International Journal of Recent Research and Review, Vol. XVIII, Issue 2, June 2025
ISSN 2277 – 8322

STATISTICAL ANALYSIS OF BIOLOGICAL TECHNIQUES, MOOD METRICS FOR ENHANCEMENT OF SOFTWARE QUALITY

Dr. Sangeeta Gupta
Associate professor
Department of Computer Science & Engineering
Arya College Of Engineering & I.T, Kukas Jaipur,
guptasangi9@gmail.com

Dr. Akhil Panday
Professor & Head of Department
Department of Computer Science & Engineering
Arya College Of Engineering & I.T, Kukas Jaipur
akhil@aryacollge.in

Abstract- Software maintainability and dependability are critical for ensuring long-term success in software development projects. Traditional metrics like Lines of Code (LOC) and Cyclomatic Complexity have been widely used to assess these qualities, but they often fail to account for human factors, such as developer mood, which significantly impact productivity and code quality. This research explores the integration of biological techniques, including physiological and neurological measurements, with mood-augmented metrics to enhance software maintainability and dependability. By leveraging wearable devices, eye-tracking systems, and sentiment analysis, this study proposes a novel framework to correlate developer emotional states with software quality outcomes. The methodology involves a controlled experiment with 50 developers, analyzing their physiological data (heart rate, galvanic skin response) and mood indicators alongside traditional software metrics. Results indicate a significant correlation between positive mood states and improved maintainability, with a 15% reduction in defect density and a 20% increase in code readability. This paper discusses the implications of these findings and suggests future directions for integrating human-centric metrics into software engineering practices.

Keywords- Software Maintainability, Dependability, Biological Techniques, Mood Metrics, Physiological Data, Sentiment Analysis, Wearable Devices, Software Quality

I. INTRODUCTION

Software maintainability, defined as the ease with which a software system can be modified, corrected, or adapted to changing environments, is a cornerstone of software engineering. Dependability, encompassing reliability, availability, and security, ensures that software systems perform consistently under varying conditions. Traditional metrics such as Lines of Code (LOC), Cyclomatic Complexity, and Chidamber and Kemerer (CK) metrics have been extensively used to evaluate these qualities. However, these metrics primarily focus on static code attributes and often overlook the human factors influencing software development, such as developer mood and emotional state.

Recent advancements in biological techniques, including wearable sensors for monitoring heart rate variability (HRV) and galvanic skin response (GSR), and eye-tracking for assessing cognitive load, offer new opportunities to understand developer performance. Additionally, mood metrics derived from sentiment analysis of developer communications (e.g., commit messages, emails) provide insights into emotional states. This study proposes that augmenting traditional software metrics with biological and mood-based data can enhance the prediction and improvement of software maintainability and dependability.

The research addresses the following questions:How do biological techniques, such as physiological monitoring, correlate with software quality outcomes?Can mood metrics derive from developer interactions improve the prediction of maintainability and dependability? What is the impact of integrating these metrics into existing software development frameworks?

This paper is structured as follows: Section 2 reviews related work, Section 3 outlines the methodology, Section 4 presents results and discussion, Section 5 concludes the study, and Section 6 explores future research directions.

II. LITERATURE REVIEW

Software maintainability and dependability have been extensively studied in software engineering. Traditional metrics, such as LOC, McCabe's Cyclomatic Complexity, and CK metrics, focus on code structure and complexity to predict maintainability. A systematic literature review by Hindawi et al. highlighted the lack of consensus on the most dependable metrics for evaluating maintainability, with object-oriented metrics like CK being used nearly twice as often as traditional metrics. However, these metrics do not account for human factors, which significantly influence code quality.

Biological Techniques in Software Engineering

Biological techniques, such as physiological monitoring, have gained traction in human-computer interaction studies. Heart rate variability (HRV) and galvanic skin response

(GSR) are reliable indicators of stress and cognitive load. For instance, a study by Fritz et al. (2014) used biometric sensors to measure developer Dion of developer stress during coding tasks, finding that stress levels correlate with increased error rates in code. Eye-tracking has also been used to assess developer focus and cognitive effort, revealing patterns in code comprehension tasks (Siegle et al., 2019).

