**Are We Testing or Being Tested? Exploring the Practical Applications of Large Language Models in Software Testing**

Robson de Souza Santos UNINASSAU Triunfo, PE, Brazil [robsonrtss@gmail.com](mailto:robsonrtss@gmail.com); Italo Santos Northern Arizona University Flagstaff, AZ, US [ids37@nau.edu](mailto:ids37@nau.edu); Cleyton Magalhaes CESAR School Recife, PE, Brazil cvcm@cesar.school; Ronnie de Souza Santos University of Calgary Calgary, AB, Canada [ronnie.desouzasantos@ucalgary.ca](mailto:ronnie.desouzasantos@ucalgary.ca)

**Aim of the Research Study:**

This study explores how software testing professionals are using Large Language Models (LLMs) such as GitHub Copilot, in different industrial settings. With the increasing integration of AI tools into the software development lifecycle, the research aims to uncover practical applications, benefits, and challenges faced by testers who employ LLMs for tasks like test case creation, debugging, and documentation. The central question the authors address is "How are software professionals using LLMs in their testing activities?".

**Methodology of the Study:**

The researchers conducted a cross-sectional survey distributed through the Prolific platform to software testers and quality assurance professionals worldwide. The survey included open- and closed-ended questions about LLM usage across different phases of the software testing lifecycle, such as requirements analysis, test planning, execution and bug fixing. Data from 80 valid responses were analyzed using both quantitative (descriptive statistics) and qualitative (thematic analysis) techniques to uncover trends and patterns in LLM adoption.

**Results of the Study:**

Findings show that 48% of respondents currently use LLMs in their testing work. LLMs are most frequently applied in early-stage activities (like requirements analysis and test planning), mid-phase tasks (like test execution and automation), and post-testing phases (bug fixing and regression testing). Participants noted LLMs' usefulness in generating test code, analyzing requirements, writing documentation, and debugging. However, the study also revealed that many testers are still cautious, often due to a lack of clear guidelines and concerns about reliability.

**Implications for Research and Practice:**

The study offers practical insights for both academia and industry. For researchers, it highlights the need to study underexplored uses of LLMs, such as in test reporting or software release processes. It also emphasizes the importance of establishing scientifically validated methods for LLM adoption. For practitioners, the paper shows real-world examples of how LLMs are already supporting various testing tasks and encourages thoughtful integration of these tools, particularly in automation and debugging, while cautioning against relying too much on them without appropriate safeguards.