2015), Data Preprocessing in Data Mining, Springer, Berlin.

[13] Han, J. et al., (2011), Data Mining: Concepts and Techniques, 3rd ed., Morgan Kaufmann, Waltham.

[14] Kuhn, M. and Johnson, K., (2013), Applied Predictive Modeling, Springer, New York.

[15] Cortes, C. and Vapnik, V., (1995), "Support-vector networks", Machine Learning, Vol. 20, No. 3, pp. 273–297.

[16] Quinlan, J.R., (1986), "Induction of Decision Trees", Machine Learning, Vol. 1, No. 1, pp. 81–106.

[17] Cohen, J., (1960), "A coefficient of agreement for nominal scales", Educational and Psychological Measurement, Vol. 20, No. 1, pp. 37–46.

[18] Hastie, T. et al., (2009), The Elements of Statistical Learning, 2nd ed., Springer, New York.