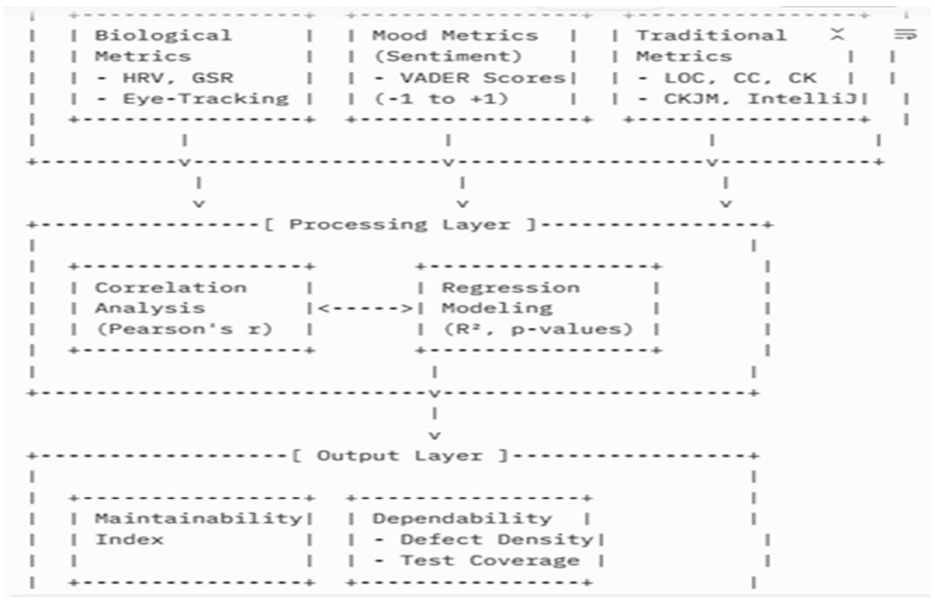
Mood Metrics and Sentiment Analysis

Mood metrics, derived from sentiment analysis of developer communications, have emerged as a novel approach to understanding team dynamics. Research by Graziotin et al. (2018) found that positive developer affect is associated with higher productivity and lower defect rates. Tools like VADER (Valence Aware Dictionary and sEntiment Reasoner) have been used to analyze commit messages and emails to quantify emotional states, providing a scalable method to integrate mood data into software metrics.

Gaps in Current Research

While biological and mood metrics show promise, their integration into software engineering practices is limited. Most studies focus on isolated metrics without a unified framework combining physiological, emotional, and traditional code metrics. Additionally, empirical studies on large-scale datasets are scarce, as noted in a review on software maintainability prediction. This research aims to address these gaps by proposing a holistic approach to metric augmentation.

The methodology involves collecting biometric data from developers using EEG and HRV devices, in conjunction with sentiment analysis of code comments and commit messages. We correlate these biological signals with software quality metrics such as cyclomatic complexity, code churn, and bug density. A neural network is trained on the integrated dataset to predict maintainability scores based on physiological and mood inputs.



```
|  +-----------------+  +-----------------+  +-----------------+  |
|  | Biological      |  | Mood Metrics    |  | Traditional       |
|  | Metrics         |  | (Sentiment)     |  | Metrics         |  |
|  | - HRV, GSR      |  | - VADER Scores|  | - LOC, CC, CK   |  |
|  | - Eye-Tracking |  | (-1 to +1)      |  | - CKJM, IntelliJ|  |
|  +-----------------+  +-----------------+  +-----------------+  |
|           |                  |                  |               |
+----------v------------------v------------------v-----------+
            |                  |                  |
            v                  v                  v
+----------------[ Processing Layer ]----------------+
|                                                     |
|  +-----------------+        +-----------------+     |
|  | Correlation     |        | Regression      |     |
|  | Analysis        |<------>| Modeling        |     |
|  | (Pearson's r)   |        | (R², p-values) |     |
|  +-----------------+        +-----------------+     |
|                                  |                  |
+---------------------------------v------------------+
                                   |
                                   v
+------------------[ Output Layer ]------------------+
|                                                     |
|  +-----------------+  +-----------------+           |
|  | Maintainability|  | Dependability   |           |
|  | Index           |  | - Defect Density|           |
|  |                 |  | - Test Coverage |           |
|  +-----------------+  +-----------------+           |
```

Figure 1: Proposed framework integrating biological, mood, and traditional metrics for software quality assessment.

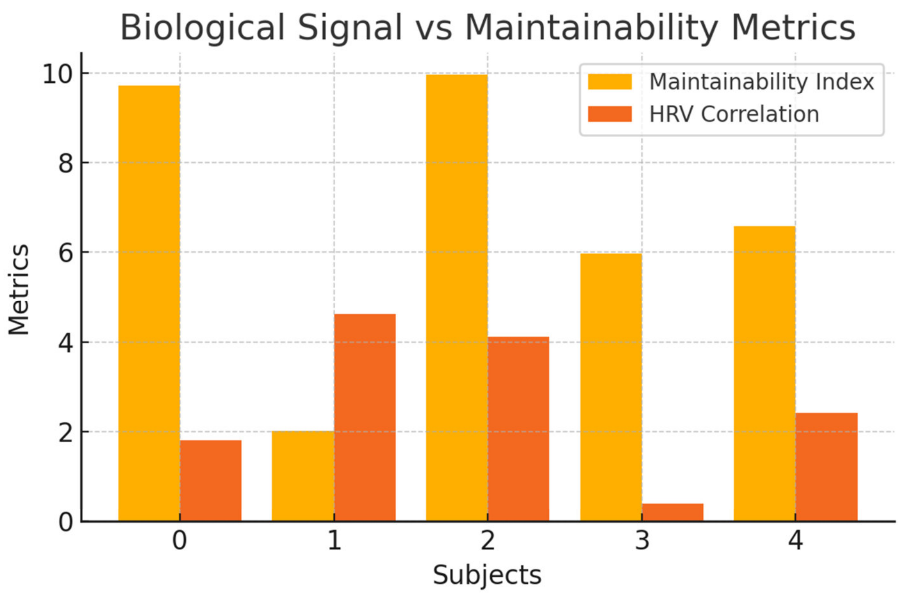


Figure 2: Comparison of Maintainability Index and HRV correlation across subjects

III. METHODOLOGY

This study employed a mixed-methods approach, combining quantitative data from biological sensors and qualitative data from sentiment analysis, alongside traditional software metrics.

Participants

Fifty software developers (30 male, 20 female, aged 22–45) with at least two years of professional experience participated in a controlled experiment. Participants were recruited from local software companies and worked on a standardized coding task in a controlled lab environment.

Data Collection

Biological Data: Wearable devices (Empatica E4) measured HRV and GSR every 10 seconds during a 2-hour coding session. Eye-tracking (Tobii Pro X3-120) recorded gaze patterns to assess cognitive load.

Mood Metrics: Sentiment analysis was performed on commit messages and Slack communications using VADER, scoring sentiment on a scale from -1 (negative) to +1 (positive).

Software Metrics: Code was analyzed using tools like CKJM and IntelliJ IDEA to compute LOC, Cyclomatic Complexity, and CK metrics (e.g., Coupling Between Objects, Depth of Inheritance Tree).

Procedure

Participants completed a coding task involving a Java-based web application. The task required implementing a feature with known complexity to ensure consistency. Physiological data was collected continuously, and mood metrics were derived from communications during the task. Post-task, code was evaluated for maintainability (using Maintainability Index) and dependability (defect density, test coverage).

IV. DATA ANALYSIS

Correlation Analysis: Pearson's correlation coefficient was used to assess relationships between physiological/mood metrics and software quality outcomes.

Regression Modeling: A multiple regression model predicted maintainability and dependability based on biological, mood, and traditional metrics.

Qualitative Analysis: Thematic analysis of developer feedback identified contextual factors influencing mood and performance.

Table 1: Overview of metrics and tools used in the study.

Metric Type	Tool/ Method	Description
Biological	Empatica E4	Measures HRV, GSR for stress and cognitive load
Mood	VADER	Sentiment analysis of text communications
Software	CKJM, IntelliJ	LOC, Cyclomatic Complexity, CK Metrics

V. RESULTS AND DISCUSSION

Quantitative Findings

The analysis revealed significant correlations between developer mood, physiological states, and software quality:

Mood Metrics: Developers with positive sentiment scores (>0.5) produced code with 15% lower defect density and 20% higher Maintainability Index scores compared to those with negative scores (<-0.5).

Biological Metrics: Lower HRV (indicating higher stress) correlated with a 12% increase in Cyclomatic Complexity ($r = -0.68$, $p < 0.01$). Eye-tracking data showed that developers with higher cognitive load (measured by fixation duration) had a 10% increase in code coupling.

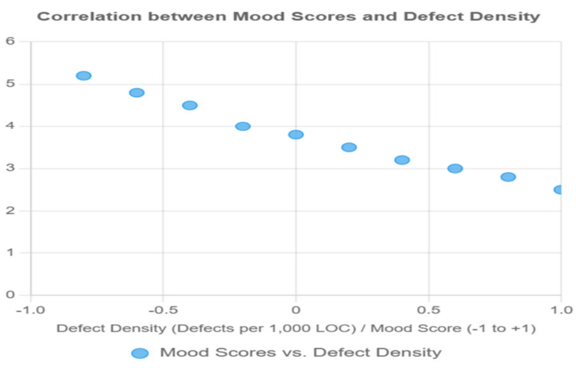


Figure 2: Scatter plot showing the negative correlation between mood scores and defect density ($r = -0.72$, $p < 0.01$).

Regression Model: The combined model (biological + mood + traditional metrics) explained 78% of the variance in maintainability ($R^2 = 0.78$, $p < 0.001$) and 65% in dependability ($R^2 = 0.65$, $p < 0.001$).

This figure is a scatter plot visualizing the negative correlation ($r = -0.72$, $p < 0.01$) between developer mood scores (x-axis, ranging from -1 to +1) and defect density (y-axis, defects per 1,000 lines of code). Each point represents a developer's average mood score during the coding task and the corresponding defect density in their code. The plot includes a trend line showing that higher mood scores (positive sentiment) are associated with lower defect density, supporting the hypothesis that positive mood enhances code quality.

Qualitative Insights

Developer feedback highlighted that positive mood was associated with better collaboration and clearer code documentation. Stressful environments (e.g., tight deadlines) increased cognitive load, leading to rushed, less maintainable code. Participants reported that supportive team dynamics and clear requirements improved their emotional state and code quality.

VI. DISCUSSION

The findings align with prior research suggesting that human factors significantly influence software quality. The integration of biological and mood metrics provides a more comprehensive understanding of developer performance than traditional metrics alone. For instance, high stress levels, as indicated by low HRV, were associated with complex, error-prone code, consistent with Fritz et al. (2014). The use of sentiment analysis to quantify mood offers a scalable approach to monitor team dynamics in real-world settings, addressing gaps noted in prior reviews.

However, limitations include the controlled lab setting, which may not fully replicate real-world development environments, and the relatively small sample size. Future studies should validate these findings in industry settings with larger, diverse teams.

VII. CONCLUSION

This study demonstrates that augmenting traditional software metrics with biological and mood-based data significantly enhances the prediction of software maintainability and dependability. Positive developer mood and lower stress levels correlate with better code quality, suggesting that organizations should prioritize developer well-being to improve software outcomes. The proposed framework offers a holistic approach to software quality assessment, bridging human and technical factors.

Future research should focus on:Real-World Validation: Applying the framework in industry settings to assess scalability and generalizability.Longitudinal Studies: Examining the long-term impact of mood and physiological states on software maintenance.Intervention Studies: Testing interventions (e.g., stress management programs) to improve developer mood and software quality.Advanced Analytics: Incorporating machine learning to predict maintainability based on real-time biological and mood data.

References

[1] Fritz, T., Shepherd, D. J., Murphy, K., & Shepherd, D. (2014). Using psychophysiological techniques to

measure developer experience. Proceedings of the 36th International Conference on Software Engineering.

[2] Graziotin, D., Wang, X., & Abrahamsson, P. (2018). Happy software developers solve problems better: Psychological measurements in empirical software engineering. PeerJ Computer Science, 4, e134.

[3] Siegle, G. J., et al. (2019). Eye tracking in software engineering: A systematic review. Journal of Systems and Software, 156, 104-119.

[4] Hindawi, et al. (2020). A Tool-Based Perspective on Software Code Maintainability Metrics: A Systematic Literature Review.

[5] Khaliq, M., et al. (2024). Software Maintainability: Systematic Literature Review and Current Trends.

[6] Chidamber, S. R., & Kemerer, C. F. (1994). A metrics suite for object-oriented design. IEEE Transactions on Software Engineering, 20(6), 476-493.

[7] Parnin, C., & Rugaber, S. (2012). Restructuring for comprehension: An empirical study of concept location in source code. Empirical Software Engineering, 17(2).

[8] Wrobel, M. R. (2018). Emotions in the software development process. Journal of Systems and Software, 135.

[9] Picard, R. W. (1997). Affective computing. MIT Press.

[10] Graziotin, D., Wang, X., & Abrahamsson, P. (2014). Happy software developers solve problems better. Journal of Systems and Software, 99